

# GT 4.2.0 Release Notes: GridFTP

## Table of Contents

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

<titleabbrev>Release Notes</titleabbrev>

## 1. Component Overview

GridFTP is a high-performance, secure, reliable data transfer protocol optimized for high-bandwidth wide-area networks. The GridFTP protocol is based on FTP, the highly-popular Internet file transfer protocol. We have selected a set of protocol features and extensions defined already in IETF RFCs and added a few additional features to meet requirements from current data grid projects.

## 2. Feature Summary

Features new in GT 4.2.0

- A new, complete reimplementation of the *server*.
- This new implementation will greatly ease new feature additions and modifications of the server (new commands, new data sources such as mass storage devices, etc.), maintainability, and resolves a licensing issue that was discovered.
- GridFTP over UDT
- SSH security for GridFTP control channel
- Running the GridFTP server with GFork GridFTP
- Multicasting / Network overlays
- Netlogger's bottleneck detection for GridFTP transfers<sup>1</sup>

Features that continue to be supported from previous versions

- GSI security: This is the PKI based, de facto standard security system used in Grid applications. Kerberos is also possible but is not supported and can be difficult to use due to divergence in the capabilities of GSI and Kerberos.

<sup>1</sup> <http://www.cedps.net/index.php/Gridftp-netlogger>

- Third-party transfers: Very common in Grid applications, this is where a *client* mediates a transfer between two servers (both likely at remote sites) rather than between the server and itself (called a *client/server transfer*).
- Cluster-to-cluster data movement or Striping: GridFTP can do coordinated data transfer by using multiple computer nodes at the source and destination.
- Partial file access: Regions of a file may be accessed by specifying an offset into the file and the length of the block desired.
- Reliability/restart: The receiving server periodically (the default is 5 seconds, but this can be changed) sends “restart markers” to the client. This marker is a messages specifying what bytes have been successfully written to the disk. If the transfer fails, the client may restart the transfer and provide these markers (or an aggregated equivalent marker), and the transfer will pick up where it left off. This can include “holes” in the file.
- Large file support: All file sizes, lengths, and offsets are 64 bits in length.
- Data channel reuse: Data channel can be held open and reused if the next transfer has the same source, destination, and credentials. This saves the time of connection establishment, authentication, and delegation. This can be a huge performance difference when moving lots of small files.
- Integrated instrumentation (Performance Markers).
- Logging/audit trail (Extensive Logging in the server).
- Parallel transfers (Multiple TCP streams between a pair of hosts).
- TCP Buffer size control (Protocol supports Manual and Automatic; Only Manual Implemented).
- Server-side computation (Extended Retrieve (ERET) / Extended Store (ESTO) commands).
- Based on Standards: RFC 959, RFC 2228, RFC 2389, IETF Draft MLST-16 , GGF GFD.020.

#### Other Supported Features

- On the client side we provide a scriptable tool called `globus-url-copy`. This tool can take advantage of all the GridFTP protocol features and can also do protocol translation between FTP, HTTP, HTTPS, and POSIX file IO on the client machine.
- We also provide a set of development libraries and APIs for developers wishing to add GridFTP functionality to their application.

#### Deprecated Features

- None

## 3. Summary of Changes in GridFTP

The following changes have occurred for GridFTP since the last stable release, 4.0.x:

- Added `GFork master plugin` to allow dynamic backend configuration.
- Added `-pp` option to `globus-url-copy` to allow pipelining of data transfers (ie, several transfers can be processed without waiting for each transfer to finish before starting the next one.)

## 4. Bug Fixes

- [Bug 1883](#)<sup>2</sup>: globus\_ftp\_control server does not decode ADAT properly.
- [Bug 1928](#)<sup>3</sup>: Strange problems: Busy wait and address in use.
- [Bug 2036](#)<sup>4</sup>: New gridftp, globus-gridftp-server, does not use globus\_gss\_assist\_map\_and\_authorize
- [Bug 2128](#)<sup>5</sup>: gridftp server checks ownership of hostcert after setuid
- [Bug 2129](#)<sup>6</sup>: Gridftp logging levels (is 7>10?)
- [Bug 2132](#)<sup>7</sup>: Server responds with "Valid credentials could not be found..."
- [Bug 2452](#)<sup>8</sup>: non-striped gridFTP not working w/ round robin
- [Bug 2463](#)<sup>9</sup>: striped server, small partial transfers hang
- [Bug 2464](#)<sup>10</sup>: striped server to non-striped server doesn't work

## 5. Known Bugs

The following problems are known to exist for GridFTP at the time of the 4.2.0 release:

- [Bug 2547](#)<sup>11</sup>: gridftp config detach option doesn't work
- GridFTP Server
  - There are some small memory leaks, though they should not grow much.
  - Threaded builds should work, but increased pollers may produce hangs (please report these).
  - Some error responses are unclear.
  - [Bug 2547](#)<sup>12</sup>: gridftp config detach option doesn't work
  - See the known issues section of source-trees/gridftp/server/src/TODO in the source installer for other issues.

### 5.1. Limitations

- [list limitations]

---

<sup>2</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=1883](http://bugzilla.globus.org/globus/show_bug.cgi?id=1883)

<sup>3</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=1928](http://bugzilla.globus.org/globus/show_bug.cgi?id=1928)

<sup>4</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2036](http://bugzilla.globus.org/globus/show_bug.cgi?id=2036)

<sup>5</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2128](http://bugzilla.globus.org/globus/show_bug.cgi?id=2128)

<sup>6</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2129](http://bugzilla.globus.org/globus/show_bug.cgi?id=2129)

<sup>7</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2132](http://bugzilla.globus.org/globus/show_bug.cgi?id=2132)

<sup>8</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2452](http://bugzilla.globus.org/globus/show_bug.cgi?id=2452)

<sup>9</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2463](http://bugzilla.globus.org/globus/show_bug.cgi?id=2463)

<sup>10</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2464](http://bugzilla.globus.org/globus/show_bug.cgi?id=2464)

<sup>11</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2547](http://bugzilla.globus.org/globus/show_bug.cgi?id=2547)

<sup>12</sup> [http://bugzilla.globus.org/globus/show\\_bug.cgi?id=2547](http://bugzilla.globus.org/globus/show_bug.cgi?id=2547)

## 6. Technology dependencies

GridFTP depends on the following GT components:

- Non-WS (General) Authentication & Authorization
- C Common Libraries
- XIO

GridFTP depends on the following 3rd party software:

- OpenSSL (version is included in release)

## 7. Tested platforms

Tested platforms for GridFTP

- i386 Linux
- ia64 Linux (TeraGrid)
- AIX 5.2
- Solaris 9
- PA-RISC HP/UX 11.11
- ia64 HP/UX 11.22
- Tru64 Unix
- Mac OS X

While the above list includes platforms on which we have tested GridFTP, it does not imply support for a specific platform. However, we are interested in hearing reports of success or bug reports on any platform.

## 8. 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 (GridFTP is not Java-based)

Schema changes since GT 4.0.x

- Not Applicable (GridFTP is not SOAP-based)

## 9. Associated Standards

Associated standards for GridFTP:

- [RFC 959 Base FTP protocol](#)<sup>13</sup>
- [RFC 2228 gssapi security extensions](#)<sup>14</sup> for FTP RFC 2389 FEAT, OPTS, etc.
- [extensions to FTP \(IETF FTP Working group draft\)](#)<sup>15</sup> for structured directory listings, SIZE, MDTM commands.
- [GFD.020 GridFTP extensions](#)<sup>16</sup>

## 10. For More Information

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

## Glossary

### C

**client** A process that sends commands and receives responses. Note that in GridFTP, the client may or may not take part in the actual movement of data.

### S

**server** A process that receives commands and sends responses to those commands. Since it is a server or service, and it receives commands, it must be listening on a port somewhere to receive the commands. Both FTP and GridFTP have IANA registered ports. For FTP it is port 21, for GridFTP it is port 2811. This is normally handled via `inetd` or `xinetd` on Unix variants. However, it is also possible to implement a daemon that listens on the specified port. This is described more fully in in the Architecture section of the GridFTP Developer's Guide.

---

<sup>13</sup> <http://rfc.net/rfc959.html>

<sup>14</sup> <http://rfc.net/rfc2228.html>

<sup>15</sup> <http://www.ietf.org/internet-drafts/draft-ietf-ftpext-mlst-16.txt>

<sup>16</sup> <http://www.ggf.org/documents/GWD-R/GFD-R.020.pdf>