GT 5.0.0 Component Guide to Public Interfaces: GSI-OpenSSH
GT 5.0.0 Component Guide to Public Interfaces: GSI-OpenSSH
# Table of Contents

1. GSI-OpenSSH Commands ................................................................. ?
   gsissh ........................................................................................................ 6
   gsiscp ........................................................................................................ 7
   gsisftp ................................................................................................... 8
   1. Configuring ......................................................................................... 1
   2. Environment variable interface ............................................................. 2
      1. Environmental variables for GSI-OpenSSH ......................................... 2
   A. Errors ..................................................................................................... 3
   Glossary ..................................................................................................... 4
List of Tables

A.1. GSI-OpenSSH Errors ..................................................................................................................... 3
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 gsish command to securely login to a remote machine.

Command syntax

gsish [-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

```bash
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/]][command]`
Chapter 1. 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 sshds 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
  ssdh_config contains options that are read by ssdh 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 2. Environment variable interface

1. Environmental variables for GSI-OpenSSH

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

The items that OpenSSHD 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
# Appendix A. Errors

## Table A.1. GSI-OpenSSH Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Definition</th>
<th>Possible Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSS-API error Failure acquiring GSSAPI credentials: GSS_S_CREDENTIALS_EXPIRED</td>
<td>This means that your proxy certificate has expired.</td>
<td>Run <code>grid-proxy-init</code> to acquire a new proxy certificate, then run <code>gsish</code> again.</td>
</tr>
<tr>
<td>...no proxy credentials...</td>
<td>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.</td>
<td>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>.</td>
</tr>
<tr>
<td>...bad file system permissions on private key; key must only be readable by the user...</td>
<td>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.</td>
<td>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).</td>
</tr>
<tr>
<td>...gssapi received empty username; failed to set username from gssapi context; Failed external-keyx for &lt;user&gt; from &lt;host&gt; &lt;port&gt;...</td>
<td>If the server was passed an &quot;implicit username&quot; (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 grid-mapfile 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.</td>
<td>Add an entry for the user to the Section 1.2, “Gridmap file”.</td>
</tr>
<tr>
<td>...INTERNAL ERROR: authenticated invalid user xxx...</td>
<td>If the subject name given in the system’s grid-mapfile points to a non-existent user, the server will give an internal error which is best caught when it is running in debugging mode.</td>
<td>Add a new account to the system matching the username pointed at by the user’s subject in the grid-mapfile.</td>
</tr>
<tr>
<td>...gssapi received empty username; no suitable client data; failed to set username from gssapi context; Failed external-keyx for &lt;user&gt; from &lt;host&gt; &lt;port&gt;...</td>
<td>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.</td>
<td>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>.</td>
</tr>
</tbody>
</table>
## Glossary

### G

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid map file</td>
<td>A file containing entries mapping certificate subjects to local user names. This file can also serve as an access control list for GSI enabled services and is typically found in <code>/etc/grid-security/grid-mapfile</code>. For more information see the Gridmap section <a href="#">here</a>.</td>
</tr>
</tbody>
</table>

### H

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host certificate</td>
<td>An EEC belonging to a host. When using GSI this certificate is typically stored in <code>/etc/grid-security/hostcert.pem</code>. For more information on possible host certificate locations see the GSI C Developer’s Guide.</td>
</tr>
</tbody>
</table>

### P

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>proxy credentials</td>
<td>The combination of a proxy certificate and its corresponding private key. GSI typically stores proxy credentials in <code>/tmp/x509up_u&lt;uid&gt;</code> , where <code>&lt;uid&gt;</code> is the user id of the proxy owner.</td>
</tr>
</tbody>
</table>