

GT 5.0.0 GSI-OpenSSH: Developer's Guide

GT 5.0.0 GSI-OpenSSH: Developer's Guide

Introduction

This document provides information for GSI-OpenSSH developers.

The changes to OpenSSH¹ to add GSI support are limited, because we build on the existing GSSAPI support in OpenSSH for Kerberos. See the GSI-OpenSSH Internals² page for more details.

¹ <http://www.openssh.org/>

² <http://dev.globus.org/wiki/GSI-OpenSSH/Internals>

Table of Contents

1. Before you begin	1
1. Feature summary	1
2. Tested platforms	1
3. Backward compatibility summary	1
4. Technology dependencies	1
5. GSI-OpenSSH Security Considerations	2
2. Usage scenarios	3
3. Tutorials	4
4. Architecture and design overview	5
I. GSI-OpenSSH Commands	?
gsssh	7
gsscp	8
gssftp	9
5. Configuring	10
6. Environment variable interface	11
1. Environmental variables for GSI-OpenSSH	11
7. Troubleshooting	12
1. Common GT Errors	12
2. Errors	13
3. Additional GSI-OpenSSH Troubleshooting	13
8. Related Documentation	14
Glossary	15

List of Tables

7.1. GSI-OpenSSH Errors	13
-------------------------------	----

Chapter 1. Before you begin

1. Feature summary

Features new in GT 5.0.0

- None.

Other Supported Features

- The **gsissh** command provides a secure remote login service with forwarding of X.509 *proxy credentials*.
- The **gsiscp** and **gsisftp** commands provide a secure file transfer service authenticated with X.509 proxy credentials, mimicking the **rcp/scp** and **ftp/sftp** commands.
- All standard OpenSSH features are supported, excluding Kerberos authentication. Kerberos authentication is *not* compatible with GSI-enabled OpenSSH.
- The GSI-OpenSSH server can replace the standard system SSH server in typical environments.
- If no username is given on the command-line, GSI-OpenSSH automatically determines the username that corresponds to the X.509 proxy *certificate subject* in the server's `grid-mapfile`.

Deprecated Features

- None

2. Tested platforms

Tested Platforms for GSI-OpenSSH

- Mac OS X 10.5
- x86/x86_64 GNU/Linux
- PPC AIX 5.3
- Sun4u Solaris 5.10

3. Backward compatibility summary

GSI-OpenSSH is backward compatible.

4. Technology dependencies

GSI-OpenSSH depends on the following GT components:

- GSLC

GSI-OpenSSH depends on the following 3rd party software:

- [OpenSSH](#)¹

5. GSI-OpenSSH Security Considerations

GSI-OpenSSH is a modified version of [OpenSSH](#)² and includes full OpenSSH functionality. For more information on OpenSSH security, see the [OpenSSH Security](#)³ page.

¹ <http://www.openssh.org/>

² <http://www.openssh.org/>

³ <http://www.openssh.org/security.html>

Chapter 2. Usage scenarios

The GSI-OpenSSH interface is through command-line tools only.

Chapter 3. Tutorials

There are no tutorials available at this time

Chapter 4. Architecture and design overview

See the [GSI-OpenSSH Internals](http://dev.globus.org/wiki/GSI-OpenSSH/Internals)¹ page.

¹ <http://dev.globus.org/wiki/GSI-OpenSSH/Internals>

GSI-OpenSSH Commands

The `gsissh(1)`, `gsiscp(1)`, and `gsisftp(1)` commands provide the same interfaces as the standard OpenSSH `ssh`, `scp`, and `sftp` commands, respectively, with the added ability to perform X.509 *proxy credential* authentication and delegation.

Name

`gsissh` -- Secure remote login

`gsissh`

Tool description

Use the `gsissh` command to securely login to a remote machine.

Command syntax

`gsissh` [-l login_name] hostname | user@hostname [command]

Name

`gsiscp` -- Secure remote file copy

`gsiscp`

Tool description

Use the `gsiscp` command to securely copy files to or from a remote machine.

Command syntax

`gsiscp [-P port] [[user@]host1:]file1 [...] [[user@]host2:]destfile`

Name

`gsisftp` -- Secure file transfer

`gsisftp`

Tool description

The *gsisftp* command provides an interactive interface for transferring files to and from remote machines.

Command syntax

`gsisftp` [[user@]host[:dir[/]]]

Chapter 5. Configuring

The GSI-enabled OpenSSH software is installed with a default set of configuration files, described below. You may want to modify the `ssh_config` file before using the clients and the `sshd_config` file before using the server.

If the GSI-enabled OpenSSH install script finds existing SSH key pairs, it will create symbolic links to them rather than generating new key pairs. The SSH key pairs are not required for GSI authentication. However, if you wish to support other SSH authentication methods, make sure the `sshd` (running as root) can read the key pair files (i.e., beware of NFS mounts with `root_squash`). If running multiple `sshd`s on a system, we recommend configuring them so they all use the same key pairs (i.e., use symbolic links) to avoid client-side confusion.

- `$GLOBUS_LOCATION/etc/ssh/moduli`
`moduli` is a crypto parameter for generating keys.
- `$GLOBUS_LOCATION/etc/ssh/ssh_config`
`ssh_config` contains options that are read by `ssh`, `scp`, and `sftp` at run-time. The installed version is the default provided by OpenSSH, with GSI authentication and `X11Forwarding` enabled. You may need to customize this file for compatibility with your system SSH installation (i.e., compare it with `/etc/ssh/ssh_config`).
- `$GLOBUS_LOCATION/etc/ssh/ssh_host_key[.pub]`
Your system's RSA public-/private-key pair for SSH protocol 1 communications.
- `$GLOBUS_LOCATION/etc/ssh/ssh_host_dsa[.pub]`
Your system's DSA public-/private-key pair for SSH protocol 2 communications.
- `$GLOBUS_LOCATION/etc/ssh/ssh_host_rsa[.pub]`
Your system's RSA public-/private-key pair for SSH protocol 2 communications.
- `$GLOBUS_LOCATION/etc/ssh/ssh_prng_cmds`
`ssh_prng_cmds` contains paths to a number of files that `ssh-keygen` may need to use if your system does not have a built-in entropy pool (like `/dev/random`).
- `$GLOBUS_LOCATION/etc/ssh/sshd_config`
`sshd_config` contains options that are read by `sshd` when it starts up. The installed version is the default provided by OpenSSH, with `X11Forwarding` enabled. You may need to customize this file for compatibility with your system SSH installation (i.e., compare it with `/etc/ssh/sshd_config`). For example, to enable PAM authentication, you may need to set "UsePAM yes" in this file.

Chapter 6. Environment variable interface

1. Environmental variables for GSI-OpenSSH

The GSI-enabled OpenSSH needs to be able to find certain files and directories in order to properly function.

The items that OpenSSH needs to be able to locate, their default location and the environment variable to override the default location are:

- *Host key*
 - Default location: /etc/grid-security/hostkey.pem
 - Override with X509_USER_KEY environment variable
- *Host certificate*
 - Default location: /etc/grid-security/hostcert.pem
 - Override with X509_USER_CERT environment variable
- *Grid map file*
 - Default location: /etc/grid-security/grid-mapfile
 - Override with GRIDMAP environment variable
- *Certificate directory*
 - Default location: /etc/grid-security/certificates
 - Override with X509_CERT_DIR environment variable

Chapter 7. Troubleshooting

1. Common GT Errors

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

2. Errors

Table 7.1. GSI-OpenSSH Errors

Error Code	Definition	Possible Solutions
GSS-API error Failure acquiring GSSAPI credentials: GSS_S_CREDENTIALS_EXPIRED	This means that your proxy certificate has expired.	Run <code>grid-proxy-init</code> to acquire a new proxy certificate, then run <code>gsissh</code> again.
...no proxy credentials...	Failing to run <code>grid-proxy-init</code> to create a user proxy with which to connect will result in the client notifying you that no local credentials exist. Any attempt to authenticate using GSI will fail in this case.	Verify that your GSI proxy has been properly initialized via <code>grid-proxy-info</code> . If you need to initialize the proxy, use the command <code>grid-proxy-init</code> .
...bad file system permissions on private key; key must only be readable by the user...	The host key that the SSH server is using for GSI authentication must only be readable by the user which owns it. Any other permissions will cause this error.	Make sure that the host key's UNIX permissions are mode 400 (that is, it should only have mode readable for the user that owns the file, and no other mode bits should be set).
...gssapi received empty username; failed to set username from gssapi context; Failed external-keyx for <user> from <host> <port>...	If the server was passed an "implicit username" (i.e. requested to map the incoming connection to a username based on some contextual clues such as the certificate's subject), and no entry exists in the <code>grid-mapfile</code> for the incoming connection's certificate subject, the server should output a clue that states it is unable to set the username against which to authenticate.	Add an entry for the user to the Section 1.2, "Gridmap file" .
...INTERNAL ERROR: authenticated invalid user xxx...	If the subject name given in the system's <code>grid-mapfile</code> points to a non-existent user, the server will give an internal error which is best caught when it is running in debugging mode.	Add a new account to the system matching the username pointed at by the user's subject in the <code>grid-mapfile</code> .
...gssapi received empty username; no suitable client data; failed to set username from gssapi context; Failed external-keyx for <user> from <host> <port>...	Should the user attempt to connect without first creating a proxy certificate, or if the user is connecting via a SSH client that does not support GSI authentication, the server will note that no GSSAPI data was sent to it. Verify that the client is able to connect through another GSI service (such as the gatekeeper) to make sure that the user's proxy has been created correctly.	Verify that you are using a GSI-enabled SSH client and that your GSI proxy has been properly initialized via <code>grid-proxy-info</code> . If you need to initialize this proxy, use the command <code>grid-proxy-init</code> .

3. Additional GSI-OpenSSH Troubleshooting

Please see the [GSI-OpenSSH Troubleshooting Page](#)¹ at NCSA.

¹ http://grid.ncsa.illinois.edu/ssh/ts_common.html

Chapter 8. Related Documentation

Please see the [GSI-OpenSSH Home Page](http://grid.ncsa.uiuc.edu/ssh/)¹ at NCSA for more information.

¹ <http://grid.ncsa.uiuc.edu/ssh/>

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.

G

grid map file

A file containing entries mapping certificate subjects to local user names. This file can also serve as a access control list for GSI enabled services and is typically found in `/etc/grid-security/grid-mapfile`. For more information see the Gridmap section [here](#).

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

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.