Globus hand-outs

Security

Preliminary steps

1. Logon to ced-ui0.datagrid.cnr.it

2. Create the .globus directory under your home directory:

   ```
   mkdir .globus
   ```

3. Copy your p12 file containing your certificate and private key:

   ```
   cp /home/Certificati/<your user name>/<your user
   name>.p12 /home/<your user name>/.globus
   ```

4. Change the current working directory to .globus:

   ```
   cd .globus
   ```

5. Extract your private key from the p12 file issuing the following command:

   ```
   openssl pkcs12 -nocerts -in <your user name>.p12 -
   out ./userkey.pem
   ```

6. Extract your certificate from the p12 file issuing the following command:

   ```
   openssl pkcs12 -clcerts -nokeys -in <your user
   name>.p12 -out ./usercert.pem
   ```

7. Change <your user name>.p12 file access permissions issuing the following command:

   ```
   chmod 0600 <your user name>.p12
   ```

8. Change your private key file access permissions issuing the following command:

   ```
   chmod 0600 userkey.pem
   ```

9. Change your certificate file access permissions issuing the following command:

   ```
   chmod 0644 usercert.pem
   ```

Create and manage proxy credentials

1. Create your proxy credentials issuing the following command:

   ```
   grid-proxy-init
   ```

   You will be prompted for the PEM passphrase. If the password is correct the proxy credentials will be created. The default life-time of your proxy is 12 hours.
   A proxy is a certificate signed with your private key and has generally a life-time of few hours.

2. Verify your proxy properties:

   ```
   grid-proxy-info -all
   ```

   You will get the following kind of output:

   ```
   subject  : /C=IT/O=CNR/OU=CED/L=Rome/CN=John
   Doe/CN=proxy
   issuer   : /C=IT/O=CNR/OU=CED/L=Rome/CN=John Doe
   type     : full
   strength : 512 bits
   path     : /tmp/x509up_u1592
   timeleft : 11:59:51
   ```

   The "path" attribute provides the absolute path of your proxy certificate file.

3. Check your proxy properties:

   ```
   openssl x509 -in /tmp/x509up_u1592 -text | more
   ```

4. Destroy your proxy credentials:

   ```
   grid-proxy-destroy
   ```

   The above command can be useful when you finish your working session before your proxy expires.

5. Create a proxy with a lifetime different from the default value (default is 12 hours):

   ```
   grid-proxy-init  -valid 24:00
   ```

   You will be prompted for the PEM passphrase. If everything is ok you will get a proxy with 24 hours of validity. You can check this by typing the grid-proxy-info command.

Information systems

Exercise 1

"grid-info-search" sends one or more queries to the MDS and displays the result on stdout. The query QUERY is a RFC-1558 compliant LDAP search filter.

Example:
Query the Top MDS with "globus-info-search" in order to retrieve information about the available resources.

1. grid-info-search –h ced-ii0.datagrid.cnr.it –p 2135 –b "mds-vo-name=local,o=grid" –x | more

Exercise 2

You can perform the same LDAP queries with the standard ldapsearch tool.
Example:
Query the Top MDS with "ldapsearch" in order to retrieve information about the available resources.

2. ldapsearch –h ced-ii0.datagrid.cnr.it –p 2135 –b "mds-vo-name=local,o=grid" –x | more

Job submission

Exercise 1

Run simple commands on a remote machine using the globus-job-run command.

1. Verify remote hostname:

   ```
   globus-job-run ced-ce0 /bin/hostname
   ```

2. Check Unix user:

   ```
   globus-job-run ced-ce0 /usr/bin/whoami
   ```

3. Check current working directory:

   ```
   globus-job-run ced-ce0 /bin/pwd
   ```

4. List remote directory content:

   ```
   globus-job-run ced-ce0 /bin/ls
   ```

Exercise 2

"globus-job-submit" allows the user to submit a job to a remote resource, using the same command-line syntax as globus-job-run does. The program translates the program arguments into an RSL (Resource Specification Language) request and uses "globusrun" to submit the job in batch submission mode, that is, after the job is submitted, no connection exists between the local host and the remote host.

"globus-job-submit" prints out a job id after successful submission of the job to a remote resource. Unless stdout/stderr are specified, the program output will be buffered at the remote site and can be retrieved at any time with globus-job-get-output.

If the output is buffered, the user must make sure to run globus-job-clean when the program output is no longer needed.

Example:
List remote directory content:

```
globus-job-submit ced-ce0.datagrid.cnr.it /bin/ls
```

If the submission is successful you will get a unique job identifier.

You can now check the status of your job:

```
globus-job-status <job-ID>
```

When the status is done, you can retrieve the standard output issuing the command

```
globus-job-get-output <job-ID>
```

and the standard error issuing the command:

```
globus-job-get-output -err <job-ID>
```

You can now remove the buffered output and error messages on the remote machine with the command:

```
globus-job-clean <job-ID>
```

Exercise 3

"globus job-submit" allows to redirect standard output and standard error to user-defined files. In this case the output and error messages are not buffered with the standard globus mechanism, hence when the status is "DONE" it is not possible to use the globus-job-get-output command. It is anyway possible to retrieve the user-defined files with gridftp commands or via normal or secure ftp.

Example:
```
globus-job-submit –stdout list.txt ced-ce0 /bin/ls
```

If the submission is successful you will get a unique job identifier.

You can now check the status of your job:

```
globus-job-status <job-ID>
```

If the status is "DONE" you can check the output on the remote machine using an appropriate globus-job-run command line:

```
globus-job-run ced-ce0 /bin/cat list.txt
```

You can now remove the buffered error messages on the remote machine with the command:

```
globus-job-clean <job-ID>
```

Data management

Exercise 1

"globus-url-copy" copies a file specified by a "source URL" to a location specified by "destination URL", using the GASS[1] transfer API. All protocols supported by GASS (local file, http, https, ...) are supported.

Example:

Create a file in your local home directory and copy it to a remote machine.

1. Create the file issuing the following command:

   touch <your user name>

2. Copy the file from the local machine to a remote machine:

   globus-url-copy file:///home/<your user name>/<your user name> gsiftp://ced-ce0.datagrid.cnr.it/tmp/<your user name>

3. Verify that the file has been copied:

   globus-job-run ced-ce0 /bin/ls /tmp/<your user name>

Exercise 2

Copy the file from a remote machine to another remote machine.

1. Copy the file:

   globus-url-copy gsiftp://ced-ce0.datagrid.cnr.it/tmp/<your user name> gsiftp://ced-se0.datagrid.cnr.it/tmp/<your user name>

2. Verify that the file has been copied:

   globus-job-run ced-se0 /bin/ls /tmp/<your user name>

---

[1] Globus Access to Secondary Storage