



Enabling Grids for E-science

# Certificates Usage and Simple Job Submission

*GILDA Tutors*

*Sofia, 22 March 2007*

*Minor changes and adaptation for Bulgarian users by  
Stanislav Spasov - [spasov@acad.bg](mailto:spasov@acad.bg)*

[www.eu-egee.org](http://www.eu-egee.org)



- **VOMS proxy usage**
- **MyProxy Usage**
- **Simple Job Submission**

## How to access the GILDA User Interface

(valid only till the end of March!

And firewalled !)

**Login** : **sofiaXX@glite-tutor.ct.infn.it**  
where **XX=01,..18**

**Passwd** : **GridSOFXX** **XX=01,..,18**

**PEM PASSPHRASE : SOFIA**

- **.globus directory contains your personal public / private keys**

```
[glite-tutor] /home/sofia21 > ls -l .globus
total 8
-rw-r--r--  1 giorgio  users  1613 Oct  4 19:30 usercert.pem
-r-----  1 giorgio  users  1914 Oct  4 19:30 userkey.pem
```

- **Pay attention to permissions**
  - `userkey.pem` contains your private key, and must be readable just by yourself (400)
  - `usercert.pem` contains your public key, which should be readable also from outside (644)

- **Main options**

**-voms** <vo-name>:[command]>

- **command** syntax is :/<voname>/group for group specify (default none)
- **command** syntax is :/<voname>/Role=<role name> for Role choice (default none)

```
voms-proxy-init --voms gilda:/gilda
voms-proxy-init --voms gilda:/gilda
```

-valid x:y, create a proxy valid for x hours and y minutes

-vomslife x, create a proxy with AC valid for x hours (max 24 h)

-cert <certfile> Non-standard location of user certificate

-key <keyfile> Non-standard location of user key

-out <proxyfile> Non-standard location of new proxy cert

-userconf <file> Non-standard location for user-defined voms server addresses

- **Default** location for voms server address file is /opt/glite/etc/vomses or \$HOME/.glite/vomses.

**Syntax** : "vo-nickname" "voms server FQDN" "port" "voms server \ certificate subject" "vo name"

Parameters for vomses are usually provided by VOs manager

**Exercise 1** : **create** a voms proxy :

```
voms-proxy-init --voms gilda
```

**then** verify obtained credentials with

```
voms-proxy-info -all
```

- **voms-proxy-info**
  - Main options :
    - all prints all proxy options
    - file specifies a different location of proxy file

```

/home/giorgio > voms-proxy-info -all
subject      : /C=IT/O=GILDA/OU=Personal
              Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it/CN=proxy
issuer       : /C=IT/O=GILDA/OU=Personal
              Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it
identity     : /C=IT/O=GILDA/OU=Personal
              Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it
type         : proxy
strength     : 512 bits
path        : /tmp/x509up_u513
timeleft    : 11:59:52
=== VO gilda extension information ===
VO          : gilda
subject     : /C=IT/O=GILDA/OU=Personal
              Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it
issuer      : /C=IT/O=GILDA/OU=Host/L=INFN
              Catania/CN=voms.ct.infn.it/Email=emidio.giorgio@ct.infn.it
attribute   : /gilda/tutors/Role=NULL/Capability=NULL
attribute   : /gilda/Role=NULL/Capability=NULL
timeleft    : 11:59:45
    
```

Standard globus attributes

Voms extensions

- **myproxy server:**
  - myproxy-init
    - Allows to create and store a long term proxy certificate:
  - myproxy-info
    - Get information about stored long living proxy
  - myproxy-get-delegation
    - Get a new proxy from the MyProxy server
  - myproxy-destroy
  - Check out them with myproxy-xxx --help option
- **A dedicated service on the RB can renew automatically the proxy**
  - contacting the myproxy server



```
[giorgio@glite-tutor:~]$ myproxy-init -s grid001.ct.infn.it
Your identity: /C=IT/O=GILDA/OU=Personal
Certificate/L=INFN/CN=Emidio
Giorgio/Email=emidio.giorgio@ct.infn.it
Enter GRID pass phrase for this identity:
Creating proxy .....
Done
Proxy Verify OK
Your proxy is valid until: Sun Jun 19 21:18:27 2005
Enter MyProxy pass phrase:
Verifying password - Enter MyProxy pass phrase:
A proxy valid for 168 hours (7.0 days) for user giorgio now exists
on grid001.ct.infn.it.
```

- **Principal options**
- **-c** hours specifies lifetime of stored credentials
- **-t** hours specifies the maximum lifetime of credentials when retrieved
- **-s** <hostname> specifies the myproxy server where to store credentials
- **-d** stores credential with the distinguished name in proxy, instead of user name (mandatory for some data management services and proxy renewal)
- For proxy renewal it's also mandatory **-n** (no passphrase). You've to specify also subject of principals that can renew a delegation (**-R** subject, or **-A** for any principal)

- Useful to retrieve info on stored credentials
- Need local credentials to be performed
- If credentials have been **initialized** with **-d** switch, you have also to specify it there

```
[giorgio@glite-tutor:~]$ myproxy-info -s grid001.ct.infn.it
username: giorgio
owner: /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio
Giorgio/Email=emidio.giorgio@ct.infn.it
timeleft: 167:55:34 (7.0 days)
```

- This command is used to retrieve a delegation from a long lived proxy stored on myproxy server
- It is independent by the machine ! You don't need to have your certificate on board
- If credentials have been initialized with `-d` switch, you have to specify it also in myproxy-get-delegation request

```
[giorgio@glite-tutor:~]$ myproxy-get-delegation \  
-s grid001.ct.infn.it
```

```
Enter MyProxy pass phrase:
```

```
A proxy has been received for user giorgio in /tmp/x509up_u513
```

- Delete, if existing, the long lived credentials on the specified myproxy server

```
[glite-tutor] /home/giorgio > myproxy-destroy \  
-s grid001.ct.infn.it
```

```
Default MyProxy credential for user giorgio was successfully  
removed.
```

- **Exercise 2 (optional)**
  - Create a myproxy on the server `grid001.ct.infn.it` , with lifetime set to 96
  - Visualize information on that
  - Create a myproxy with `-d` option
  - Which differences you note ?
  - Destroy both

- myproxy doesn't support natively VOMS
- To allow storing of voms ext., myproxy client has been modified,
- The faculty of choosing VO and group/roles has been added, while the previous options have all been kept

```
myproxy-init --voms gilda
```

- Proxies then retrieved with `myproxy-get-delegation` will have the requested voms extension but...
- There's a limitation, due to voms extensions lifetime: typically it's limited, and it's not renewed when performing `myproxy-get-delegation`



Studying solutions to extend voms extension renew in get-delegation

- The “modified” client is available only on GILDA UI's
- Will be largely deployed when the above issues will be solved

```
[ui-test] /home/giorgio > myproxy-get-delegation -s
grid001.ct.infn.it
Enter MyProxy pass phrase:
A proxy has been received for user giorgio in /tmp/x509up_u500
[ui-test] /home/giorgio > voms-proxy-info -all
subject      : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it/CN=proxy/CN=proxy/CN=prox
              y
issuer       : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it/CN=proxy/CN=proxy
identity    : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it/CN=proxy/CN=proxy
type        : unknown
strength    : 512 bits
path        : /tmp/x509up_u500
timeleft    : 12:00:09
=== VO gilda extension information ===
VO          : gilda
subject     : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio
              Giorgio/Email=emidio.giorgio@ct.infn.it
issuer      : /C=IT/O=GILDA/OU=Host/L=INFN
              Catania/CN=voms.ct.infn.it/Email=emidio.giorgio@ct.infn.it
attribute   : /gilda/Role=NULL/Capability=NULL
attribute   : /gilda/tutors/Role=NULL/Capability=NULL
timeleft    : 23:59:57
```

Voms extension  
lifetime

## Workload Management System

- In the glite middleware a user can submit and cancel jobs, query their status, and retrieve their output. These tasks go under the name of *Workload Management*.
- There are two different User Interfaces to accomplish these tasks. One is the Command Line Interface and the other is the Graphical User Interface.



## Exercise 3

### Simple Job Submit

- Create a simple job
- Submit it to the GRID for execution
- Check the status of the job
- Retrieve output of the job

Create or modify **hello.jdl** with following contents :

```
Executable = "hello.sh";  
StdError = "test.err";  
StdOutput = "test.out";  
InputSandbox = {"hello.sh"};  
OutputSandbox = {"test.err", "test.out"};
```

---

Create or modify **hello.sh** with following contents :

```
#!/bin/sh  
date  
hostname  
echo "*** Hello World***"  
ls -al  
echo "End."
```

---

Just to speed up the exercise—the files can be obtained like this:

```
cd ~
```

```
mkdir test
```

```
cd test
```

```
wget http://people.acad.bg/~spasov/hello.sh
```

```
wget http://people.acad.bg/~spasov/hello.jdl
```

---

Make hello.sh script executable with

```
chmod +x hello.sh
```

Before actually submitting the job, it is possible to see which CEs are eligible to run a job specified by a given JDL file using the command

**glite-job-list-match hello.jdl**

Selected Virtual Organisation name (from proxy certificate extension): gilda  
Connecting to host glite-rb2.ct.infn.it, port 7772

\*\*\*\*\*

COMPUTING ELEMENT IDs LIST

The following CE(s) matching your job requirements have been found:

```

*CEId*
ce-nano-37.to.infn.it:2119/jobmanager-lcgpbs-infinite
ce-nano-37.to.infn.it:2119/jobmanager-lcgpbs-long
ce-nano-37.to.infn.it:2119/jobmanager-lcgpbs-short
grid0021.datagrid.cnr.it:2119/jobmanager-lcgpbs-infinite
grid0021.datagrid.cnr.it:2119/jobmanager-lcgpbs-long
grid0021.datagrid.cnr.it:2119/jobmanager-lcgpbs-short
grid010.ct.infn.it:2119/jobmanager-lcgpbs-infinite
grid010.ct.infn.it:2119/jobmanager-lcgpbs-long
grid010.ct.infn.it:2119/jobmanager-lcgpbs-short
grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-infinite
grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-long
grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-short
iceage-ce-01.ct.infn.it:2119/jobmanager-lcgpbs-infinite
iceage-ce-01.ct.infn.it:2119/jobmanager-lcgpbs-long
iceage-ce-01.ct.infn.it:2119/jobmanager-lcgpbs-short
tc06.nesc.ed.ac.uk:2119/jobmanager-lcgpbs-infinite
tc06.nesc.ed.ac.uk:2119/jobmanager-lcgpbs-long
tc06.nesc.ed.ac.uk:2119/jobmanager-lcgpbs-short
gn0.hpcc.sztaki.hu:2119/jobmanager-lcgpbs-gilda
trigrd-ce00.unime.it:2119/jobmanager-lcgpbs-infinite
trigrd-ce00.unime.it:2119/jobmanager-lcgpbs-long
trigrd-ce00.unime.it:2119/jobmanager-lcgpbs-short

```

- **Job Submission**

- Perform the job submission to the Grid.

```
$ glite-job-submit [options] <jdl_file>
```

- where <jdl file> is a file containing the job description, usually with extension .jdl.

**--vo** <vo name> : perform submission with a different VO than the UI default one.

**--output, -o** <output file> save jobId on a file.

**--resource, -r** <resource value> specify the resource for execution.

**--nomsgi** neither message nor errors on the stdout will be displayed.

```
$ glite-job-submit -o test.id hello.jdl
```

If the request has been correctly submitted this is the typical output that you can get:

```
Selected Virtual Organisation name (from proxy certificate extension): gilda  
Connecting to host glite-rb2.ct.infn.it, port 7772  
Logging to host glite-rb2.ct.infn.it, port 9002
```

```
===== glite-job-submit Success =====
```

```
The job has been successfully submitted to the Network Server.  
Use glite-job-status command to check job current status. Your job identifier is:
```

```
- https://glite-rb2.ct.infn.it:9000/bNIXtU0b8hus1VUD3TdqzQ
```

```
The job identifier has been saved in the following file:  
/home/sofia21/test/test.id
```

```
=====
```

In case of failure, an error message will be displayed instead, and an exit status different from zero will be returned.

If the command returns the following error message:

```
**** Error: API_NATIVE_ERROR ****
```

```
Error while calling the "NSClient::multi" native api
```

```
AuthenticationException: Failed to establish security context...
```

```
**** Error: UI_NO_NS_CONTACT ****
```

```
Unable to contact any Network Server
```

it means that there are authentication problems between the UI and the *Network Server* (check your proxy or contact the site administrator).

After a job is submitted, it is possible to see its status using the `glite-job-status` command.

```
$ glite-job-status -i test.id
```

```
*****
```

## BOOKKEEPING INFORMATION:

```
Status info for the Job : https://glite-rb2.ct.infn.it:9000/bNIXtU0b8hus1VUD3TdqzQ
```

```
Current Status:    Scheduled
```

```
Status Reason:    Job successfully submitted to Globus
```

```
Destination:      grid0021.datagrid.cnr.it:2119/jobmanager-lcgpbs-short
```

```
Submitted:        Thu Mar 22 08:11:31 2007 CET
```

```
*****
```



More information on **glite-job-status**

The option **-i <file path>** can be used to specify a file with a list of job identifiers (saved previously with the **-o** option of **glite-job-submit**).

**glite-job-status -i jobs.list**

---

```
1 : https://lxshare0234.cern.ch:9000/UPBqN2s2ycxt1TnuU3kzEw
2 : https://lxshare0234.cern.ch:9000/8S6IwPW33AhyxhkSv8Nt9A
3 : https://lxshare0234.cern.ch:9000/E9R0Yl4J7qgsq7FYTnhmsA
4 : https://lxshare0234.cern.ch:9000/Tt80pBn17AFPJyUSN9Qb7Q
a : all
q : quit
```

---

Choose one or more `edg_jobId(s)` in the list - [1-4]all:

If the **--all** option is used instead, the status of all the jobs owned by the user submitting the command is retrieved.

Even More information on **glite-job-status** :

The **--status <state>** (-s) option makes the command retrieve only the jobs that are in the specified state, and the

**--exclude <state>** (-e) option makes it retrieve jobs that are not in the specified state.

This two lasts options are mutually exclusive, although they can be used with **--from** and **--to**.

Example: All jobs of the user that are in the state **DONE** or **RUNNING** are retrieved.

```
glite-job-status --all -s Done -s Running
```

A job can be canceled before it ends using the command `glite-job-cancel`.

`glite-job-cancel -i test.id`

```
Are you sure you want to remove specified job(s)? [y/n]n :y
```

```
===== glite-job-cancel Success=====
```

```
The cancellation request has been successfully submitted for the following  
job(s)
```

```
- https://lxshare0234.cern.ch:9000/dAE162is6EStca0VqhVkog
```

```
=====
```

After the job has finished (it reaches the DONE status), its output can be copied to the UI

```
$glite-job-output --dir ./ -i test.id
```

Retrieving files from host: glite-rb2.ct.infn.it ( for https://glite-rb2.ct.infn.it:9000/bNIXtU0b8hus1VUD3TdqzQ )

\*\*\*\*\*

## JOB GET OUTPUT OUTCOME

Output sandbox files for the job:

- https://glite-rb2.ct.infn.it:9000/bNIXtU0b8hus1VUD3TdqzQ  
have been successfully retrieved and stored in the directory:  
/home/sofia21/test/sofia21\_bNIXtU0b8hus1VUD3TdqzQ

\*\*\*\*\*

By default, the output is stored under /tmp, but it is possible to specify in which directory to save the output using the **--dir <path name>** option.

## Check the output files

```
ls -la  
cd sofia*  
cat test.out
```

If the proxy is not necessary anymore  
consider destroying it !

```
voms-proxy-destroy
```

