

# Table of Contents

<b>Administrator's TroubleShooting</b> .....	1
Problem: Python commands issue SSLError .....	1
Problem: SSH DSA keys slowdown/failures .....	2
Problem: Authentication fail .....	2
Problem: PBS jobs history disabled .....	3
Problem: The user cannot browse server or submit jobs on cluster .....	6
Problem: The wiki is slow and sometimes does not display images .....	7
Problem: RVEC notification do not work .....	7
Problem: Error during fetch in .....	8
Problem: Application stay blocked, having Job status in SUBMITTING .....	8
Problem: Python commands issue SSLError .....	9
Problem: SSH DSA keys slowdown/failures .....	10
Problem: Authentication fail .....	10
Problem: PBS jobs history disabled .....	11
Problem: The user cannot browse server or submit jobs on cluster .....	14
Problem: The wiki is slow and sometimes does not display images .....	15
Problem: RVEC notification do not work .....	16
Problem: Error during fetch in .....	16
Problem: Application stay blocked, having Job status in SUBMITTING .....	16
Problem: Gateway License expired .....	17



# Administrator's Troubleshooting

This is a list of problems and solutions that could be encountered by an administrator when using HPC Gateway. This list is not exhaustive and will be enhanced based on the returns of HPC Gateway usage. It is highly recommended that you build and share your own trouble shooting wiki page in your local area for your administrators as well.

You can also consult the [User's Troubleshooting](#).

---

## Problem: Python commands issue SSLError

If the python command are not working and issuing errors related to SSL, it may be a configuration problem between Python and SSL.

If the HTTP interface is also opened, you can confirm the problem by setting the environment variable HPCG\_BASE\_URL to aim at the HTTP interface instead of the HTTPS interface. This should make the command work by removing the need of SSL. If you want to use this trick as a workaround, you can change the value of HPCG\_BASE\_URL in `/opt/hpcg/core/etc/profile.sh`.

Check if the python you are using is the one deployed with HPC Gateway installer and that the SSL module is correctly installed.

```
[hpcgadmin@hpcgdemo ~]$ which python
/opt/hpcg/external/python2.7/bin/python

[hpcgadmin@hpcgdemo ~]$ python -c "import ssl; print ssl.OPENSSL_VERSION"
OpenSSL 1.0.1e-fips 11 Feb 2013
```

If the SSL module is not correctly installed, you must install the openssl and openssl-devel packages and reinstall HPC Gateway or just recompile/install python.

We have also experience some problems with the very specific version 1.0.1e-15 of OpenSSL on RedHat 6.5. You can check what is the verion installed on your system.

```
[hpcgadmin@rnd01 ~]$ rpm -qa | grep openssl
openssl-1.0.1e-15.el6.x86_64
openssl-devel-1.0.1e-15.el6.x86_64 # <== This is a mandatory package that
must be installed before HPC Gateway installation
```

If you are using 1.0.1e-15, you should upgrade it with the package manager.

Note: openssl-devel is **mandatory** when installing (compiling) python, otherwise the ssl module cannot be installed and therefore cannot be used at runtime.

```
# yum search openssl-devel
...
```

```
# yum install openssl-devel.x86_64
```

```
...
```

---

## Problem: SSH DSA keys slowdown/failures

We have observed problems when using DSA keys with recent OpenSSH daemons (openHPC and Ubuntu > 16). In addition DSA keys are considered too weak and should be regarded as deprecated.

We recommend using SSH RSA keys and not DSA. If you are experiencing some problem with the SSH/SFTP commands issued by HPC Gateway, make sure you are not using DSA keys.

---

## Problem: Authentication fail

There are several reasons why a user can not login into HPC Gateway:

- The password is wrong
- The user is not defined in the database
- The user's uid is too low

### The password is wrong

Make sure the password you type is the good one and the keyboard is not in caps locked or using another language.

### The user is not defined in the database

HPC Gateway authentication and authorisation mechanism are described in [identity management wiki](#).

Check the database using RoboMongo tool (if installed) or `hpcg_dbase_print.py` command line.

```
$ hpcg_dbase_print.py -k configs.webserver.settings
[{'key': u'autoPopulate', 'value': u'true'}, {'key': u'defaultTeam',
'value': u'568e3a3cddff3a6ccdaf92c8'}]

$ hpcg_dbase_print.py -k users.hpcgadmin
{'info': {'phone': u'', 'role': u'admin', 'address': u''},
'creationDate': 1464357258L, 'wikiURI': u'', 'tags': [u'hpc',
u'hpcgadmin', u'admin'], 'admin': True, 'statusLifecycle': u'active',
'teams': [{'role': u'admin', 'name': u'Public', 'id':
u'568e3a3cddff3a6ccdaf92c8'}], 'modificationDate': 1464357258L,
'fullName': u'hpcgadmin', '_id': ObjectId('5748518a0ec8114072f11cfe'),
```

```
u'email': u'', u'projects': [], u'name': u'hpcgadmin'}
```

If the user is not set and autoPopulate is False, then consider to:

- Set autoPopulate to True
- Add the user in the database using hpcg\_users.py command line

---

## Problem: PBS jobs history disabled

In order to work correctly, HPC Gateway requires PBS job\_history to be enabled. If not, the jobs disappear from PBS history as soon as they are complete and HPC Gateway will never have the opportunity to 'see' them as finished.

As a consequence, the tasks will remain RUNNING forever.

You can use the following command in order to check this if job\_history PBS parameter is configured:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 0
    state_count = Transit:0 Queued:0 Held:0 Waiting:0 Running:0
Exiting:0 Be
gun:0
    default_queue = workq
    log_events = 511
    mail_from = adm
    query_other_jobs = True
    resources_default.ncpus = 1
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
    scheduler_iteration = 600
    FLicenses = 2109
    resv_enable = True
    node_fail_requeue = 310
    max_array_size = 10000
    pbs_license_info = 6200@pbs-lic.hpc.qanet
    pbs_license_min = 1
    pbs_license_max = 2147483647
    pbs_license_linger_time = 31536000
    license_count = Avail_Global:1917 Avail_Local:192 Used:0
High_Use:192 Av
ail_Sockets:0 Unused_Sockets:0
    pbs_version = PBSPro_13.1.0.160576
```

```
eligible_time_enable = False
max_concurrent_provision = 5
```

If there is no `job_history` parameters in the output of the previous command, it means that it is set to no meaning disabled.

### How to fix:

Use the following command and check again the configuration:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr -c "set server job_history_enable
= True"

[root@myhead ~]# /opt/pbs/default/bin/qmgr
Max open servers: 49
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 0
    state_count = Transit:0 Queued:0 Held:0 Waiting:0 Running:0
Exiting:0 Begun:0
    default_queue = workq
    log_events = 511
    mail_from = adm
    query_other_jobs = True
    resources_default.ncpus = 1
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
    scheduler_iteration = 600
    FLicenses = 2109
    resv_enable = True
    node_fail_requeue = 310
    max_array_size = 10000
    pbs_license_info = 6200@pbs-lic.hpc.qanet
    pbs_license_min = 1
    pbs_license_max = 2147483647
    pbs_license_linger_time = 31536000
    license_count = Avail_Global:1917 Avail_Local:192 Used:0
High_Use:192 Avail_Sockets:0 Unused_Sockets:0
    pbs_version = PBSPro_13.1.0.160576
    eligible_time_enable = False
    job_history_enable = True
    max_concurrent_provision = 5
```

The default value for `job_history_duration` is 336 hours meaning 2 weeks. HPC Gateway needs a minimum of 10 minutes.

Unless you need more history duration in your cluster use case, here after you will find how to set the job history duration to 10 minutes:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr -c "set server
job_history_duration=00:10:00 "
```

Then you can check if the new `job_history_duration` is correctly set:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 29
    state_count = Transit:0 Queued:0 Held:1 Waiting:0 Running:0
Exiting:0 Begun:0
    default_queue = workq
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
    scheduler_iteration = 600
    FLicenses = 0
    resv_enable = True
    node_fail_requeue = 310
    max_array_size = 10000
    default_qsub_arguments = -V
    pbs_license_info = /opt/Altair/license/altair_pbspro_lic.dat
    pbs_license_min = 0
    pbs_license_max = 2147483647
    pbs_license_linger_time = 31536000
    license_count = Avail_Global:0 Avail_Local:0 Used:0 High_Use:0
Avail_Sockets:34 Unused_Sockets:3
    pbs_version = 14.2.1.20170124052131
    eligible_time_enable = False
    job_history_enable = True
    job_history_duration = 00:10:00
    max_concurrent_provision = 5
```

Finally, you must restart the cluster agent then open `job_submitter` and check task status after submission. Note that the previous executed tasks and related jobs will stay forever with the UNKNOWN status.

### The user's id is too low

For security reason, there is a minimum id for a user to be able to connect to HPC Gateway. For instance, we do not want that root connect to HPC Gateway, and more generally we want to avoid that system user can connect. This limit is set in jetty configuration file:

`${HPCG_HOME}/core/jetty/etc/login.conf`. The id of the user must be greater than `uidMin`.

```
$ cat ${HPCG_HOME}/core/jetty/etc/login.conf
...
ssh-login-module {
    com.fujitsu.fse.torii.authentication.SshLoginModule required
    debug="true"
    hostname="localhost"
    port="22"
    uidMin="1000";
};
```

Note that the user id is defined in `/etc/passwd` and the system user id limit is defined in `/etc/login.defs`

```
$ grep hpcgadmin /etc/passwd
hpcgadmin:x:1008:1008::/home/hpcgadmin:/bin/bash

$ grep UID_MIN /etc/login.defs
UID_MIN                1000
SYS_UID_MIN            201
```

---

## **Problem: The user cannot browse server or submit jobs on cluster**

Follow the following steps. If one of these steps fail, then it is a problem and it must be resolved.

- Step 1 : "hpcgadmin" must be able to open a SSH session on behalf of the user

Run the following command:

```
$ id # make sure you are hpcgadmin
uid=10020(hpcgadmin) gid=10020(hpcgadmin) groups=10020(hpcgadmin)

$ ssh -i /opt/hpcg/repo/etc/sys/root/id_rsa_hpcg_<server> <user>@<server>
```

If the connection fails, then this is the problem: hpcgadmin must be able to connect to the user session using HPC Gateway private key. This is normally setup when the user first connect to HPC Gateway through the script `/opt/hpcg/core/etc/profile.d/hpcg.profile.sh`.

Check the script and check if there are errors when the user logs in using a standard ssh connection.

1. Step 2 : the command "groups" must return a non zero output

Run the following command:

```
$ id # make sure you are hpcgadmin
uid=10020(hpcgadmin) gid=10020(hpcgadmin) groups=10020(hpcgadmin)
```



```
$ ssh -i /opt/hpcg/repo/etc/sys/root/id_rsa_hpcg_<server> <user>@<server>
"groups ; echo \$?"
```

HPC Gateway execute "groups" command to assess rights on folders and files during the file system browsing. For example to allow file editing (or not) using the text editor.

---

## Problem: The wiki is slow and sometimes does not display images

The wiki is based on top of dokuwiki embedded in jetty, that use php-cgi. It is possible to configure php-cgi to better serve the http requests.

- Step 1: check PHP\_FCGI\_CHILDREN and PHP\_FCGI\_MAX\_REQUESTS environment variables

You can check if you have defaults in /opt/hpcg/etc/setenv\_php.sh

```
$ cat /opt/hpcg/etc/setenv_php.sh

# This file is generated by HPC Gateway installer

export hpcg_php_home=/usr/bin
export HPCG_PHP_HOME=/usr/bin

export PHP_FCGI_CHILDREN=4
export PHP_FCGI_MAX_REQUESTS=128
```

If PHP\_FCGI\_CHILDREN and PHP\_FCGI\_MAX\_REQUESTS do not exist or the value are too low, you can consider to overwrite them in your local profile located in /opt/hpcg/repo/etc/profile.<hostname>.sh

You will need to source the updated profile and restart the php process after such update.

- Step 2: increase memory\_limit in /etc/php.ini

```
memory_limit = 128M
```

---

## Problem: RVEC notification do not work

The RVEC server is allocated, but the color stays green. This means the notification from RVEC server to HPC Gateway does not work. Check that:

- RVEC servlet do not have security constraints in /opt/hpcg/core/jetty/webapps/torii/WEB-INF/web.xml.
- Port 8080 is set for the Manager port in the RVEC Management settings. If it is not set to 8080, the RVEC manager uses by default the same port the the connection one. If your are using 8443 (and you should), then the notification can not work without the setting.
- Sometimes, if we have a string proxy, the IP doesn't work with the connection. So we have to set in the Manager IP directly the name (ssf.fujitsu.fr and not the IP 69.244.109.5).

## Problem: Error during fetch in

The input data are not copied in the run directory. This means the local “hpcg\_copy.py” script did not work. You can get more information by

- Opening the task details
- Go to the runlog directory
- Open <batch>-<jobid>.out

Here are some common errors and recovery:

- hpcg\_copy.py not found: the script is not in /home/hpcgadmin/hpcgateway/utilities directory. Restart the cluster agent and check it the script is in the right location.
- the name of the server is not identical to the hostname: hpcg\_copy rely on the name of the server to run scp command. Then, the name of the server must be in /etc/hosts to be recognized as the server

## Problem: Application stay blocked, having Job status in SUBMITTING

The Gateway service named cluster is stopped or is not responsive.

Logged as hpcgadmin check the cluster service status:

```
[hpcgadmin@head ~]$ source /opt/hpcg/core/etc/profile.sh
[hpcgadmin@head ~]$ hpcg.sh
=> status of mongo          on head          : running [1524]
=> status of jetty          on head          : running [1565]
=> status of php            on head          : running [1605]
=> status of cluster        on head          : stopped - not
running but pid file exists
```

You can see that cluster service is stopped, but a related pid file still exists.

The pid file does'nt allow to start the cluster service with the simple start order.

But the service can be restart anyway with the restart order.

Restart the Gateway cluster service:

```
[hpcgadmin@head ~]$ source /opt/hpcg/core/etc/profile.sh
[hpcgadmin@head ~]$
[hpcgadmin@YaKiToRi ~]$ hpcg.sh -s restart -l cluster -c "Restart cluster
after head node reboot"
=> stop    of cluster        on head          : unknown
=> start   of cluster        on head          : started successfully

*****
```

```
* Gateway processes ['cluster'] *
*****

=> cluster | 7268 | hpcgadmin | 2018-09-27 12:02:18 |
/opt/hpcg/external/jdk1.8.0_60/bin/java -
agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=7008 -Xms512m -
Xmx2048m -Dtorii.agent.cluster.name=yakitori -
Djava.util.logging.config.file=/opt/hpcg/repo/etc/logging.yakitori.properties
-Dtorii.mongo.host.name=127.0.0.1 -Dtorii.mongo.host.port=27017 -
Dtorii.mongo.database=Torii -Dtorii.mongo.usessl=false -
Dtorii.python.interpreter=/home/hpcgadmin/hpcgateway/external/python2.7/bin/
python -Dtorii.location=main -jar /opt/hpcg/core/cluster/lib/torii-cluster-
agent.jar
```

This cluster service stopped can be the consequence of:

- a voluntary cluster service stop
- a cluster service crash
- a voluntary a head node restart.

This restart could be:

- a brutal power off using the directly the power button
- a power outage
- a simple init command
- ...

## Problem: Python commands issue SSLError

If the python command are not working and issuing errors related to SSL, it may be a configuration problem between Python and SSL.

If the HTTP interface is also opened, you can confirm the problem by setting the environment variable HPCG\_BASE\_URL to aim at the HTTP interface instead of the HTTPS interface. This should make the command work by removing the need of SSL. If you want to use this trick as a workaround, you can change the value of HPCG\_BASE\_URL in /opt/hpcg/core/etc/profile.sh .

Check if the python you are using is the one deployed with HPC Gateway installer and that the SSL module is correctly installed.

```
[hpcgadmin@hpcgdemo ~]$ which python
/opt/hpcg/external/python2.7/bin/python

[hpcgadmin@hpcgdemo ~]$ python -c "import ssl; print ssl.OPENSSL_VERSION"
OpenSSL 1.0.1e-fips 11 Feb 2013
```

If the SSL module is not correctly installed, you must install the openssl and openssl-devel packages and reinstall HPC Gateway or just recompile/install python.

We have also experience some problems with the very specific version 1.0.1e-15 of OpenSSL on RedHat 6.5. You can check what is the verion installed on your system.

```
[hpcgadmin@rnd01 ~]$ rpm -qa | grep openssl
openssl-1.0.1e-15.el6.x86_64
openssl-devel-1.0.1e-15.el6.x86_64 # <== This is a mandatory package that
must be installed before HPC Gateway installation
```

If you are using 1.0.1e-15, you should upgrade it with the package manager.

Note: openssl-devel is **mandatory** when installing (compiling) python, otherwise the ssl module cannot be installed and therefore cannot be used at runtime.

```
# yum search openssl-devel
...

# yum install openssl-devel.x86_64
...
```

---

## Problem: SSH DSA keys slowdown/failures

We have observed problems when using DSA keys with recent OpenSSH daemons (openHPC and Ubuntu > 16). In addition DSA keys are considered too weak and should be regarded as deprecated.

We recommend using SSH RSA keys and not DSA. If you are experiencing some problem with the SSH/SFTP commands issued by HPC Gateway, make sure you are not using DSA keys.

---

## Problem: Authentication fail

There are several reasons why a user can not login into HPC Gateway:

- The password is wrong
- The user is not defined in the database
- The user's uid is too low

### The password is wrong

Make sure the password you type is the good one and the keyboard is not in caps locked or using another language.

### The user is not defined in the database

HPC Gateway authentication and authorisation mechanism are described in [identity management wiki](#).

Check the database using RoboMongo tool (if installed) or `hpcg_dbase_print.py` command line.

```
$ hpcg_dbase_print.py -k configs.webserver.settings
[{'u'key': u'autoPopulate', u'value': u'true'}, {'u'key': u'defaultTeam',
u'value': u'568e3a3cddff3a6ccdaf92c8'}]

$ hpcg_dbase_print.py -k users.hpcgadmin
{'u'info': {'u'phone': u'', u'role': u'admin', u'address': u''},
u'creationDate': 1464357258L, u'wikiURI': u'', u'tags': [u'hpc',
u'hpcgadmin', u'admin'], u'admin': True, u'statusLifecycle': u'active',
u'teams': [{'u'role': u'admin', u'name': u'Public', u'id':
u'568e3a3cddff3a6ccdaf92c8'}], u'modificationDate': 1464357258L,
u'fullName': u'hpcgadmin', u'_id': ObjectId('5748518a0ec8114072f11cfe'),
u'email': u'', u'projects': [], u'name': u'hpcgadmin'}
```

If the user is not set and `autoPopulate` is `False`, then consider to:

- Set `autoPopulate` to `True`
- Add the user in the database using `hpcg_users.py` command line

---

## Problem: PBS jobs history disabled

In order to work correctly, HPC Gateway requires `PBS job_history` to be enabled. If not, the jobs disappear from PBS history as soon as they are complete and HPC Gateway will never have the opportunity to 'see' them as finished.

As a consequence, the tasks will remain `RUNNING` forever.

You can use the following command in order to check this if `job_history` PBS parameter is configured:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 0
    state_count = Transit:0 Queued:0 Held:0 Waiting:0 Running:0
Exiting:0 Be
gun:0
    default_queue = workq
    log_events = 511
    mail_from = adm
    query_other_jobs = True
    resources_default.ncpus = 1
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
```

```
scheduler_iteration = 600
FLicenses = 2109
resv_enable = True
node_fail_requeue = 310
max_array_size = 10000
pbs_license_info = 6200@pbs-lic.hpc.qanet
pbs_license_min = 1
pbs_license_max = 2147483647
pbs_license_linger_time = 31536000
license_count = Avail_Global:1917 Avail_Local:192 Used:0
High_Use:192 Av
ail_Sockets:0 Unused_Sockets:0
pbs_version = PBSPro_13.1.0.160576
eligible_time_enable = False
max_concurrent_provision = 5
```

If there is no job\_history parameters in the output of the previous command, it means that it is set to no meaning disabled.

### How to fix:

Use the following command and check again the configuration:

```
[root@myhead ~]# /opt/pbs/default/bin/qmgr -c "set server job_history_enable
= True"

[root@myhead ~]# /opt/pbs/default/bin/qmgr
Max open servers: 49
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 0
    state_count = Transit:0 Queued:0 Held:0 Waiting:0 Running:0
Exiting:0 Begun:0
    default_queue = workq
    log_events = 511
    mail_from = adm
    query_other_jobs = True
    resources_default.ncpus = 1
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
    scheduler_iteration = 600
    FLicenses = 2109
    resv_enable = True
    node_fail_requeue = 310
    max_array_size = 10000
```

```

pbs_license_info = 6200@pbs-lic.hpc.qanet
pbs_license_min = 1
pbs_license_max = 2147483647
pbs_license_linger_time = 31536000
license_count = Avail_Global:1917 Avail_Local:192 Used:0
High_Use:192 Avail_Sockets:0 Unused_Sockets:0
pbs_version = PBSPro_13.1.0.160576
eligible_time_enable = False
job_history_enable = True
max_concurrent_provision = 5

```

The default value for `job_history_duration` is 336 hours meaning 2 weeks. HPC Gateway needs a minimum of 10 minutes.

Unless you need more history duration in your cluster use case, here after you will find how to set the job history duration to 10 minutes:

```

[root@myhead ~]# /opt/pbs/default/bin/qmgr -c "set server
job_history_duration=00:10:00 "

```

Then you can check if the new `job_history_duration` is correctly set:

```

[root@myhead ~]# /opt/pbs/default/bin/qmgr
Qmgr: list server
Server myhead
    server_state = Active
    server_host = myhead.hpc.zone
    scheduling = True
    total_jobs = 29
    state_count = Transit:0 Queued:0 Held:1 Waiting:0 Running:0
Exiting:0 Begun:0
    default_queue = workq
    default_chunk.ncpus = 1
    resources_assigned.mpiprocs = 0
    resources_assigned.ncpus = 0
    resources_assigned.nodect = 0
    scheduler_iteration = 600
    FLicenses = 0
    resv_enable = True
    node_fail_requeue = 310
    max_array_size = 10000
    default_qsub_arguments = -V
    pbs_license_info = /opt/Altair/license/altair_pbspro_lic.dat
    pbs_license_min = 0
    pbs_license_max = 2147483647
    pbs_license_linger_time = 31536000
    license_count = Avail_Global:0 Avail_Local:0 Used:0 High_Use:0
Avail_Sockets:34 Unused_Sockets:3
    pbs_version = 14.2.1.20170124052131
    eligible_time_enable = False
    job_history_enable = True

```

```
job_history_duration = 00:10:00
max_concurrent_provision = 5
```

Finally, you must restart the cluster agent then open `job_submitter` and check task status after submission. Note that the previous executed tasks and related jobs will stay forever with the UNKNOWN status.

## The user's id is too low

For security reason, there is a minimum id for a user to be able to connect to HPC Gateway. For instance, we do not want that root connect to HPC Gateway, and more generally we want to avoid that system user can connect. This limit is set in jetty configuration file: `${HPCG_HOME}/core/jetty/etc/login.conf`. The id of the user must be greater than `uidMin`.

```
$ cat ${HPCG_HOME}/core/jetty/etc/login.conf
...
ssh-login-module {
    com.fujitsu.fse.torii.authentication.SshLoginModule required
    debug="true"
    hostname="localhost"
    port="22"
    uidMin="1000";
};
```

Note that the user id is defined in `/etc/passwd` and the system user id limit is defined in `/etc/login.defs`

```
$ grep hpcgadmin /etc/passwd
hpcgadmin:x:1008:1008::/home/hpcgadmin:/bin/bash

$ grep UID_MIN /etc/login.defs
UID_MIN          1000
SYS_UID_MIN      201
```

---

## Problem: The user cannot browse server or submit jobs on cluster

Follow the following steps. If one of these steps fail, then it is a problem and it must be resolved.

- Step 1 : "hpcgadmin" must be able to open a SSH session on behalf of the user

Run the following command:

```
$ id # make sure you are hpcgadmin
uid=10020(hpcgadmin) gid=10020(hpcgadmin) groups=10020(hpcgadmin)
```



```
$ ssh -i /opt/hpcg/repo/etc/sys/root/id_rsa_hpcg_<server> <user>@<server>
```

If the connection fails, then this is the problem: hpcgadmin must be able to connect to the user session using HPC Gateway private key. This is normally setup when the user first connect to HPC Gateway through the script /opt/hpcg/core/etc/profile.d/hpcg.profile.sh.

Check the script and check if there are errors when the user logs in using a standard ssh connection.

1. Step 2 : the command "groups" must return a non zero output

Run the following command:

```
$ id # make sure you are hpcgadmin
uid=10020(hpcgadmin) gid=10020(hpcgadmin) groups=10020(hpcgadmin)

$ ssh -i /opt/hpcg/repo/etc/sys/root/id_rsa_hpcg_<server> <user>@<server>
"groups ; echo \${?}"
```

HPC Gateway execute "groups" command to assess rights on folders and files during the file system browsing. For example to allow file editing (or not) using the text editor.

---

## Problem: The wiki is slow and sometimes does not display images

The wiki is based on top of dokuwiki embedded in jetty, that use php-cgi. It is possible to configure php-cgi to better serve the http requests.

- Step 1: check PHP\_FCGI\_CHILDREN and PHP\_FCGI\_MAX\_REQUESTS environment variables

You can check if you have defaults in /opt/hpcg/etc/setenv\_php.sh

```
$ cat /opt/hpcg/etc/setenv_php.sh

# This file is generated by HPC Gateway installer

export hpcg_php_home=/usr/bin
export HPCG_PHP_HOME=/usr/bin

export PHP_FCGI_CHILDREN=4
export PHP_FCGI_MAX_REQUESTS=128
```

If PHP\_FCGI\_CHILDREN and PHP\_FCGI\_MAX\_REQUESTS do not exist or the value are too low, you can consider to overwrite them in your local profile located in /opt/hpcg/repo/etc/profile.<hostname>.sh

You will need to source the updated profile and restart the php process after such update.

- Step 2: increase memory\_limit in /etc/php.ini

```
memory_limit = 128M
```

## Problem: RVEC notification do not work

The RVEC server is allocated, but the color stays green. This means the notification from RVEC server to HPC Gateway does not work. Check that:

- RVEC servlet do not have security constraints in `/opt/hpcg/core/jetty/webapps/torii/WEB-INF/web.xml`.
- Port 8080 is set for the Manager port in the RVEC Management settings. If it is not set to 8080, the RVEC manager uses by default the same port the the connection one. If your are using 8443 (and you should), then the notification can not work without the setting.
- Sometimes, if we have a string proxy, the IP doesn't work with the connection. So we have to set in the Manager IP directly the name (ssf.fujitsu.fr and not the IP 69.244.109.5).

---

## Problem: Error during fetch in

The input data are not copied in the run directory. This means the local “hpcg\_copy.py” script did not work. You can get more information by

- Opening the task details
- Go to the runlog directory
- Open `<batch>-<jobid>.out`

Here are some common errors and recovery:

- `hpcg_copy.py` not found: the script is not in `/home/hpcgadmin/hpcgateway/utilities` directory. Restart the cluster agent and check it the script is in the right location.
- the name of he server is not identical to the hostname: `hpcg_copy` rely on the name of the server to run `scp` command. Then, the name of the server must be in `/etc/hosts` to be recognized as the server

## Problem: Application stay blocked, having Job status in SUBMITTING

The Gateway service named cluster is stopped or is not responsive.

Logged as `hpcgadmin` check the cluster service status:

```
[hpcgadmin@head ~]$ source /opt/hpcg/core/etc/profile.sh
[hpcgadmin@head ~]$ hpcg.sh
=> status of mongo           on head           : running [1524]
=> status of jetty           on head           : running [1565]
=> status of php             on head           : running [1605]
=> status of cluster         on head           : stopped - not
running but pid file exists
```

You can see that cluster service is stopped, but a related pid file still exists.

The pid file doesn't allow to start the cluster service with the simple start order.

But the service can be restart anyway with the restart order.

Restart the Gateway cluster service:

```
[hpcgadmin@head ~]$ source /opt/hpcg/core/etc/profile.sh
[hpcgadmin@head ~]$
[hpcgadmin@YaKiToRi ~]$ hpcg.sh -s restart -l cluster -c "Restart cluster
after head node reboot"
=> stop      of cluster      on head      : unknown
=> start     of cluster      on head      : started successfully

*****
* Gateway processes ['cluster'] *
*****

=> cluster | 7268      | hpcgadmin | 2018-09-27 12:02:18 |
/opt/hpcg/external/jdk1.8.0_60/bin/java -
agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=7008 -Xms512m -
Xmx2048m -Dtorii.agent.cluster.name=yakitori -
Djava.util.logging.config.file=/opt/hpcg/repo/etc/logging.yakitori.properties
-Dtorii.mongo.host.name=127.0.0.1 -Dtorii.mongo.host.port=27017 -
Dtorii.mongo.database=Torii -Dtorii.mongo.usessl=false -
Dtorii.python.interpreter=/home/hpcgadmin/hpcgateway/external/python2.7/bin/
python -Dtorii.location=main -jar /opt/hpcg/core/cluster/lib/torii-cluster-
agent.jar
```

This cluster service stopped can be the consequence of:

- a voluntary cluster service stop
- a cluster service crash
- a voluntary a head node restart.

This restart could be:

- a brutal power off using the directly the power button
- a power outage
- a simple init command
- ...

## Problem: Gateway License expired

The Gateway license is expired.

An expired license triggers a message at service restart and in Gateway cluster agent log file.

Here after you can see the type of message displayed at Gateway cluster agent restart:

```
[hpcgadmin@head hpcgadmin]$ source /opt/hpcg/core/etc/profiles.sh
[hpcgadmin@head hpcgadmin]$ hpcg.sh -s restart -l all -c "license checking"
=> stop    of cluster.head      on head      : stop successful
=> stop    of scheduler         on head      : stop successful
=> stop    of php               on head      : stop successful
=> stop    of jetty             on head      : stop successful
=> stop    of mongo             on head      : stop successful
=> start   of mongo             on head      : started
successfully
=> start   of jetty             on head      : started
successfully
=> start   of php               on head      : started
successfully
=> start   of scheduler         on head      : started
successfully
=> start   of cluster.head      on head      : started
successfully
2019/01/11 16:37:15 - ERROR - The license has expired on 2018-12-30

2019/01/11 16:37:15 - ERROR - Contact your local Fujitsu support
2019/01/11 16:37:15 - ERROR - Or visit Fujitsu support web site
...
```

In order to confirm, you can display with less or more details the current installed license:

```
[hpcgadmin@head hpcgadmin]$ hpcg_license.py

License summary
=====
License max tokens           : 400
License validity (days)     : 90
License installation date    : 2018-10-01
License expiration date      : 2018-12-30

Token usage                   : 20 (5%)

[hpcgadmin@head hpcgadmin]$ hpcg_license.py --details

License details
=====
Collection                   :      Doc   Token/   Total   Total
                             :      num   doc     token  cost

CORE
Clusters                     :         1     15     15     1,500
Servers                      :         1      5      5      500

ADD-ON
```

```
Total
=====
Token usage      =    20
Annual cost      =   2,000 euros
```

```
License summary
=====
License max tokens      : 400
License validity (days) : 90
License installation date : 2018-10-01
License expiration date  : 2018-12-30

Token usage              : 20 (5%)
```

Fujitsu must then provide a license file that must be installed on Gateway host. The license must be transferred on Gateway host.

Use the command `hpcg_license_py` as following in order to install the given license:

```
[hpcgadmin@head hpcgadmin]$ hpcg_license.py --
license_file=/tmp/license.<customer_name>.<XXX>tokens.<date>.dat
2019/01/11 16:53:45 - INFO - License successfully updated
```

```
License summary
=====
License max tokens      : 320
License validity (days) : None
License installation date : 2019-01-11
License expiration date  : 2019-09-01

Token usage              : 20 (6%)
```

The restart all the services including Gateway cluster agent:

```
[hpcgadmin@head hpcgadmin]$hpcg.sh -s restart -l all -c "license checking"
=> stop    of cluster.head      on head      : stop successful
=> stop    of scheduler         on head      : stop successful
=> stop    of php               on head      : stop successful
=> stop    of jetty             on head      : stop successful
=> stop    of mongo             on head      : stop successful
=> start   of mongo             on head      : started
successfully
=> start   of jetty             on head      : started
successfully
=> start   of php               on head      : started
successfully
=> start   of scheduler         on head      : started
successfully
=> start   of cluster.head      on head      : started
successfully
```

...

The previous message regarding expired license should not appear anymore

In order to confirm, you can display with less or more details the new installed license:

```
[hpcgadmin@head hpcgadmin]$ hpcg_license.py
```

License summary

=====

```
License max tokens           : 320
License validity (days)     : None
License installation date    : 2019-01-11
License expiration date     : 2019-09-01
```

```
Token usage                  : 20 (6%)
```

```
[hpcgadmin@hpcgadmin hpcgadmin]$ hpcg_license.py --details
```

License details

=====

Collection	:	Doc num	Token/doc	Total token	Total cost
CORE					
Clusters	:	1	15	15	1,500
Servers	:	1	5	5	500

ADD-ON

Total

=====

```
Token usage    = 20
Annual cost    = 2,000 euros
```

License summary

=====

```
License max tokens           : 320
License validity (days)     : None
License installation date    : 2019-01-11
License expiration date     : 2019-09-01
```

```
Token usage                  : 20 (6%)
```

From:  
<http://docs.fujitsu.fr/HPCGateway/current/> - **HPC Gateway**

Permanent link:  
[http://docs.fujitsu.fr/HPCGateway/current/doku.php?id=fujitsu:hpcgateway:guides:faq:troubleshooting\\_admin](http://docs.fujitsu.fr/HPCGateway/current/doku.php?id=fujitsu:hpcgateway:guides:faq:troubleshooting_admin)

Last update: **2019/04/05 13:58**

