LHC COMPUTING GRID

LCG PROXY MANUAL INSTALLATION AND CONFIGURATION

Document identifier:	**LCG-GIS-MI-PXY**
EDMS id:	**none**
Version:	**v1.2.0**
Date:	**August 20, 2004**
Section:	**LCG Grid Deployment**
Document status:	**Status: ACTIVE**
Author(s):	Alessandro Usai, Antonio Retico
File:	**PXY**

Abstract: *LCG Proxy Manual Installation Guide*

CONTENTS

REFERENCES

[1] A. Retico and A. Usai. Lcg grid-mapfile handling on lcg nodes, 2004. `http://www.cern.ch/grid-deployment/gis/release-docs/GMF-index.html`

ABSTRACT

Manual steps for the installation and configuration of a LCG PXY Node based on a standard (non CERN-like) Red Hat 7.3 installation

Link to this document:
This document is available on the *Grid Deployment* web site

>   http://www.cern.ch/grid-deployment/gis/release-docs/PXY-index.html

There you need to select the version tag before downloading the corresponding version of the guide.
Oter documents labeled with "cvs:", in the *References*, are maintained in txt version into the LCG cvs repository.
They can be downloaded via web by browsing the LCG Deployment home page

>   http://www.cern.ch/grid-deployment/

following then the **CVS Deployment** link on the left side of the page.

1. PREREQUISITES FOR INSTALLATION

In order to install the PXY software you need:

- a working Linux Redhat7.3 on the Proxy (This procedure has been tested on a basic RH7.3 and should not differ for more or less equivalent versions of RH. Should it be so please contact us)

- root account on PXY machine

- references to software packages to be installed (see below)

2. PROXY SOFTWARE INSTALLATION

In order to have a running PXY the following software must be installed

- PXY package

- Other software which a group may rely on

Two different methods are supported for PXY software installation

- Centrally managed PXY installation using the Debian *apt-get* tool

- Manual PXY installation using the Red Hat *rpm* tool.

The first method is simple and straightforward. Node-specific meta-packages define and hide the list of rpms to be installed. Dependencies among rpms, in particular, are easily managed by the *apt-get* tool. This may help if on your site you are running a customized version of Red Hat and you prefer not to solve dependencies manually.
On the other side, the installation using *apt-get* gives to site administrators poor control on installation and on the way the dependencies are solved.

The second method gives full control to site administrators for the installation, because it is based on an explicitly declared set of rpms to be installed on top of Red Hat 7.3
On the other hand, the installation procedure is slower and needs each unmanaged dependency to be manually solved.

2.1. PXY INSTALLATION USING *apt-get*

in order to install your Proxy using *apt-get* please refer to the following document:

http://www.cern.ch/grid-deployment/gis/aptDB/

2.2. PXY INSTALLATION USING *rpm*

All the tools needed for LCG releases are available on the *Grid Deployment* web site

http://www.cern.ch/grid-deployment/gis/release-docs/PXY-index.html

You can start browsing by selecting the relevant version tag.
Once selected your tag, you will be moved to a page where you can find:

PXY.pdf: this guide

PXY_manual_install_rpm: plain full list of those rpms needed to install the PXY software on top of a plain RH7.3 installation, whose dependencies are already dealt with.

PXY_manual_install_forced_rpm: plain full list of those rpms needed to install the PXY software on top of a plain RH7.3 installation, and whose dependencies are NOT dealt with, therefore requiring a forced installation. If this list is not present or void it means that no rpm needs to be forced

PXY_manual_install_wget: a shell script which downloads, from the LCG mirror site at CERN, the rpms included into the list *PXY_manual_install_rpm*

PXY_manual_install_extra_rpm: a short list of rpms which RH7.3 does not install by default, but which have been turned out to be required in order to have the SE software installed. This list is provided just for reference, since the rpms from this list are already included into the list *PXY_manual_install_rpm*

If any file of the mentioned list is not in the repository it means that it is not applicable within the installation of the current tag. So each further reference to it in this guide can be simply ignored.

All the above mentioned files are grouped in the archive file **PXY_tools.tar.gz**. So they can be easily downloaded and used on target machines.

The provided *PXY_manual_install_rpm* includes:

- all the packages which are needed in order to have a production Proxy.

- all the general purpose objects (i.e. shared libraries or common applications) which the PXY installation needs in order to run and which are not included in the standard operative system. The composition of this group of rpms may vary depending on what is considered "basic OS" at each site, and so it cannot be easily defined in a static way. Given the previous considerations, this part of the provided list may be not exhaustive.

A possible way to use the provided tools is decribed in the next sections

Download rpms from the repository in a local directory

- log in as *root* into the node to be installed and copy there the file **PXY_tools.tar.gz**.

- extract the archived files.

    > gunzip PXY_tools.tar.gz
    > tar xvf PXY_tools.tar

- give execution permissions and run the script *PXY_manual_install_wget*, which will then fetch all the rpms required into the subdirectories *PXY_INSTALL* and *PXY_INSTALL_FORCED*

    > cd PXY_tools
    > chmod u+x PXY_manual_install_wget
    > ./PXY_manual_install_wget

Install the downloaded rpms

- In the directory **PXY_INSTALL**, where the rpms listed in *PXY_manual_install_rpm* have been downloaded, run the command

  ```
  > rpm -Uvh "*.rpm"
  ```

- In the directory **PXY_INSTALL_FORCED** where the rpms listed in *PXY_manual_install_forced_rpm* have been downloaded, run the command

  ```
  > rpm -Uvh --nodeps "*.rpm"
  ```

  Due to possible missing rpms this operation might yield some missing dependencies which have to be analyzed and solved by adding/removing rpms to/from the *PXY_manual_install_rpm* list.

Should you prefer to do an upgrade of the PXY software already installed at your site instead of a new installation, the necessary steps are listed in the upgrade procedure usually reported in the last section of the PXY installation guide.

This procedure, when available, refers to changes occurring between the current version and the previous one. This also means that the procedure is not applicable if the software at your site is more than two tags old.

2.3. INSTALLING THE CERTIFICATION AUTHORITY RPMS

As the list and structure of Certification Authorities (CA) accepted by the LCG project can change independently of the middleware releases, the rpm list related to the CAs certificates and URLs has been decoupled from the standard LCG release procedure. You should consult the page

http://markusw.home.cern.ch/markusw/lcg2CAlist.html

in order to ascertain what the version number of the latest set of CA rpms is. At time of writing it is 1.6, so the full set of rpms can be downloaded with the following command;

```
> wget -l1 -nd -r http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/security/index_r1.6.html
```

These rpms should then be installed in the usual way;

```
> rpm -Uvh *.rpm
```

Changes and updates to these settings will be announced on the LCG-Rollout mailing list.

3. NTP CLIENT INSTALLATION & CONFIGURATION

A general requirement for the LCG nodes is that they should be synchronized. This requirement may be fulfilled in several ways. If your nodes run under AFS most likely they are already synchronized. Otherwise, you can use the NTP protocol with a time server.

Instructions and examples for a NTP client configuration are provided in this section. If you are not planning to use a time server on your machine you can just skip it.

3.1. NTP SOFTWARE INSTALLATION

In order to install the NTP client, you need the following rpms to be installed:

- ntp

- libcap

- libcap-devel

The following versions of the above said rpms have been proven to work on our OS configuration (the list includes the corresponding links to download sites):

- ntp-4.1.1-1

  ```
  http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/ntp-4.1.1-1.i386.rpm
  ```

- libcap-devel-1.10-8

  ```
  http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/libcap-devel-1.10-8.i386.rpm
  ```

- libcap-1.10-8

  ```
  http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/libcap-1.10-8.i386.rpm
  ```

A complete example of NTP rpms download and installation has been provided in 3.3..

3.2. NTP CONFIGURATION

- Configure the file */etc/ntp.conf* by adding the lines dealing with your time server configuration such as, for instance:

  ```
  restrict <time_server_IP_address> mask 255.255.255.255 nomodify notrap noquery
  server <time_server_name>
  ```

  Additional time servers can be added for better performance results. For each server, the hostname and IP address are required. Then, for each time-server you are using, add a couple of lines similar to the ones shown above into the file */etc/ntp.conf*.

- Edit the file */etc/ntp/step-tickers* adding a list of your time server(s) hostname(s), as in the following example:

```
137.138.16.69
137.138.17.69
```

- If you are running a kernel firewall, you will have to allow inbound communication on the NTP port. If you are using ipchains, you can do this by adding the following to */etc/sysconfig/ipchains*

```
-A input -s <NTP-serverIP-1> -d 0/0 123 -p udp -j ACCEPT
-A input -s <NTP-serverIP-2> -d 0/0 123 -p udp -j ACCEPT
```

  Remember that ipchains rules are parsed in order, so ensure that there are no matching REJECT lines preceding those that you add. You can then reload the firewall

```
> /etc/init.d/ipchains restart
```

- Activate the ntpd service with the following commands:

```
> ntpdate <your ntp server name>
> service ntpd start
> chkconfig ntpd on
```

  A complete example of NTP configuration has been provided in 3.3.

3.3. EXAMPLE OF NTP CLIENT INSTALLATION AND CONFIGURATION

This section reports a complete example of NTP server configuration done in the CERN context.

- install NTP

```
> wget http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/ntp-4.1.1-1.i386.rpm
> wget http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/libcap-devel-1.10-8.i386.rpm
> wget  http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/release/libcap-1.10-8.i386.rpm
> rpm -ivh libcap-1.10-8.i386.rpm libcap-devel-1.10-8.i386.rpm ntp-4.1.1-1.i386.rpm
```

- Configure the file */etc/ntp.conf*:

  the lines

```
restrict 137.138.16.69 mask 255.255.255.255 nomodify notrap noquery
server 137.138.16.69
restrict 137.138.17.69 mask 255.255.255.255 nomodify notrap noquery
server 137.138.17.69
```

  have been added to the file */etc/ntp.conf*

- Edit the file */etc/ntp/step-tickers*:

  the lines

  ```
  137.138.16.69
  137.138.17.69
  ```

  have been added to the file */etc/ntp/step-tickers*

- Edit */etc/sysconfig/ipchains* adding

  ```
  -A input -s 137.138.16.69 -d 0/0 123 -p udp -j ACCEPT
  -A input -s 137.138.17.69 -d 0/0 123 -p udp -j ACCEPT
  ```

  then reload the firewall

  ```
  > /etc/init.d/ipchains restart
  ```

- Activate the ntpd server

  ```
  > ntpdate ip-time-1.cern.ch
  > service ntpd start
  > chkconfig ntpd on
  ```

- You can check ntpd's status by running the following command

  ```
  > ntpq -p
  ```

4. LCG COMMON CONFIGURATION

4.1. SET-UP HOST CERTIFICATES

CE, SE, PROXY, RB nodes require the host certificate/key files before you start their installation.
Contact your national certification authority (CA) to understand how to obtain a host certificate and a private key if you do not have them already.
Relevant information on all the trusted CAs can be found in

http://marianne.in2p3.fr/datagrid/ca/ca-table-ca.html

which lists the CAs used by edg, from which you should choose a CA close to you.

Once you have obtained a valid certificate, i.e. a file

hostcert.pem

containing the machine host certificate and a file

hostkey.pem

containing the machine private key, make sure to place the two files into the directory

/etc/grid-security

with the following permissions

```
> chmod 400 /etc/grid-security/hostkey.pem

> chmod 644 /etc/grid-security/hostcert.pem
```

It is IMPORTANT that permissions be set as shown, as otherwise certification errors will occur!!!

4.2. CONFIGURE DYNAMIC LINKER RUN-TIME BINDINGS

If not already present, append the following lines to the file */etc/ld.so.conf*

```
/opt/gcc-3.2.2/lib
/opt/globus/lib
/opt/edg/lib
/usr/local/lib
```

- Run the command:

```
    > /sbin/ldconfig -v
```

(It produces a huge amount of output)

4.3. CREATE TOP-LEVEL EDG CONFIGURATON FILE

This file is parsed by EDG daemons to locate the EDG root directory and various other global properties.

Create and edit the file */etc/sysconfig/edg as follows*:

```
# Root directory for EDG software. (mandatory)
# Usual value: /opt/edg
EDG_LOCATION=/opt/edg

# Directory for machine-specific files.
# Usual value: $EDG_LOCATION/var

EDG_LOCATION_VAR=/opt/edg/var

# World writable directory for temporary files.  (mandatory)
# Usual value: /tmp
EDG_TMP=/tmp

# The directory containing trusted certificates and CRLs (CERTDIR).
# Usual value: /etc/grid-security/certificates


# Host certificate (X509_USER_CERT) for services which don't have their own.

# Usual value: /etc/grid-security/hostcert.pem

X509_USER_CERT=/etc/grid-security/hostcert.pem


# Host key (X509_USER_KEY) for services which don't have their own.

# Usual value: /etc/grid-security/hostkey.pem

X509_USER_KEY=/etc/grid-security/hostkey.pem
```

```
# Location of the grid mapfile (GRIDMAP).
# Usual value: /etc/grid-security/grid-mapfile

GRIDMAP=/etc/grid-security/grid-mapfile


# Location of the grid map directory for pooled accounts (GRIDMAPDIR).

# Usual value: /etc/grid-security/gridmapdir

GRIDMAPDIR=/etc/grid-security/gridmapdir/
```

4.4. CREATE TOP-LEVEL GLOBUS CONFIGURATON FILE

Create and edit the file */etc/sysconfig/globus* as follows:

```
GLOBUS_LOCATION=/opt/globus
GLOBUS_CONFIG=/etc/globus.conf
GLOBUS_TCP_PORT_RANGE="20000 25000"
```

4.5. CREATE GLOBUS CONFIGURATON FILE

Create and edit the file /etc/globus.conf as follows:

```
[common]
GLOBUS_LOCATION=/opt/globus
globus_flavor_name=gcc32dbg
x509_user_cert=/etc/grid-security/hostcert.pem
x509_user_key=/etc/grid-security/hostkey.pem
gridmap=/etc/grid-security/grid-mapfile
gridmapdir=/etc/grid-security/gridmapdir/
[mds]
globus_flavor_name=gcc32dbgpthr
user=edginfo


[mds/gris/provider/edg]
```

```
[mds/gris/registration/site]
regname=cerntestlcg2
reghn=<GRIS_NAME>


[gridftp]
log=/var/log/globus-gridftp.log


[gatekeeper]
default_jobmanager=fork
job_manager_path=$GLOBUS_LOCATION/libexec

jobmanagers="fork "

[gatekeeper/fork]
type=fork
job_manager=globus-job-manager
```

where <GRIS_NAME> is the machine on which the Proxy is registered so as to be visible via the GRIS configuration (to be explained later). This machine can for example be a Computing Element.

4.6. LAUNCH GLOBUS INIZIALIZATION SCRIPT

- Define "GLOBUS_LOCATION" environment variable (if needed)

```
> export GLOBUS_LOCATION=/opt/globus
```

- Launch setup script

```
> /opt/globus/sbin/globus-initialization.sh
```

This Globus script is meant to be launched on all the LCG nodes, as a general configuration step.
Because of this it can yield warning messages as well as some error ones. An example of output to be considered "normal" in a RB configuration has been produced in Appendix 1.
Since the script produces a huge output it is recommended to re-direct the standard output into a log file:

```
> /opt/globus/sbin/globus-initialization.sh > globus-init.log 2>&1
```

4.7. GSI ACCESS CONTROL LIST (GRID-MAPFILE)

The plain text file

/etc/grid-security/grid-mapfile

maps a GSI (Grid Security Infrastructure) Credential to a local user's login name.

The information in this file is verified and if needed periodically updated by the cron job

```
> /opt/edg/sbin/edg-mkgridmap
```

whose configuration file is

/opt/edg/etc/edg-mkgridmap.conf

If this in turn is not properly configured, the relevant information of the authorised users will not be fetched or updated, resulting in the node not working properly!!!

Instructions on how to edit edg-mkgridmap.conf can be found in [1].

5. POOL ACCOUNTS

The group "sshd" and the user "sshd" should be created as shown

```
> groupadd -g 74 sshd
> useradd  -M -u 74 -g 74 -d /var/empty/sshd sshd -s /bin/false
```

6. GRIS CONFIGURATION

This section deals with the configuration of the Gris which publishes information about the Proxy node, rendering it visible within the Grid.

6.1. CREATE GROUP AND USER *edginfo*

Create both the user and group *edginfo* on the CE.
The user home directory should be */opt/edg/var*

```
> groupadd -g 999 edginfo
> useradd -c "EDG Info user" -u 999 -g 999 -d /opt/edg/var edginfo
```

(the proposed choice of group ID and user ID is not mandatory)

6.2. INFO-PROVIDER.CONF

Create the directory */opt/edg/var/info/*

```
> mkdir /opt/edg/var/info
```

Create and edit the file */opt/edg/var/info/info-provider.conf* as follows

```
EDG_LOCATION=/opt/edg
GRID_INFO_USER=
REP_MET_PRESENT=no
REP_LOC_PRESENT=no
REP_OPT_PRESENT=no
SE_PRESENT=yes
```

6.3. LCG-INFO-WRAPPER

Create and edit the file */opt/lcg/libexec/lcg-info-wrapper* as follows

```
#!/bin/sh
/opt/lcg/libexec/lcg-info-generic /opt/lcg/var/lcg-info-generic.conf
```

after which the command

```
> chmod a+wxr /opt/lcg/libexec/lcg-info-wrapper
```

should be launched.

6.4. INFORMATION PROVIDER

The information provider needs to be configured. In order to do so create and edit

/opt/lcg/var/lcg-info-generic.conf as follows:

```
ldif_file=/opt/lcg/var/lcg-info-static.ldif
generic_script=/opt/lcg/libexec/lcg-info-generic
wrapper_script=/opt/lcg/libexec/lcg-info-wrapper
template=/opt/lcg/etc/GlueService.template

dn: GlueServiceURI= <NODE_NAME>:7512,Mds-vo-name=local,o=grid
GlueServiceAccessPointURL: <NODE_NAME>:7512
GlueServiceType: MyProxy
GlueServicePrimaryOwnerName: LCG
GlueServicePrimaryOwnerContact: grid-testbed-managers@host.invalid
GlueServiceHostingOrganization: <HOSTING_ORG>
GlueServiceMajorVersion: 1
GlueServicePatchVersion: 1
GlueServiceAccessControlRule: <BDII_NODE>
GlueServiceAccessControlRule: <RB_NODE>
GlueServiceInformationServiceURL: ldap://<NODE_NAME>:2135/mds-vo-name=local,o=grid
GlueServiceStatus: running
```

where <NODE_NAME> is the Proxy Node Name (FULL NAME!); <HOSTING_ORG> is the Hosting Organization Name; <BDII_NODE> is the full name of your BDII and <RB_NODE> is the full name of your Resource Broker.
WARNING: PLEASE MAKE SURE THAT THE END OF LINE CHARACTER IS PRESENT AT THE END OF THE LAST LINE. IF NOT THE INFORMATION WILL BE BADLY PROCESSED AND IN THE END DISREGARDED!!!

A production example follows:

```
ldif_file=/opt/lcg/var/lcg-info-static.ldif
generic_script=/opt/lcg/libexec/lcg-info-generic
wrapper_script=/opt/lcg/libexec/lcg-info-wrapper
template=/opt/lcg/etc/GlueService.template

dn: GlueServiceURI= adc0006.cern.ch:7512,Mds-vo-name=local,o=grid
GlueServiceAccessPointURL: adc0006.cern.ch:7512
GlueServiceType: MyProxy
GlueServicePrimaryOwnerName: LCG
GlueServicePrimaryOwnerContact: mailto:hep-project-grid-cern-testbed-managers@cern.ch
GlueServiceHostingOrganization: CERN-TEST-LCG2
GlueServiceMajorVersion: 1
GlueServicePatchVersion: 1
GlueServiceAccessControlRule: adc0009.cern.ch
GlueServiceAccessControlRule: lxshare0410.cern.ch
GlueServiceInformationServiceURL: ldap://adc0006.cern.ch:2135/mds-vo-name=local,o=grid
GlueServiceStatus: running
```

The script *lcg-info-generic-config* should then be launched in order to create the right configuration:

```
> /opt/lcg/sbin/lcg-info-generic-config  /opt/lcg/var/lcg-info-generic.conf
```

Notice that the file */opt/lcg/var/lcg-info-static.ldif* will be created as a result. This file SHOULD NOT BE EDITED MANUALLY.

6.5. INFO PROVIDER INITIALIZATION SCRIPT

This step is required so as to define which Information Provider should be used by the GRIS.
In order to do so launch the setup script

```
> /opt/edg/sbin/edg-info-provider-setup
```

and after that the globus initialization script

```
> /opt/globus/sbin/globus-initialization.sh > globus-init.log 2>&1
```

6.6. START THE INFORMATION SERVICE

```
> /sbin/chkconfig globus-mds on
> service globus-mds start
```

6.7. GRIS AND GIIS TEST

Test the GRIS and GIIS as follows:

```
> ldapsearch -h <PROXY-Hostname> -p 2135 -b "mds-vo-name=local,o=grid" -x
> ldapsearch -h <GIIS Node> -p 2135 -b "mds-vo-name=<your-registration-name>,o=grid" -x
```

NOTE CAREFULLY:
In order for the Proxy node to be *visible* the BDII has to contain the information relative to the GIIS node (e.g. a computing element) which also requires that the Proxy node be entered in the file */etc/globus* of the GIIS node itself.

Also notice that should there be a sizeable time difference between the Proxy node and BDII, the latter one will not update the Proxy information, which is therefore lost.

7. PROXY SPECIFIC CONFIGURATION

Edit the file

> /opt/edg/etc/edg-myproxy.conf

entering all the authorised renewers (e.g. Resource Brokers) which can renew the proxy validity.
A production example follows:

```
authorized_renewers "/C=CH/O=CERN/OU=GRID/CN=host/lxshare0410.cern.ch"
```

Run the script

> /etc/init.d/myproxy start

The script will create the file

> /etc/myproxy-server.config

which contains all the accepted credentials (taken from the directory */etc/grid-security/certificates/*) and authorized renewers (taken from the file */opt/edg/etc/edg-myproxy.conf*).

8. SERVERS START UP

Launch the following command:

```
> /opt/globus/sbin/myproxy-server -c /etc/myproxy-server.config --verbose
```

The server takes care to "listen" to any incoming requests, making sure to comply with those requests coming from accredited ones only.

9. CRON TABLE CONFIGURATION

The following lines should be added to the crontab:

```
PATH=/sbin:/bin:/usr/sbin:/usr/bin

28 1,7,13,19 * * * /opt/edg/etc/cron/edg-fetch-crl-cron
```

10. UPGRADE FROM THE PREVIOUS VERSION

This procedure is meant to help site administrators upgrade the node without starting an installation "from scratch", whenever possible.

The aim is to define those "delta" steps dealing both with installation and configuration phases. It is worthy of reminding that here "delta" refers to two contiguous software tags, the latter of which is the current one. Therefore, the procedure is not applicable to software versions more than two tags old.

WARNING: As a requirement for a successful upgrade, all the active servers which might be affected by the upgrade should be stopped.

The servers should be restarted once the installation and configuration described in the following sub paragraphs have been successfully done.

10.1. SOFTWARE INSTALLATION

The list "PXY_manual_install_rpm", which has been described in 2., contains the latest version of the PXY-related rpms to be installed. If the *rpm -i* command were launched as described in that section, a large number of conflicts would be issued because most of the packages would be reported to be already installed, some of which with the same version number, some others with an older one.

The easiest way to handle this situation is to download those rpms which should update the ones already installed on the node as well as any new ones.

The list of these rpms is contained in the file

PXY_upgrade_rpm

whereas the list with the complete command lines for the download is contained in the file

PXY_upgrade_wget

WARNING: This script will create a directory tree where the rpms are downloaded and needs to be launched once only.

After downloading the rpms, the first thing to do is to update the rpms within the directory *PXY_UPGRADE*, by issuing the command

```
> rpm -U --oldpackage "*.rpm"
```

A further list of rpms is provided, namely

PXY_upgrade_forced_rpm

containing those new rpms whose dependecies are not dealt with, therefore requiring the installation to be "forced".

The same "wget" file

PXY_upgrade_wget

takes also care to download the rpms of this final list in the directory

PXY_UPGRADE_FORCED

The rpms listed in this directory should be installed with the command

```
> rpm -Uvh --force --nodeps "*.rpm"
```

Finally the list of rpms *PXY_remove_rpm*

is provided which contains those packages which need to be removed; this is particularly likely to be necessary with CA-related rpms.
The rpm command to remove the listed rpms is

> rpm -e 'cat PXY_remove_rpm'

10.2. INSTALLING THE UP-TO-DATE CERTIFICATION AUTHORITY RPMS

As the list and structure of Certification Authorities (CA) accepted by the LCG project can change independently of the middleware releases, the rpm list related to the CAs certificates and URLs has been decoupled from the standard LCG release procedure. You should consult the page

http://markusw.home.cern.ch/markusw/lcg2CAlist.html

in order to ascertain what the version number of the latest set of CA rpms is. At time of writing it is 1.6, so the full set of rpms can be downloaded with the following command;

```
> wget -l1 -nd -r http://grid-deployment.web.cern.ch/grid-deployment/download/RpmDir/security/index_r1.6.html
```

These rpms should then be installed in the usual way;

```
> rpm -Uvh *.rpm
```

Changes and updates to these settings will be announced on the LCG-Rollout mailing list.

10.3. UPGRADE CONFIGURATION

No configuration change is needed to upgrade your Proxy from the previous tag to the current one.

CHANGE HISTORY

Table 1: Change History

version	date	description
v0.2	29/Mar/2004	
v1.0	5/Apr/2004	Final draft
v1.1	7/Apr/2004	Fully Working Version
v1.2.0	3/Aug/04	2.,10.: rpms dealing with CAs no more included in rpm lists.