

Table of Contents

Notes about Installation and Configuration of a MyProxy server - EMI-2 - SL6 x86_64.....	1
References.....	1
Service installation.....	1
O.S. and Repos.....	1
yum install.....	1
Service configuration.....	2
site-info.def.....	2
glite-px.....	2
host certificate required.....	4
Service configuration.....	4
yaim check.....	4
yaim config.....	4
Know Issue and Workaround.....	4
Service checks.....	4
myproxy-init.....	4
myproxy-info.....	5
myproxy-get-delegation.....	5
Additional notes.....	5
MyProxy server High Availability.....	6
Master-Master configuration.....	6
Master-Slave configuration.....	6
Configuring the Master Server.....	6
Configuring Slave Servers.....	6
Revision.....	7

Notes about Installation and Configuration of a MyProxy server - EMI-2 - SL6 x86_64

- These notes are provided by site admins on a best effort base as a contribution to the IGI communities and **MUST** not be considered as a substitute of the Official IGI documentation .
- This document is addressed to site administrators responsible for middleware installation and configuration.
- The goal of this page is to provide some hints and examples on how to install and configure an IGI **myproxy** service based on EMI-2 middleware on SL6.

NB: The myproxy service is a **CORE** service, it should not be installed at Resource Center level. The official endpoint provided by IGI is `myproxy.cnaf.infn.it` and **MUST** be used by all Resource Centers and Services part of the IGI infrastructure.

References

1. About IGI - Italian Grid infrastructure
2. About IGI Release
3. EMI-2 Release
4. Yaim Guide
5. site-info.def yaim variables
6. site-BDII yaim variables
7. Site Certification GIIS Check
8. Troubleshooting Guide for Operational Errors on EGI Sites
9. Grid Administration FAQs page

Service installation

O.S. and Repos

- Starts from a fresh installation of Scientific Linux 6.x (x86_64).

```
# cat /etc/redhat-release
Scientific Linux release 6.2 (Carbon)
```

* Install the additional repositories: EPEL, Certification Authority, EMI-2

```
# yum install yum-priorities yum-protectbase epel-release
# rpm -ivh http://emisoft.web.cern.ch/emisoft/dist/EMI/2/sl6/x86_64/base/emi-release-2.0.0-1.sl6.
# cd /etc/yum.repos.d/
# wget http://repo-pd.italiangrid.it/mrepo/repos/egi-trustanchors.repo
```

- Be sure that SELINUX is disabled (or permissive). Details on how to disable SELINUX are here :

```
# getenforce
Disabled
```

yum install

```
# yum clean all
Loaded plugins: downloadonly, kernel-module, priorities, protect-packages, protectbase, security,
Cleaning up Everything
```

```
# yum install ca-policy-egi-core
# yum install emi-px
```

Service configuration

The configuration file for this service is really basic. For authorization:

- DN list of authorized renewals (WMS and nagios)
- DN list of trusted retrievers (nagios)

site-info.def

```
# cp -vr /opt/glite/yaim/examples/siteinfo /root/
~/opt/glite/yaim/examples/siteinfo' -> ~/root/siteinfo'
~/opt/glite/yaim/examples/siteinfo/site-info.def' -> ~/root/siteinfo/site-info.def'
~/opt/glite/yaim/examples/siteinfo/services' -> ~/root/siteinfo/services'
~/opt/glite/yaim/examples/siteinfo/services/glite-px' -> ~/root/siteinfo/services/glite-px'
~/opt/glite/yaim/examples/siteinfo/services/glite-bdii_site' -> ~/root/siteinfo/services/glite-bdii_site'

# cat /root/siteinfo/site-info.def
SITE_NAME=IGI-BOLOGNA
PX_HOST=`hostname -f`
BDII_DELETE_DELAY=0
```

glite-px

```
# cat siteinfo/services/glite-px
GRID_AUTHORIZED_RETRIEVERS="\*"

GRID_AUTHORIZED_RENEWERS="
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=gridit-wms-01.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-wms-01.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-01.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-02.cnaf.infn.it'
'/C=IT/O=INFN/OU=grid014.ct.infn.it/L=Catania/CN=grid014.ct.infn.it/emailAddress=giuseppe.platani'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=gridit-cert-rb.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=eumed-rb-1.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=euchina-rb-1.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-03.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-04.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-05.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-06.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=gridit-rb-01.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=egrid-rb-01.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=prod-rb-01.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=prod-rb-02.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=prod-wms-01.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=eu-india-02.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=sc2.cr.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=Bari/CN=wms1.ba.infn.it'
'/C=IT/O=INFN/OU=Host/L=Bari/CN=wms2.ba.infn.it'
'/C=IT/O=INFN/OU=Host/L=Bari/CN=wms3.ba.infn.it'
'/C=CH/O=CERN/OU=GRID/CN=host/lxn1185.cern.ch'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-07.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-08.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee-rb-09.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=cert-rb-06.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=glite-rb-00.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=glite-rb-01.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel07.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel09.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel10.cnaf.infn.it'
```

```
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel11.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel12.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel14.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel18.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel19.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=devel20.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=Padova/CN=cream-06.pd.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms001.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms002.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms003.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms004.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms005.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms006.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms007.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms008.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms009.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms011.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms012.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms013.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms014.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms015.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms016.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=wms017.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=cert-02.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=pps-fts.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=tigerman.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=Milano/CN=egee-rb-01.mi.infn.it'
'/C=IT/O=INFN/OU=Host/L=CIRMMP/CN=wms-enmr.cern.unifi.it'
'/DC=ch/DC=cern/OU=computers/CN=wms101.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms102.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms103.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms104.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms105.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms106.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms107.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms108.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms109.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms110.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms111.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms112.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms113.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms114.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms115.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms116.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms117.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms118.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms119.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms121.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms122.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms123.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms124.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms125.cern.ch'
'/DC=ch/DC=cern/OU=computers/CN=wms126.cern.ch'
'/O=dutchgrid/O=hosts/OU=nikhef.nl/CN=graszode.nikhef.nl'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=mon-it.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=mon-cnaf.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=bbrbuild01.cr.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=bbr-serv09.cr.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee017.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=sb-serv01.cr.cnaf.infn.it'
"

GRID_TRUSTED_RETRIEVERS="
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=mon-it.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=mon-cnaf.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=bbrbuild01.cr.cnaf.infn.it'
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=bbr-serv09.cr.cnaf.infn.it'
```

```
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=egee017.cnaf.infn.it'  
'/C=IT/O=INFN/OU=Host/L=CNAF/CN=sb-serv01.cr.cnaf.infn.it'  
"
```

host certificate required

```
# ll /etc/grid-security/host*  
-rw-r--r-- 1 root root 1440 Dec 29 09:30 /etc/grid-security/hostcert.pem  
-r----- 1 root root 887 Dec 29 09:30 /etc/grid-security/hostkey.pem
```

Service configuration

yaim check

```
# chmod -R 600 /root/siteinfo  
  
# /opt/glite/yaim/bin/yaim -v -s /root/siteinfo/site-info.def -n glite-PX  
INFO: Using site configuration file: /root/siteinfo/site-info.def  
[...]  
INFO: YAIM terminated succesfully.
```

yaim config

Please use the debug flag ("-d 6") to configure the services in order to have detailed information. For your convenience you can save all the configuration information in a log file you can look at any time, separated from the yaimlog default one.

```
# /opt/glite/yaim/bin/yaim -c -d 6 -s /root/siteinfo/site-info.def -n glite-PX  
DEBUG: Checking siteinfo dir is not world readable  
[...]  
INFO: Configuration Complete. [ OK ]  
INFO: YAIM terminated succesfully.
```

Know Issue and Workaround

Al momento il servizio non parte al boot (baco di yaim, notificato in GGUS .

```
# chkconfig myproxy-server on
```

Service checks

myproxy-init

On a user interface:

```
# $ myproxy-init -s myproxy.cnaf.infn.it -k veronesi-test  
username: veronesi  
owner: /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/CN=Paolo Veronesi  
name: veronesi-test  
timeleft: 167:55:38 (7.0 days)  
[veronesi@ui ~]$ myproxy-init -s myproxy.cnaf.infn.it -k veronesi-test  
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/CN=Paolo Veronesi  
Enter GRID pass phrase for this identity:  
Creating proxy .....  
Proxy Verify OK  
Your proxy is valid until: Thu Jan 5 10:03:38 2012  
Enter MyProxy pass phrase:  
Verifying - Enter MyProxy pass phrase:  
A proxy valid for 168 hours (7.0 days) for user veronesi now exists on myproxy.cnaf.infn.it.
```

On the MyProxy server:

```
# tail -f /var/log/messages
Dec 29 10:03:40 myproxy myproxy-server[9119]: Connection from 131.154.101.141
Dec 29 10:03:41 myproxy myproxy-server[9119]: Authenticated client /C=IT/O=INFN/OU=Personal Certi
Dec 29 10:03:42 myproxy myproxy-server[9119]: Received PUT request for username veronesi
Dec 29 10:03:43 myproxy myproxy-server[9119]: Client /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/

# ls -ltr /var/lib/myproxy/
total 36
-rw----- 1 myproxy myproxy 132 Dec 29 10:03 veronesi-veronesi-test.data
-rw----- 1 myproxy myproxy 5912 Dec 29 10:03 veronesi-veronesi-test.creds
```

myproxy-info

On a user interface:

```
# myproxy-info -s myproxy.cnaf.infn.it -k veronesi-test
username: veronesi
owner: /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/CN=Paolo Veronesi
  name: veronesi-test
  timeleft: 167:55:38 (7.0 days)
```

On the MyProxy server:

```
# tail -f /var/log/messages
Dec 29 10:42:08 myproxy myproxy-server[9209]: Connection from 131.154.101.141
Dec 29 10:42:08 myproxy myproxy-server[9209]: Authenticated client /C=IT/O=INFN/OU=Personal Certi
Dec 29 10:42:08 myproxy myproxy-server[9209]: Received INFO request for username veronesi
Dec 29 10:42:08 myproxy myproxy-server[9209]: Client /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/
```

myproxy-get-delegation

On a user interface:

```
$ myproxy-get-delegation -s myproxy.cnaf.infn.it -k veronesi-test
Enter MyProxy pass phrase:
A credential has been received for user veronesi in /tmp/x509up_u23019.
```

On the MyProxy server:

```
# tail -f /var/log/messages
Dec 29 11:01:05 myproxy myproxy-server[31270]: Connection from 131.154.101.141
Dec 29 11:01:05 myproxy myproxy-server[31270]: Authenticated client /C=IT/O=INFN/OU=Personal Cert
Dec 29 11:01:08 myproxy myproxy-server[31270]: Received GET request for username veronesi
Dec 29 11:01:08 myproxy myproxy-server[31270]: credential passphrase matched
Dec 29 11:01:08 myproxy myproxy-server[31270]: Delegating credentials for /C=IT/O=INFN/OU=Persona
Dec 29 11:01:08 myproxy myproxy-server[31270]: Client /C=IT/O=INFN/OU=Personal Certificate/L=CNAF/
```

Additional notes

In order to make the WMS renewal function it is necessary:

1. To include the DN of the WMS that process the jobs among the authorized renewers on the MyProxy server, i.e. to add `authorized_renewers` DN to the configuration and restart the server;
2. Upload the proxy of the job submitter in the MyProxy server using `myproxy-init -s myproxy_server -d -n`
3. Submit the job with the MyProxy server hostname being given in the JDL

MyProxy server High Availability

- Reference

Master-Master configuration

This configuration is recommended in a lan.

- Two (or more) myproxy servers must share the directory `/var/lib/myproxy`.
- Round Robin DNS can be used to balance the load of the myproxy instances. An arbiter (like nagios) must check the availability of each instance and eventually remove the failing instance.

Master-Slave configuration

This configuration is recommended in a wan. **The slave server acts as read only server.**

Configuring the Master Server

- Add to `/etc/myproxy-server.config` the slaves instances:

```
slave_servers <server1>;<server2>;...
```

The MyProxy server which has been configured as the primary is also responsible for replication of its repository. Replication is accomplished via the `myproxy-replicate` utility. This utility will read the repository on the local machine as specified by its arguments. Each stored credential found in the repository will be sent to any configured secondary servers according to the `slave_servers` tag found in `myproxy-server.config`. It is recommended that this utility be called via cron or some other automatic mechanism. For example, the following script can be used to call `myproxy-replicate`:

```
#!/bin/sh

# The MyProxy replicate cron script calls the replicate utility and logs output.
export GLOBUS_LOCATION=/usr/share/globus/
. /usr/share/globus/globus-script-initializer

# The -r and -c options need to be changed to correct location for the server.
# [-storage|-r]=<path to repository> Directory of the MyProxy repository.
# [-config|-c]=<path to config file> Directory of the MyProxy Server
/usr/sbin/myproxy-replicate -r /var/lib/myproxy -c /etc/myproxy-server.config 2>&1 | logger -t my

exit 0
```

This would then be scheduled with cron by updating the `/etc/crontab` file with something like the following.

```
SHELL=/bin/sh
0,10,20,30,40,50 * * * * /root/myproxy-replicate.cron >/dev/null 2>&1
```

This line causes the replicate script to be run every 10 minutes. The time interval is at the discretion of the MyProxy administrator but the granularity should be small enough to avoid large deltas between repositories.

Configuring Slave Servers

Setting up a machine to run as a secondary MyProxy server is simple and straight forward. As with the primary, MyProxy software should be installed and configured as described in the installation instructions. Once this is done, changes need to be made to the `myproxy-server.config`.

The secondary server should only accept credentials from the primary server, so a user will never be able to store directly to a secondary and cause an inconsistency in the repository. User interaction with secondary servers is limited to `myproxy-logon` and `myproxy-retrieve`. All other commands must be performed via the primary. In normal operation, all commands should be sent to the primary, and users should not need to know about servers running as secondary machines. Secondary servers should only be accessed by clients when the primary is unreachable. In order to limit credential storage on the secondary to only the primary server, the value of `accepted_credentials` must be set to the DN of the primary. All other `myproxy-server.config` values should be set as they are on the primary. It is simplest and safest to copy the `myproxy-server.config` file from the primary to the secondary and change the value of `accepted_credentials`. The following shows a simple secondary configuration. Only the primary MyProxy server on `myproxy.cnaf.infn.it` is allowed to modify the credentials in the repository of the secondary.

```
# cat /etc/myproxy-server.config
accepted_credentials "'/C=IT/O=INFN/OU=Host/L=CNAF/CN=myproxy01.cnaf.infn.it', '/C=IT/O=INFN/OU=H
authorized_retrievers "*"
default_retrievers "*"
authorized_renewers "*"
default_renewers "none"
```

The secondary server is now ready to be run.

Revision

Date	Comment
2012-06-12	myproxy fresh installation - EMI 2 Matterhorn Products - gLite-proxyrenewal v. 1.3.25

This topic: SiteAdminCorner > NotesAboutMyProxy-EMI-2-SL6

Topic revision: r3 - 2012-06-21 - PaoloVeronesi



Copyright © 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback