XtreemOS

Integrated Project
BUILDING AND PROMOTING A LINUX-BASED OPERATING SYSTEM TO SUPPORT VIRTUAL
ORGANIZATIONS FOR NEXT GENERATION GRIDS

Installing XtreemOS on a Virtual Machine

XtreemOS Technical Report # 6

Yvon Jégou[a]

Report Registration Date: October 26, 2010

Version 1 / Last edited by Yvon Jégou / October 26, 2010

Project co-funded by the European Commission within the Sixth Framework Programme		
Dissemination Level		
PU	Public	√
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

[a]Yvon.Jegou@inria.fr

Revision history:

Version	Date	Authors	Institution	Section affected, comments
0.1	29/09/10	Yvon Jégou	INRIA	Initial document

Contents

List of Figures

1   Introduction

This document describes, step by step, all actions necessary to create an XtreemOS Virtual Machine Grid from the 2.1.2 ISO with KVM and libvirt tools. A core node is first installed and configured. The second part of the document shows the installation of a resource node from the ISO and its configuration from the previously installed core node. In the last part, this resource is cloned to produce a second resource node.

2   VM creation and ISO installation

2.1   Configuring Virtual Machine networking in KVM

It is possible to define the networking environment of virtual machines in KVM. In this document, we consider VMs running with private IP addresses and accessing Internet through NAT. VMs can be configured with static IP addresses or can get this address from KVM using DHCP. Here is a sample network configuration file for KVM:

```
yjegou@host:~$ sudo virsh net-dumpxml default
<network>
  <name>default</name>
  <uuid>f7c4b917-29ae-7614-1a3e-8c542f5c7205</uuid>
  <forward mode='nat'/>
  <bridge name='virbr0' stp='on' forwardDelay='0' />
  <ip address='192.168.122.1' netmask='255.255.255.0'>
    <dhcp>
      <range start='192.168.122.2' end='192.168.122.254' />
      <host mac='54:52:00:00:10:00' ip='192.168.122.10' />
      <host mac='54:52:00:00:10:01' ip='192.168.122.11' />
      <host mac='54:52:00:00:10:02' ip='192.168.122.12' />
      <host mac='54:52:00:00:10:03' ip='192.168.122.13' />
      <host mac='54:52:00:00:10:04' ip='192.168.122.14' />
      <host mac='54:52:00:00:10:05' ip='192.168.122.15' />
    </dhcp>
  </ip>
</network>
```

You can edit this configuration using `virsh net-edit`. First, stop the network with the command:

```
yjegou@host:~$ sudo virsh net-destroy default
```

Then, edit the configuration:

```
yjegou@host:~$ sudo virsh net-edit default
```

Finally, restart the network:

```
yjegou@host:~$ sudo virsh net-start default
```

The KVM host automatically gets a local IP address in this network. This address can be used by VMs as a gateway to Internet and as a DNS server. The host can communicate with the local VMs using this interface.

```
yjegou@host:~$ ifconfig
...
virbr0    Link encap:Ethernet  HWaddr 1e:47:15:a8:9a:2a
          inet addr:192.168.122.1  Bcast:192.168.122.255  ...
...
```

The local DNS resolves names from the host `/etc/hosts` file and from the host resolver. So, it is enough to declare all virtual machines in the host `/etc/hosts` file:

```
yjegou@host:~$ cat /etc/hosts
127.0.0.1 host localhost.localdomain localhost
127.0.1.1 host

192.168.122.10   xos-core.xtreemos.eu xos-core
192.168.122.11   xos-node1.xtreemos.eu xos-node1
192.168.122.12   xos-node2.xtreemos.eu xos-node2
192.168.122.13   xos-node3.xtreemos.eu xos-node3
192.168.122.14   xos-node4.xtreemos.eu xos-node4
```

Figure 1 shows the network configuration in the Virtual Machine Manager when it is connected (right-click on `localhost (QEMU)` and select `Connect`); and then from menu `Edit > Host Details`, select the tab `Virtual Networks`.
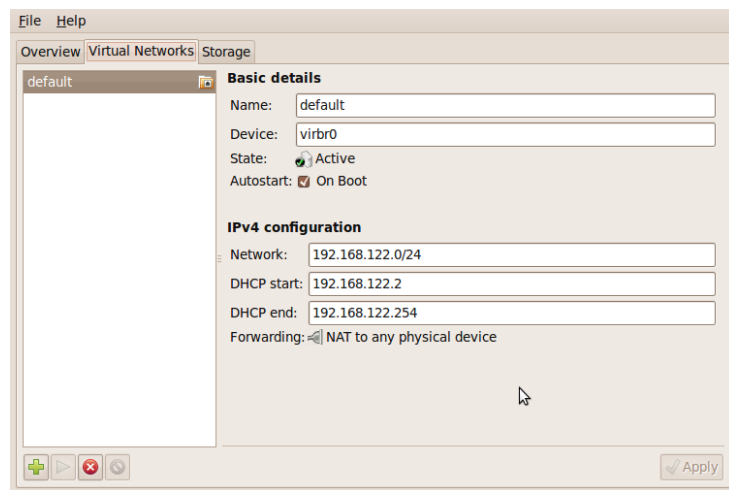
Figure 1: The default network configuration of KVM in the manager panel

2.2   Installing the 2.1.2 ISO

The installation of XtreemOS in a VM from an ISO can be started either from command line (cf Section 2.2.1) or from graphical interface (cf Section 2.2.2).

2.2.1   Command line installation

If you want to spare some disk space, you may want to define a sparse disk image using the qcow2 format.

```
yjegou@host:~$ qemu-img create -f qcow2 xos-core.img 10G
Formatting 'xos-core.img', fmt=qcow2 size=10737418240 encryption=off cluster_size=0

yjegou@host:~$ virt-install  --connect qemu:///system --name xos-core  --ram 800\
 --os-type linux  --os-variant mandriva2009  --cdrom /path/to/XtreemOS-2.1.2-x86_64.iso\
 --disk xos-core.img  --network network=default,mac=54:52:00:00:10:00
```

Or you may directly create the VM, the raw disk image (note the required `size` argument in the `disk` parameter) and boot on the ISO image.

```
yjegou@host:~$ virt-install --connect qemu:///system  --name xos-core  --ram 800\
 --os-type linux  --os-variant mandriva2009  --cdrom /path/to/XtreemOS-2.1.2-x86_64.iso\
 --disk xos-core.img,size=10  --network network=default,mac=54:52:00:00:10:00
```

When the Virtual Manager Viewer starts, it displays the interactive installer and you can continue the installation procedure from Figure 11.

2.2.2 Graphical installation with the Virtual Machine Manager

In this section, we consider the installation of a virtual machine using the graphical interface of the Virtual Machine Manager. The installation is also possible using the command line interface.
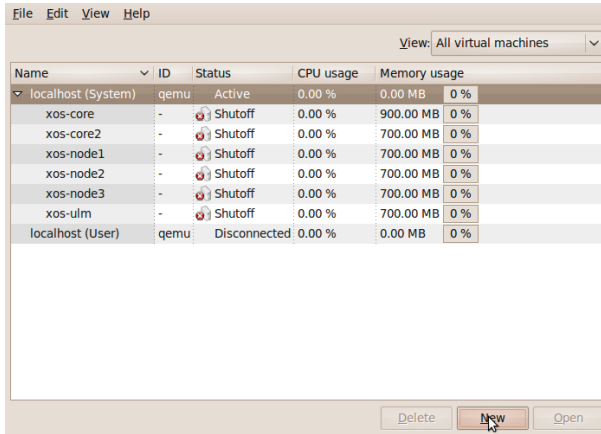
Figure 2: Create a new virtual machine
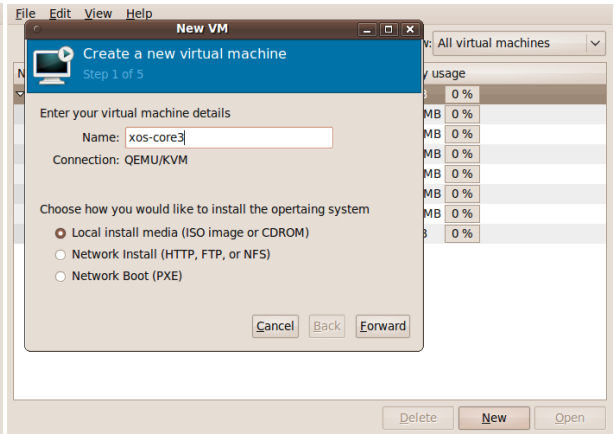
Figure 3: Configure new VM
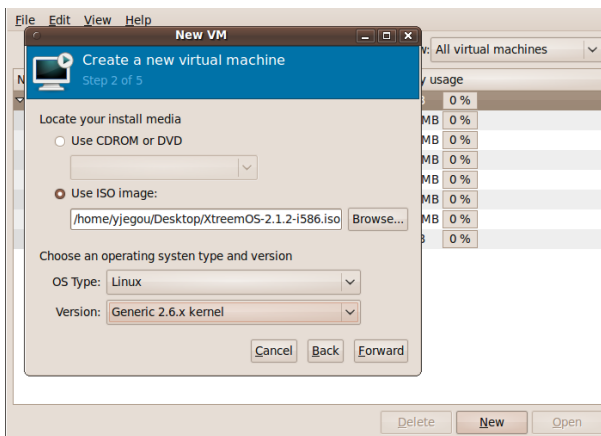
Figure 4: Configure installer
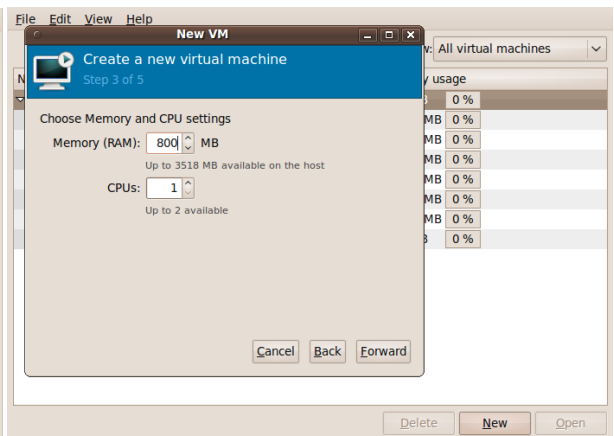
Figure 5: Configure VM resources
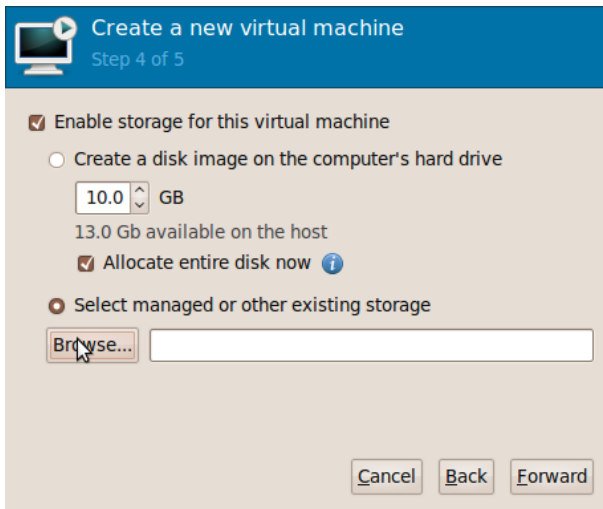
Figure 6: Configure VM image
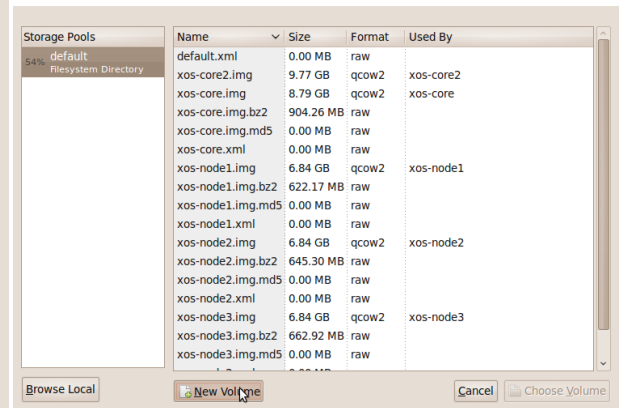
Figure 7: Create VM disk image
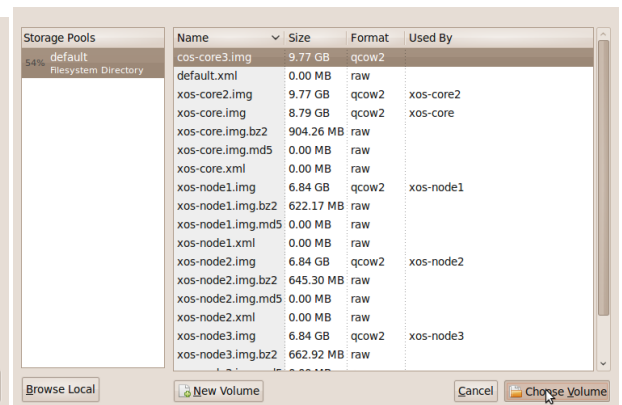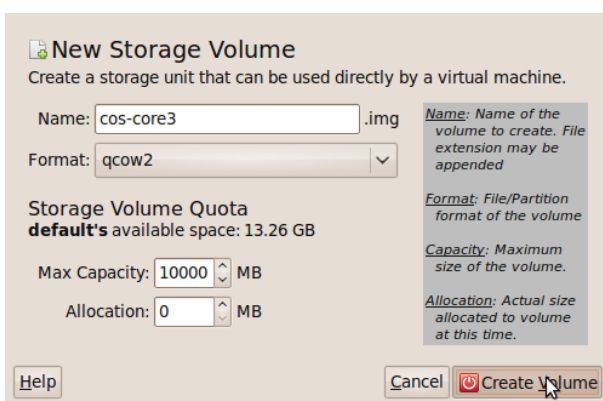
Figure 8: Create VM Storage (see Section 8.1) and select this volume for installation
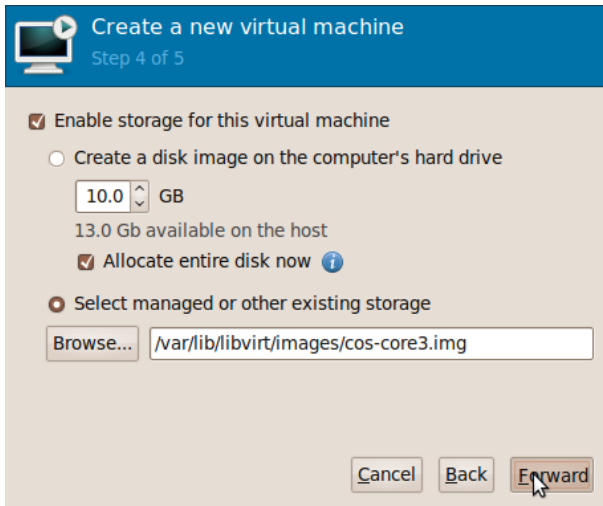
Figure 9: Configure disk size and path
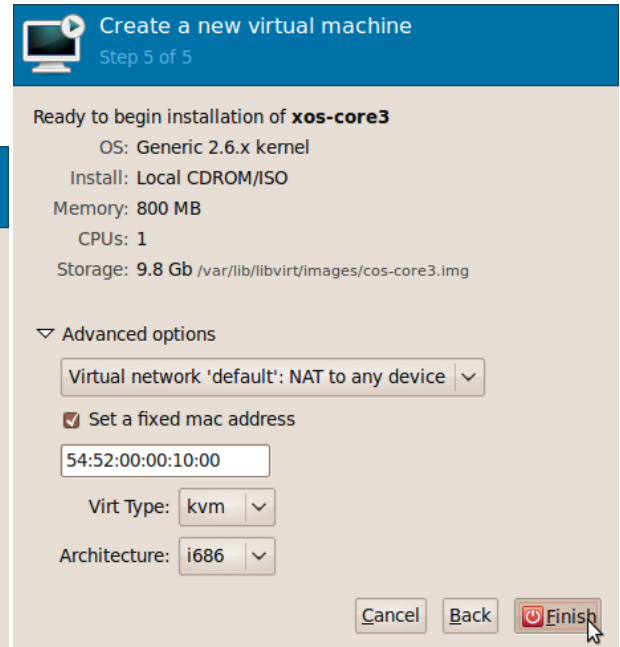
Figure 10: Define networking options (NAT), set a fixed MAC address and start installation
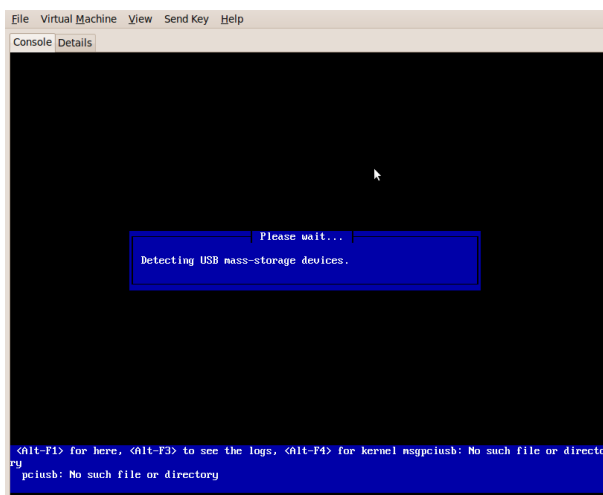
Figure 11: Installer is booting. . .
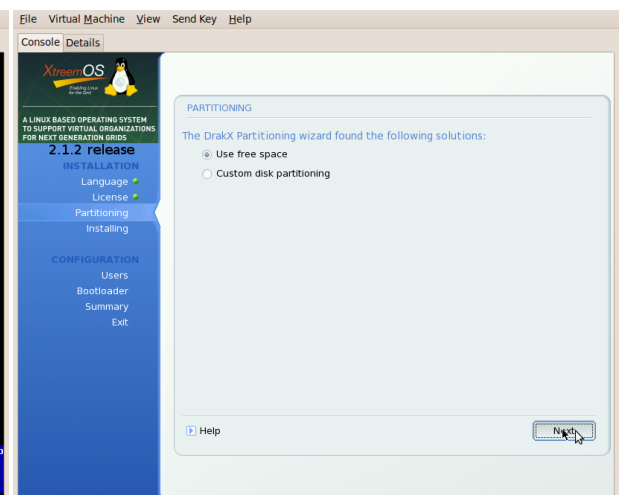
Figure 12: VM disk partitioning
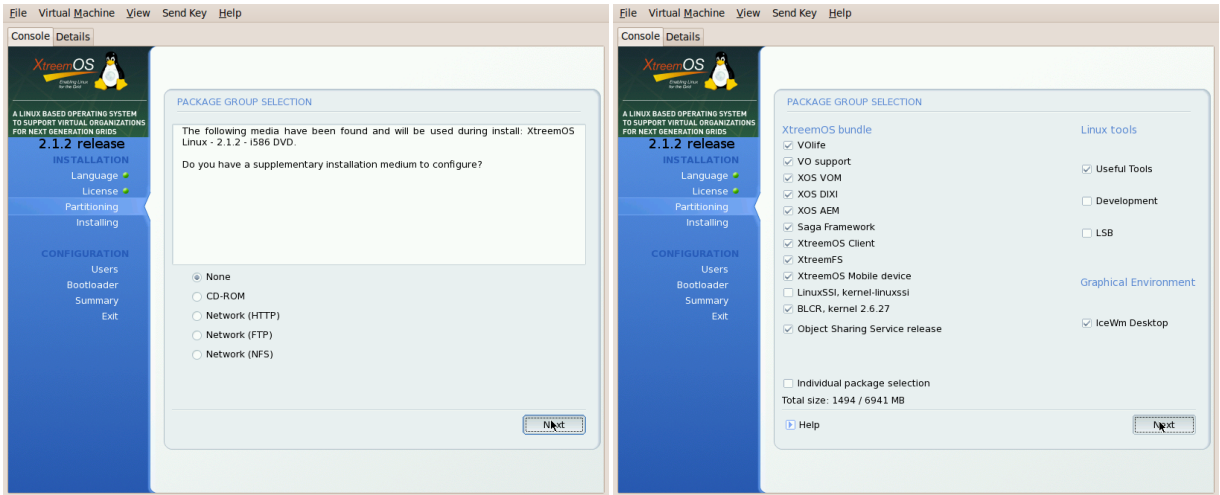
Figure 13: Media and package selection. For this core+resource node, all packages are installed. Note that it is also possible to de-select all XtreemOS packages at this step, since, later, the `xosautoconfig` tool will install all necessary packages depending on the chosen configuration.
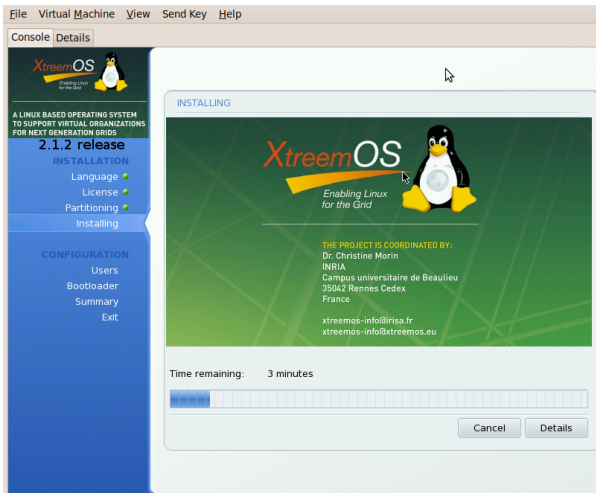
Figure 14: Installation is starting...
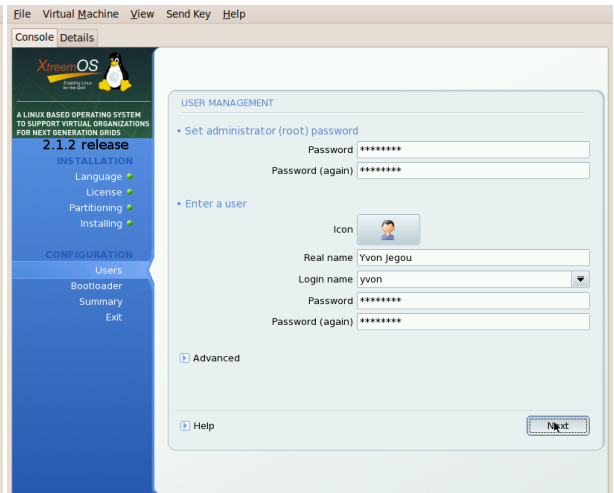
Figure 15: Users configuration: root and user
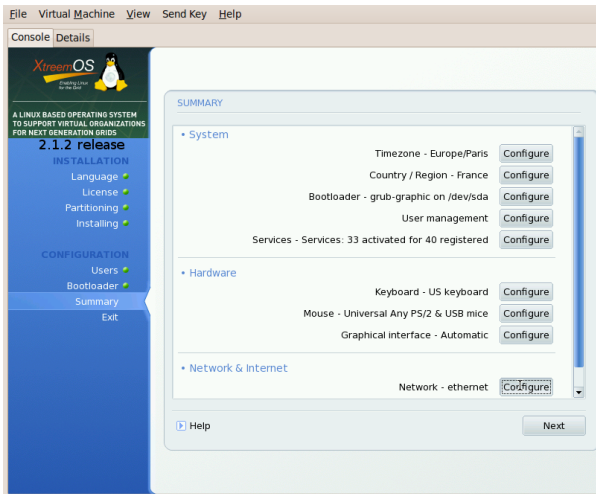
Figure 16: Operating system configuration: timezone, country, services, networking
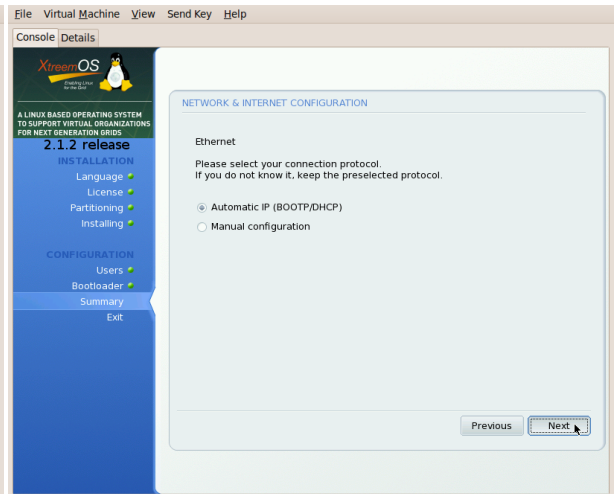
Figure 17: Operating system configuration, networking: configure for DHCP. It is also possible to statically configure the network at this step.
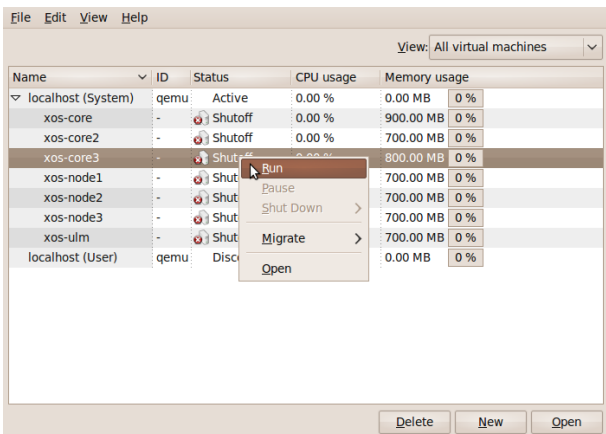
Figure 18: Installation is terminated, the installer has shut down. If the VM does not automatically reboot, it can be started from the manager window. After reboot, it is possible to log in as root or as the default user using this console.

3 XtreemOS core node configuration

The following steps detail the configuration of the core node from a terminal on the VM host. This method is possible using KVM as the host gets an IP address on the same network as the client VMs. Note that it is also possible to log in and configure the node from the console.

3.1 First connection as simple user

The default configuration of XtreemOS does not allow `root` to log in through `ssh` using a simple password. In this document, we will first log in as the normal user defined during the installation process.

```
yjegou@host:~$ ssh yvon@xos-core
yvon@xos-core's password:
[yvon@xos-core ~]$
```

For possible errors or warnings, see Section 8.2 and Section 8.3.

3.2 **SSH keys initialization**

Once logged in on the node as normal user, configure `SSH`.

```
[yvon@xos-core ~]$ mkdir .ssh
[yvon@xos-core ~]$ chmod g-w .ssh
```

The second line is important as using the default configuration of Mandriva, directories are created with write access to group, and `SSH` strictly controls access rights on configuration files.

Copy the user `SSH` public key:

```
yjegou@host:~$ scp /home/yjegou/.ssh/authorized_keys2  yvon@xos-core:.ssh/
yvon@xos-core's password:
authorized_keys2                               100%  397    0.4KB/s   ...
```

3.3 **root account setup**

```
[yvon@xos-core ~]$ su -
Password:
[root@xos-core ~]# mkdir .ssh
[root@xos-core ~]# chmod g-w .ssh
[root@xos-core ~]# cp ~yvon/.ssh/authorized_keys2 .ssh
```

3.4 Grid certificates

A basic set of certificates is necessary to operate a simple XtreemOS grid: a root certificate as well as service certificates for `cda`, `vops`, `rca` and XtreemFS services. This document does not detail how to run XtreemFS with certificates. You can either get a copy of an existing set of certificates or generate your own set yourself.

3.5 Generating your own set of certificates

The XtreemOS environment provides packages `rootca-config` for installing and configuring a root certificate authority and `create-csr` for managing certificate requests.

3.5.1 Certificate management packages

Packages `rootca-config` and `create-csr` are located on the XtreemOS mirrors and are not installed by default. As the virtual machine has been installed from a DVD ISO, it is necessary to first reconfigure the XtreemOS package management system to fetch new packages from the Mandriva mirrors. One solution is to execute `xosautoconfig` as following:

```
[root@xos-core ~]# xosautoconfig --linuxonly
```

Parameter `--linuxonly` limits the node configuration to the standard Linux part of the system. No XtreemOS service is started. By the way, `xosautoconfig` initiates mirror selection process for Mandriva repositories. See Section 8.4 in case of problems.

Another solution is to manually setup the package management system using the following commands:

```
[root@xos-core ~]# urpmi.removemedia -a
[root@xos-core ~]# urpmi.addmedia --distrib http://ftp.free.fr/mirrors/\
ftp.mandriva.com/MandrivaLinux/official/2009.0/i586/
[root@xos-core ~]# urpmi.addmedia --wget xtreemos http://ftp.free.fr/mirrors/\
ftp.mandriva.com/MandrivaLinux/devel/xtreemos/2009.0/i586/media/xtreemos/release/
```

Finally, install packages `rootca-config` and `create-csr` on the XtreemOS core node and follow the admin guide.

```
[root@xos-core ~]# urpmi rootca-config
[root@xos-core ~]# urpmi create-csr
```

3.5.2 Service certificate generation

Initiate a certification authority.

```
[root@xos-core ~]# create-rootca /opt/xtreemosca
Generating a 2048 bit RSA private key
........+++
...........................................+++
writing new private key to '/opt/xtreemosca/private/xtreemos.key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
Root CA Private key written to /opt/xtreemosca/private/xtreemos.key - keep this\
 private key secure
Root CA Public key certificate written to /opt/xtreemosca/public/xtreemos.crt (valid\
  until Sep 30 13:19:12 2012 GMT).
This is the XtreemOS root certificate to be installed on all machines in this Grid
It can be published on the VOLife home page for this Grid
Root CA public key certificate copied to /etc/xos/truststore/certs/xtreemos.crt.
```

Create a directory for managing the service certificates.

```
[root@xos-core ~]# mkdir Certificates
[root@xos-core ~]# cd Certificates/
```

Generate a certificate request (csr) for the `cda`.

```
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" cda
Generating a 1024 bit RSA private key
...............++++++
..............++++++
writing new private key to 'xos-core.xtreemos.eu-cda.key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
```

Similarly, generate certificate requests for all services.

```
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" cda
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" rca
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" vops
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" dir
```

```
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" mrc
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" osd
[root@xos-core Certificates]# create-csr xos-core.xtreemos.eu "XtreemOS" xtfs_mount
[root@xos-core Certificates]# ls *.csr
xos-core.xtreemos.eu-cda.csr   xos-core.xtreemos.eu-rca.csr
xos-core.xtreemos.eu-dir.csr   xos-core.xtreemos.eu-vops.csr
xos-core.xtreemos.eu-mrc.csr   xos-core.xtreemos.eu-xtfs_mount.csr
xos-core.xtreemos.eu-osd.csr
```

Process all certificate requests.

```
[root@xos-core Certificates]# for i in *.csr; do process-csr /opt/xtreemosca $i; done
Using configuration from /etc/xos/config/openssl/process-csr.conf
Enter pass phrase for /opt/xtreemosca/private/xtreemos.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 8 (0x8)
...
            X509v3 Key Usage:
                Digital Signature, Key Encipherment, Certificate Sign
Certificate is to be certified until Oct  1 14:06:36 2011 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Created certificate in xos-core.xtreemos.eu-cda.crt
...
Created certificate in xos-core.xtreemos.eu-dir.crt
Created certificate in xos-core.xtreemos.eu-mrc.crt
Created certificate in xos-core.xtreemos.eu-osd.crt
Created certificate in xos-core.xtreemos.eu-rca.crt
Created certificate in xos-core.xtreemos.eu-vops.crt
Created certificate in xos-core.xtreemos.eu-xtfs_mount.crt
```

Record all service certificates and their private keys in a `tar` file.

```
[root@xos-core Certificates]# mkdir public
[root@xos-core Certificates]# cp /opt/xtreemosca/public/xtreemos.crt ./public/
[root@xos-core Certificates]# tar zcf VMcerts.tgz *.crt *.key public/xtreemos.crt
```

3.6 Certificates installation

```
yjegou@host: VirtNodesCA$ tar ztf VMcerts.tgz
public/xtreemos.crt
xos-core.xtreemos.eu-cda.crt
xos-core.xtreemos.eu-cda.key
xos-core.xtreemos.eu-dir.crt
xos-core.xtreemos.eu-dir.key
xos-core.xtreemos.eu-mrc.crt
xos-core.xtreemos.eu-mrc.key
xos-core.xtreemos.eu-osd.crt
xos-core.xtreemos.eu-osd.key
xos-core.xtreemos.eu-rca.crt
xos-core.xtreemos.eu-rca.key
xos-core.xtreemos.eu-vops.crt
xos-core.xtreemos.eu-vops.key
xos-core.xtreemos.eu-xtfs_mount.crt
xos-core.xtreemos.eu-xtfs_mount.key
```

3.6.1 Certificate on the core node

The service certificates and the corresponding keys have been assembled in a `tar` file on the VM host (or on the core node) where they have been generated. Copy the certificates on the core node.

```
yjegou@host: VirtNodesCA$ scp VMcerts.tgz yvon@xos-core:./
VMcerts.tgz                                      100%   16KB  16.0KB/s  ...
```

Unpack.

```
[yvon@xos-core ~]$ ls
public/                      xos-core.xtreemos.eu-osd.crt
tmp/                         xos-core.xtreemos.eu-osd.key
VMcerts.tgz                  xos-core.xtreemos.eu-rca.crt
xos-core.xtreemos.eu-cda.crt  xos-core.xtreemos.eu-rca.key
xos-core.xtreemos.eu-cda.key  xos-core.xtreemos.eu-vops.crt
xos-core.xtreemos.eu-dir.crt  xos-core.xtreemos.eu-vops.key
xos-core.xtreemos.eu-dir.key  xos-core.xtreemos.eu-xtfs_mount.crt
xos-core.xtreemos.eu-mrc.crt  xos-core.xtreemos.eu-xtfs_mount.key
xos-core.xtreemos.eu-mrc.key
```

3.6.2 Certificates in `xosautoconfig`

Log in the node as `root` and install the certificates in the configuration tool.

```
[root@xos-core ~]# cd /etc/xos/xosautoconfig/conf/etc/xos/truststore/certs/
[root@xos-core certs]# /bin/rm *
[root@xos-core certs]# cp ~yvon/public/xtreemos.crt ./xtreemos.crt
[root@xos-core certs]# cp ~yvon/xos-core.xtreemos.eu-cda.crt ./cda.crt
[root@xos-core certs]# cp ~yvon/xos-core.xtreemos.eu-rca.crt ./rcaserver.crt
[root@xos-core certs]# cp ~yvon/xos-core.xtreemos.eu-vops.crt ./vops.crt
[root@xos-core certs]# ls
cda.crt  rcaserver.crt  vops.crt  xtreemos.crt
[root@xos-core certs]# cd ../private/
[root@xos-core private]# /bin/rm *
[root@xos-core private]# cp ~yvon/xos-core.xtreemos.eu-cda.key ./cda.key
[root@xos-core private]# cp ~yvon/xos-core.xtreemos.eu-rca.key ./rcaserver.key
[root@xos-core private]# cp ~yvon/xos-core.xtreemos.eu-vops.key ./vops.key
```

It is also possible to directly install the certificates in their final destination. In this case, it is important to remove all certificates from the configuration tool to avoid conflicts (if certificates are stored in `xosautoconfig` configuration directory, `xosautoconfig` moves them to their final locations).

The certificate private keys delivered by `create-csr` are protected by a password. It is possible to remove this password to simplify certificate installation using `openssl rsa` (see `man rsa`). The standard solution is to store the password in a configuration file. The default password of all keys configured by `xosautoconfig` is `xtreemos`. If your passwords are different, go to directory `/etc/xos/xosautoconfig/conf/etc/xos/config` and store the passwords in files

- `VOPSConfig.conf` for VOPS;
- `RCAServerConfig.conf` for RCA server;
- `cdaserver/cdaserver.properties` and `volife/volife.properties` for CDA.

3.7 Configure `xosautoconfig`

3.7.1 File `localDefs`

File `localDefs` defines attributes of the nodes: host name, ... `xosautoconfig` can guess the values associated to MYHOSTNAME, MYIP, MYINTERFACE from the networking configuration and to MYNODETYPE, from the other configuration files. So, these values should be left unassigned. This allows `xosautoconfig`

to automatically adapt the configuration to changes in the node environment (change in hostname/IP, clone, ...). With SETMEDIA set to true, xosautoconfig configures the Mandriva mirrors used for package retrieval. xosautoconfig automatically resets this variable to false when run. If a value is assigned to CONFIGUREVO, xosautoconfig tries to configure the node for this VO. As no VO is defined for this first configuration, this variable should be left unassigned. NOPROMPT=true allows xosautoconfig to run in non-interactive mode.

```
[root@xos-core xosautoconfig]# cd /etc/xos/xosautoconfig/
[root@xos-core xosautoconfig]# cat localDefs
#local definitions

SETMEDIA=true
CONFIGUREVO=
MYHOSTNAME=
MYIP=
MYINTERFACE=
MYDISK=/dev/sda6
MYNODETYPE=
XOSDADDRESSEXTERNALADDRESS=$MYIP
XOSDADDRESSHOST=$MYIP
ADDRESSHOST=$MYIP
NOPROMPT=true
```

3.7.2   File `globalDefs`

File globalDefs defines attributes common to all nodes of the same grid. For a simple grid configuration (all core services on the same node), all IP addresses and host names except for PROXY and NTP should be the IP and host name of the nodes being configured. Variable USESSL should be false as long as all nodes of the grid have not been configured (restriction for all releases of XtreemOS until 2.1.2, at least).

```
[root@xos-core xosautoconfig]# cat globalDefs
#global definitions

#PROXY=http://paradonf.irisa.fr:3128
PROXY=noProxy

NTP=ntp1.irisa.fr

GLOBALVOPSIP=192.168.122.10
SCALARISBOOTIP=192.168.122.10
OWBOOTSTRAPIP=192.168.122.10
RSSBOOTSTRAPIP=192.168.122.10
DIXIROOTHOST=xos-core.xtreemos.eu
DIXIROOTIP=192.168.122.10
DIRHOSTIP=192.168.122.10
MRCHOSTIP=192.168.122.10
OSDHOSTIP=192.168.122.10
USESSL=false
```

A future release of xosautoconfig should consider a third configuration file for site-related attributes. Variables NTP and PROXY should be moved to this new file.

3.7.3   File `services`

File services defines all services to be configured for each node type. Each line of this file defines one (or more) types (a string) and a list of services to be configured for this node type. A type can be listed on multiple lines. It is also possible to use a host name (FQDN) as a type. Type all-nodes allows to list services to be configured on all nodes of the grid.

```
[root@xos-core xosautoconfig]# cat services
# Service description
...
head-node: JobDirectory JobMng RCAServer ResAllocator ReservationManager
head-node: ResMng VOPS ExecMng RCAClient ResAllocator ResourceMonitor
head-node: SRDSMng VOLife xvoms cdaserver cdaclient
head-node: xtreemfs-dir xtreemfs-mrc xtreemfs-osd
head-node: amsd nsspam openssh xtreemos-openssh ntp xtreemfs-client

resource-node: ExecMng RCAClient ResAllocator ResourceMonitor SRDSMng
resource-node: ResMng cdaclient
resource-node: amsd nsspam openssh xtreemos-openssh ntp xtreemfs-client

all-nodes: CronDaemon DaemonGlobal XMLExtractor
```

3.7.4 File `nodeTypes`

File `nodeTypes` associates host names to node types. Each line contains a node type and a list of host names.
The special type `default-node-type` allows to specify which node type should be associated by default.

```
[root@xos-core xosautoconfig]# cat nodeTypes
head-node: xos-core.xtreemos.eu
resource-node: xos-node1.xtreemos.eu xos-node2.xtreemos.eu
default-node-type: resource-node
```

3.7.5 Folder `/root/.ssh/`

Files `authorized_keys` and `config-xos` of folder `/root/.ssh/` are configured by `xosautoconfig`
from files located in `/etc/xos/xosautoconfig/conf/root/.ssh`.

```
[root@xos-core .ssh]# cp ~yvon/.ssh/authorized_keys2 \
                         /etc/xos/xosautoconfig/conf/root/.ssh/
```

File `/root/.ssh/config-xos` is read by `ssh-xos` and specifies the location of the user certificate.

```
[root@xos-core .ssh]# cat /etc/xos/xosautoconfig/conf/root/.ssh/config-xos
 XosProxyFile /root/.xos/truststore/certs/user.crt
 XosPrivKeyFile /root/.xos/truststore/private/user.key
 XosVoName     VO-Yvon
```

3.7.6 Folder `/root/.xos/`

File `XATICAConfig.conf` of folder `/root/.xos` configures commands linked to library `libXATICA` and
file `XATIConfig.conf` configures `java` commands linked to the DIXI bus when they are run by `root`. These
files specify the location of the DIXI bus and of the certificates. When `xosautoconfig` is run, it first up-
dates keys `xosdaddress.host` and `address.host` of these files from the values defined in `localDefs`.
Then each line of these configuration files replace the corresponding line in folder `/root/.xos/`. Note that
`xosautoconfig` requests the DIXI interface to auto-generate these files if they are not present.

```
[root@xos-core .xos]# cat XATICAConfig.conf
xosdaddress.host=192.168.122.110
address.host=192.168.122.110
certificateLocation=/root/.xos/truststore/certs/user.crt
privateKeyLocation=/root/.xos/truststore/private/user.key
trustStoreSSL=/etc/xos/truststore/certs/
useSSL=false
userCertificateFile=/root/.xos/truststore/certs/user.crt
```

16

```
[root@xos-core .xos]# cat XATIConfig.conf
loadPrivateKey=false
useSSL=false
xosdaddress.externalAddress=192.168.122.110
xosdaddress.host=192.168.122.110
privateKeyLocation=/root/.xos/truststore/private/user.key
userKeyFile=/root/.xos/truststore/private/user.key
networkInterface=eth0
trustStoreSSL=/etc/xos/truststore/certs/
address.host=192.168.122.110
userCertificateFile=/root/.xos/truststore/certs/user.crt
certificateLocation=/root/.xos/truststore/certs/user.crt
```

3.7.7 File `/etc/hosts`

In the case where networking is defined statically, it is possible to ask xosautoconfig to configure file /etc/ hosts on all nodes using file /etc/xos/xosautoconfig/conf/etc/hosts.

3.8 Saving the grid configuration

It is possible to setup a simple grid from the same set of configuration files. Saving the whole xosautoconfig tree allows to replicate the configuration on other nodes. Note that if host names or IP addresses have been fixed in the localDefs file, this file should not be replicated.

```
[root@xos-core xos]# cd /etc/xos
[root@xos-core xos]# tar zcf MyGrid.tgz xosautoconfig/
```

3.9 Running `xosautoconfig`

Running xosautoconfig as root from a terminal should configure this first core node.

```
[root@xos-core xos]# xosautoconfig
* Checking configuration files and directories...
/etc:                                                    [ OK ]
/root:                                                   [ OK ]
...
```

Note that it is possible to pass various attributes to xosautoconfig from the command line. See file /usr/share/doc/xosautoconfig/README of the XtreemOS node.

3.10 Checking that XtreemOS is up

After the execution of xosautoconfig, this first node should be ready. Some verifications are useful at this stage to avoid some problems later.

3.10.1 XtreemFS

Check that all XtreemFS services are running.

```
[root@xos-core xos]# ps -aef | grep xtreemfs
xtreemfs 13836     1  0 14:46 pts/0    00:00:02 /usr/bin/java -cp \
 /usr/share/java/XtreemFS.jar:/usr/share/java/BabuDB.jar:/usr/share /java/yidl.jar\
 org.xtreemfs.dir.DIR /etc/xos/xtreemfs/dirconfig.properties
xtreemfs 13904     1  0 14:46 pts/0    00:00:04 /usr/bin/java -cp \
 /usr/share/java/XtreemFS.jar:/usr/share/java/BabuDB.jar:/usr/share /java/yidl.jar\
 org.xtreemfs.mrc.MRC /etc/xos/xtreemfs/mrcconfig.properties
xtreemfs 13938     1  0 14:46 pts/0    00:00:04 /usr/bin/java -cp \
 /usr/share/java/XtreemFS.jar:/usr/share/java/yidl.jar\
 org.xtreemfs.osd.OSD /etc/xos/xtreemfs/osdconfig.properties
```

Figure 19: Web interface of XtreemFS dir service

Figures 19, 20 and 21 show the web interface expected for the XtreemFS `dir` (`http://xos-core:30638/`), `mrc` (`http://xos-core:30636/`) and `osd` (`http://xos-core:30640/`) services.

Figure 20: Web interface of XtreemFS mrc service

Figure 21: Web interface of XtreemFS osd service

3.10.2 DIXI

```
[root@xos-core xos]# ps -aef | grep dixi
root      14277     1  1 14:46 pts/0    00:00:13 java -cp :/usr/share/dixi/libs/\
log4j.jar:/usr/share/dixi/libs/jrpcgen.jar:...-r /usr/share/dixi
```

SRDS is in general launched as a DIXI service. Figure 22 shows the web interface expected for SRDS (http://xos-core:9000/).
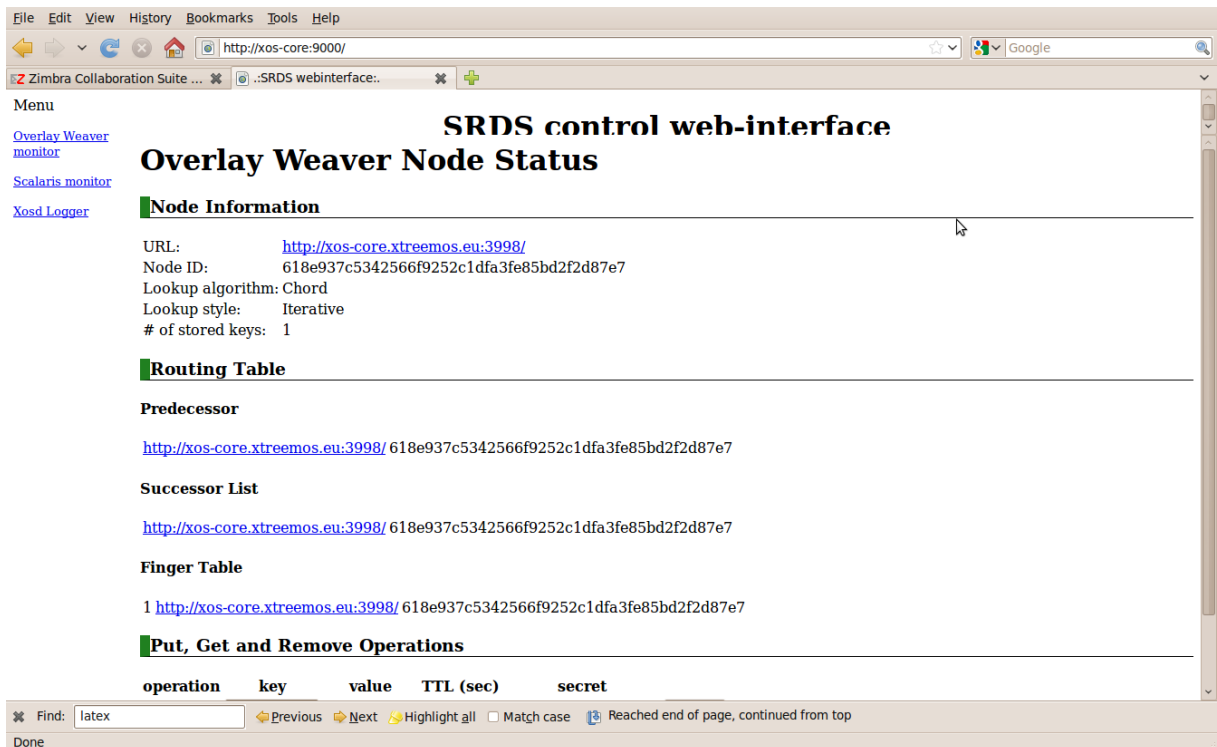
Figure 22: Web interface of SRDS

3.10.3 Scalaris

```
[root@xos-core xos]# ps -aef | grep erlang
root       4419     1  0 09:55 ?        00:00:00 /usr/lib/erlang/erts-5.6.4/bin/epmd\
 -daemon
root      14383     1  0 14:46 pts/0    00:00:00 /usr/lib/erlang/erts-5.6.4/bin/beam\
 -- -root /usr/lib/erlang -progname erl ...  -hidden -name boot -s boot
```

Scalaris is in general launched as a DIXI service. Figure 23 shows the web interface (http://xos-core:9001/) expected for Scalaris on the bootstrap node.

3.10.4 `cdaserver`

Check that the cdaserver is running:

```
[root@xos-core ~]# ps -aef | grep cdaserver
cdauser    3528     1  0 12:08 ?        00:00:14 /usr/bin/java -ea -server\
 -Djava.net.preferIPv4Stack=true\
 -cp /etc/xos/config/xvoms:/etc/xos/config/cdaserver:/usr/share/java/CdaUtil.jar:...\
 eu.xtreemos.security.cda.server.CdaServer
```
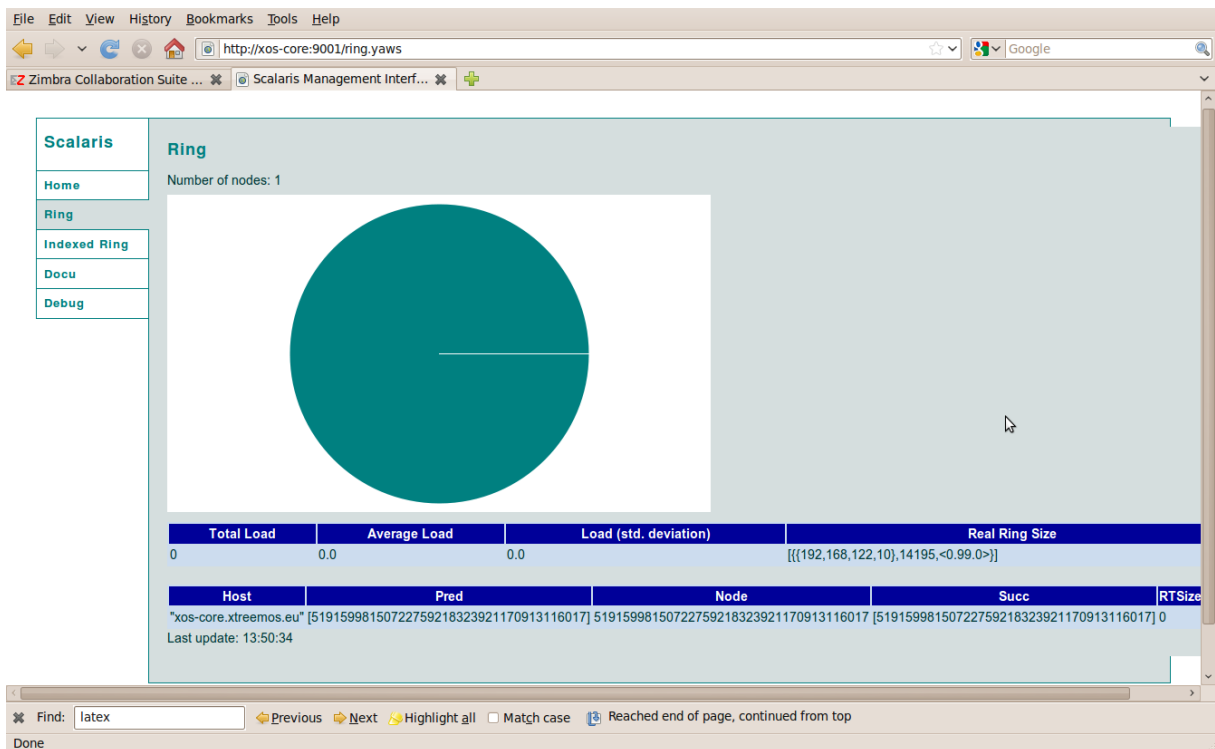
Figure 23: Web interface of Scalaris

3.10.5 VOlife

The VOlife service presents two interfaces to the user: a web interface and a command-line interface. Figure 24 shows the web interface (`http://xos-core:8080/volifecycle/`) expected for VOlife on the core node.

The correct installation of VOlife can also be tested using the command-line interface:

```
[root@xos-core ~]# volife_run.sh --list-all-users
id=1,realname=XtreemOS VOuser,guid=ea9a7366-e34f-4a99-9e31-277430366475,\
 username=xtreemos-vouser,password=ea49b4f00f3a638886aa4045a4f4666d,\
 email=VOuser.XtreemOS@other.org,status=approved,affiliation=Other,\
 description=,expiryDate=2012-05-01 00:00:00.0,vos=\
 [2c0e8cb2-4453-46fe-85b7-74874e76e7c2],ownedvos=\
 [2c0e8cb2-4453-46fe-85b7-74874e76e7c2],vogroups=[group1],voroles=[],\
 actor=VOUSER,requests=[],rcas=[],reserved=,recordedBy=SimpleEntity,\
 recordDate=2008-12-10 17:17:45.0,recordVersion=1
id=2,realname=XtreemOS Admin,guid=8b432b5e-f812-42ca-b90a-d3dc1538dff5,\
 username=admin,password=e10e097d7c31a283b4229729b4dbde44,email=none@no.org,\
 status=approved,affiliation=Other,description=,expiryDate=2012-05-01 \
 00:00:00.0,vos=[],ownedvos=[],vogroups=[],voroles=[],actor=VOUSER,\
 requests=[],rcas=[],reserved=,recordedBy=SimpleEntity,recordDate=\
 2009-05-22 12:05:52.0,recordVersion=1
```

If these interfaces do not operate correctly, for instance, if it is not possible to create an account using the web interface or it is not possible to list the users using the command-line interface, check some hints in Section 8.5.

4   Creating users and VOs, ready to submit jobs

User and VOs can be created using the `VOlife` web interface. In this document, we use the command-line interface which allows easy scripting. Note however that this command-line interface is not fully secure as it allows access to the XVOMS data-base without checking user credentials.

21
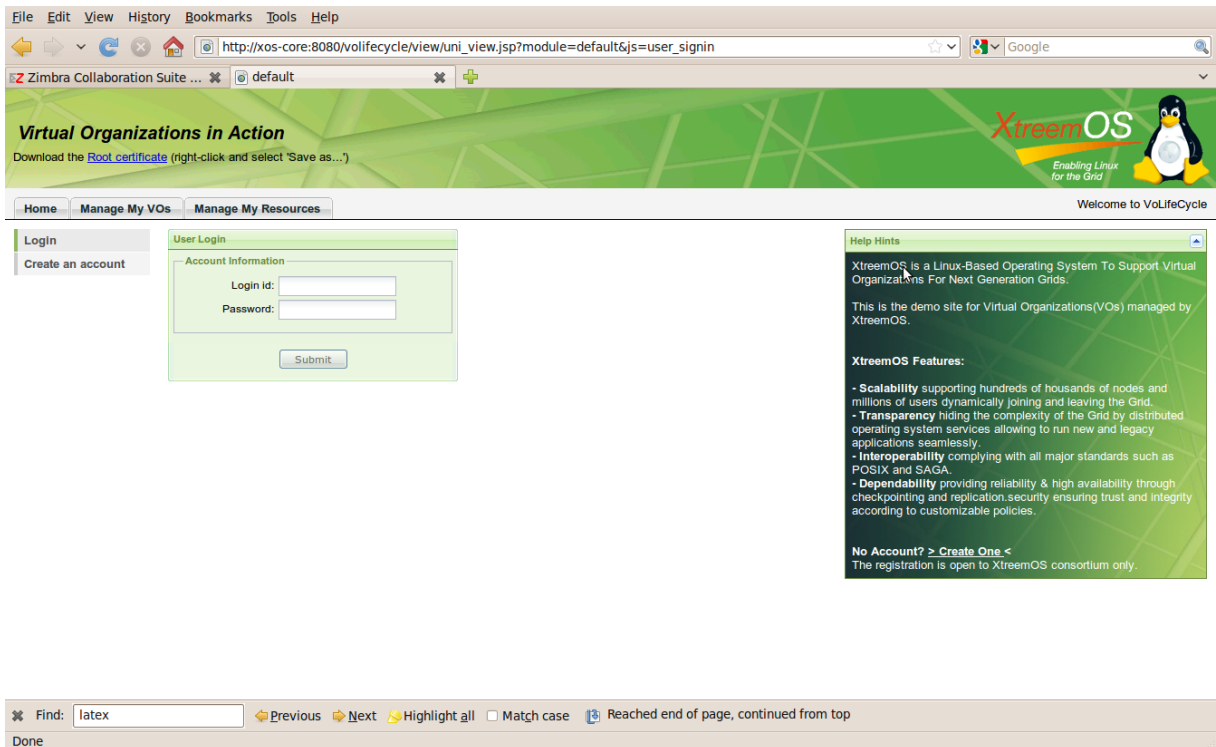
Figure 24: Web interface of VOlife

4.1 First users

First step, register users `yvon-admin` and `yvon` (or any other users of your choice).

```
[root@xos-core ~]# volife_run.sh –create–user yvon-admin xtreemos Yvon Jegou INRIA\
 Yvon.Jegou@inria.fr
id=4,realname=Yvon Jegou,guid=55c8acc0-0745-4df7-bced-7a8c98bfe8ed,\
 username=yvon-admin,password=ea49b4f00f3a638886aa4045a4f4666d,\
 email=Yvon.Jegou@inria.fr,status=pending,affiliation=INRIA,description=,\
 expiryDate=Sat Sep 24 16:08:05 CEST 2011,vos=[],ownedvos=[],\
 vogroups=[],voroles=[],actor=VOUSER,requests=[],rcas=[],reserved=,\
 recordedBy=SimpleEntity,recordDate=Fri Sep 24 16:08:05 CEST 2010,\
 recordVersion=1

[root@xos-core ~]# volife_run.sh –create–user yvon xtreemos Yvon\
 Jegou INRIA Yvon.Jegou@inria.fr
id=5,realname=Yvon Jegou,guid=663a6798-3d16-4923-a5da-e829a03f057e,\
 username=yvon,password=ea49b4f00f3a638886aa4045a4f4666d,email=\
 Yvon.Jegou@inria.fr,status=pending,affiliation=INRIA,description=,\
 expiryDate=Sat Sep 24 16:08:23 CEST 2011,vos=[],ownedvos=[],\
 vogroups=[],voroles=[],actor=VOUSER,requests=[],rcas=[],reserved=,\
 recordedBy=SimpleEntity,recordDate=Fri Sep 24 16:08:23 CEST 2010,\
 recordVersion=1
```

Second step, approve user registration.

```
[root@xos-core ~]# volife_run.sh –approve–user yvon-admin
id=4,realname=Yvon Jegou,guid=55c8acc0-0745-4df7-bced-7a8c98bfe8ed,\
 username=yvon-admin,password=ea49b4f00f3a638886aa4045a4f4666d,email=\
 Yvon.Jegou@inria.fr,status=approved,affiliation=INRIA,description=,\
 expiryDate=2011-09-24 16:08:05.0,vos=[],ownedvos=[],vogroups=[],\
```

```
voroles=[],actor=VOUSER,requests=[],rcas=[],reserved=,recordedBy=\
SimpleEntity,recordDate=2010-09-24 16:08:05.0,recordVersion=1
```

```
[root@xos-core ~]# volife_run.sh -approve-user yvon
id=5,realname=Yvon Jegou,guid=663a6798-3d16-4923-a5da-e829a03f057e,\
 username=yvon,password=ea49b4f00f3a638886aa4045a4f4666d,email=\
 Yvon.Jegou@inria.fr,status=approved,affiliation=INRIA,description=,\
 expiryDate=2011-09-24 16:08:23.0,vos=[],ownedvos=[],vogroups=[],\
 voroles=[],actor=VOUSER,requests=[],rcas=[],reserved=,recordedBy=\
 SimpleEntity,recordDate=2010-09-24 16:08:23.0,recordVersion=1
```

4.1.1 Create a VO and register a user in the VO

User `yvon-admin` creates a first VO named `vo-yvon`:

```
[root@xos-core ~]# volife_run.sh -create-vo vo-yvon vo-yvon-test yvon-admin
id=2,name=vo-yvon,owner=yvon-admin,gvid=\
 70ac51ce-d716-4d5b-9522-076d1b7f1396,description=vo-yvon-test,\
 users=[yvon-admin],vogroups=[],resources=[],requests=[],reserved=,\
 recordedBy=SimpleEntity,recordDate=Fri Sep 24 16:10:22 CEST 2010,\
 recordVersion=1
```

User `yvon` generates a request for joining `vo-yvon`:

```
[root@xos-core ~]# volife_run.sh -create-user-req vo-yvon yvon
id=3,description=,type=user_request,targetvo=70ac51ce-d716-4d5b-9522-076d1b7f1396,\
 owner=yvon,reserved=,recordedBy=,recordDate=,recordVersion=0
```

The VO owner (`yvon-admin`) approves the request.

```
[root@xos-core ~]# volife_run.sh -approve-user-req 3
```

Note that the VO owner needs to know the request ID (`id=3` in our example). This ID is provided in result of the request for joining the VO. He can also list all pending requests and get these ids using

```
[root@xos-core ~]# volife_run.sh -list-user-req <vo_name|vo_gvid>
```

4.1.2 Generate a user certificate

First, generate a new key pair. This operation may take some time.

```
[root@xos-core ~]# volife_run.sh -gen-keypair yvon xtreemos
certs/663a6798-3d16-4923-a5da-e829a03f057e.pem
```

Second step, generate a certificate.

```
[root@xos-core ~]# volife_run.sh -gen-xoscert yvon vo-yvon xtreemos 90
16:11:54,886 DEBUG XVOMSUtil:? - Convert from User to VOUser:\
 [663a6798-3d16-4923-a5da-e829a03f057e,70ac51ce-d716-4d5b-9522-076d1b7f1396,\
 null,[null],[null],null,null,null]\
 certs/663a6798-3d16-4923-a5da-e829a03f057e-70ac51ce-d716-4d5b-9522-\
 076d1b7f1396.pem
```

Optional step: manage groups. For instance, create the group `users` in VO `vo-yvon`.

```
[root@xos-core ~]# volife_run.sh -add-group vo-yvon users
id=2,description=users,ggid=a9199374-3b98-40a0-8ed6-2d74cffd46f3,\
 vo=70ac51ce-d716-4d5b-9522-076d1b7f1396,users=[],voroles=[],\
 reserved=,recordedBy=SimpleEntity,recordDate=Fri Sep 24 16:12:13\
 CEST 2010,recordVersion=1
```

Last step, get the certificate.

```
[root@xos-core ~]# get-xos-cert xos-core:6730 vo-yvon users -u yvon\
 -p xtreemos -k /root/.xos/truststore/private/yvon.key \
 -c /root/.xos/truststore/certs/yvon.crt
Passphrase to protect private key (at least 8 characters long):
Type passphrase again to confirm:
Generating a new public/private key pair
Warning: certificate presented by remote host xos-core belongs to
 xos-core.xtreemos.eu -
carrying on as you requested to ignore CDA host certificates which\
 don't belong to the CDA server you connected to.
You should only use any credentials issued for testing purposes.

Saving certificate chain (user+CDA) in /root/.xos/truststore/certs/yvon.crt.
```

Check that the certificates have been stored in the correct location and that they can be verified.

```
[root@xos-core ~]# ls /root/.xos/truststore/certs/
yvon.crt
[root@xos-core ~]# ls /root/.xos/truststore/private/
yvon.key
[root@xos-core ~]# openssl verify \
    -CAfile /etc/xos/truststore/certs/xtreemos.crt  \
    -CApath /etc/xos/truststore/certs \
    /root/.xos/truststore/certs/yvon.crt
/root/.xos/truststore/certs/yvon.crt: OK
```

Finally, provide the correct pathnames for the certificates:

```
[root@xos-core ~]# cd /root/.xos/truststore/certs/
[root@xos-core certs]# ln -s yvon.crt user.crt
[root@xos-core certs]# cd /root/.xos/truststore/private/
[root@xos-core private]# ln -s yvon.key user.key
```

Note that it is also possible to provide the final pathnames to `get-xos-cert`.

4.2 Configure a VO on the node

First step, check that the resource certificate has been installed by `xosautoconfig`.

```
[root@xos-core ~]# ls -al /etc/xos/truststore/certs/resource.crt
-rw-r--r-- 1 root root 1151 2010-09-24 14:46 \
                              /etc/xos/truststore/certs/resource.crt
```

The location of this certificate is specified in files `/etc/xos/config/RCAClientConfig.conf` and `/etc/xos/config/XOSdConfig.conf` in field `certificateLocation`. If the resource certificate has not been generated, see Section 8.8.

Second step, add the new VO to the RCA service.

```
[root@xos-core ~]# rca_vo l
List empty.
[root@xos-core ~]# rca_vo a 70ac51ce-d716-4d5b-9522-076d1b7f1396
Adding the RCA to VO 70ac51ce-d716-4d5b-9522-076d1b7f1396
```

Command `rca_vo l` prints the list of registered VOs. Command `rca_vo a` registers a new VO specified by its VOID. The VOID can be obtained from various `volife_run.sh` commands inside fields `gvid`. For instance,

```
[root@xos-core ~]# volife_run.sh -list-vo vo-yvon
id=2,name=vo-yvon,owner=yvon-admin,\
 gvid=70ac51ce-d716-4d5b-9522-076d1b7f1396,description=vo-yvon-test,\
 users=[yvon-admin, yvon],vogroups=[users],resources=[],requests=[],\
 reserved=,recordedBy=SimpleEntity,recordDate=2010-09-24 16:10:22.0,\
 recordVersion=1
```

Third step, add the node to the list of resources of a registered VO.

```
[root@xos-core ~]# rca_resource_vo a 70ac51ce-d716-4d5b-9522-076d1b7f1396
Adding self to the VO.
Added resource to VO 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/incoming/.
[root@xos-core ~]# rca_resource_vo c 70ac51ce-d716-4d5b-9522-076d1b7f1396
The RCA client received the certificate for VO\
 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/.
```

The first request to rca_resource_vo adds the node to the VO and returns a certificate in /etc/xos/truststore/certs/incoming/. It is possible to copy this certificate to its final location /etc/xos/truststore/certs/. Another solution is to re-execute rca_resource_vo with option c. This call will renew the certificate and store it directly in its final location. If this command returns an error message indicating that the VO is not a registered VO, see Section 8.9.

4.3 Configure the local policies on the node

The node local policies must be configured to accept the execution of user codes in the context of VOs:

```
[root@xos-core ~]# xos-policy-admin-am \
                   -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396 \
                   --force
[root@xos-core ~]# xos-policy-admin-gm \
                   -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396 \
                   --force
```

The correct configuration of the policies can be checked using xos-policy-admin-chk.

```
[root@xos-core ~]# xos-policy-admin-chk  \
                     -pem /root/.xos/truststore/certs/yvon.crt
dn = [/CN=663a6798-3d16-4923-a5da-e829a03f057e], \
vo = [70ac51ce-d716-4d5b-9522-076d1b7f1396], role = [null]
Sucess in PAM checking !
```

If xos-policy-admin-chk fails, see Section 8.10.

4.4 Check **ssh-xos**

Re-log in the node using ssh-xos:

```
[root@xos-core certs]# ssh-xos localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
RSA key fingerprint is ec:42:b0:4f:61:58:dc:7b:de:7a:8e:43:ed:f5:15:fd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
Enter passphrase for key '/root/.xos/truststore/private/user.key':
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' \
already exists in Directory Service, \
id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)

-bash-3.2$ pwd
/home/663a6798-3d16-4923-a5da-e829a03f057e
-bash-3.2$ id
uid=60000(/CN=663a6798-3d16-4923-a5da-e829a03f057e) \
 gid=60169(xosuser_g60169) groups=60169(xosuser_g60169)
```

Ignore errno = 17: see Section 8.11. The user is logged with his grid IDs as shown by id and the user home volume is his home-directory. Also check that it is possible to specify the host name to ssh-xos:

```
[root@xos-core ~]# ssh-xos xos-core
The authenticity of host 'xos-core (192.168.122.10)' can't be established.
RSA key fingerprint is ec:42:b0:4f:61:58:dc:7b:de:7a:8e:43:ed:f5:15:fd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'xos-core,192.168.122.10' (RSA) to the \
 list of known hosts.
Enter passphrase for key '/root/.xos/truststore/private/user.key':
Last login: Fri Sep 24 16:27:33 2010 from localhost
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' \
 already exists in Directory Service, \
 id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
-bash-3.2$
```

If ssh-xos requests the user password after requesting the passphrase for the certificate key, check that the local policies are configured for his VO on the remote node (see Section 4.3). Another possible source of failure for ssh-xos is the presence of the /etc/pki/tls/cert.pem certificate on the resource node (see Section 8.10.1).

4.5 Submit a first job as **root**

First check that grid commands can be executed:

```
[root@xos-core ~]# xps -a
[root@xos-core ~]# xreservation -qf
Address = [://192.168.122.10:60000]: * : *
```

If these commands fail, (return an error or do not return), check that xosd as well as all core services are running. Check that there is no major error in file /var/log/xosd/xosd.log.

The default XtreemOS distribution provides a sample job in /etc/skel/psx.jsdl. Submit this job using xsub, check that it has been executed using xps and check that the job has returned some result in file psx.out in the home volume using ssh-xos.

```
[root@xos-core ~]# xsub -f /etc/skel/psx.jsdl
Job submitted succesfully: cc2d382e-17ad-4e78-ad36-f14be07c9477
[root@xos-core ~]# xps -a
cc2d382e-17ad-4e78-ad36-f14be07c9477 @ 1285338693435 :
                jobID = cc2d382e-17ad-4e78-ad36-f14be07c9477
                userDN = 663a6798-3d16-4923-a5da-e829a03f057e
                VO = 70ac51ce-d716-4d5b-9522-076d1b7f1396
                jobStatus = Done
                submitTime = Fri Sep 24 16:31:26 CEST 2010
[root@xos-core ~]# ssh-xos xos-core
Enter passphrase for key '/root/.xos/truststore/private/user.key':
Last login: Fri Sep 24 16:30:07 2010 from xos-core.xtreemos.eu
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' \
 already exists in Directory Service, \
 id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
-bash-3.2$ ls -al
total 0
drwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e root          \
    0 2010-09-24 16:31 ./
-rw------- 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60009 \
   12 2010-09-24 16:29 .bash_history
-rwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60009 \
   37 2010-09-24 16:31 psx.err*
-rwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60009 \
22965 2010-09-24 16:31 psx.out*
drwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60009 \
    0 2010-09-24 16:27 tmp/
-bash-3.2$ exit
logout
```

4.6 Configure a user environment and submit a job

For each new user, we need to create an account on some node, initialize some local configuration files, register to some VO and download certificates. For this simple installation guide, we initialize the user environment from `root`.

```
[root@xos-core ~]# cp -ar /root/.xos ~yvon/
[root@xos-core ~]# chown -R yvon:yvon ~yvon/.xos/
[yvon@xos-core ~]$ cat /dev/null > .xos/xosd-xati.log
```

The last command erases the `xati` logs. All path names must be updated in the configuration files: replace all occurrences of `/root/.xos/` by `/home/yvon/.xos` in files `~yvon/.xos/XATICAConfig.conf` and `~yvon/.xos/XATIConfig.conf`. Note that the system does not currently interpret environment variables or "~" in configuration files: use absolute paths.

An alternative solution to initialize a user environment is that the user executes both `xps -a` and `xconsole_dixi`. Both commands will end in error as no configuration file is present. But these commands will install default configuration files in the user environment. These files must be updated by the user before submitting requests.

Check that the user can execute grid requests.

```
[yvon@xos-core ~]$ xps -a
```

Submit a first job.

```
[yvon@xos-core ~]$ xsub -f /etc/skel/psx.jsdl
Job submitted succesfully: 265d4168-345d-4b42-9ccf-4ef35d7c42c8
[yvon@xos-core ~]$ xps -a
265d4168-345d-4b42-9ccf-4ef35d7c42c8 @ 1285339311014 :
            jobID = 265d4168-345d-4b42-9ccf-4ef35d7c42c8
            userDN = 663a6798-3d16-4923-a5da-e829a03f057e
            VO = 70ac51ce-d716-4d5b-9522-076d1b7f1396
            jobStatus = Done
            submitTime = Fri Sep 24 16:41:48 CEST 2010
```

And check that the user can read the results using `ssh-xos`.

```
[yvon@xos-core ~]$ ssh-xos xos-core
The authenticity of host 'xos-core (192.168.122.10)' can't be established.
RSA key fingerprint is ec:42:b0:4f:61:58:dc:7b:de:7a:8e:43:ed:f5:15:fd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'xos-core,192.168.122.10' (RSA) to \
 the list of known hosts.
Enter passphrase for key '/home/yvon/.xos/truststore/private/user.key':
Last login: Fri Sep 24 16:31:44 2010 from xos-core.xtreemos.eu
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' \
already exists in Directory Service, \
 id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
-bash-3.2$ ls -al
total 0
drwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e root           \
    0 2010-09-24 16:31 ./
-rw------- 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60214 \
   24 2010-09-24 16:31 .bash_history
-rwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60214 \
   37 2010-09-24 16:41 psx.err*
-rwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60214 \
23151 2010-09-24 16:41 psx.out*
drwx------ 1 /CN=663a6798-3d16-4923-a5da-e829a03f057e xosuser_g60214 \
    0 2010-09-24 16:27 tmp/
-bash-3.2$ id
uid=60000(/CN=663a6798-3d16-4923-a5da-e829a03f057e) \
 gid=60214(xosuser_g60214) groups=60214(xosuser_g60214)
```

If `ssh-xos` requests a user password after having requested the certificate password, the user cannot log in the node as a grid user. Check that the policies are configured correctly on the node. Another possible error is described in Section 8.12.

5 Adding a resource VM to the grid

This section describes the installation of a resource node in a VM from the ISO.

5.1 Installing from the ISO

The first steps of the installation are identical to the core node case (cf Section 2.2). Small differences:

- machine name is `xos-node1` (cf Figure 3)

- MAC address is `54:52:00:00:10:01` (cf Figure 10)

- do not select core services (VOlife, XtreemFS server, . . .) in the package selecting step (cf Figure 13), since the corresponding services will not be exploited. Anyway, the `xosautoconfig` tool will download and install the required packages as needed.

Proceed until Section 3. Run the steps in sections 3.1, 3.2 and 3.3.

5.2 Configuring with `xosautoconfig`

This resource node will be configured using the same `xosautoconfig` configuration file as the core node.
First step: get the configuration files saved in Section 3.8.

```
yjegou@host:~$ scp root@xos-core:/etc/xos/MyGrid.tgz /tmp
MyGrid.tgz                                    100%   43KB   43.4KB/s   00:00

yjegou@host:~$ scp /tmp/MyGrid.tgz yvon@xos-node1:./
MyGrid.tgz                                    100%   43KB   43.4KB/s   00:00
```

As root, install these files in `xosautoconfig` directory.

```
[root@xos-node1 ~]# cp ~yvon/MyGrid.tgz /etc/xos
[root@xos-node1 ~]# cd /etc/xos
[root@xos-node1 xos]# tar zxf MyGrid.tgz
```

The `xosautoconfig` are now identical to the `xos-core` version. Run `xosautoconfig`.

```
[root@xos-node1 ~]# xosautoconfig
* Checking configuration files and directories...
/etc:                                                        [ OK ]
/root:                                                       [ OK ]
...
```

The execution of `xosautoconfig` should end with:

```
...
Starting xosd: nohup: redirecting stderr to stdout
                                                             [  OK  ]
Service eu.xtreemos.xosd.security.rca.server.RCAServer is not running\
 on any known node.
* apply for a resource certificate
Requesting a new certificate...
Resource ResourceID = [IP=192.168.122.11:60000] not registered.
* configuration of this node is suspended:
```

```
run script confirmResource on the rca node

and terminate the configuration with\
 script finishConfig on this node
* note: a rca_apply request has already been issued. First check with
rca_list_pending that it has been\
 recorded. If the list is empty
execute rca_apply before running\
 confirmResource on the rca node
* xosautoconfig finished its job. To review all changed configuration files
* head out to /etc/xos/xosautoconfig/backup-100928-0953
```

Check that a certificate request has been emitted.

```
[root@xos-node1 ~]# rca_list_pending
Listing pending resources:
ResourceID = [IP=192.168.122.11:60000]: [hostIP=Address =\
 [://192.168.122.11:60000(192.168.122.11)],\
 hostUniqueID=xos-node1.xtreemos.eu, operatingSystemName=Linux,\
 processorArchitecture=x86, CPUCount=1.0, RAMSize=7.21420288E8,\
 cpuLoadLast15Min=12, cpuLoadLast5Min=22, cpuLoadLast1Min=47]
```

If the resource node does not appear in this list, rerun the rca_apply request. Approve the request on the core node.

```
[root@xos-core ~]# /usr/lib/xos/xosautoconfig/confirmResource
 I am running the RCA server, confirm rca_apply
rca_confirm 192.168.122.11:60000
```

Note that you need to provide the absolute path name of confirmResource. See bug https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=258. Now, request the resource certificate from the resource node.

```
[root@xos-node1 ~]# /usr/lib/xos/xosautoconfig/finishConfig
/usr/lib/xos/xosautoconfig/finishConfig: line 27: ./localDefs:\
 No such file or directory
Listing pending resources:
  List empty.
Listing registered resources:
ResourceID = [IP=192.168.122.11:60000]: [hostIP=Address =\
 [://192.168.122.11:60000(192.168.122.11)],\
 hostUniqueID=xos-node1.xtreemos.eu, operatingSystemName=Linux,\
 processorArchitecture=x86, CPUCount=1.0, RAMSize=7.21420288E8,\
 cpuLoadLast15Min=12, cpuLoadLast5Min=22, cpuLoadLast1Min=47],\
 ResourceID = [IP=192.168.122.10:60000]: [hostIP=Address =\
 [://192.168.122.10:60000(192.168.122.10)],\
 hostUniqueID=xos-core.xtreemos.eu, operatingSystemName=Linux,\
 processorArchitecture=x86, CPUCount=1.0, RAMSize=8.25229312E8,\
 cpuLoadLast15Min=6, cpuLoadLast5Min=19, cpuLoadLast1Min=43]
Requesting a new certificate...
Identity certificate:
  DN: C=FR,L=Rennes,OU=Myriads,O=INRIA,CN=Address =\
 [://192.168.122.11:60000(192.168.122.11)]
  serial number: 1285660787785
  issuer DN: O=INRIA,OU=rca,CN=xos-core.xtreemos.eu/rca
  validity start: Tue Sep 28 09:54:47 CEST 2010
  validity end: Thu Oct 28 10:04:47 CEST 2010
Attributes of attribute certificate:
 (attributes in extensions)
```

```
  MemorySize = 7.21420288E8
  Service =
    eu.xtreemos.system.communication.redirector.ServiceCallRedirector
    eu.xtreemos.xosd.resallocator.ResAllocator
    eu.xtreemos.ads.connection.dixi.SRDSMng
    eu.xtreemos.xosd.resourcemonitor.ResourceMonitor
    eu.xtreemos.xosd.security.rca.client.RCAClient
    eu.xtreemos.xosd.resmng.ResMng
    eu.xtreemos.xosd.daemon.Daemon
    eu.xtreemos.xosd.xmlextractor.XMLExtractor
    eu.xtreemos.xosd.crondaemon.CronDaemon
    eu.xtreemos.xosd.execMng.ExecMng

  CPUCount = 1
  CPUSpeed = 2.927624192E9

configuration terminated,
                good luck :-)
[root@xos-node1 ~]# ls -al /etc/xos/truststore/certs/resource.crt
-rw-r--r-- 1 root root 1151 2010-09-28 10:00\
                                /etc/xos/truststore/certs/resource.crt
```

Rename the `/etc/pki/tls/cert.pem` certificate (see 8.10.1).

```
[root@xos-node1 ~]# cd /etc/pki/tls
[root@xos-node1 tls]# mv cert.pem cert.pem-org
[root@xos-node1 tls]# cd
```

5.3   Attaching the resource node to a VO

In order to exploit this node as a resource for some VO, it is necessary to generate a VO certificate and to configure the local policies. The VO considered in this section is the VO configured in the previous steps on the core node.

Add the resource to a VO:

```
[root@xos-node1 ~]# rca_resource_vo a 70ac51ce-d716-4d5b-9522-076d1b7f1396
Adding self to the VO.
Added resource to VO 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/incoming/.
[root@xos-node1 ~]# rca_resource_vo c 70ac51ce-d716-4d5b-9522-076d1b7f1396
The RCA client received the certificate for VO\
 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/.
```

Check that the certificate has been generated in `/etc/xos/truststore/certs/`. If not, repeat the requests.

Configure local policies:

```
[root@xos-node1 ~]# xos-policy-admin-am  -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396\
                                  --force
[root@xos-node1 ~]# xos-policy-admin-gm  -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396\
                                  --force
```

In order to check the local policy, copy a user certificate from the core node. Note that, in order to directly copy the certificate from the core node to this new node, it is necessary to forward the SSH agent (parameter -A of ssh) from the host.

```
yjegou@host:~$ ssh -A root@xos-node1
Last login: Wed Oct  6 16:47:51 2010 from 192.168.122.1

[root@xos-node1 ~]# scp -r root@xos-core:.xos/truststore/certs/user.crt /tmp
user.crt                                    100% 2656     2.6KB/s 00:00
```

Check that local policies accept this certificate.

```
[root@xos-node1 ~]# xos-policy-admin-chk -pem /tmp/user.crt
dn = [/CN=663a6798-3d16-4923-a5da-e829a03f057e], vo =\
 [70ac51ce-d716-4d5b-9522-076d1b7f1396], role = [null]
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e'\
 already exists in Directory Service,\
 id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
Sucess in PAM checking !
```

Check that `ssh-xos` from the core node is OK.

```
[root@xos-core ~]# ssh-xos xos-node1
...
Enter passphrase for key '/root/.xos/truststore/private/user.key':
Last login: Tue Sep 28 10:23:06 2010 from xos-node1.xtreemos.eu
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e'\
 already exists in Directory Service,\
 id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
-bash-3.2$ ls
psx.err*  psx.out*  tmp/
```

The core node is ready to execute jobs for the configured VO: check that it appears in the list of resource node from the core node.

```
[root@xos-core ~]# xreservation -qf
Address = [://192.168.122.10:60000]: * : *
Address = [://192.168.122.11:60000]: * : *
```

6 Cloning an XtreemOS node

The previous section described the setup of a new resource node from an ISO. This section describes cloning an existing virtual machine.

6.1 Cloning a VM

First step, create a new volume to store the VM image. A right-click on the `localhost(System)` line of the Virtual Machine Manager, followed by `Details` will result in the window of Figure 25.
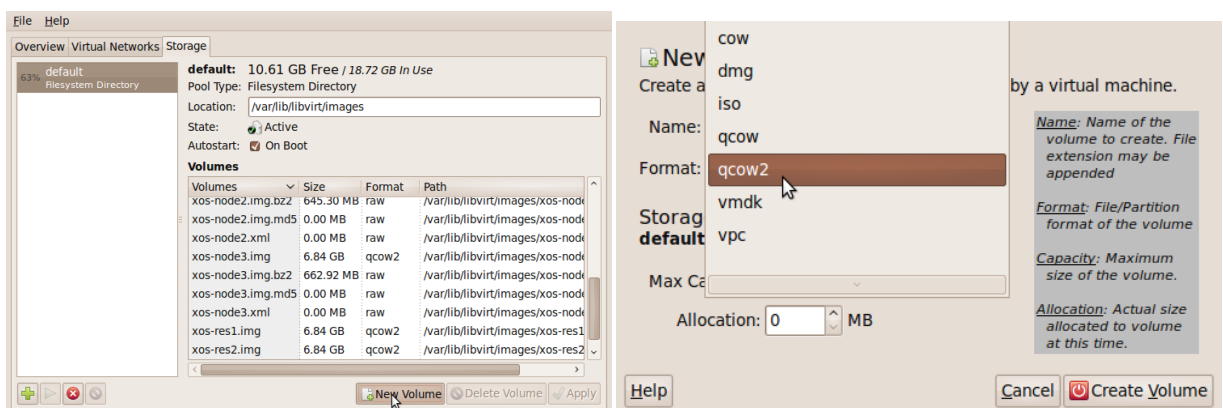

Figure 25: Volume storage creation for a new virtual machine

The virtual machine to be cloned must be stopped:

```
[root@xos-node1 ~]# poweroff
```

31

The `xos-node1` virtual machine is cloned as following:

```
yjegou@host:~$ sudo virt-clone -o xos-node1 -n xos-node2 \
               -f /var/lib/libvirt/images/xos-node2.img -m 54:52:00:00:10:02
This will overwrite the existing path '/var/lib/libvirt/images/xos-node2.img'!
Do you really want to use this disk (yes or no)? yes
Cloning /var/lib/libvirt/ 100% |=========================| 6.8 GB    01:59

Clone 'xos-node2' created successfully.
```

The new virtual machine can be booted now from the KVM virtual machine manager. This virtual machine gets an IP address from the host using DHCP and the MAC address provided to `virt-clone`. This virtual machine comes with users and ssh already configured.

```
yjegou@host:~$ ssh yvon@xos-node2
The authenticity of host 'xos-node2 (192.168.122.12)' can't be established.
RSA key fingerprint is ed:8b:df:f3:89:07:e5:54:c9:e4:65:6c:cd:ae:32:c6.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'xos-node2,192.168.122.12' (RSA) to the list of known hosts.
Last login: Tue Sep 28 11:25:28 2010 from 192.168.122.1
[yvon@xos-node1 ~]$ su -
Password:
[root@xos-node1 ~]# ifconfig
eth1      Link encap:Ethernet  HWaddr 54:52:00:00:10:02
          inet addr:192.168.122.12  Bcast:192.168.122.255  Mask:255.255.255.0
          inet6 addr: fe80::5652:ff:fe00:1002/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
...
```

Note that, also the node is assigned its own IP address, the host name is still the source host name of the clone. The host name will be fixed later when `xosautoconfig` is run. The IP address is assigned to interface `eth1`. This may result in problems as some of the XtreemOS services expect sockets to be bound to a fixed interface. The node is using `eth1` because the MAC address in use by the cloned node remains associated to `eth0`:

```
[root@xos-node1 rules.d]# cat 70-persistent-net.rules
# This file was automatically generated by the /lib/udev/write_net_rules
# program run by the persistent-net-generator.rules rules file.
#
# You can modify it, as long as you keep each rule on a single line.

# Drakx-net rule for eth0 (54:52:00:00:10:01)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTRaddress=="54:52:00:00:10:01",\
 ATTRtype=="1",  KERNEL=="eth*", NAME="eth0"

# PCI device 0x10ec:0x8139 (8139too)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTRaddress=="54:52:00:00:10:02",\
 ATTRtype=="1",  KERNEL=="eth*", NAME="eth1"

# PCI device 0x10ec:0x8139 (8139too)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",  ATTRaddress=="54:52:00:00:10:02",\
 ATTRtype=="1",  KERNEL=="eth*", NAME="eth1"
```

The solution in a Mandriva distribution is to edit these rules and to assign `eth0` to the local MAC address:

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",  ATTRaddress=="54:52:00:00:10:02",\
 ATTRtype=="1",  KERNEL=="eth*", NAME="eth0"
```

And reboot.

```
[root@xos-node1 rules.d]# reboot; exit

Broadcast message from root (pts/1) (Tue Sep 28 17:53:04 2010):
```

```
The system is going down for reboot NOW!
logout

[yvon@xos-node1 ~]$ Connection to xos-node2 closed by remote host.
Connection to xos-node2 closed.
yjegou@host:~$ ssh root@xos-node2
Last login: Tue Sep 28 10:53:53 2010 from 192.168.122.1
[root@xos-node2 ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 54:52:00:00:10:02
          inet addr:192.168.122.12  Bcast:192.168.122.255  Mask:255.255.255.0
          inet6 addr: fe80::5652:ff:fe00:1002/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

After reboot, interface eth0 is in use. However, the node still keeps references to the original node in configuration files. For instance:

```
[root@xos-node2 ~]# cat /root/.xos/XATICAConfig.conf
xosdaddress.host=192.168.122.11
xosdaddress.port=55000
address.host=192.168.122.11
address.port=10001
...
```

6.2 Re-configuring with `xosautoconfig`

Running xosautoconfig regenerates the configuration files:

```
[root@xos-node2 ~]# xosautoconfig
* Checking configuration files and directories...
/etc:                                                       [ OK ]
/root:                                                      [ OK ]
/etc/xos/xosautoconfig/conf:                                [ OK ]
...
Starting xosd: nohup: redirecting stderr to stdout
                                                            [  OK  ]
Service eu.xtreemos.xosd.security.rca.server.RCAServer is not running\
 on any known node.
* apply for a resource certificate
Requesting a new certificate...
Resource ResourceID = [IP=192.168.122.12:60000] not registered.
* configuration of this node is suspended:

run script confirmResource on the rca node

and terminate the configuration with\
 script finishConfig on this node
* note: a rca_apply request has already been issued. First check with
rca_list_pending that it has been\
 recorded. If the list is empty
execute rca_apply before running\
 confirmResource on the rca node
* xosautoconfig finished its job. To review all changed configuration files
* head out to /etc/xos/xosautoconfig/backup-100928-1800
```

Note that the resource certificate must also be changed.

```
[root@xos-core ~]# /usr/lib/xos/xosautoconfig/confirmResource
 I am running the RCA server, confirm rca_apply
rca_confirm 192.168.122.12:60000
```

```
root@xos-node2 ~]# /usr/lib/xos/xosautoconfig/finishConfig
...
Requesting a new certificate...
Identity certificate:
  DN: C=FR,L=Rennes,OU=Myriads,O=INRIA,CN=Address = \
 [://192.168.122.12:60000(192.168.122.12)]
  serial number: 1285689940553
  issuer DN: O=INRIA,OU=rca,CN=xos-core.xtreemos.eu/rca
  validity start: Tue Sep 28 18:00:40 CEST 2010
  validity end: Thu Oct 28 18:10:40 CEST 2010
Attributes of attribute certificate:
 (attributes in extensions)
  MemorySize = 7.21420288E8
  Service =
    eu.xtreemos.system.communication.redirector.ServiceCallRedirector
    eu.xtreemos.xosd.resallocator.ResAllocator
    eu.xtreemos.ads.connection.dixi.SRDSMng
    eu.xtreemos.xosd.resourcemonitor.ResourceMonitor
    eu.xtreemos.xosd.security.rca.client.RCAClient
    eu.xtreemos.xosd.resmng.ResMng
    eu.xtreemos.xosd.daemon.Daemon
    eu.xtreemos.xosd.xmlextractor.XMLExtractor
    eu.xtreemos.xosd.crondaemon.CronDaemon
    eu.xtreemos.xosd.execMng.ExecMng

  CPUCount = 1
  CPUSpeed = 2.927624192E9
Attribute VO certificate /etc/xos/truststore/certs/attrcert70ac51ce\
-d716-4d5b-9522-076d1b7f1396ext.crt:
  MemorySize = 7.21420288E8
  Service =
    eu.xtreemos.system.communication.redirector.ServiceCallRedirector
    eu.xtreemos.xosd.resallocator.ResAllocator
    eu.xtreemos.ads.connection.dixi.SRDSMng
    eu.xtreemos.xosd.resourcemonitor.ResourceMonitor
    eu.xtreemos.xosd.security.rca.client.RCAClient
    eu.xtreemos.xosd.resmng.ResMng
    eu.xtreemos.xosd.daemon.Daemon
    eu.xtreemos.xosd.xmlextractor.XMLExtractor
    eu.xtreemos.xosd.crondaemon.CronDaemon
    eu.xtreemos.xosd.execMng.ExecMng

  VO = 70ac51ce-d716-4d5b-9522-076d1b7f1396
  CPUCount = 1
  CPUSpeed = 2.927624192E9

configuration terminated,
              good luck :-)
```

No need to configure the local policies.

```
[root@xos-node2 ~]# ssh-xos localhost
Enter passphrase for key '/root/.xos/truststore/private/user.key':
Last login: Tue Sep 28 10:23:06 2010 from xos-node1.xtreemos.eu
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' already ...
-bash-3.2$ ls
psx.err*  psx.out*  tmp/
-bash-3.2$
```

6.3 Attaching the clone node to a VO

The resource certificates have been renewed by `xosautoconfig`. But the VO certificate of the cloned node is still in place and should be replaced:

```
[root@xos-node2 ~]# openssl x509 -noout -text -in \
 /etc/xos/truststore/certs/attrcert70ac51ce-d716-4d5b-9522-076d1b7f1396ext.crt
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            01:2b:57:98:2e:ee
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: O=INRIA, OU=rca, CN=xos-core.xtreemos.eu/rca
        Validity
            Not Before: Sep 28 09:01:52 2010 GMT
            Not After : Oct 28 09:11:52 2010 GMT
        Subject: C=FR, L=Rennes, OU=Myriads, O=INRIA,\
  CN=Address = [://192.168.122.11:60000(192.168.122.11)]
        Subject Public Key Info:
```

Use `rca_resource_vo` to renew the VO certificate.

```
[root@xos-node2 ~]# rca_resource_vo a 70ac51ce-d716-4d5b-9522-076d1b7f1396
Adding self to the VO.
Added resource to VO 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/incoming/.
[root@xos-node2 ~]# rca_resource_vo c 70ac51ce-d716-4d5b-9522-076d1b7f1396
The RCA client received the certificate for VO\
 70ac51ce-d716-4d5b-9522-076d1b7f1396. Please check /etc/xos/truststore/certs/.
```

It is now possible to submit jobs.

Final step: edit all user's config files `~/.xos/XATICAConfig.conf` and `~/.xos/XATIConfig.conf`. Replace the IP address of the cloned node by the current node address in both files.

And good luck!

7 User environments

Sections 5 and 6 described the configuration of resource nodes. This section describes the configuration of a user account on an XtreemOS node which allows the user to log in using `ssh` and then to submit and monitor jobs in a VO. Those not willing to configure such a user environment can skip this section.

First, register a local user (UNIX) on the node.

```
yjegou@host:~$ ssh -A root@xos-node1
Last login: Wed Oct  6 18:08:50 2010 from 192.168.122.1

[root@xos-node1 ~]# adduser xuser
[root@xos-node1 ~]# passwd xuser
Changing password for user xuser.
New UNIX password:
BAD PASSWORD: it is based on a dictionary word
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

Copy the SSH authorized key.

```
yjegou@host:~$ scp /home/yjegou/.ssh/authorized_keys2 xuser@xos-node1:.ssh/
xuser@xos-node1's password:
authorized_keys2                                100%  397     0.4KB/s   00:00
```

Get certificates from a VO.

```
[xuser@xos-node1 ~]$ get-xos-cert xos-core:6730 vo-yvon users -u yvon -p xtreemos\
 -k ~/.xos/truststore/private/user.key -c ~/.xos/truststore/certs/user.crt
Passphrase to protect private key (at least 8 characters long):
Type passphrase again to confirm:
Generating a new public/private key pair
Warning: certificate presented by remote host xos-core belongs to xos-core.xtreemos.eu -
carrying on as you requested to ignore CDA host certificates which don't belong to the\
 CDA server you connected to.
You should only use any credentials issued for testing purposes.

Saving certificate chain (user+CDA) in /home/xuser/.xos/truststore/certs/user.crt.
```

Generate ~/.xos/XATICAConfig.conf configuration file: calling xps -a (or any other grid command) generates a default file. Update this file. Note that it is also possible to copy this file from /root/.xos and then update the user path.

```
[xuser@xos-node1 ~]$ cat .xos/XATICAConfig.conf
xosdaddress.host=192.168.122.11
xosdaddress.port=55000
address.host=192.168.122.11
address.port=10001
certificateLocation=/etc/xos/truststore/certs/
privateKeyLocation=/etc/xos/truststore/private/
trustStoreSSL=/etc/xos/truststore/certs/
useSSL=false
userCertificateFile=/home/xuser/.xos/truststore/certs/user.crt
passwd=12345678
```

Check that the configuration file is OK.

```
[xuser@xos-node1 ~]$ xps -a
```

Execute xconsole_dixi in order to generate the default ~/.xos/XATIConfig.conf file.

```
[xuser@xos-node1 ~]$ xconsole_dixi
XtreemOS Console

$ exit
Bye
```

Edit this file.

```
[xuser@xos-node1 ~]$ cat .xos/XATIConfig.conf
#Properties File for the client application
#Thu Oct 07 15:34:39 CEST 2010
loadPrivateKey=false
useSSL=false
xosdaddress.externalAddress=192.168.122.11
sslPrivateKeyPassword=12345678
xosdaddress.host=192.168.122.11
privateKeyLocation=/home/xuser/.xos/truststore/private/user.key
userKeyFile=/home/xuser/.xos/truststore/private/user.key
schemasLocation=/usr/share/dixi/XMLExtractor/Schemas/
networkInterface=
trustStoreSSL=/etc/xos/truststore/certs/
address.host=192.168.122.11
userCertificateFile=/home/xuser/.xos/truststore/certs/user.crt
xosdaddress.port=60000
address.port=10000
certificateLocation=/home/xuser/.xos/truststore/certs/user.crt
```

Edit `~/.ssh/config-xos`: replace `$HOME` by user's home directory.

```
[xuser@xos-node1 ~]$ cat /etc/skel/.ssh/config-xos
XosProxyFile /home/xuser/.xos/truststore/certs/user.crt
XosPrivKeyFile /home/xuser/.xos/truststore/private/user.key
XosVoName XXX
```

Check that `ssh-xos` is OK.

```
[xuser@xos-node1 ~]$ ssh-xos localhost
Enter passphrase for key '/home/xuser/.xos/truststore/private/user.key':
Last login: Thu Oct  7 10:26:03 2010 from xos-core.xtreemos.eu
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e'\
 already exists in Directory Service, id='f0c660d6-71b1-4d48-9557-50d6fac83e14'\
 (errno=17)
-bash-3.2$ ls
psx.err*  psx.out*  tmp/
```

Note that files `psx.err` and `psx.out` are the outputs of the jobs initially submitted from the core node during configuration.

8   Hints and Troubleshooting

8.1   Storage Volume Format

Using the `raw` storage volume format on KVM has been reported to result in disk corruption on some distributions. See http://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=176. Format `qcow2` does not have the same issues.

8.2   SSH identification change

SSH refuses to connect with the following message.

```
yjegou@host:~$ ssh yvon@xos-core
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!    @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
21:60:89:d2:4f:6d:54:f5:db:f0:0a:21:eb:02:91:eb.
Please contact your system administrator.
Add correct host key in /home/yjegou/.ssh/known_hosts to get rid of this ...
Offending key in /home/yjegou/.ssh/known_hosts:98
RSA host key for xos-core has changed and you have requested strict checking.
Host key verification failed.
```

You have already logged on some node with the same name or IP address. SSH keeps track of already known nodes and expects a node to always connect using the same key. To solve this problem, remove the offending line from file `~/.ssh/known_hosts` using your favorite text editor.

8.3   SSH identification change

`SSH` prints the following warning the first time it logs in some node.

```
yjegou@host:~$ ssh yvon@xos-core
The authenticity of host 'xos-core (192.168.122.10)' can't be established.
RSA key fingerprint is 21:60:89:d2:4f:6d:54:f5:db:f0:0a:21:eb:02:91:eb.
```

```
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'xos-core,192.168.122.10' (RSA) to the list of ...
yvon@xos-core's password:
[yvon@xos-core ~]$
```

8.4 Mandriva mirror selection

Tool xosautoconfig initiates a mirror selection process on Mandriva repositories the first time it is executed. This process may fail if the node has not full access to the Internet, for instance when it is located behind a firewall. Setting field SETMEDIA to false in file localDefs of xosautoconfig disables this automatic mirror selection process. It is then possible to manually configure the XtreemOS repository using urpmi.removemedia and urpmi.addmedia. On the other hand, setting SETMEDIA to true in file localDefs will force a new selection process of XtreemOS mirrors next time xosautoconfig is executed.

8.5 VOlife does not run

```
[root@xos-core xos]# volife_run.sh --create-user yvon-admin xtreemos \
Yvon Jegou INRIA Yvon.Jegou@inria.fr

15:10:31,194  WARN ThreadPoolAsynchronousRunner:608 - com.mchange.v2.\
async.ThreadPoolAsynchronousRunner$DeadlockDetector@20f443 -- APPARENT\
DEADLOCK!!! Creating emergency threads for unassigned pending tasks!
15:10:31,206  WARN ThreadPoolAsynchronousRunner:624 - com.mchange.v2.\
async.ThreadPoolAsynchronousRunner$DeadlockDetector@20f443 -- APPARENT\
DEADLOCK!!! Complete Status:
        Managed Threads: 3
        Active Threads: 3
        Active Tasks:
                com.mchange.v2.resourcepool.BasicResourcePool$Acquire\
Task@671f95 (com.mchange.v2.async.ThreadPoolAsynchronousRunner$PoolThread-#0)
```

If VOlife does not behave correctly (web interface as well as command-line interfaces) on a fresh install, try to re-initialize the XVOMS data-base:

```
[root@xos-core ~]# /usr/share/xvoms/bin/xvoms_init.sh
 - Create xvoms database entry and password
Enter password for MySQL root user:
Retype password for MySQL root user:
Loading default data into xvoms database

 - Mysqld configuration (accept network connection)
Shutting down MySQL: .....                                       [  OK  ]
Starting MySQL: .                                               [  OK  ]
```

8.6 cdaserver does not run

A possible reason for the cdaserver service to fail is an incorrect access right to certificates. Service cdaserver runs with ID/group cdauser:cdauser: access rights to the XtreemOS root certificate(s) and cda certificate/key must allow read for others. The cdaserver logs are located in /var/log/cdaserver/cdaserver.log. Insufficient access rights to the certificate result in:

```
[root@xos-core cdaserver]# less cdaserver.log
...
24 Sep 2010 14:46:15,874  INFO cda:340 - started
java.io.FileNotFoundException: /etc/xos/truststore/certs/xtreemos.crt\
 (Permission denied)
```

8.7 XtreemFS services do not start

Check the `xtreemfs` service logs in `/var/log/xtreemfs`. A possible cause of error on a fresh install is a failure during the execution of the XtreemFS post-install scripts. For instance the following error means that the `database` directory for the `mrc` service was not created in `/var/lib/xtreemfs` during installation.

```
...
[ E | MRCRequestDispatcher | MRC | 1 | Sep 20 11:42:53 ] STARTUP FAILED!
[ E | MRCRequestDispatcher | MRC | 1 | Sep 20 11:42:53 ] org.xtreemfs.mrc.\
database.DatabaseException: org.xtreemfs.babudb.BabuDBException: cannot\
 start database operations logger (error code: IO_ERROR)
 ... org.xtreemfs.mrc.database.babudb.BabuDBVolumeManager.init\
(BabuDBVolumeManager.java:112)
 ... org.xtreemfs.mrc.MRCRequestDispatcher.startup(MRCRequestDispatcher.\
java:325)
 ... org.xtreemfs.mrc.MRC.<init>(MRC.java:57)
 ... org.xtreemfs.mrc.MRC.main(MRC.java:122)
[ E | MRCRequestDispatcher | MRC | 1 | Sep 20 11:42:53 ] root cause:\
 org.xtreemfs.babudb.BabuDBException: cannot start database operations\
 logger (error code: IO_ERROR)
 ... org.xtreemfs.babudb.BabuDB.<init>(BabuDB.java:199)
 ... org.xtreemfs.babudb.BabuDBFactory.createBabuDB(BabuDBFactory.java:31)
 ... org.xtreemfs.mrc.database.babudb.BabuDBVolumeManager.init\
(BabuDBVolumeManager.java:109)
 ... org.xtreemfs.mrc.MRCRequestDispatcher.startup(MRCRequestDispatcher.\
java:325)
 ... org.xtreemfs.mrc.MRC.<init>(MRC.java:57)
 ... org.xtreemfs.mrc.MRC.main(MRC.java:122)
```

Creating this directory manually will correct the problem. An alternative is to re-run the post-install script located in `http://code.google.com/p/xtreemfs/source/browse/branches/XtreemFS-1.2.2/packaging/postinstall_setup.sh`. See bug `https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=288` for more info.

8.8 Certificate `/etc/xos/truststore/certs/resource.crt` is missing

Certificate `/etc/xos/truststore/certs/resource.crt` of a core node is generated by the sequence of requests `rca_apply`, `rca_confirm` and `rca_request`. Request `rca_confirm` must be run on the node running the RCA server.

8.9 `rca_resource_vo c VOID` returns "resource not member of VO"

From time to time, `rca_resource_vo c VOID` prints message `Resource ResourceID = [IP=192.-168.122.10:60000] is not a member of VO VOID`. In this case, try to execute `rca_vo a VOID` on the core node before executing the sequence `rca_resource_vo a` and `rca_resource_vo c` on the node.

8.10 `xos-policy-admin-chk` fails

First, note that `xos-policy-admin-chk` must be run as `root`. If `xos-policy-admin-chk` returns `errno = 17, ...` before printing `Sucess in PAM checking !`, simply ignore this message. The configuration is OK. The message is due to a (missing) feature of XtreemFS client described in Section 8.11.

If `xos-policy-admin-chk` returns with message `Oops: Permission denied`, the local policies have not been configured to accept users of this VO:

```
[root@xos-core ~]# xos-policy-admin-chk  \
                         -pem /root/.xos/truststore/certs/yvon.crt
dn = [/CN=663a6798-3d16-4923-a5da-e829a03f057e], \
vo = [70ac51ce-d716-4d5b-9522-076d1b7f1396], role = [null]
PAM:fail in mapping connect !
```

```
        * a)Please check whether AMS daemon is running correctly *
        * b)Please check whether mapping rules are correct.      *
        *     If not, try:                                       *
        *        xos-policy-admin-am  -vo <vo> --force            *
        *        xos-policy-admin-gm  -vo <vo> --force            *
        * c)Please check whether setting rule is correct.        *
        *     If not, try:                                       *
        *        xos-policy-admin-set -uidmax <num> -uidmin <num> *
        *                            -gidmax <num> -gidmin <num> *
Oops: Permission denied
```

You must re-run the following commands as `root`. The missing VO ID can be copied from the previous command.

```
[root@xos-core ~]# xos-policy-admin-am \
                   -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396 \
                   --force
[root@xos-core ~]# xos-policy-admin-gm \
                   -vo 70ac51ce-d716-4d5b-9522-076d1b7f1396 \
                   --force
```

Note that the VOID to specify in these commands is printed by `xos-policy-admin-chk`.

8.10.1 `/etc/pki/tls/cert.pem`

On a fresh installation, `xos-policy-admin-chk` can result in the following error also all certificates seem OK and validated using `openssl verify`:

```
[root@xos-core ~]# xos-policy-admin-chk  \
                            -pem /root/.xos/truststore/certs/yvon.crt
dn = [/CN=663a6798-3d16-4923-a5da-e829a03f057e], \
vo = [70ac51ce-d716-4d5b-9522-076d1b7f1396], role = [null]
Error: unable to get local issuer certificate
Error verifying the certificate
Oops: Permission denied

Hints:
(1) Have you correctly configured /etc/pam.d/pam_app_conv ?
(2) Have you had valid certificate ?
```

This error is related to the presence of certificate `/etc/pki/tls/cert.pem` in the default Mandriva distribution (https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=193). This certificate is used as the default trust anchor by openssl. The same problem happens with `openssl verify` if the root anchor is not specified:

```
[root@xos-core ~]# openssl verify -CApath /etc/xos/truststore/certs/ \
                            /root/.xos/truststore/certs/yvon.crt
/root/.xos/truststore/certs/yvon.crt:   /CN=663a6798-3d16-4923-a5da-e829a03f057e
error 20 at 0 depth lookup:unable to get local issuer certificate
```

The solution is to remove (or rename) this certificate:

```
[root@xos-core ~]# cd /etc/pki/tls
[root@xos-core tls]# mv cert.pem cert.pem-org
[root@xos-core tls]# cd
[root@xos-core ~]# openssl verify -CApath /etc/xos/truststore/certs/ \
                            /root/.xos/truststore/certs/yvon.crt
/root/.xos/truststore/certs/yvon.crt: OK
```

The removal of this certificate allows a correct behavior of `openssl`.

8.11 XtreemFS

The XtreemFS client provides no means to check if a volume exists, which prevents the XtreemOS automounter to check if the home volume must be created the first time it is mounted (https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=105). This missing feature results in error 17 emitted by the automounter.

```
errno = 17, strerror = volume 'vol-663a6798-3d16-4923-a5da-e829a03f057e' \
already exists in Directory Service, \
id='f0c660d6-71b1-4d48-9557-50d6fac83e14' (errno=17)
```

8.12 `ssh-xos`

Bug https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=249 prevents users to log in a node with ssh-xos if a UNIX account corresponding to his user name does not exist on the remote node. This bug does not prevent root to log in as root account exists on all nodes. The workaround to this bug is to specify root as user name in ssh-xos requests as in the following example:

```
[yvon@xos-core ~]$ ssh-xos root@xos-core
```

Note that this bug should not affect our installation as long as there a single node in our grid.

8.13 Grid account mapping

Bug https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=246 can result in an incorrect mapping of user credentials on a local account: the result is a user logged in the account of some other user or, in some cases, as root. This bug affects job execution (the application is run using an incorrect account) as well as ssh-xos (user get logged in the account of another user, possibly root).

8.14 Missing services

If some services managed by xosd seems to be missing, for instance service JobMng when you call xps, you should restart service xosd.

```
[root@xos-node1 .xos]# xps -a
xps: Error getting user jobs: -30 (Service not running)
Service eu.xtreemos.xosd.jobmng.JobMng is not running on any known node.
[root@xos-node1 .xos]# service xosd restart
Stopping xosd:                                              [  OK  ]
Starting xosd: nohup: redirecting stderr to stdout
                                                           [  OK  ]
[root@xos-node1 .xos]# xps -a
```

8.15 No job run on some resource node

If some resource node never receive jobs for execution, check that a valid certificate is present.

```
[root@xos-node1 .xos]# xconsole_dixi
$ xps -a
 JobID                           -         Submit Time -     Job State
    * Resource Address:port
           + PID    - User Time - Sys Time - Proc State
 de5c258f-d99f-4f07-b90b-5061cc8c4de6 -  Tue Sep 28 11:04:23 -        Done
    * ://192.168.122.10:60000(192.168.122.10)
 aefdbb39-b467-4923-ba3d-f967cca417ae -  Tue Sep 28 11:05:22 -        Done
    * ://192.168.122.10:60000(192.168.122.10)
 f8288812-34aa-45a7-813c-067de3566c66 -  Tue Sep 28 11:05:30 -        Done
    * ://192.168.122.10:60000(192.168.122.10)
```

All job seem to go to xos-core (192.168.122.10). Try the following.

```
[root@xos-node1 .xos]# xconsole_dixi
$ xrs -jsdl /etc/skel/psx.jsdl
Listing resources matching JSDL query:
  Address = [://192.168.122.10:60000]
$ exit
Bye
```

xos-node1 (192.168.122.11) does not appear in the list of nodes accepting jobs. Check that the VO certificate corresponding to the VO of the user certificate is present in /etc/xos/truststore/certs.

```
[root@xos-node1 .xos]# ls /etc/xos/truststore/certs/
052cc605.0@      b1479a7e.0@  incoming/        vops.crt
863d161c.0@      cda.crt      rcaserver.crt  xtreemos.crt
attrextcert.crt  d833eb57.0@  resource.crt
```

No VO certificate here. Re-run the rca_resource_vo for the VO:

```
[root@xos-node1 .xos]#  rca_resource_vo c 70ac51ce-d716-4d5b-9522-076d1b7f1396
Resource ResourceID = [IP=192.168.122.11:60000] is not a member of VO\
 70ac51ce-d716-4d5b-9522-076d1b7f1396.
[root@xos-node1 .xos]#  rca_resource_vo a 70ac51ce-d716-4d5b-9522-076d1b7f1396
Adding self to the VO.
Added resource to VO 70ac51ce-d716-4d5b-9522-076d1b7f1396. Please check\
 /etc/xos/truststore/certs/incoming/.
[root@xos-node1 .xos]# ls /etc/xos/truststore/certs/incoming/
attrcert70ac51ce-d716-4d5b-9522-076d1b7f1396ext.crt
[root@xos-node1 .xos]#  rca_resource_vo c 70ac51ce-d716-4d5b-9522-076d1b7f1396
The RCA client received the certificate for VO 70ac51ce-d716-4d5b-9522-076d1b7f1396.\
 Please check /etc/xos/truststore/certs/.
[root@xos-node1 .xos]# ls /etc/xos/truststore/certs/
052cc605.0@                                          b1479a7e.0@  rcaserver.crt
863d161c.0@                                          cda.crt      resource.crt
attrcert70ac51ce-d716-4d5b-9522-076d1b7f1396ext.crt  d833eb57.0@  vops.crt
attrextcert.crt                                      incoming/    xtreemos.crt
```

Try first to run rca_resource_vo c <VOID>. If this request fails (see bug https://sourceforge.net/apps/mantisbt/xtreemos/view.php?id=274), run rca_resource_vo a <VOID>. The certificate if OK when a file named attrcert<VOID>ext.crt is present in directory /etc/xos/truststore/certs/. Here is the correct behavior.

```
[root@xos-node1 .xos]# xconsole_dixi
XtreemOS Console

$ xps -a
There are no jobs running in the system
$ xrs -jsdl /etc/skel/psx.jsdl
Listing resources matching JSDL query:
  Address = [://192.168.122.10:60000]
  Address = [://192.168.122.11:60000]
$ xsub -f /etc/skel/psx.jsdl
Job submitted succesfully: 04991ea7-c5bb-4eba-a28a-76f11f081eb5
$ xsub -f /etc/skel/psx.jsdl
Job submitted succesfully: 6d59230a-c34f-4f16-97dc-6ead9643427f
$ xsub -f /etc/skel/psx.jsdl
Job submitted succesfully: bf05dabe-f7a2-4429-b755-ba45bda774e0
$ xps -a
 JobID                             -          Submit Time -      Job State
    * Resource Address:port
            + PID    - User Time - Sys Time - Proc State
 6d59230a-c34f-4f16-97dc-6ead9643427f -  Tue Sep 28 11:22:46  -        Done
```

```
  * ://192.168.122.11:60000(192.168.122.11)
 04991ea7-c5bb-4eba-a28a-76f11f081eb5 -  Tue Sep 28 11:22:43  -         Done
  * ://192.168.122.10:60000(192.168.122.10)
 bf05dabe-f7a2-4429-b755-ba45bda774e0 -  Tue Sep 28 11:22:48  -         Done
  * ://192.168.122.11:60000(192.168.122.11)
$ exit
Bye
```

Nodes `xos-core` (`192.168.122.10`) and `xos-node1` (`192.168.122.11`) have executed jobs.

9   Conclusion

Good luck!