Installing GT 5.2.3
Installing GT 5.2.3

Introduction

This guide is the starting point for everyone who wants to install Globus Toolkit 5.2.3. It will take you through a basic installation that installs the following basic services: a security infrastructure (GSI), GridFTP, and Execution Services (GRAM5).

This guide is also available as a PDF\(^1\). However, each component includes online reference material, which this guide sometimes links to.
Table of Contents

1. Before you begin ........................................................................................................... 1
   1. Typographical Conventions ......................................................................................... 1
2. Installing GT 5.2.3 ........................................................................................................... 2
   1. Installing from Native Linux Packages ................................................................. 2
      2. Installation from Source Installer ................................................................... 4
3. Basic Security Configuration ............................................................................................ 8
   1. Obtain host credentials ......................................................................................... 8
      2. Add authorization .................................................................................................. 9
      3. Verify Basic Security ............................................................................................. 9
      4. Firewall configuration ....................................................................................... 10
4. Basic Setup for GT 5.2.3 ............................................................................................... 14
5. Platform Notes .............................................................................................................. 15
   1. Platform Notes ..................................................................................................... 15
6. Appendix ....................................................................................................................... 16
Glossary ............................................................................................................................. 17
List of Tables

3.1. Summary of Globus Toolkit Traffic ................................................................. 12
Chapter 1. Before you begin

Before you start installing the Globus Toolkit 5.2.3, there are a few things you should consider. The toolkit contains several components, and you may only be interested in some of them.

The Globus Toolkit version 5.2.3 includes:

- **GSI**: security
- **GridFTP**: file transfer
- **GRAM**: job execution/resource management
- **MyProxy**: credential repository/certificate authority
- **GSI-OpenSSH**: GSI secure single sign-on remote shell

⚠ Important

These all run on Unix platforms only.

If you are new to the toolkit and want to experiment with the components, you may want to use a supported RedHat based or Debian based Linux system. With the new supported native packaging installs, they are the simplest platforms on which to install GT services.

For the purposes of this documentation, Globus is being installed on a machine called elephant.

1. Typographical Conventions

Where there is a command to be typed, it will be preceded by one of the following prompts:

```
elephant# , donkey#  Run this command as the root super-user, on the elephant or donkey hosts respectively. You might have to use a command like su(8) or sudo(8) to start a root shell before executing the command.

myproxy@elephant%  Run this command as the myproxy user, on the elephant host. This user is created automatically when the myproxy-server package is installed.

quser@elephant% , quser@donkey%  Run this command as the normal user account you are intending to interact with your Globus services, on the elephant or donkey hosts. In this document, we use the quser account for this, but if you have another user, you can use it for that purpose.
```

Commands themselves will be typeset as **run-this-command -with-arguments**, and responses to the commands like this **Some Response Text**. If there is some portion of a command which should be replaced by value, such as a version number, it will be typeset like this: **REPLACEME**.

Finally, in some cases you will be prompted for a passphrase. When that occurs, the entry of the passphrase will be indicated by ******, even though nothing will be printed to the screen.
Chapter 2. Installing GT 5.2.3

1. Installing from Native Linux Packages

1.1. Pre-requisites

We distribute the Globus Toolkit 5.2 as a set of RPM and Debian packages for Linux systems, as well as a source installer which can be used on other operating systems. In this quickstart, we will be installing RPM packages. Thus, it is a pre-requisite for following this quickstart that you are running a distribution for which we provide RPMs. If you are running a supported Debian or Ubuntu system, the process is very similar, but you'll need to use the **apt-get** or similar tools to install the packages. For the source installer, there is more work involved, and you'll need to consult the full installation guide.

First, we will to set up our system to use the Globus RPM repository. This repository contains the Globus software packages, signed by our build manager. We provide RPM packages that contain a yum source configuration file and the public key which can be used to verify the packages. If your distribution has Globus 5.2.3 packages within its repository, you can skip to the next section.

The RPMs for the various binary repositories can be found at the repo config download page.¹

To install from binary RPMs, get the linux distribution-specific rpm from the link above, install it with

```
elephant# rpm -hUv \       
    http://www.globus.org/ftppub/gt5/5.2/5.2.3/installers/repo/Globus-5.2.stable-config-DISTRO.noarch.rpm
```

For example, to install the Globus repository for Fedora 17, the command would be;

```
elephant# rpm -hUv \       
    http://www.globus.org/ftppub/gt5/5.2/5.2.3/installers/repo/Globus-5.2.stable-config-fedora-17-1.noarch.rpm
```

**Note**

The Debian and Ubuntu repositories have a different naming, and is based on the operating system release codenames. The pattern is `http://www.globus.org/ftppub/gt5/5.2/5.2.3/installers/repo/globus-repository-5.2-stable-CODENAME_0.0.3_all.deb`. So for Debian 6, codename `squeeze`, the repository package is located at `http://www.globus.org/ftp-pub/gt5/5.2/5.2.3/installers/repo/globus-repository-5.2-stable-squeeze_0.0.3_all.deb`.

Once you've installed the Globus repository package, you can use your operating system's packaging tools, such as **yum**, to install the Globus components.

**Important**

For operating systems based on RHEL (such as Red Hat Enterprise Linux, CentOS, and Scientific Linux), the compatible EPEL repository must be enabled before installing myproxy. For OS versions 5.x, install the **EPEL 5 package**, and for OS version 6.x, use **6 package**.

For information about installing these, see the **EPEL FAQ**.

¹ [http://www.globus.org/ftppub/gt5/5.2/5.2.3/installers/repo/](http://www.globus.org/ftppub/gt5/5.2/5.2.3/installers/repo/)
⁴ [http://fedoraproject.org/wiki/EPEL/FAQ#How_can_I_install_the_packages_from_the_EPEL_software_repository](http://fedoraproject.org/wiki/EPEL/FAQ#How_can_I_install_the_packages_from_the_EPEL_software_repository)
This step is not needed for Fedora, Debian, or Ubuntu systems.

1.2. Installing the Toolkit

The components of the toolkit can be installed separately, or all at once. This section will show how to install various components, on both RPM based and Debian based Linux systems.

For RPM-based systems, used the `yum` command to install the Globus components and their dependencies. For Debian-based systems, used the `apt-get` or `aptitude` commands.

For example, to install the GridFTP client tools, do the following for RPM-based systems:

```
elephant# yum install globus-data-management-client
```

Do the following for Debian-based systems:

```
elephant# apt-get install globus-data-management-client
```

1.2.1. Package Groups

The Globus Toolkit distribution includes several high-level package groups that can be used to install multiple packages to enable full client or server functionality of some Globus Toolkit component.

These packages are:

- `globus-gridftp` GridFTP client and server tools
- `globus-gram5` GRAM5 client and server tools
- `globus-gsi` Globus Security Infrastructure tools for managing certificates and proxies
- `globus-data-management-client` Client Tools for data management, including the GridFTP client programs and `globus-url-copy`
- `globus-data-management-sdk` Development headers and documentation for writing applications using the GridFTP APIs.
- `globus-resource-management-server` Server tools for deploying a GRAM5 resource manager
- `globus-resource-management-client` Client tools for resource management, including the globusrun tool, and the `globus-job-*` tools.
- `globus-resource-management-sdk` Development headers and documentation for writing applications using the GRAM5 APIs.

1.3. Updating a Globus Installation

Starting with GT 5.2, the package repositories included with the repo configuration packages will have updates enabled. That means that all major bug fixes and security issues for GT 5.2.3 can be easily installed via `yum` or `apt-get`. These updates will be published in the GT 5.2 updates rss feed. Also, this means that when the next point release is made, collecting other minor bug fixes, the upgrade can be done via `yum` or `apt-get` without installing a new repository definition package.

1. http://www.globus.org/toolkit/rss/advisories/5.2.rss
2. Installation from Source Installer

To build the Globus Toolkit from the source installer, first download the source from download page\(^1\), and be sure you have all of the following prerequisites installed.

This table shows specific package names (where available) for systems supported by GT 5.2.3:

<table>
<thead>
<tr>
<th>Prerequisite</th>
<th>Reason</th>
<th>RedHat-based Systems</th>
<th>Debian-based Systems</th>
<th>Solaris 11</th>
<th>Mac OS X</th>
</tr>
</thead>
<tbody>
<tr>
<td>C Compiler</td>
<td>Most of the toolkit is written in C, using C99 and POSIX.1 features and libraries.</td>
<td>gcc</td>
<td>gcc</td>
<td>pkg:/developer/gcc-45 or Solaris Studio(^7) 12.3</td>
<td>XCode(^8)</td>
</tr>
<tr>
<td>GNU or BSD tar</td>
<td>GPT uses the (-z) option to manipulate compressed tar files.</td>
<td>tar</td>
<td>tar</td>
<td>pkg:/archiver/gnu-tar</td>
<td>(included in OS)</td>
</tr>
<tr>
<td>GNU or BSD sed</td>
<td>Standard sed does not support long enough lines to process autoconf-generated scripts and Makefiles</td>
<td>sed</td>
<td>sed</td>
<td>pkg:/text/gnu-sed</td>
<td>(included in OS)</td>
</tr>
<tr>
<td>GNU Make</td>
<td>Standard make does not support long enough</td>
<td>make</td>
<td>make</td>
<td>pkg:/developer/build/gnu-make</td>
<td>(included in XCode)</td>
</tr>
</tbody>
</table>

---

\(^1\) [http://www.globus.org/toolkit/downloads/5.2.3](http://www.globus.org/toolkit/downloads/5.2.3)


\(^8\) [https://developer.apple.com/xcode/](https://developer.apple.com/xcode/)
## Installing GT 5.2.3

<table>
<thead>
<tr>
<th>Prerequisite</th>
<th>Reason</th>
<th>RedHat-based Systems</th>
<th>Debian-based Systems</th>
<th>Solaris 11</th>
<th>Mac OS X</th>
</tr>
</thead>
<tbody>
<tr>
<td>lines to process autoconf-generated makefiles</td>
<td>libtool-ltdl-dev</td>
<td>libltdl-dev</td>
<td>pkg:/library/libtool/libltdl</td>
<td>Included in XCode for MacOS X 10.5-10.7; for newer versions, you must install it yourself. See OS X Platform Notes for more information.</td>
<td></td>
</tr>
<tr>
<td>The Globus Toolkit uses this library to portably load shared libraries.</td>
<td>openssl-devel</td>
<td>libssl-dev</td>
<td>pkg:/library/security/openssl</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GSI security uses OpenSSL's implementation of the SSL protocol and X.509 certificates.</td>
<td>perl</td>
<td>perl</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT and parts of GRAM5 are written in Perl</td>
<td>perl-Archive-Tar</td>
<td>perl-modules</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses Archive::Tar to manipulate packages</td>
<td>perl-Compress-Zlib</td>
<td>perl-modules</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses Compress::Zlib to deal with compressed packages.</td>
<td>perl</td>
<td>perl</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses Digest::MD5 to compute package digests.</td>
<td>perl</td>
<td>perl</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses File::Spec indirectly via Pod::Parser</td>
<td>perl</td>
<td>perl-base</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses IO::Zlib to deal with compressed packages.</td>
<td>perl-IO-Zlib</td>
<td>perl-modules</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
<tr>
<td>GPT uses Pod::Parser to generate command-line help screens.</td>
<td>perl</td>
<td>perl-modules</td>
<td>pkg:/runtime/perl-512</td>
<td>(included in base OS)</td>
<td></td>
</tr>
</tbody>
</table>
### Prerequisite

<table>
<thead>
<tr>
<th>RedHat-based Systems</th>
<th>Debian-based Systems</th>
<th>Solaris 11</th>
<th>Mac OS X</th>
</tr>
</thead>
<tbody>
<tr>
<td>perl-Test-Simple</td>
<td>perl-modules</td>
<td>Install <code>Test::Simple</code> from CPAN</td>
<td>(included in base OS)</td>
</tr>
<tr>
<td>perl-XML-Parser</td>
<td>libxml-parser-perl</td>
<td>pkg://library/perl-5/xml-parser-512</td>
<td>(included in base OS)</td>
</tr>
</tbody>
</table>

### Note

In order to use the GNU versions of `sed`, `tar`, and `make` on Solaris, put `/usr/gnu/bin` at the head of your path. Also, to use all of the `perl` executables, add `/usr/perl5/bin` to your path.

### 2.2. Installing from Source Installer

1. Create a user named `globus`. This non-privileged user will be used to perform administrative tasks, deploying services, etc. Pick an installation directory, and make sure this account has read and write permissions in the installation directory.

   ![Tip](image)

   You might need to create the target directory as root, then chown it to the `globus` user:

   ```bash
   elephant# mkdir /usr/local/globus-5.2.3
   elephant# chown globus:globus /usr/local/globus-5.2.3
   ```

   ![Important](image)

   If for some reason you do not create a user named `globus`, be sure to run the installation as a non-root user. In that case, make sure to pick an install directory that your user account has write access to.

2. Download the required software noted in Section 2.1, “Required software”.

3. The Globus Toolkit Source Installer sets the installation directory by default to `/usr/local/globus-5.2.3`, but you may replace `/usr/local/globus-5.2.3` with whatever directory you wish to install to, by setting the prefix when you configure.

   As the `globus` user, run:

   ```bash
   globus@elephant% ./configure --prefix=YOUR_PREFIX_DIRECTORY
   ```

   You can use command line arguments to `./configure` for a more custom install. Here are the lines to enable features which are disabled by default:

   ```bash
   Optional Packages:
   [...] --with-gsiopenshargs="args"
   Arguments to pass to the build of GSI-OpenSSH, like
   --with-tcp-wrappers
   ```

   ^9 http://search.cpan.org/search?mode=all&query=Test%3A%3ASimple
Installing GT 5.2.3

For a full list of options, see `./configure --help`. For a list of GSI-OpenSSH options, see Optional Build-Time Configuration for GSI-OpenSSH. For more information about our packaging or about choosing a flavor, see Packaging Details for Installing GT.

4. The source installer will build all of the globus toolkit packages in the default make rule. The same `package groups` as the native packages may be used to build and install a subset of the toolkit.

   Run:

   ```bash
   globus@elephant% make PACKAGE-GROUPS
   
   Note that this command can take a while to complete. If you wish to have a log file of the build, use `tee`:
   
   globus@elephant% make 2>&1 | tee build.log
   
   The syntax above assumes a Bourne shell. If you are using another shell, redirect stderr to stdout and then pipe it to `tee`.
   
   Note
   
   Using make in parallel mode (-j) does not work with the installer makefile.
   
5. Finally, run:

   ```bash
   globus@elephant% make install
   
   This completes your installation. Now you may move on to the configuration sections of the following chapters.
   
   We recommend that you install any security advisories available for your installation, which are available from the Advisories page. You may also be interested in subscribing to some mailing lists for general discussion and security-related announcements.

2.3. Updating an Installation

The updates available in the native packages described above are also published as GPT source packages on the updates page. To install update packages, use the command

   ```bash
   globus@elephant% gpt-build -update package-name flavors
   
   For the update command, `package-name` is the full path to the update tarball you've downloaded, and `flavors` is the list of binary flavors that you have installed (typically gcc32dbg or gcc64dbg).

---

2 [http://dev.globus.org/wiki/Mailing_Lists](http://dev.globus.org/wiki/Mailing_Lists)
Chapter 3. Basic Security Configuration

1. Obtain host credentials

You must have X.509 certificates to use the GT 5.2.3 software securely (referred to in this documentation as host certificates). For an overview of certificates for GSI (security) see GSI Configuration Information and GSI Environment Variables.

If you will need to be interoperable with other sites, you will need to obtain certs from a trusted Certificate Authority, such as those that are included in IGTF. If you are simply testing the software on your own resources, SimpleCA offers an easy way to create your own certificates (see section below).

Host credentials must:

• consist of the following two files: hostcert.pem and hostkey.pem
• be in the appropriate directory for secure services: /etc/grid-security/
• match the hostname for a the machine. If the machine is going to be accessed remotely, the name on the certificate must match the network-visible hostname.

You have the following options:

1.1. Request a certificate from an existing CA

Your best option is to use an already existing CA. You may have access to one from the company you work for or an organization you are affiliated with. Some universities provide certificates for their members and affiliates. Contact your support organization for details about how to acquire a certificate. You may find your CA listed in the TERENA Repository.

If you already have a CA, you will need to follow their configuration directions. If they include a CA setup package, follow the CAs instruction on how to install the setup package. If they do not, you will need to create an /etc/grid-security/certificates directory and include the CA cert and signing policy in that directory. See Configuring a Trusted CA for more details.

This type of certificate is best for service deployment and Grid inter-operation.

1.2. SimpleCA

SimpleCA provides a wrapper around the OpenSSL CA functionality and is sufficient for simple Grid services. Alternatively, you can use OpenSSL’s CA.sh command on its own. Instructions on how to use the SimpleCA can be found in Installing SimpleCA.

SimpleCA is suitable for testing or when a certificate authority is not available.

If you install the globus-simpleca native package, it will automatically create a CA and host certificate if you don't have one configured yet. Otherwise, you'll need to use grid-ca-create to create the CA and grid-default-ca to make that the default for requesting credentials.

---

1 http://www.igtf.net
2 http://www.tacar.org/
To create user credentials, you can run the command `grid-cert-request` as a user that you want to create a credential for. You can then run the `grid-ca-sign` command as the `simpleca` user to sign the certificate.

### 2. Add authorization

Installing Globus services on your resources doesn't automatically authorize users to use these services. Each user must have their own user certificate, and each user certificate must be mapped to a local account.

To add authorizations for users, you'll need to update the `grid-mapfile` database to include the mapping between the credentials and the local user accounts.

You'll need two pieces of information:

- the subject name of a user's certificate
- the local account name that the certificate holder can access.

To start with, if you have created a user certificate, you can run the `grid-cert-info` command to get the certificate's subject name, and `id -un` to get the account name:

```bash
globus@elephant% grid-cert-info -subject
/O=Grid/OU=GlobusTest/OU=simpleCA-elephant.globus.org/CN=Globus User

globus@elephant% id -un
globus
```

You may add the line by running the following command as root:

```bash
elephant# grid-mapfile-add-entry -dn "/O=Grid/OU=GlobusTest/OU=simpleCA-elephant.globus.org/CN=Globus User" -ln gtuser
```

**Important**

The quotes around the subject name are *important*, because it contains spaces.

### 3. Verify Basic Security

Now that you have installed a trusted CA, acquired a hostcert and acquired a usercert, you may verify that your security setup is complete. As your user account, run the following command:

```bash
gtuser$ grid-proxy-init -verify -debug
```

- **User Cert File:** `/home/gtuser/.globus/usercert.pem`
- **User Key File:** `/home/gtuser/.globus/userkey.pem`
- **Trusted CA Cert Dir:** `/etc/grid-security/certificates`
- **Output File:** `/tmp/x509up_u506`

Your identity: `/DC=org/DC=doegrids/OU=People/CN=GT User 332900`

Enter GRID pass phrase for this identity:
Basic Security Configuration

Creating proxy ...++++++++++++
..................++++++++++++
Done
Proxy Verify OK
Your proxy is valid until: Fri Jan 28 23:13:22 2005

There are a few things you can notice from this command. Your usercert and key are located in
$HOME/.globus/. The proxy certificate is created in /tmp/. The "up" stands for "user proxy", and the _u506
will be your UNIX userid. It also prints out your distinguished name (DN), and the proxy is valid for 12 hours.

If this command succeeds, your single node is correctly configured.

If you get an error, or if you want to see more diagnostic information about your certificates, run the following:

gtuser$ grid-cert-diagnostics

For more troubleshooting information, see the GSI troubleshooting guide

4. Firewall configuration

There are four possible firewall scenarios that might present themselves: restrictions on incoming and outgoing ports
for both client and server scenarios.

This section divides sites into two categories: client sites, which have users that are acting as clients to Grid services,
and server sites, which are running Grid services. Server sites also often act as client sites either because they also
have users on site or jobs submitted by users to the site act as clients to other sites by retrieving data from other sites
or spawning sub-jobs.

4.1. Client Site Firewall Requirements

This section describes the requirements placed on firewalls at sites containing Globus Toolkit clients. Note that often
jobs submitted to sites running Globus services will act as clients (e.g. retrieving files needed by the job, spawning
subjobs), so server sites will also have client site requirements.

4.1.1. Allowed Outgoing Ports

Clients need to be able to make outgoing connections freely from ephemeral ports on hosts at the client site to all
ports at server sites.

4.1.2. Allowed Incoming Ports

As described in Section 3, "Job State Callbacks and Polling", the Globus Toolkit GRAM service uses callbacks to
communicate state changes to clients and, optionally, to stage files to/from the client. If connections are not allowed
back to the Globus Toolkit clients, the following restrictions will be in effect:

• You cannot do a job submission request and redirect the output back to the client. This means the globus-job-run
  command won't work. globus-job-submit will work, but you cannot use globus-job-get-output. globusrun with the
  -o option also will not work.

• Staging to or from the client will also not work, which precludes the -s and -w options.

• The client cannot be notified of state changes in the job, e.g. completion.

To allow these callbacks, client sites should allow incoming connection in the ephemeral port range. Client
sites wishing to restrict incoming connections in the ephemeral port range should select a port range for their
site. The size of this range should be approximately 10 ports per expected simultaneous user on a given host, though this may vary depending on the actual usage characteristics. Hosts on which clients run should have the GLOBUS_TCP_PORT_RANGE environment variable set for the users to reflect the site’s chosen range.

### 4.1.3. Network Address Translation (NAT)

Clients behind NATs will be restricted as described in Section 4.1.2, “Allowed Incoming Ports” unless the firewall and site hosts are configured to allow incoming connections.

This configuration involves:

- Select a separate portion of the ephemeral port range for each host at the site on which clients will be running (e.g. 45000-45099 for host A, 45100-45199 for host B, etc.).

- Configure the NAT to direct incoming connections in the port range for each host back to the appropriate host (e.g., configure 45000-45099 on the NAT to forward to 45000-45099 on host A).

- Configure the Globus Toolkit clients on each site host to use the selected port range for the host using the techniques described in Section 2.1, “If client is behind a firewall”.

- Configure Globus Toolkit clients to advertise the firewall as the hostname to use for callbacks from the server host. This is done using the GLOBUS_HOSTNAME environment variable. The client must also have the GLOBUS_HOSTNAME environment variable set to the hostname of the external side of the NAT firewall. This will cause the client software to advertise the firewall's hostname as the hostname to be used for callbacks causing connections from the server intended for it to go to the firewall (which redirects them to the client).

### 4.2. Server Site Firewall Requirements

This section describes firewall policy requirements at sites that host Grid services. Sites that host Grid services often host Grid clients, however the policy requirements described in this section are adequate for clients as well.

#### 4.2.1. Allowed Incoming Ports

A server site should allow incoming connections to the well-known Grid Service Ports as well as ephemeral ports. These ports are 22/tcp (for gsi-enabled openssh), 2119/tcp (for GRAM) and 2811/tcp for GridFTP.

A server not allowing incoming connections in the ephemeral port range will have the following restrictions:

- If port 2119/tcp is open, GRAM will allow jobs to be submitted, but further management of the jobs will not be possible.

- While it will be possible to make GridFTP control connections if port 2811/tcp is open, it will not possible to actually get or put files.

Server sites wishing to restrict incoming connections in the ephemeral port range should select a range of port numbers. The size of this range should be approximately 20 ports per expected simultaneous user on a given host, though this may vary depending on the actual usage characteristics. While it will take some operational experience to determine just how big this range needs to be, it is suggested that any major server site open a port range of at least a few hundred ports. Grid Services should configured as described in Section to reflect the site’s chosen range.

#### 4.2.2. Allowed Outgoing Ports

Server sites should allow outgoing connections freely from ephemeral ports at the server site to ephemeral ports at client sites as well as to Grid Service Ports at other sites.
4.2.3. Network Address Translation (NAT)

Grid services are not supported to work behind NAT firewalls because the security mechanisms employed by Globus require knowledge of the actual IP address of the host that is being connected to.

We do note there have been some successes in running GT services behind NAT firewalls.

4.3. Summary of Globus Toolkit Traffic

Table 3.1. Summary of Globus Toolkit Traffic

<table>
<thead>
<tr>
<th>Application</th>
<th>Network Ports</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRAM Gatekeeper(to start jobs)</td>
<td>To 2119/tcp on server from controllable ephemeral port on client</td>
<td>Connections back to client (controllable ephemeral port to controllable ephemeral port) required if executable or data staged from client or output from job sent back to client. Port 2119/tcp defined by IANA</td>
</tr>
<tr>
<td>GRAM Job-Manager</td>
<td>From controllable ephemeral port on client to controllable ephemeral port on server.</td>
<td>Port on server selected when original connection made by the client to the Gatekeeper and returned to the client in a URL. May result in connection back to client from ephemeral port on server to controllable ephemeral port on client.</td>
</tr>
<tr>
<td>GridFTP</td>
<td>From controllable ephemeral port on client to port 2811/tcp on server for control channel.</td>
<td>Port 2811/tcp defined by IANA.</td>
</tr>
<tr>
<td>GSI-Enabled SSH</td>
<td>From ephemeral port on client to port 22/tcp on server.</td>
<td>Same as standard SSH. Port 22/tcp defined by IANA.</td>
</tr>
<tr>
<td>MyProxy</td>
<td>From ephemeral port on client to port 7512/tcp on server.</td>
<td>Default. Can be modified by site.</td>
</tr>
</tbody>
</table>

4.4. Controlling The Ephemeral Port Range

Controllable ephemeral ports in the Globus Toolkit can be restricted to a given range. Setting the environment variable GLOBUS_TCP_PORT_RANGE can restrict ephemeral ports. The value of this variable should be formatted as min,max (a comma separated pair). This will cause the GT libraries (specifically GlobusIO) to select port numbers for controllable ports in that specified range.

```bash
% GLOBUS_TCP_PORT_RANGE=40000,40010
% export GLOBUS_TCP_PORT_RANGE
% globus-gass-server
https://globicus.lbl.gov:40000
^C
%
```

This environment variable is respected by both clients and servers that are started from within the environment in which it is set. There are better ways, however, to configure a globus-job-manager or a GridFTP server to restrict its port range.
• globus-job-manager has an option, -globus-tcp-port-range PORT_RANGE that acts in the same manner as the environment variable. It can be specified on the command line or in the configuration file. See the job manager documentation for all of its options.

• See the GridFTP documentation for information about using GridFTP with firewalls.
Chapter 4. Basic Setup for GT 5.2.3

The Quickstart Guide walks you through setting up basic services on multiple machines.
Chapter 5. Platform Notes

1. Platform Notes

1.1. Mac OS X 10.8 (Mountain Lion)

The libtool library is no longer distributed with MacOS in Mountain Lion. Install the latest libtool from the GNU libtool source mirror\(^1\) prior to building Globus from the source installer. To do so in a way that will work, you'll need to configure libtool with --program-prefix=g to cause the libtool script to be named glibtool to avoid conflicts with the OS X libtool program which provides different functionality than GNU libtool.

If you install libtool in a directory other than your Globus installation directory, you'll need to add it to your build environment, by adding CPPFLAGS="-LLIBTOOL-INSTALLDIR/include" and LDFLAGS="-LLIBTOOL-INSTALLDIR/lib" to your environment when compiling with the installer.

\(^1\) [http://ftpmirror.gnu.org/libtool/](http://ftpmirror.gnu.org/libtool/)
Chapter 6. Appendix

The Install Guide appendix can be found here.
Glossary

G

| Grid Security Infrastructure (GSI) | GSI stands for Grid Security Infrastructure and is used to describe the original infrastructure of GT security, which is comprised of SSL, PKI and proxy certificates. |