Louis Poncet
System Administration
12 Oct 2009

GRID SITE SYSTEM ADMINISTRATION

THIS DOCUMENT IS STILL A DRAFT AND FOR NOW ONLY COVER THE CERTIFICATION TESTBED FOR EGI PROJECT.

Introduction

A grid site is a computing center running computing resources and storage resources with a GRID middleware to reach them. The target of this document is to explain how to install and design a grid site without using any fabric management software. This method permit to have a totally neutral method that can be integrate in any Fabric Management software.

Fabric management software are used in computing centre to install software, manage updates of the Os and of the software, manage the configurations : users, authorization shared file system ... We can take as example quattor, CFengine and lot others. We need a document for installation without a fabric management software to be compatible with all of them.

The middleware installation is complex and need a large knowledge in the middleware that you have to install. A site is mainly a set of resources for calculation and storage, the biggest amount of machines are worker nodes and storage machines. We can imagine to automatize completely the installation configuration of those two types of resources and managing the installation of supervisor nodes by hand, also in large CC.

Tools

What do we need to do, concretely we need to be able to send command on multiple machines in parallel, we need to be able to share the configuration files in a secure way, manage our users and machine x509 certificates. We need to test the basic functionalities and monitor our resources. The process for installation is : installation of the Os, installation of the middleware, configuration of the middleware, basic testing and configuration of the monitoring system to add this new resources.

For the Operating system installation we need to install a network based installation system i recommend to use the standard one provide by the Operating system. In case of redhat based Os , we need install a PXE, TFTP and NFS server as installation service. The description of the machine settings used Kickstart description process by anaconda the Redhat installation software

Official redhat configuration documentation for network installation : http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/sysadmin-guide/pt-install-info.html

Operating system installation

Service Nodes

Services nodes are all the nodes expect WNs and User Interfaces. For those resources a basic Linux installation is require. We don't need to install extra packages for user usage like developments tools, users libraries. The service nodes are maintainable by hand for sites small sites. The Os will contain only basic required packages and system administration.

Kistart configuration file packages list :
%packages
@base
@Administration Tools

Worker nodes and UI

The UIs are installed by the users or as an environment settings for central linux services. For the WNs we have a cluster of machines with the same configuration. Those nodes system configuration has to match the customer (user) request. Development tools, libraries specific user commands and other settings.

Installation and configuration for small site and testing oriented sites

To install small sites we just need to follow line per line the generic installation guide. The two problematic point in the install guides are the two copyrighted packages, java and oracle instant client.

How to install

- Os installation
- Host certificates creation and installation
- Getting the .repo configuration files for yum
- Installation of the meta package or the group for the node
- Configuration of the service trough YAIM
- Test of the service
- Monitoring reconfiguration to add this resource

Installing resources for the shared testbed in EGI

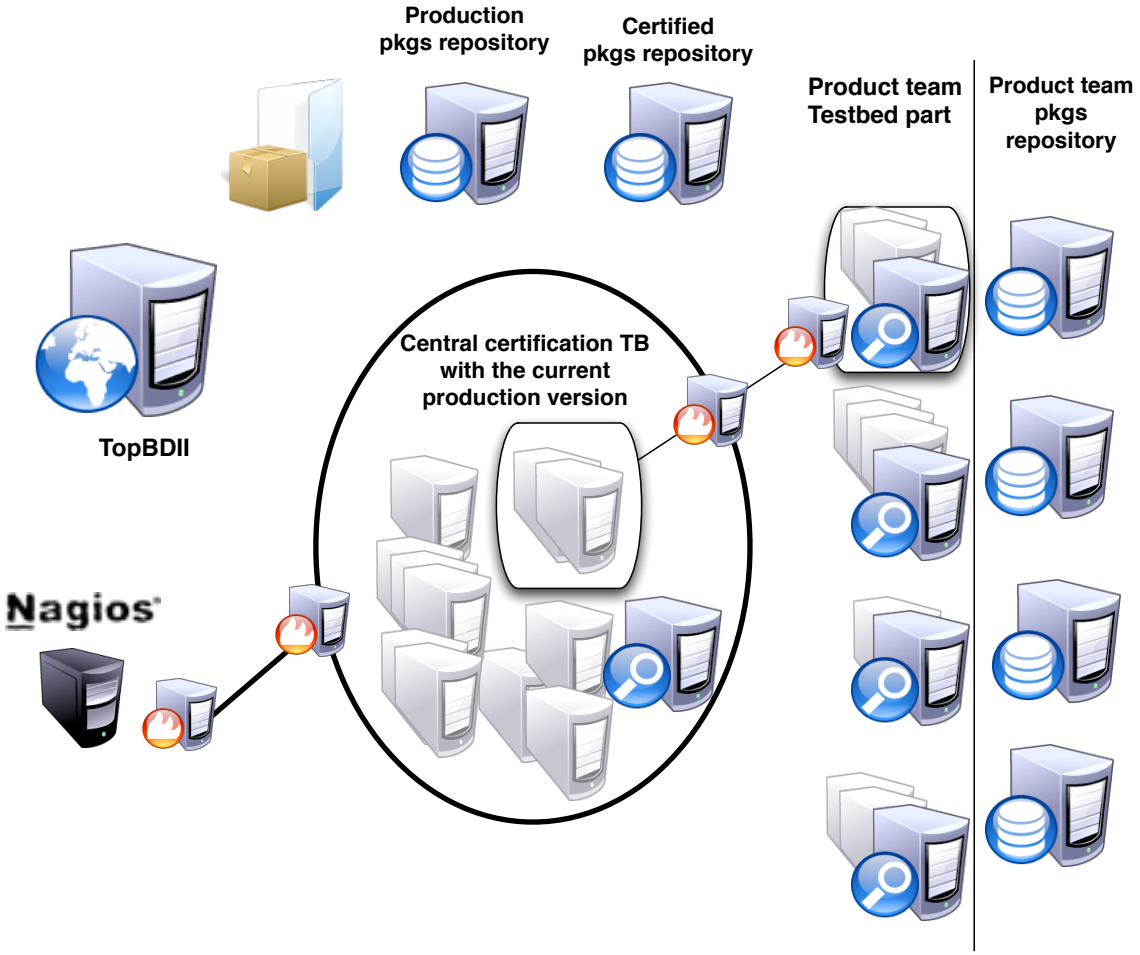
Product team provider of service

Product Name (Lead Manager & Partner)	Software Components (Partners)	Allocated Node Type(s)
Authorization (Christoph Witzig, SWITCH)	Authz Service (SWITCH, HIP, INFN, NIKHEF) Shibboleth interoperability (SWITCH)	ARGUS SLCS_client
VO Management (Vincenzo Ciaschini, INFN)	VOMS (INFN) VOMSAdmin (INFN)	VOMS
Security Infrastructure Product Team (John White, HIP)	Delegation Framework (CERN, HIP, STFC) Trustmanager (HIP) Util-Java (HIP) Hydra (HIP) DICOM (HIP) myProxy Integration (HIP) LCAS/LCMAPS (NIKHEF) glExec (NIKHEF) SCAS (NIKHEF) Gridsite (STFC)	Hydra PX SCAS GLEXEC_wn
Information Systems (Laurence Field, CERN)	BDII (CERN) GLUE Schema (CERN)	BDII
Compute Element (Massimo Sgaravatto, INFN)	CREAM (INFN) CEMon (INFN) BLAH (INFN)	CREAM
Job Management (Marco Cecchi, INFN)	WMS (INFN, ED)	WMS
Logging & Bookkeeping (Ales Krenek, CESNET)	Proxy and attribute certificate renewal (CESNET) Logging &Bookkeeping (CESNET) Gsoap-plugin (CESNET)	LB
Data Management (Ákos Frohner, CERN)	CGSI_gSOAP (CERN) DPM (CERN) GFAL /lcg_util (CERN) LFC (CERN) FTS (CERN)	FTS (various) DPM (various) LFC
Integrated Clients (Andreas Unterkircher, CERN)	Proxy Renewal (Elisa @ ???) GSI-SSH (External - TBC)	UI WN VO Box
Batch System Integration (Jan Just Keijser, NIKHEF)	Torque SA3, NIKHEF) LSF – unsupported Condor(PIC) SGE (CESGA)	<LRMS>_utils TORQUE_server
MPI (John Walsh, TCD)	MPI Task Force	MPI
dCache (Patrick Fuhrmann, DESY)	dCache (DESY)	dcache (various)
AMGA (Soonwook Hwang, KISTI)	AMGA (KISTI)	AMGA

All Product will have to produce a GIP string to register on the central siteBDII. All informations for the configuration to create the required site-info files for YAIM will be maintain by SA3 getting the informations is a safe way with and web-based text editor. The firewall set-up will be really complex, On those resources that are on different network and need ports open to some type of nodes but not the other. Each team has to create from the site-info information the firewall settings per node type for there computing centre.

The final list of product team and the way to test the Batch system is totally clear today, it can be each Batch system team manage is own CE or CREAM CE team provide 1 CREAM CE per Batch system team, at least 1 CREAM CE and 1 Batch server should be in the main testbed.

Shared testbed architecture



In this graphic a representation of the testbed site

The main testbed is the reliable reference site, each expert in products have the hand on his own resource. Each resources are located in each CC and we have to set a precise firewall settings to manage the fact that we are not on the same site. Or we can localize the resources is one place and give admin keys per node type per product team.

Configuration

YAIM configuration has to be coordinate, each product team has to provide the required information about the resources that its provide. The twiki page will be fulfilled by each product team to provide those information and others, then we can have a web page editor like the one use for the BDII to merge configuration informations. https://twiki.cern.ch/twiki/bin/view/EGEE/EgiTestbed has to be filled following the template in it by each product team.

Graphic of the bdii web editor

The coordination place to merge site-info information and BDII configuration should be done by the teams that are responsible for those two products. In this screenshot we have the configuration for the site, cert-tb-cern the Top BDII on the Testbed and the files site-info, site-info-wms and myproxy one.

Security issue

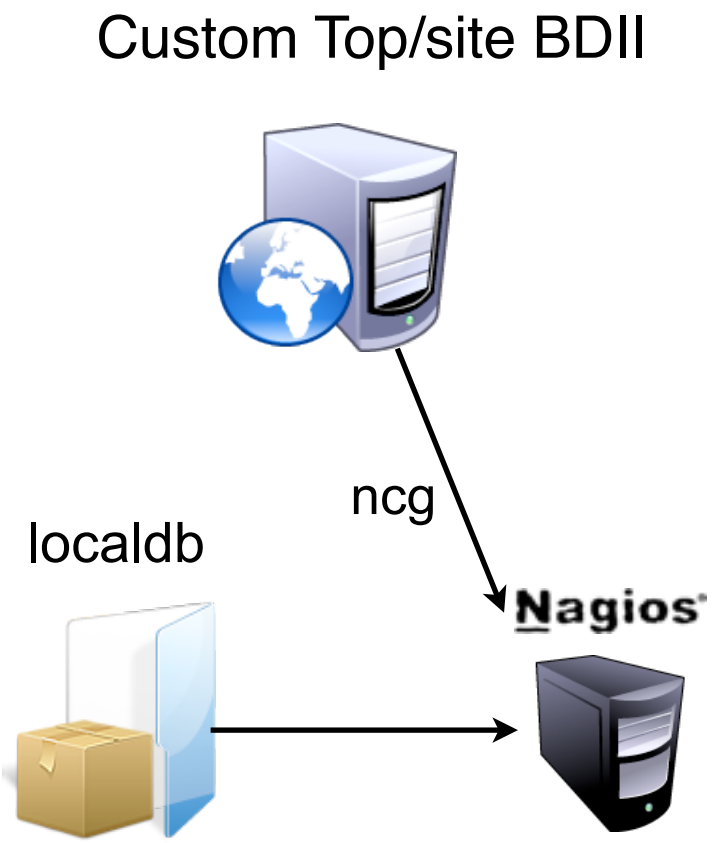
All resources on the testbed require a proper firewall setting to allow external hosts to reach the resources that are normally on site. A good example is the siteBDII normally the port 2170 of resources machines does not allow external connection on this port, for our setting the siteBDII maintain by it associate product team will have to be allow to reach those resources, each product team will have to allow the proper port to the proper hosts. The siteBDII team will have to give information about that each product team will have to fill its part in the twiki about firewall setting.

NAGIOS monitoring

For the main testbed we need a reliable monitoring system, the testbed is a site. We can use the standard NAGIOS setting provide by the WILCG Nagios team. To do this configuration we can redo a GOC db like but we can also use the options provide by NCG. NCG is the tool develop by the monitoring team to configure NAGIOS, it can take its source from 3 differents "database", GOCDB, BDII and the localdb (a text file).

In our case we can use the BDII + the local db, We take a snapshot of the main testbed when we consider it as ok, we just add the metric check_yum in the localdb with all the hosts used for the certification.

NAGIOS LCG team : https://twiki.cern.ch/twiki/bin/view/EGEE/OAT_EGEE_III



NCG extracting data from a BDII and localdb

The NCG localdb can permit us to install easily new metrics before giving them to the Nagios monitoring team for a generic integration, this method permit also for each product team to develop its own NAGIOS sensor.

For example in this piece of configuration file ncg.conf :

SITENAME = cert-tb-cern

MAIN_DB_FILE=/etc/ncg/ncg.localdb

BDII = lxbra2306.cern.ch <<< SITE BDII

GLITE_VERSION=3.1.0

.....

<LDAP>

LDAP_ADDRESS=$BDII

ADD_HOSTS=1

</LDAP>

This is using the BDII as db to get the machines info. But i need also basic monitoring (like a ping). For that i am using the ncg-localdb, this file permit to add or remove settings of bcg. In this example i define the rest of my resources as host (a regular ping).

HOST_SERVICE!lxb7608v1.cern.ch!Host

HOST_SERVICE!lxb7608v2.cern.ch!Host

HOST_SERVICE!lxb7608v3.cern.ch!Host

HOST_SERVICE!lxb7608v4.cern.ch!Host

HOST_SERVICE!lxb7608v5.cern.ch!Host

HOST_SERVICE!lxb8075.cern.ch!Host

HOST_SERVICE!lxb8076.cern.ch!Host

HOST_SERVICE!lxbra1910.cern.ch!Host

Using the testbed for certification of new version

All the settings that we did before to have a stable testbed permit to have a reference to complete the resources that we need to certify one product without installing everything, i don't want to install a WMS to run the certification of a Batch system, but i need one. I'll use the one from the main testbed.

The pkgs repository available will be :

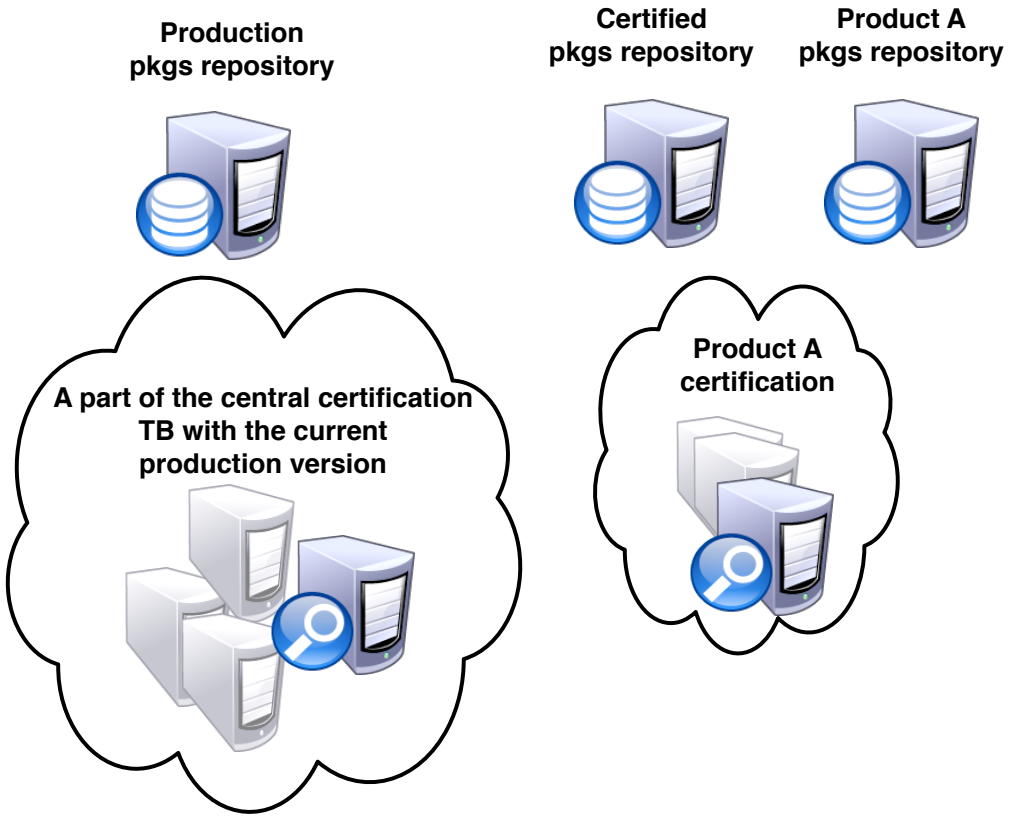
- Production
- Internal

With this set each product team can manage the main testbed stable nodes with the production repository, and adjust his settings for the next release of its product. For example we can have a certification of product A with its own new packages + internal + production to install the node that will be certify.

Certification testing architecture

The certification of a new version of a product needs resources that are not only in the node type directly concern by the product.

How can i design a testbed with my choice of resources to put in ?

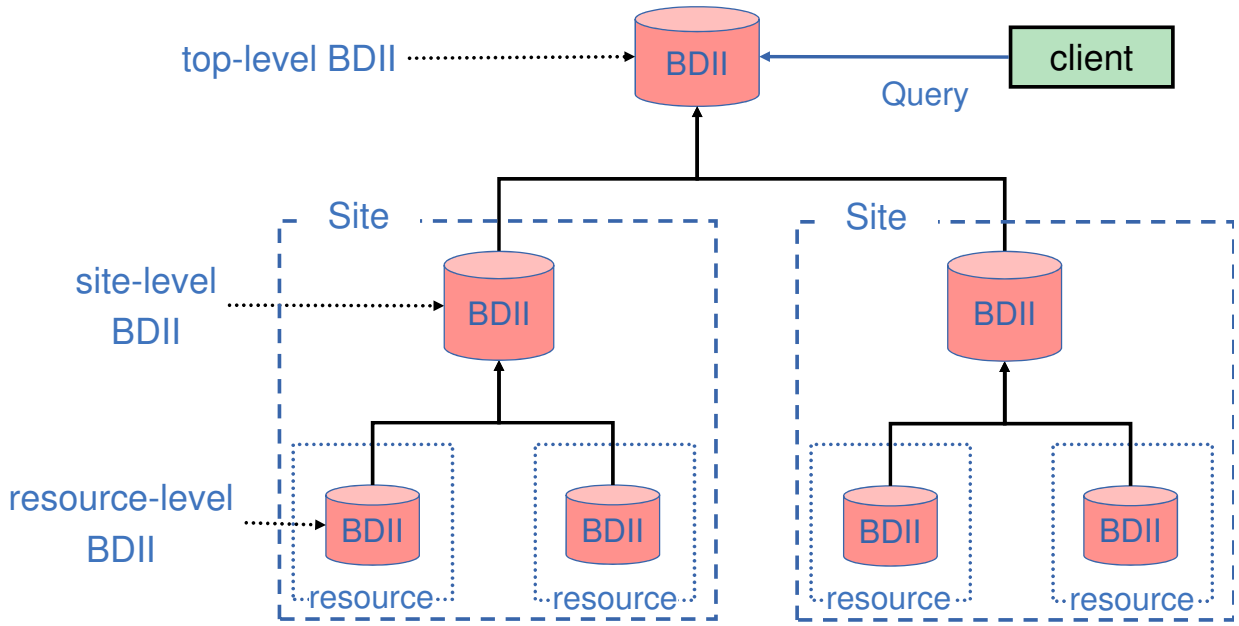


List of what other products i need for my certification, pick up from the testbed resources that i need or integrate my new resources, i can create and create my own BDII or insert my new resources on the stable BDII (cf. Informations system adjustment).

The architecture require for the certification is ready, i can now run my test with the new version of the middleware on one host and the stable still available.

Informations system adjustment

A site is a set of resources available from a UI and visible trough BDII. The BDII is the place where will be able to design our special site for a certification. The site BDII get information from node BDII on the site and the top BDII get information from the site BDIIs.



The configuration is a set of ldap strings, if we take the top BDII of the certification testbed site BDII we have :

Site A BDII Conf File

CE ldap://lxbra2307.cern.ch:2170/mds-vo-name=resource,o=grid

CTBSA3INFN ldap://cream-37.pd.infn.it:2170/mds-vo-name=resource,o=grid

BDII ldap://lxbra2306.cern.ch:2170/mds-vo-name=resource,o=grid

CREAMCE ldap://lxbra2308.cern.ch:2170/mds-vo-name=resource,o=grid

SE ldap://lxbra1910.cern.ch:2170/mds-vo-name=resource,o=grid

DPM ldap://lxb7608v1.cern.ch:2170/mds-vo-name=resource,o=grid

LFC ldap://lxb7608v3.cern.ch:2170/mds-vo-name=resource,o=grid

PX ldap://lxbra2304.cern.ch:2170/mds-vo-name=resource,o=grid

FTS ldap://lxbra2310.cern.ch:2170/mds-vo-name=resource,o=grid

VOBOX ldap://lxb7607v2.cern.ch.cern.ch:2170/mds-vo-name=resource,o=grid

Let's take as example the certification of the new version of the SE dpm, i need 2 new SEs on the testbed a remove the production one for my test. I will create my BDII to set what i need and use it for my test, the configuration in this case for my custom BDII will be :

```
# Site A BDII Conf File
CE  ldap://lxbra2307.cern.ch:2170/mds-vo-name=resource,o=grid
CTBSA3INFN ldap://cream-37.pd.infn.it:2170/mds-vo-name=resource,o=grid
BDII  ldap://lxbra2306.cern.ch:2170/mds-vo-name=resource,o=grid

CREAMCE  ldap://lxbra2308.cern.ch:2170/mds-vo-name=resource,o=grid
LFC  ldap://lxb7608v3.cern.ch:2170/mds-vo-name=resource,o=grid
PX  ldap://lxbra2304.cern.ch:2170/mds-vo-name=resource,o=grid
FTS  ldap://lxbra2310.cern.ch:2170/mds-vo-name=resource,o=grid
VOBOX  ldap://lxb7607v2.cern.ch.cern.ch:2170/mds-vo-name=resource,o=grid
SE ldap://HOSTNAME_OF_THE_NEW_SE_1:2170/mds-vo-name=resource,o=grid
SE2 ldap://HOSTNAME_OF_THE_NEW_SE_2:2170/mds-vo-name=resource,o=grid
```

This new BDII will not list the production SE but list the news ones HOSTNAME_OF_THE_NEW_SE_1 and HOSTNAME_OF_THE_NEW_SE_1.

The resource BDII of each nodes is protect by a firewall for external access only the site and top BDII are reachable from any site. In this case we need to create some information provider specific to extract the information that we need from the site BDII of another site. We need an information provider. A GIP (Grid Information Provider) is a little software that publish the informations that the BDII will publish following the GLUE schema format.

For example i need SEs for my testing and i want to add them to my custom BDII. If i want to get this list using an ldapsearch request i'll do :

```
ldapsearch -x -LLL -H "ldap://siteBDII.cern.ch:2170" -b "Mds-Vo-name=SITE_NAME,o=grid" objectClass=GlueSE
```

This request will extract the SE from the site SITE_NAME using the site BDII siteBDII.cern.ch the option -LLL permit to respect the format require by the BDII, we have the information that we want and from that we'll create a information provider for my site, I'll also need to replace SITE_NAME by MY_SITE_NAME. The request will be :

```
ldapsearch -x -LLL -H "ldap://siteBDII.cern.ch:2170" -b "Mds-Vo-name=SITE_NAME,o=grid" objectClass=GlueSE | sed 's/SITE_NAME/MY_SITE_NAME/g'
```

I just create a shell script with this string and i copy it in /opt/glite/etc/gip/provider of my site BDII, the new resources are now available.

To monitor my new site i have to re-run ncg with the BDII address and it will index all the resources list in the BDII (that ncg can index). Restart NAGIOS few minutes later you know the state of your "new" site.

Conclusion What do we need.

- To certify all type of resources we need a set of test user member of different VO resister in one of the main testbed VOMS, a CA for our purpose increase the security for the environment If this CA is compromised we don't need to pass by IGTF to revocate the certificate of it. It is natural that the security team responsible for VOMS handle that.

- A sysadmin per product team able to set correctly the BDII for the stable testbed and to plug there resources for certifications.

- Monitoring system per site