xCAT 2.0 Roadrunner Cookbook

5/7/2008

Table of Contents

1.0 Release Description

xCAT 2.0 is a complete rewrite of xCAT 1.2/1.3, implementing a new architecture. See the xCAT2.0 Cookbook for more details about the 2.0 product: http://xcat.svn.sourceforge.net/svnroot/xcat/xcat-core/trunk/xCAT-client-2.0/share/doc/xCAT2.0.pdf . All commands are client/server, authenticated, logged and policy driven. The clients can be run on any OS with Perl, including Windows. The code has been completely rewritten in Perl, and table data is now stored in a relational database.

2.0 Installing the Management Node

Before beginning, ensure that your networks are setup correctly.

2.1 Download Fedora 8 and Create Repository

1. Get Fedora ISOs and place in a directory, for example /root/xcat2:
```
mkdir /root/xcat2
cd /root/xcat2
wget
   ftp://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/x86_64/iso/Fedora-8-x86_64-DVD.iso
wget
   ftp://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/ppc/iso/Fedora-8-ppc-DVD.iso
```

2. Create YUM repository for Fedora RPMs:

```
   mkdir /root/xcat2/fedora8
   mount -r -o loop /root/xcat2/Fedora-8-x86_64-DVD.iso /root/xcat2/fedora8

   cd /etc/yum.repos.d
   mkdir ORIG
   mv fedora*.repo ORIG

   Create fedora.repo with contents:

   [fedora]
   name=Fedora $releasever - $basearch
   baseurl=file:///root/xcat2/fedora8
   enabled=1
   gpgcheck=0
```

3. Install createrepo:
   `yum install createrepo`

2.2 Downloading and Installing xCAT 2.0

2.2.1 If Your Management Node <u>Has</u> Internet Access:

2.2.1.1 Download Repo Files

YUM can be pointed directly to the xCAT download site.

```
cd /etc/yum.repos.d
wget http://xcat.sf.net/yum/core-snap/xCAT-core-snap.repo
wget http://xcat.sf.net/yum/dep-snap/rh5/x86_64/xCAT-dep-snap.repo
```

1.1.1.1 Set Up Repo File for Fedora Site

Create fedora-internet.repo:

```
[fedora-everything]
name=Fedora $releasever - $basearch
failovermethod=priority
#baseurl=http://download.fedora.redhat.com/pub/fedora/linux/releases/
   $releasever/Everything/$basearch/os/
mirrorlist=http://mirrors.fedoraproject.org/mirrorlist?repo=fedora-
   $releasever&arch=$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora file:///etc/pki/rpm-gpg/RPM-GPG-
   KEY
```

Continue now at step Install xCAT 2.0 software & Its Dependencies.

2.2.2 If Your Management Node Does Not Have Internet Access:

2.2.2.1 Download xCAT2.0 and Its Dependencies

Note: do the wget's on a machine with internet access and copy the files to the management node.

```
cd /root/xcat2
wget http://xcat.sf.net/yum/core-rpms-snap.tar.bz2
wget http://xcat.sf.net/yum/dep-rpms-snap.tar.bz2
tar jxvf core-rpms-snap.tar.bz2
tar jxvf dep-rpms-snap.tar.bz2
```

2.2.2.2 Get Fedora 8 OSS dependencies

```
cd /root/xcat2/dep-snap/rh/x86_64

wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-Net-SNMP-5.2.0-1.fc8.1.noarch.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-XML-Simple-2.17-1.fc8.noarch.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-Crypt-DES-2.05-4.fc7.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/net-snmp-perl-5.4.1-4.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/ksh-20070628-1.1.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-IO-Socket-INET6-2.51-2.fc8.1.noarch.rpm
wget
http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/os
   /Packages/dhcp-3.0.6-10.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/syslinux-3.36-7.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/mtools-3.9.11-2.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/expect-5.43.0-9.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-DBD-SQLite-1.12-2.fc8.1.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-Expect-1.20-1.fc8.1.noarch.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-IO-Tty-1.07-2.fc8.1.x86_64.rpm
```

```
wget
   http://mirrors.usc.edu/pub/linux/distributions/fedora/linux/releases/8/Everythin
   g/x86_64/os/Packages/scsi-target-utils-0.0-1.20070803snap.fc8.x86_64.rpm
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86_64/
   os/Packages/perl-Net-Telnet-3.03-5.1.noarch.rpm
```

2.2.2.3 Setup YUM repositories for xCAT and Dependencies

```
cd /root/xcat2/dep-snap/rh5/x86_64
./mklocalrepo.sh
cd /root/xcat2/core-snap
./mklocalrepo.sh
```

2.2.3 Install xCAT 2.0 software & Its Dependencies

```
yum clean metadata
yum install xCAT.x86_64
```

2.2.4 Test xCAT installation

```
source /etc/profile.d/xcat.sh
tabdump site
```

2.2.5 Update xCAT 2.0 software

If you need to update the xCAT 2.0 rpms later, download the new version of http://xcat.sf.net/yum/core-rpms-snap.tar.bz2 (if the management node does not have access to the internet) and then run:

```
yum update '*xCAT*'
```

If you have a service node stateless image, don't forget to update the image with the new xCAT rpms (see Build the service node stateless image):

```
cp -pf /etc/yum.repos.d/*.repo
   /install/netboot/fedora9/x86_64/service/rootimg/etc/yum.repos.d
yum --installroot=/install/netboot/fedora8/x86_64/service/rootimg update '*xCAT*'
packimage -o fedora8 -p service -a x86_64
```

2.2.6 Setup Yum for Fedora8 Node Installs

```
umount /root/xcat2/fedora8
cd /root/xcat2
copycds Fedora-8-x86_64-DVD.iso
copycds Fedora-8-ppc-DVD.iso
```

The copycds commands will copy the contents of the DVDs to /install/fedora8/<arch>.

Edit /etc/yum.repos.d/fedora.repo and change:

```
baseurl=file:///root/xcat2/fedora8
```

to

```
baseurl=file:///install/fedora8/x86_64
```

3.0 xCAT Hierarchy using Service nodes

In large clusters it is desirable to have more than one node (the Management Node) handle the installation of the compute nodes. We call these additional nodes service nodes. You can have one or more service nodes setup to install groups of compute nodes.

The service nodes need to communicate with the xCAT2.0 database on the Management Node and run xCAT command to install the nodes. The service node will be installed with the xCAT code and required the PostgreSQL Database be setup instead of SQLite Default database. PostgreSQL allows a client to be setup on the service node such that the service node can access (read/write) the database on the Management Node (Master Node) from the service node.

If you do not plan on using service nodes, you can skip this section 3 and continue to use the SQLite Default database setup during the installation.

3.1 Switching to PostgreSQL Database

To setup the postgresql database on the Management Node follow these steps.

This example assumes:

- 192.168.0.1: ip of master
- xcatdb: database name
- xcatadmin: database role (aka user)
- cluster: database password
- 192.168.0.10 & 192.168.0.11: service nodes

Substitute your address and desired userid , password and database name as appropriate.

The following rpms should be installed from the Fedora8 media on the Management Node (and service node when installed). These are required for postgresql.

```
1. yum install perl-DBD-Pg postgresql-server postgresql
```

2. Initialize the database :

```
service postgresql initdb
```

3. `service postgresql start`

4. `su - postgres`

5.
   ```
   createuser -P xcatadmin
   Enter password for new role: cluster
   Enter it again: cluster
   Shall the new role be a superuser? (y/n) n
   Shall the new role be allowed to create databases? (y/n) n
   Shall the new role be allowed to create more new roles? (y/n) n
   ```

6. `createdb -O xcatadmin xcatdb`

7. `exit`

8. `cd /var/lib/pgsql/data/`

9. `vi pg_hba.conf`

   Lines should look like this (with your IP addresses substituted). This allows the service nodes to access the DB.

   ```
   local all all ident sameuser
   # IPv4 local connections:
   host all all 127.0.0.1/32 md5
   host all all 192.168.0.1/32 md5
   host all all 192.168.0.10/32 md5
   host all all 192.168.0.11/32 md5
   ```

   where 192.168.0.10 and 11 are service nodes.

10. `vi postgresql.conf`

    set listen_addresses to '*':

    ```
    listen_addresses = '*'   This allows remote access.
    ```

    Note: Be sure to uncomment the line

11. `service postgresql restart`

12. `chkconfig postgresql on`

13. Backup your data to migrate to the new database

    ```
    mkdir -p ~/xcat-dbback
    dumpxCATdb -p ~/xcat-dbback
    ```

14. /etc/sysconfig/xcat should contain these lines, substitute your cluster facing address for 192.168.0.1, and user and password are xcatadmin cluster in this instance – skip this??

    ```
    XCATCFG='Pg:dbname=xcatdb;host=192.168.0.1|xcatadmin|cluster'
    export XCATCFG
    ```

```
XCATROOT=/opt/xcat
export XCATROOT
```

15. copy /etc/sysconfig/xcat to /install/postscripts/sysconfig/xcat for installation on the service nodes.
    – skip this??
    ```
    chmod 700 /etc/sysconfig/xcat
    ```

16. /etc/xcat/cfgloc should contain the following line, again substituting your info. This points the xCAT database access code to the new database.

    ```
    Pg:dbname=xcatdb;host=192.168.0.1|xcatadmin|cluster
    ```

17. copy /etc/xcat/cfgloc to /install/postscripts/etc/xcat/cfgloc for installation on the service nodes.
    ```
    mkdir -p /install/postscripts/etc/xcat
    cp /etc/xcat/cfgloc /install/postscripts/etc/xcat/cfgloc
    ```

18. `chmod 700 /etc/xcat/cfgloc`

19. source /etc/sysconfig/xcat #read the text into the current shell – skip this??

20. You can add . /etc/sysconfig/xcat to a setup shell script in /etc/profile.d, so the XCATROOT and XCATCFG environment variables are setup when you login. – skip this??

21. Restore your database to postgresql:
    ```
    XCATBYPASS=1 restorexCATdb -p ~/xcat-dbback
    ```

22. Start the xcatd daemon using the postgresql database
    ```
    service xcatd restart
    ```

23. Run this command to get correct Master node name known by ssl:
    ```
    openssl x509 -text -in /etc/xcat/cert/server-cert.pem -noout|grep Subject:
    ```
    this will display something like:
    ```
    Subject: CN=mgt.cluster
    ```

24. Update the policy table with mgt.cluster output from the command:
    ```
    chtab priority=5 policy.name=<mgt.cluster> policy.rule=allow
    ```
    Note: this name must be an MN name that is known by the service nodes.

25. Make sure the site table has the following settings (using tabdump, tabedit, chtab):

    ```
    #key,value,comments,disable
    "xcatiport","3002",,
    "xcatdport","3001",,
    "master","mn20",,
    ```

where 11.16.0.1 and mn20 are the ip address and hostname of the management node as known by the service nodes.

26. Verify the policy table contains:

```
#priority,name,host,commands,noderange,parameters,time,rule,comments,disable
"1","root",,,,,,"allow",,
"2",,,"getbmcconfig",,,,"allow",,
"3",,,"nextdestiny",,,,"allow",,
"4",,,"getdestiny",,,,"allow",,
"5","mn20",,,,,,"allow",,
```

3.2 Define the service nodes in the database

For this example, we have two service nodes rra000 and rrb000. To add the service nodes to the database run the following commands to add and update the service nodes' attributes in the site, nodelist and noderes tables. Note: service nodes are required to be defined with group “service”. The commands below are using the group “service” to update all service nodes.

Note: For table attribute definitions run tabdump -d <table name>

3.2.1 Define Service Nodes and bmc in nodelist table

```
nodeadd rra000,rrb000 groups=service,ipmi,all
```

3.2.2 Define Service Nodes in noderes table

```
chtab node=service noderes.netboot=pxe noderes.installnic=eth0
   noderes.primarynic=eth0
```

3.2.3 Define Service Nodes in ipmi table

```
chtab node=service ipmi.bmc='|^(.+)$|($1)bmc|' ipmi.username=USERID
   ipmi.password=PASSW0RD
```

3.2.4 Define Service Nodes in nodehm table

```
chtab node=service nodehm.cons=ipmi nodehm.mgt=ipmi  nodehm.serialspeed=19200
   nodehm.serialflow=hard nodehm.serialport=0
```

3.2.5 Define Service Nodes in nodetype table

```
chtab node=service nodetype.arch=x86_64 nodetype.os=fedora8 nodetype.nodetype=osi
```

3.2.6 Set Necessary Attributes in site Table

```
chtab key=defserialport site.value=0
chtab key=defserialspeed site.value=19200
chtab key=xcatservers site.value=rra000,rrb000
```

Note: this last line should be removed when the code no longer uses this.

3.2.7 Define Service Node OS and Profile attributes

```
chtab node=service nodetype.os=fedora8 nodetype.profile=service
```

4.0 Setup Services

4.1 Setup networks Table

All networks in the cluster must be defined in the networks table. When xCAT was installed, makenetworks ran which created an entry in this table for each of the networks the management node is on. We will update the entry for the network for the management node and create one for each CU.

```
chtab net=11.16.0.0 networks.netname=mvnet networks.mask=255.255.0.0
   networks.mgtifname=eth4 networks.gateway=9.114.88.190
   networks.dhcpserver=11.16.0.1 networks.tftpserver=11.16.0.1
   networks.nameservers=11.16.0.1  networks.dynamicrange=11.16.1.210-11.16.1.250
chtab net=11.17.0.0 networks.netname=cuanet networks.mask=255.255.0.0
   networks.mgtifname=eth1 networks.gateway=11.17.255.254
   networks.dhcpserver=11.17.0.1 networks.tftpserver=11.17.0.1
   networks.nameservers=11.17.0.1  networks.dynamicrange=11.17.1.200-11.17.1.250
chtab net=11.18.0.0 networks.netname=cubnet networks.mask=255.255.0.0
   networks.mgtifname=eth1 networks.gateway=11.18.255.254
   networks.dhcpserver=11.18.0.1 networks.tftpserver=11.18.0.1
   networks.nameservers=11.18.0.1  networks.dynamicrange=11.18.1.200-11.18.1.250
```

Disable the entry for the public network (connected to the outside world):

```
chtab net=9.114.88.160 networks.netname=public networks.disable=1
```

4.2 Setup DNS

Set nameserver, forwarders and domain in the site table:

```
chtab key=nameservers site.value=11.16.0.1 # IP of mgmt node
chtab key=forwarders site.value=9.114.8.1,9.114.8.2 # site DNS servers
chtab key=domain site.value=cluster.net   # domain part of the node hostnames
```

Edit /etc/hosts to be similar to:

```
127.0.0.1       localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
192.168.2.100   b7-eth0
192.168.100.1   b7
192.168.100.10  blade1
192.168.100.11  blade2
192.168.100.12  blade3
172.30.101.133  amm3
```

Run:

```
makedns
```

Setup /etc/resolv.conf:

```
search cluster.net
nameserver 11.16.0.1
```

Start DNS:

```
service named start
chkconfig --level 345 named on
```

4.3 Define AMMs as nodes

```
nodeadd bca01-bca04 groups=mm,all
nodeadd bcb01-bcb04 groups=mm,all
#nodeadd swa01-swa04 groups=mm,all   # do not think we need to define these??
#nodeadd swb01-swb04 groups=mm,all   # do not think we need to define these??
chtab node=mm nodehm.mgt=blade
chtab node=mm mp.mpa='|(.*)|($1)|'
```

4.4 Setup AMM

Note: currently the network settings on the MM (both for the MM itself and for the switch module) need to be set up with your own customized script. Eventually, this will be done by xCAT through lsslp, finding it on the switch, looking in the switch table, and then setting it in the MM.

```
rspconfig mm snmpcfg=enable sshcfg=enable
rspconfig mm pd1=redwoperf pd2=redwoperf
rpower mm reset
```

Note: should also set ntp settings using rspconfig, once MN is configured as NTP server.

Test the ssh set up with:

```
psh -l USERID@mm info -T mm[1]
```

TIP to update firmware:

Put CNETCMUS.pkt in /tftpboot

```
telnet AMM
env -T mm[1]
update -v -i TFTP_SERVER_IP -l CNETCMUS.pkt
```

TIP for SOL to work best telnet to nortel switch (default pw is “admin”) and type:

```
/cfg/port int1/gig/auto off
```

Do this for each port (I.e. int2, int3, etc.)

4.5 Startup TFTP

```
mknb x86_64
service tftpd restart
```

5.0 Define Compute Nodes in the Database

5.1 Setup the nodelist Table

The nodelist table contains a node definition for each node in the cluster. We have provided a script to automate these definitions for the RR cluster.

/opt/xcat/share/xcat/tools/mkrrnodes will allow you to automatically define as many nodes as you would like to and setup nodegroups needed to manage those nodes. See man mkrrnodes.

For example, running mkrrnodes will define the following nodes with the assigned groups in the nodelist table. These nodegroups will be used in additional xCAT Table setup so that an entry does not have to be made for every node.

```
/opt/xcat/share/xcat/tools/mkrrnodes -C a -R 001,012
/opt/xcat/share/xcat/tools/mkrrnodes -C b -R 001,012
```

adds to the nodelist table entries like the following:

```
"rra001a","rra001,ls21,cua,opteron,compute,tb,all,rack01",,,
"rra001b","rra001,qs22,cua,cell,cell-b,compute,all,tb,rack01",,,
"rra001c","rra001,qs22,cua,cell,cell-c,compute,all,tb,rack01",,,
"rra002a","rra002,ls21,cua,opteron,compute,tb,all,rack01",,,
"rra002b","rra002,qs22,cua,cell,cell-b,compute,all,tb,rack01",,,
"rra002c","rra002,qs22,cua,cell,cell-c,compute,all,tb,rack01",,,
```

5.2 Set Up the nodehm table

```
chtab node=opteron-cua nodehm.cons=blade nodehm.mgt=blade nodehm.conserver=rra000
   nodehm.serialspeed=19200 nodehm.serialflow=hard nodehm.serialport=1
chtab node=cell-cua nodehm.cons=blade nodehm.mgt=blade nodehm.conserver=rra000
   nodehm.serialspeed=19200 nodehm.serialflow=hard nodehm.serialport=0
chtab node=opteron-cub nodehm.cons=blade nodehm.mgt=blade nodehm.conserver=rrb000
   nodehm.serialspeed=19200 nodehm.serialflow=hard nodehm.serialport=1
chtab node=cell-cub nodehm.cons=blade nodehm.mgt=blade nodehm.conserver=rrb000
   nodehm.serialspeed=19200 nodehm.serialflow=hard nodehm.serialport=0
```

When Table.pm supports where strings on rows that contain regex's (or I think when the servicenode table is used to determine if conserver should be set up??), the above lines can be replaced with:

```
chtab node=tb nodehm.cons=blade nodehm.mgt=blade nodehm.conserver='|rr(.).*|
   rr($1)000|'  nodehm.serialspeed=19200 nodehm.serialflow=hard
```

5.3 Set Up the mp Table

```
chtab node=opteron mp.mpa="|rr(.)(\d+)\D|bc(\$1)(sprintf('%02d',((\$2-1)/3+1)))|"
   mp.id='|rr.(\d+)\D|((($1-1)%3)*4+1)|'
chtab node=cell-b mp.mpa="|rr(.)(\d+)\D|bc(\$1)(sprintf('%02d',((\$2-1)/3+1)))|"
   mp.id='|rr.(\d+)\D|((($1-1)%3)*4+3)|'
chtab node=cell-c mp.mpa="|rr(.)(\d+)\D|bc(\$1)(sprintf('%02d',((\$2-1)/3+1)))|"
   mp.id='|rr.(\d+)\D|((($1-1)%3)*4+4)|'
```

5.4 Setup Conserver

```
makeconservercf
service conserver stop
service conserver start
```

Test a few nodes with rpower and rcons.

5.5 Setup the noderes Table

The noderes table will define for the node or nodegroup, the service node used to service the node or group, the type of network booting supported, the node which is the tftpserver, dhcpserver,etc as known by the node.

If you <u>are</u> using Service Nodes:
For each node or nodegroup defined in the noderes table change the service node attribute in the noderes table to point to the name or ip address of it's service node.

```
chtab node=opteron noderes.netboot=pxe noderes.servicenode='|rr(.).*|rr($1)000|'
   noderes.xcatmaster='|rr(.).*|rr($1)000-eth1|' noderes.installnic=eth0
   noderes.primarynic=eth0
chtab node=cell noderes.netboot=yaboot noderes.servicenode='|rr(.).*|rr($1)000|'
   noderes.xcatmaster='|rr(.).*|rr($1)000-eth1|' noderes.installnic=eth0
   noderes.primarynic=eth0

# Will not need this when it properly defaults to xcatmaster
chtab node=opteron noderes.tftpserver='|rr(.).*|rr($1)000-eth1|'
chtab node=cell noderes.tftpserver='|rr(.).*|rr($1)000-eth1|'
```

If you are <u>not</u> using Service Nodes:

```
chtab node=opteron noderes.netboot=pxe noderes.xcatmaster=mn20 nodehm.serialport=1
   noderes.installnic=eth0 noderes.primarynic=eth0
chtab node=cell noderes.netboot=yaboot noderes.xcatmaster=mn20
   nodehm.serialport=0 noderes.installnic=eth0 noderes.primarynic=eth0
```

5.5.1 Sample noderes table

Your noderes table will end up looking like this (if you use service nodes):

```
#node,servicenode,netboot,tftpserver,nfsserver,monserver,kernel,initrd,kcmdline,nf
   sdir,serialport,installnic,primarynic,xcatmaster,current_osimage,next_osimage,co
   mments,disable
"opteron","|rr(.).*|rr($1)000|","pxe","|rr(.).*|rr($1)000-
   eth1|",,,,,,,"1","eth0","eth0","|rr(.).*|rr($1)000-eth1|",,,,
"cell","|rr(.).*|rr($1)000|","yaboot","|rr(.).*|rr($1)000-
   eth1|",,,,,,,"1","eth0","eth0","|rr(.).*|rr($1)000-eth1|",,,,
"service",,"pxe",,,,,,,,"1","eth0","eth0",,,,,
```

5.5.2 Setting up which services run on the Service Nodes

Note: if in the noderes table you have an assigned servicenode for a node, you must make sure you have that service started on that service node by having an entry for it in the servicenode table. If you leave a field blank in the servicenode table, the default is to start that service on that node. For RoadRunner we are starting all services on all service nodes. To do this, run:

```
chtab node=service servicenode.nameserver=1 servicenode.dhcpserver=1
   servicenode.tftpserver=1 servicenode.nfsserver=1 servicenode.conserver=1
   servicenode.monserver=1 servicenode.ldapserver=1 servicenode.ntpserver=1
   servicenode.comments='Starts all services on all service nodes'
```

5.6 Setup nodetype table

Define the OS and profile type for building the stateless image.

```
chtab node=opteron nodetype.os=fedora8 nodetype.arch=x86_64
   nodetype.profile=compute nodetype.nodetype=osi
chtab node=cell nodetype.os=fedora8 nodetype.arch=ppc64 nodetype.profile=compute
   nodetype.nodetype=osi
```

5.6.1.1 Sample nodetype table

Your nodetype table will look something like this:

```
#node,os,arch,profile,nodetype,comments,disable
"service","fedora8","x86_64","service","osi",,
"opteron","fedora8","x86_64","compute","osi",,
"cell","fedora8","ppc64","compute","osi",,
```

5.7 Setup passwords in passwd table

Add needed passwords to the passwd table to support installs.

```
chtab key=system passwd.username=root passwd.password=cluster
chtab key=blade passwd.username=USERID passwd.password=PASSW0RD
chtab key=ipmi passwd.username=USERID passwd.password=PASSW0RD
```

5.8 Verify the tables

To verify that the tables are set correctly, run lsdef on a service node, opteron blade, and cell blade:

```
lsdef rra000,rra001a,rra001b
```

5.9 Setup deps Table for proper boot sequence

The following is an example of how you can setup the deps table to ensure the triblades boot up in the proper sequence. The 1st row tells xCAT the opteron blades should not be powered on until the corresponding cell blades are powered on. The 2nd row tells xCAT the cell blades should not be powered off until the corresponding opteron blades are powered off.

```
chtab node=opteron deps.nodedep='|rr(.\d+)a|rr($1)b,rr($1)c|' deps.msdelay=5000
   deps.cmd=on
chtab node=cell deps.nodedep='|rr(.\d+).|rr($1)a|' deps.msdelay=5000 deps.cmd=off
```

Verify the dependencies are correct:

```
nodels rra001a deps.nodedep
nodels rra001b deps.nodedep
```

5.10 Set Up Postscripts to be Run on the Nodes

Add names of postscripts that should be run for all nodes by using the xcatdefaults row of the postscripts table. (xCAT automatically fills in this table with defaults during a new install of the xCAT software on the management node, so you may not have to do this step.)

```
chtab node=xcatdefaults postscripts.postscripts=syslog,remoteshell
```

Also add postscripts that should be run on the service nodes:

```
chtab node=service
   postscripts.postscripts=configeth,servicenode,xcatserver,xcatclient
```

5.11 Get MAC addresses

```
getmacs tb
```

Don't think we need this anymore:

```
rinv tb macs | perl -pi -e 's/([^:]*):.*?ress (\d): (00(:[0-9A-F]{2}){5})/nodech \
   1 mac.mac=\3 #\2/' | grep \#1 > /tmp/setmacs.sh
source /tmp/setmacs.sh
```

To verify mac addresses in table:

```
tabdump mac
```

5.12 Setup DHCP

The dynamic ranges for the networks were set up already in chapter 4.

Define dhcp interfaces in site table. We use this weird value because our MN uses eth4 to communicate with the service nodes, and the service nodes use eth1 to communicate with the compute nodes.

```
chtab key=dhcpinterfaces site.value='mn20|eth4;service|eth1'
```

Ensure dhcpd is running:

```
service dhcpd start
```

Configure DHCP:

```
makedhcp -n
service dhcpd restart
```

6.0 Build the service node stateless image

The service node stateless images must contain not only the OS, but also the xCAT software. In addition, a number of files are added to the image to support the postgresql database access from the service node to the Management node, and ssh access to the nodes that the service nodes services. Note: the following example assumes you are building the stateless image on the Management Node.

1. Check the service node packaging to see if it has all the rpms required:
   ```
   cd /opt/xcat/share/xcat/netboot/fedora/
   vi service.pkglist service.exlist
   ```

   Make sure service.pkglist has the following packages:

   ```
   vi
   dhcp
   atftp
   bind
   nfs-utils
   ```

   Edit service.exlist and include things you may need by removing the corresponding line. Make sure you remove:

   ```
   ./usr/lib/perl5*
   ```

   While you are here, edit compute.pkglist and compute.exlist, adding and removing as necessary.

2. Run image generation:
   ```
   cd /opt/xcat/share/xcat/netboot/fedora/
   ./genimage -i eth0 -n tg3,bnx2 -o fedora8 -p service
   ```
3. Install xCAT code into the service node image:
   ```
   rm -f /install/netboot/fedora8/x86_64/service/rootimg/etc/yum.repos.d/*
   cp -pf /etc/yum.repos.d/*.repo /install/netboot/fedora8/x86_64/service/rootimg/
      etc/yum.repos.d
   yum --installroot=/install/netboot/fedora8/x86_64/service/rootimg install
      xCATsn
   ```
4. Prevent DHCP from starting up until xcatd has had a chance to configure it:
   ```
   chroot /install/netboot/fedora8/x86_64/service/rootimg
   chkconfig dhcpd off
   chkconfig dhcrelay off
   exit
   ```
5. Pack the image
   ```
   packimage -o fedora8 -p service -a x86_64
   ```
6. To update the xCAT software in the image at a later time:

```
yum --installroot=/install/netboot/fedora8/x86_64/service/rootimg update '*xCAT*'
packimage -o fedora8 -p service -a x86_64
```

6.1 Configure the Service Node BMCs??

It seems like we missed a step of having the rack mounted nodes automatically discovered when powered on, by having xcat respond to the dhcp request, check the switch for the port #, assign the hostname, and configure the BMC on the node. This also requires that the switch table be set up.

6.2 Install the Service Nodes

```
nodeset service netboot
rpower service boot
wcons service      # make sure DISPLAY is set to your X server/VNC or
     rcons <one-node-at-a-time>    # or do rcons for each node
tail -f /var/log/messages
```

6.3 Test Service Node installation

- ssh to the service nodes.
- Check to see that the xcat daemon xcatd is running.
- Run some database command on the service node, e.g tabdump site, or nodels, and see that the database can be accessed from the service node.
- Check that /install and /tftpboot are mounted on the service node from the Management Node.

Right now, have to start conserver, until Ling fixes that bug:

```
service conserver start
```

7.0 iSCSI Install QS22 Blade

Note: in these instructions, substitute your management node IP address for 11.16.0.1.

```
yum install yaboot-xcat scsi-target-utils
chtab key=iscsidir site.value=/install/iscsi
```

Pick a QS22 blade for the iSCSI install that can access the management node. Add it as a node (and its management module, if necessary):

```
nodeadd mvqs21b groups=compute,iscsi
nodeadd bca2 groups=mm2
```

Make sure the root userid and password are in the iscsi table

```
chtab node=mvqs21b iscsi.userid=root iscsi.passwd=cluster iscsi.server=11.16.0.1
```

Other table settings:

```
chtab node=mvqs21b noderes.nfsserver=11.16.0.1 nodehm.serialport=0
   noderes.netboot=yaboot noderes.installnic=eth0 noderes.primarynic=eth0
chtab node=mvqs21b nodetype.os=fedora8 nodetype.arch=ppc64 nodetype.profile=iscsi
   nodetype.nodetype=osi iscsi.server=11.16.0.1
chtab node=mvqs21b nodehm.mgt=blade nodehm.cons=blade nodehm.serialspeed=19200
   nodehm.serialflow=hard
chtab node=bca2 nodehm.mgt=blade
chtab node=mvqs21b mp.mpa=bca2 id=2
chtab node=bca2 mp.mpa=bca2

getmacs mvqs21b
```

Put mvqs21b and bca2 in /etc/hosts, then:

```
makedns
makedhcp -n

service tgtd restart
nodech mvqs21b iscsi.file=
setupiscsidev -s8192 mvqs21b

nodeset mvqs21b install
rpower mvqs21b boot
```

NOTE: for reinstall:

```
chtab node=mvqs21b nodetype.profile=iscsi   # don't need this?
nodeset mvqs21b install
rpower mvqs21b boot
```

7.1 Build QS22 Stateless image

1. Logon to the qs22 blade:

   ```
   ssh mvqs21b
   mkdir /install
   mount 11.16.0.1:/install /install
   ```

2. Create fedora.repo:

   ```
   cd /etc/yum.repos.d
   rm -f *.repo
   ```

   Put the following lines in /etc/yum.repos.d/fedora.repo:

   ```
   [fedora]
   name=Fedora $releasever - $basearch
   baseurl=file:///install/fedora8/ppc64
   enabled=1
   gpgcheck=0
   ```

3. Test with: `yum search gcc`

4. Copy the executables and files needed from the Management Node:

   ```
   cd /root
   scp 11.16.0.1:/opt/xcat/share/xcat/netboot/fedora/genimage .
   scp 11.16.0.1:/opt/xcat/share/xcat/netboot/fedora/geninitrd .
   scp 11.16.0.1:/opt/xcat/share/xcat/netboot/fedora/compute.ppc64.pkglist .
   ```

5. Generate the image:

   ```
   ./genimage -i eth0 -n tg3 -o fedora8 -p compute
   ```

   NOTE: iSCSI, QS22, tg3, all slow, take a nap

7.2 Boot the QS22 Stateless image

On the Management Node:

1. Adding Service Node ssh keys – **Don't think we need to do this**, because the private & public keys on the service node will be the same as the management node
   If you wish to be able to ssh from your service nodes to their compute nodes, you will have to follow these steps to add the additional required keys to the install image on the service node before the image is installed.

   ```
   ssh rra000
   ssh-keygen -t rsa
   ```

```
#take defaults and answer no to passcode/passphrase message when generating
   keys
cat /root/.ssh/id_rsa.pub >>
   /install/netboot/fedora8/ppc64/compute/rootimg/root/.ssh/authorized_keys
```

Repeat the above steps for rrb000

2. Edit fstab in the image:

```
cd /install/netboot/fedora8/ppc64/compute/rootimg/etc
cp fstab fstab.ORIG
```

Edit fstab. **Change:**

```
devpts  /dev/pts devpts   gid=5,mode=620 0 0
tmpfs   /dev/shm tmpfs    defaults       0 0
proc    /proc    proc     defaults       0 0
sysfs   /sys     sysfs    defaults       0 0
```

to:

```
proc             /proc                  proc         rw 0 0
sysfs            /sys                   sysfs        rw 0 0
devpts           /dev/pts               devpts       rw,gid=5,mode=620 0 0
#tmpfs           /dev/shm               tmpfs        rw 0 0
compute_ppc64    /                      tmpfs        rw 0 1
none             /tmp                   tmpfs        defaults,size=10m 0 2
none             /var/tmp               tmpfs        defaults,size=10m 0 2
```

3. Pack the image:

```
packimage -o fedora8 -p compute -a ppc64
```

4. Install the image on all the QS22 blades:

```
nodeset cell netboot
rpower cell boot
```

7.3 To Update QS22 Stateless image

1. Before YUM/RPM commands (or genimage):

```
rm /install/netboot/fedora8/ppc64/compute/rootimg/var/lib/rpm/__db.00*
```

2. To update image using YUM:

```
rm -f /install/netboot/fedora8/ppc64/compute/rootimg/etc/yum.repos.d/*
cp /etc/yum.repos.d/*.repo
   /install/netboot/fedora8/ppc64/compute/rootimg/etc/yum.repos.d
```

Now install vi into the image:

```
yum --installroot=/install/netboot/fedora8/ppc64/compute/rootimg install vi
```

3. To update image using RPM:

```
rpm --root /install/netboot/fedora8/ppc64/compute/rootimg -Uvh
   /install/fedora8/ppc64/Packages/vim-minimal-7.1.135-1.fc8.ppc.rpm
```

4. To update the image by running genimage, add packages to compute.ppc64.pkglist and rerun genimage

```
5. packimage -o fedora8 -p compute -a ppc64
```

7.4 Build and Install QS22 Compressed Image

On the QS22 blade:

```
yum install kernel-devel gcc squashfs-tools
```

On internet connected node:

```
svn co http://xcat.svn.sf.net/svnroot/xcat/xcat-dep/trunk/aufs
```

7.4.1 Build aufs

```
cd aufs
tar jxvf aufs-2-6-2008.tar.bz2
cd aufs
mv include/linux/aufs_type.h fs/aufs/
cd fs/aufs/
patch -p1 < ../../../aufs-standalone.patch
chmod +x build.sh
./build.sh

# ls -lh aufs.ko
-rw-r--r-- 1 root root 3.5M 2008-03-10 14:20 aufs.ko

strip -g aufs.ko
cp aufs.ko /root
```

7.4.2 Generate the compressed image

```
cd /opt/xcat/share/xcat/netboot/fedora
./geninitrd -i eth0 -n tg3,squashfs,aufs,loop -o fedora8 -p compute -l $(expr
   100 \* 1024 \* 1024)
```

7.4.3 Pack and install the compressed image

On the Management Node:

```
yum install squashfs-tools
```

```
packimage -a ppc64 -o fedora8 -p compute -m squashfs
chtab node=cell   nodetype.profile=compute nodetype.os=fedora8
nodeset cell  netboot
rpower cell boot
```

7.4.4 Check Memory Usage

```
# ssh left "echo 3 > /proc/sys/vm/drop_caches;free -m;df -h"
             total       used       free     shared    buffers     cached
Mem:          3961         99       3861          0          0         61
-/+ buffers/cache:         38       3922
Swap:            0          0          0
Filesystem            Size  Used Avail Use% Mounted on
compute_ppc64         100M  220K  100M   1% /
none                   10M     0   10M   0% /tmp
none                   10M     0   10M   0% /var/tmp
```

Max for / is 100M, but only 220K being used (down from 225M), but wheres the OS?

Look at cached. 61M compress OS image. 3.5x smaller

As files change in hidden OS they get copied to tmpfs (compute_ppc64) with a copy on write. To reclaim space reboot. The /tmp and /var/tmp is for MPI and other Torque and user related stuff. if 10M is too small you can fix it. To reclaim this space put in epilogue:

```
umount /tmp /var/tmp; mount -a
```

Reboot cell as stateless, from management node to reclaim space:

```
nodeset cell  netboot
xdsh cell reboot #be nice, iSCSI is still stateful, be kind to the state??
```

7.4.5 To Switch a Compute Blade to iSCSI for More Setup

To reboot rra047b as iscsi for more stateless setup fun:

```
nodech rra047b nodetype.profile=iscsi
nodeset rra047b  iscsiboot
rpower rra047b  boot
```

8.0 Build LS21 Stateless image

The LS21 image can be built on the Management Node since it is of the same architecture.

On the Management Node:

1. Check the compute node packaging to see if it has all the rpms required.

```
cd /opt/xcat/share/xcat/netboot/fedora/
vi compute.exlist and compute.pklist
```

For example to add vi to be installed on the node, add the name of the vi rpm to compute.pkglist

```
echo vi >>compute.pkglist
```

Include things you may need which are excluded, by editing compute.exlist.
For example, if you require ./usr/lib/perl5, remove the following line:

```
./usr/lib/perl5*
```

2. Run image generation:

```
./genimage -i eth0 -n tg3,bnx2 -o fedora8 -p compute
```

3. Adding Service Node ssh keys – **Don't need to do this**, because the private & public keys on the service node will be the same as the management node
   If you wish to be able to ssh from your service nodes to their compute nodes, you will have to follow these steps to add the additional required keys to the install image on the service node before the image is installed. (The keys were already generated on the service nodes in chapter 7.)

```
ssh rra000
cat id_rsa.pub >>
   /install/netboot/fedora8/x86_64/compute/rootimg/root/.ssh/authorized_keys
```

   Repeat the above steps for rrb000

4. Edit fstab in the image

```
cd /install/netboot/fedora8/x86_64/compute/rootimg/etc
cp fstab fstab.ORIG
```

Edit fstab:

Change:

```
devpts  /dev/pts devpts   gid=5,mode=620 0 0
tmpfs   /dev/shm tmpfs    defaults       0 0
proc    /proc    proc     defaults       0 0
sysfs   /sys     sysfs    defaults       0 0
```

to:

```
proc             /proc                         proc          rw 0 0
sysfs            /sys                          sysfs         rw 0 0
devpts           /dev/pts                      devpts        rw,gid=5,mode=620 0
   0
#tmpfs           /dev/shm                      tmpfs         rw 0 0
```

```
compute_x86_64   /                           tmpfs        rw 0 1
none             /tmp                        tmpfs        defaults,size=10m 0
   2
none             /var/tmp                    tmpfs        defaults,size=10m 0
   2
```

5. Package the image
   ```
   packimage -o fedora8 -p compute -a x86_64
   ```
6. Install the image on all the LS21 blades
   ```
   nodeset opteron netboot
   rpower opteron boot
   ```

8.1 To Update the LS21 Stateless Image Later On

1. Before running YUM/RPM commands:
   ```
   rm /install/netboot/fedora8/x86_64/compute/rootimg/var/lib/rpm/__db.00*
   ```
2. To update image using YUM:
   ```
   rm -f /install/netboot/fedora8/x86_54/compute/rootimg/etc/yum.repos.d/*
   cp /etc/yum.repos.d/*
      /install/netboot/fedora8/x86_64/compute/rootimg/etc/yum.repos.d
   ```
   Now install, for example, vi into the image:
   ```
   yum --installroot=/install/netboot/fedora8/x86_64/compute/rootimg install vi
   ```
3. To update image using RPM:
   ```
   rpm --root /install/netboot/fedora8/x86_64/compute/rootimg -Uvh blah.rpm
   ```
4. Repackage:
   ```
   packimage -o fedora8 -p compute -a x86_64
   ```
5. Install on all LS21 blades:
   ```
   nodeset opteron netboot
   rpower opteron boot
   ```

8.2 Build and Install LS21 Compressed Image

On Management Node:

```
yum install kernel-devel gcc squashfs-tools
```

8.2.1 Build aufs

```
svn co http://xcat.svn.sf.net/svnroot/xcat/xcat-dep/trunk/aufs
```

```
cd aufs
tar jxvf aufs-2-6-2008.tar.bz2
cd aufs
mv include/linux/aufs_type.h fs/aufs/
cd fs/aufs/
patch -p1 < ../../../aufs-standalone.patch
chmod +x build.sh
./build.sh

ls -lh aufs.ko
-rw-r--r-- 1 root root 3.2M 2008-02-27 13:09 aufs.ko

strip -g aufs.ko
cp aufs.ko /opt/xcat/share/xcat/netboot/fedora/
```

8.2.2 Generate and pack the compressed image

```
cd /opt/xcat/share/xcat/netboot/fedora
./geninitrd -i eth0 -n tg3,bnx2,squashfs,aufs,loop -o fedora8 -p service -l $(expr
   100 \* 1024 \* 1024)
packimage -a x86_64 -o fedora8 -p compute -m squashfs
```

NOTE: The -l and -t is the size of the / and /tmp,/var/tmp file systems in RAM

NOTE: To unsquash:

```
   cd  /install/netboot/fedora8/x86_64/service
   rm -f rootimg.sfs
   packimage -a x86_64 -o fedora8 -p service -m cpio
```

8.2.3 Install the image

```
nodeset opteron netboot
rpower opteron  boot
```

8.2.4 Check Memory Usage

```
#ssh middle "echo 3 > /proc/sys/vm/drop_caches;free -m;df -h"
             total       used       free     shared    buffers     cached
Mem:          3969         82       3887          0          0         43
-/+ buffers/cache:         38       3930
Swap:            0          0          0
Filesystem            Size  Used Avail Use% Mounted on
compute_x86_64        100M  216K  100M   1% /
none                   10M     0   10M   0% /tmp
none                   10M     0   10M   0% /var/tmp
```

3x smaller.

9.0 Building QS22 Image for 64K pages

On Management Node:

```
cd /opt/xcat/share/xcat/netboot/fedora
cp compute.exlist compute.exlist.4k
echo "./lib/modules/2.6.23.1-42.fc8/*" >>compute.exlist

wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/source/SRPM
   S/kernel-2.6.23.1-42.fc8.src.rpm
nodech mvqs21b nodetype.profile=iscsi
nodeset mvqs21b iscsiboot
rpower mvqs21b boot
```

On the blade:

```
ssh mvqs21b
mkdir /install
mount mgmt:/install /install
yum install rpm-build redhat-rpm-config ncurses ncurses-devel kernel-devel gcc
   squashfs-tools
rpm -Uivh kernel-2.6.23.1-42.fc8.src.rpm
rpmbuild -bp --target ppc64 /usr/src/redhat/SPECS/kernel.spec
cd /usr/src/redhat/BUILD/kernel-2.6.23
cp -r linux-2.6.23.ppc64 /usr/src/
cd /usr/src/kernels/$(uname -r)-$(uname -m)
find . -print | cpio -dump /usr/src/linux-2.6.23.ppc64/
cd /usr/src/linux-2.6.23.ppc64
make mrproper
cp configs/kernel-2.6.23.1-ppc64.config .config
make menuconfig

Kernel options  --->
[*] 64k page size
Platform support  --->
[ ] Sony PS3
<exit><exit><save>

Edit Makefile suffix:
EXTRAVERSION = .1-42.fc8-64k

make -j4
make modules_install
strip vmlinux
mv vmlinux /boot/vmlinuz-2.6.23.1-42.fc8-64k
cd /lib/modules/2.6.23.1-42.fc8-64k/kernel
find . -name "*.ko" -type f -exec strip -g {} \;
#mkinitrd /boot/initrd-2.6.23.1-42.fc8-64k.img 2.6.23.1-42.fc8-64k
#rm -f /boot/vmlinuz-2.6.23.1-42.fc8 /boot/initrd-2.6.23.1-42.fc8.img
#rm -rf /lib/modules/2.6.23.1-42.fc8
```

9.1 Rebuild aufs

Rebuild aufs.so:

```
rm -rf aufs
tar jxvf aufs-2-6-2008.tar.bz2
cd aufs
mv include/linux/aufs_type.h fs/aufs/
cd fs/aufs/
patch -p1 < ../../../aufs-standalone.patch
chmod +x build.sh
./build.sh 2.6.23.1-42.fc8-64k
strip -g aufs.ko
cp aufs.ko /root
```

NOTE: patch genimage (??)

On blade:

```
cd /root
./genimage -i eth0 -n tg3 -o fedora8 -p compute
cd /lib/modules
cp -r 2.6.23.1-42.fc8-64k
   /install/netboot/fedora8/ppc64/compute/rootimg/lib/modules/
cd /boot
cp vmlinuz-2.6.23.1-42.fc8-64k /install/netboot/fedora8/ppc64/compute/kernel
```

9.2 Test unsquashed:

On blade:

```
cd /root
./geninitrd -i eth0 -n tg3 -o fedora8 -p compute -k 2.6.23.1-42.fc8-64k
```

On Management Node:

```
rm -f /install/netboot/fedora8/ppc64/compute/rootimg.sfs
packimage -a ppc64 -o fedora8 -p compute -m cpio
nodech mvqs21b nodetype.profile=compute nodetype.os=fedora8
gnodeset mvqs21b netboot
rpower mvqs21b boot
```

9.2.1 Check memory

```
# ssh left "echo 3 > /proc/sys/vm/drop_caches;free -m;df -h"
             total       used       free     shared    buffers     cached
Mem:          4012        495       3517          0          0        429
-/+ buffers/cache:         66       3946
Swap:            0          0          0
Filesystem            Size  Used Avail Use% Mounted on
compute_ppc64         2.0G  432M  1.6G  22% /
```

```
none                       10M     0   10M   0% /tmp
none                       10M     0   10M   0% /var/tmp
```

9.3 Test squash

On mvqs21b:

```
cd /root
./geninitrd -i eth0 -n tg3,squashfs,aufs,loop -o fedora8 -p compute -k
   2.6.23.1-42.fc8-64k -l $(expr 100 \* 1024 \* 1024)
```

On Management Node:

```
rm -f /install/netboot/fedora8/ppc64/compute/rootimg.sfs
packimage -a ppc64 -o fedora8 -p compute -m squashfs #bug, must remove sfs first
nodech left nodetype.profile=compute nodetype.os=fedora8
nodeset left netboot
rpower left boot
```

9.3.1 Check memory

```
# ssh left "echo 3 > /proc/sys/vm/drop_caches;free -m;df -h"
             total       used       free     shared    buffers     cached
Mem:          4012        127       3885          0          0         65
-/+ buffers/cache:         61       3951
Swap:            0          0          0
Filesystem            Size  Used Avail Use% Mounted on
compute_ppc64         100M  1.7M   99M   2% /
none                   10M     0   10M   0% /tmp
none                   10M     0   10M   0% /var/tmp
```

./lib/modules/* in compute.exlist: (??)

9.4 To Switch Back to 4K Pages

On blade:

```
cd /boot
cp -f vmlinuz-2.6.23.1-42.fc8 /install/netboot/fedora8/ppc64/compute/kernel
cd /root
./geninitrd -i eth0 -n tg3 -o fedora8 -p compute
```

OR

```
./geninitrd -i eth0 -n tg3,squashfs,aufs,loop -o fedora8 -p compute -l $(expr
   100 \* 1024 \* 1024)
```

From Management Node:

```
rm -f /install/netboot/fedora8/ppc64/compute/rootimg.sfs
packimage -a ppc64 -o fedora8 -p compute -m cpio
```

OR

```
packimage -a ppc64 -o fedora8 -p compute -m squashfs
nodech mvqs21b nodetype.profile=compute nodetype.os=fedora8
nodeset mvqs21b netboot
rpower mvqs21b boot
```

10.0 Using NFS Hybrid for the Diskless Images

1. Make sure you have latest xCAT installed (later than Thu Apr 24 17:34:48 UTC 2008)

2. Get stateless cpio or squashfs set up and test, e.g.:

   ```
   cd /opt/xcat/share/xcat/netboot/fedora
   ./genimage -i eth0 -n tg3,bnx2 -o fedora8 -p compute
   ```

   cpio

   ```
   ./geninitrd -i eth0 -n tg3,bnx2,loop -o fedora8 -p compute
   packimage -a x86_64 -o fedora8 -p compute -m cpio
   ```

   Overhead: (large image with development tools installed)

   ```
                total       used       free     shared    buffers     cached
   Mem:          3969        740       3229          0          0        680
   -/+ buffers/cache:         59       3910
   Swap:            0          0          0
   Filesystem            Size  Used Avail Use% Mounted on
   compute_x86_64        2.0G  695M  1.3G  35% /
   none                   10M     0   10M   0% /tmp
   none                   10M     0   10M   0% /var/tmp
   e326001:/home          32G   15G   16G  47% /home
   ```

   squashfs (note order of modules important)

   ```
   ./geninitrd -i eth0 -n tg3,bnx2,squashfs,aufs,loop -o fedora8 -p compute
   packimage -a x86_64 -o fedora8 -p compute -m squashfs
   ```

   Overhead: (large image with development tools installed)

   ```
                total       used       free     shared    buffers     cached
   Mem:          3969        243       3726          0          0        206
   -/+ buffers/cache:         37       3932
   Swap:            0          0          0
   Filesystem            Size  Used Avail Use% Mounted on
   compute_x86_64        2.0G  260K  2.0G   1% /
   ```

```
none                   10M     0   10M   0% /tmp
none                   10M     0   10M   0% /var/tmp
e326001:/home          32G   15G   16G  47% /home

nodeset noderange netboot
rpower noderange boot
```

3. Patch kernel and build new aufs.ko:

Get AUFS from CVS:

```
cd /tmp
mkdir aufs
cd /tmp/aufs
cvs -d:pserver:anonymous@aufs.cvs.sourceforge.net:/cvsroot/aufs login #CVS
   password is empty
cvs -z3 -d:pserver:anonymous@aufs.cvs.sourceforge.net:/cvsroot/aufs co aufs
cd /tmp/aufs/aufs
cvs update
```

Install stuff

```
yum install rpm-build redhat-rpm-config ncurses ncurses-devel kernel-devel gcc
   squashfs-tools
```

Kernel notes (x86_64 and ppc64):

```
cd /tmp
wget
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/source/
   SRPMS/kernel-2.6.23.1-42.fc8.src.rpm
rpm -Uivh kernel-2.6.23.1-42.fc8.src.rpm
yum install redhat-rpm-config
rpmbuild -bp --target $(uname -m) /usr/src/redhat/SPECS/kernel.spec
cd /usr/src/redhat/BUILD/kernel-2.6.23
cp -r linux-2.6.23.$(uname -m) /usr/src/
cd /usr/src/kernels/$(uname -r)-$(uname -m)
find . -print | cpio -dump /usr/src/linux-2.6.23.$(uname -m)/
cd /usr/src/linux-2.6.23.$(uname -m)
make mrproper
cp configs/kernel-2.6.23.1-$(uname -m).config .config
patch -p0 < /tmp/aufs/aufs/patch/put_filp.patch
cd /tmp/aufs/aufs
cp -r include /usr/src/linux-2.6.23.$(uname -m)
cp -r fs/aufs /usr/src/linux-2.6.23.$(uname -m)/fs
cd /usr/src/linux-2.6.23.$(uname -m)
```

Edit fs/Kconfig and change (at end):

```
source "fs/nls/Kconfig"
source "fs/dlm/Kconfig"
```

To:

```
source "fs/nls/Kconfig"
source "fs/dlm/Kconfig"
source "fs/aufs/Kconfig"
```

Append to: fs/Makefile

```
obj-$(CONFIG_AUFS) += aufs/

make menuconfig

File system --->
  <M> Another unionfs
  ---   These options are for 2.6.23.1-42.fc8
  [ ]   Use simplified (fake) nameidata
        Maximum number of branches (127)  --->
  [*]   Use <sysfs>/fs/aufs
  [ ]   Use inotify to detect actions on a branch
  [ ]   NFS-exportable aufs
  [ ]   Aufs as an readonly branch of another aufs
  [ ]   Delegate the internal branch access the kernel thread
  [ ]   show whiteouts
  [*]   Make squashfs branch RR (real readonly) by default
  [ ]   splice.patch for sendfile(2) and splice(2)
  [*]   put_filp.patch for NFS branch
  [ ]   lhash.patch for NFS branch
  [ ]   fsync_super-2.6.xx.patch was applied or not
  [ ]   deny_write_access.patch was applied or not
  [ ]   Special handling for FUSE-based filesystem
  [*]   Debug aufs
  [ ]   Compatibility with Unionfs (obsolete)
Exit, Exit, Save

make -j4
make modules_install
make install
```

Whew!

4. Remove old aufs.ko:

```
cd /opt/xcat/share/xcat/netboot/fedora
rm -f aufs.ko
```

5. Boot NFS:

Create ifcfg-eth0:

```
cd /install/netboot/fedora8/x86_64/compute/rootimg/etc/sysconfig/networks-
   scripts
```

Put in ifcfg-eth0:

```
ONBOOT=yes
BOOTPROTO=none
DEVICE=eth0
```

(This solves an intermittent problem where DHCP hoses IP long enough to hose NFS and then nothing works. It's also one less DHCP and it boots faster.)

Update modules:

```
cd /lib/modules
   cp -r * /install/netboot/fedora8/x86_64/compute/rootimg/lib/modules
```

Dump services:

```
cd /install/netboot/fedora8/x86_64/compute/rootimg/
chroot .
chkconfig --del messagebus
chkconfig --del mdmonitor
chkconfig --del netconsole
exit
```

Edit fstab:

```
cd /install/netboot/fedora8/x86_64/compute/rootimg/etc
```

add this line:

```
sunrpc            /var/lib/nfs/rpc_pipefs rpc_pipefs rw 0 0
```

```
yum --installroot=/install/netboot/fedora8/x86_64/compute/rootimg install nfs-
   utils
cd /opt/xcat/share/xcat/netboot/fedora
./geninitrd -i eth0 -n tg3,bnx2,aufs,loop,sunrpc,lockd,nfs_acl,nfs -o fedora8 -
   p compute
packimage -a x86_64 -o fedora8 -p compute -m nfs
```

Notice helpful message:

```
NOTE: Contents of /install/netboot/fedora8/x86_64/compute/rootimg
MUST be available on all service and management nodes and NFS exported.

nodeset noderange netboot
rpower noderange boot
```

Overhead: (large image with development tools installed)

```
             total       used       free     shared    buffers     cached
Mem:          3969         47       3922          0          0          8
-/+ buffers/cache:         38       3931
Swap:            0          0          0
Filesystem           Size  Used Avail Use% Mounted on
compute_x86_64       2.0G  300K  2.0G   1% /
none                  10M     0   10M   0% /tmp
none                  10M     0   10M   0% /var/tmp
e326001:/home         32G   15G   16G  47% /home
```

11.0 Installing OpenLDAP

11.1 Setup LDAP Server

On the management node:

1. export /home (rw) for testing

    ```
     echo '/home *(rw,no_root_squash,sync)' >> /etc/exports
     exportfs -a
    ```

2. add a test userid “ibm”

    ```
    useradd ibm
    mkdir ~ibm/.ssh
    mkdir ~ibm/.pbs_spool
    ```

3. Generate root ssh keys for mn20 and give ibm id root ssh authority

    ```
    ssh-keygen -t rsa -q -N "" -f ~ibm/.ssh/id_rsa
    cp ~ibm/.ssh/id_rsa.pub ~ibm/.ssh/authorized_keys

    vi  ~ibm/.ssh/config
    ```

    Add the following lines:

    ```
    ForwardX11 yes
    StrictHostKeyChecking no
    FallBackToRsh no
    BatchMode yes
    ConnectionAttempts 5
    UsePrivilegedPort no
    Compression no
    Cipher blowfish
    UserKnownHostsFile /dev/null
    CheckHostIP no
    ```

4. Set permissions :

    ```
    chown -R ibm.ibm ~ibm
    chmod 700 ~ibm/.ssh
    chmod 600 ~ibm/.ssh/*
    ```

11.1.1 Install the LDAP rpms

```
yum install openldap-servers  <-- could not get this package. I had to download
   it from the fedora website .
   http://download.fedora.redhat.com/pub/fedora/linux/releases/8/Everything/x86
   _64/os/Packages/
( located in /tmp/lissa/tools/ldap)
```

The following rpms should be installed:

```
openldap-*
openldap-devel-*
openldap-clients-*
openldap-servers-*
```

11.1.2 Configure LDAP

```
cd /etc/openldap
```

5. edit slapd.conf

   Comment out the following the two lines that start with suffix and rootdn:

   ```
   database       bdb
   #suffix       "dc=my-domain,dc=com"
   #rootdn        "cn=Manager,dc=my-domain,dc=com"
   ```

   Add the following information to the end of the file:

   ```
   #xCAT start
   #cluster.net:
   suffix          "dc=cluster,dc=net"

   #root access
   rootdn          "cn=root,dc=cluster,dc=net"

   #passwd generated with: slappasswd  cluster  - you should generate your own
      password and use  in the ldap commands below in place of cluster.

   rootpw       {SSHA}ChnyPaCrc2uUZ0O6iRHy3zk4rEH7HNpL



   # password hash algorithm
   password-hash {SSHA}

   # The userPassword by default can be changed by the entry owning it if they
   # are authenticated. Others should not be able to see it, except the admin.
   access to attrs=userPassword
           by dn="uid=admin,ou=People,dc=cluster,dc=net" write
           by anonymous auth
           by self write
           by * none

   #
   ##password aging
   access to attrs=shadowLastChange
           by dn="uid=admin,ou=People,dc=cluster,dc=net" write
           by self write
   ```

```
        by * read
```

6. `cp /etc/openldap/DB_CONFIG.example /var/lib/ldap/DB_CONFIG`

7. start ldap

   ```
   service ldap start
   ```

11.1.3 Migrate Users

```
cd /usr/share/openldap/migration
cp migrate_common.ph migrate_common.ph.save
```

Edit migrate_common.ph and change the following lines to be:

```
$DEFAULT_MAIL_DOMAIN = "cluster.net";
$DEFAULT_BASE = "dc=cluster,dc=net";
$EXTENDED_SCHEMA = 1;
```

Run:

```
./migrate_base.pl >/tmp/base.ldif
./migrate_passwd.pl /etc/passwd >>/tmp/base.ldif
./migrate_group.pl /etc/group >>/tmp/base.ldif
cd /var/lib/ldap
service ldap stop
slapadd -l /tmp/base.ldif
chown ldap.ldap *
service ldap start
```

Test the database by searching for a the user ibm:

```
ldapsearch -x -v -D  "cn=root,dc=cluster,dc=net" -w cluster  -b
   "ou=People,dc=cluster,dc=net" "uid=ibm"
```

Output should be as follows:

```
ldap_initialize( <DEFAULT> )
filter: uid=ibm
requesting: All userApplication attributes
# extended LDIF
#
# LDAPv3
# base <ou=People,dc=cluster,dc=net> with scope subtree
# filter: uid=ibm
# requesting: ALL
#

# ibm, People, cluster.net
dn: uid=ibm,ou=People,dc=cluster,dc=net
uid: ibm
cn: ibm
sn: ibm
mail: ibm@cluster.net
objectClass: person
```

```
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword:: e2NyeXB0fSEh
shadowLastChange: 13998
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 501
gidNumber: 501
homeDirectory: /home/ibm

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

11.1.4 Add a User to LDAP

Setup a new user adduser.ldif file with the following contents:

```
dn: uid=ibm4,ou=People,dc=cluster,dc=net
uid: ibm4
cn: ibm4
sn: ibm4
mail: ibm4@cluster.net
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 13998
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 504
gidNumber: 504
homeDirectory: /home/ibm4
```

Run:

```
   ldapadd -x -c -D  "cn=root,dc=cluster,dc=net" -w cluster -f adduser1.ldif

adding new entry "uid=ibm4,ou=People,dc=cluster,dc=net"
```

Verify:

```
ldapsearch -x -v -D  "cn=root,dc=cluster,dc=net" -w cluster  -b
   "ou=People,dc=cluster,dc=net" "uid=ibm4"
```

11.2 Setup LDAP Client

11.2.1 Install LDAP into the Image

```
yum --installroot=/install/netboot/fedora8/x86_64/compute/rootimg \
        install openldap-clients nss_ldap nfs-utils vi
```

11.2.2 Update the LDAP Configuration in the Image

```
cd /install/netboot/fedora8/x86_64/compute/rootimg
```

Edit /etc/ldap.conf with these changes:

```
uri ldap://11.16.0.1/
host 11.16.0.1
base dc=cluster,dc=net
nss_base_passwd ou=People,dc=cluster,dc=net
nss_base_shadow ou=People,dc=cluster,dc=net
nss_base_group  ou=Group,dc=cluster,dc=net
```

Edit /etc/openldap/ldap.conf with these changes:

```
URI  ldap://11.16.0.1
BASE dc=cluster,dc=net
```

Add ldap to the following three lines in /etc/nsswitch

```
passwd:     files ldap
shadow:     files ldap
group:      files ldap
```

Edit /etc/pam.d/system-auth, change (order important!):

change:

```
account     required      pam_unix.so
```

to:

```
account     sufficient    pam_ldap.so
account     required      pam_unix.so
```

Add to fstab to Mount /home for testing:

```
11.16.0.1:/home /home nfs timeo=14,intr 1 2
```

11.2.3 Build the Image and Install

Add the following rpms to the image for testing. Note: the order of modules in the geninitrd command is important!

```
cd /opt/xcat/share/xcat/netboot/fedora
./geninitrd -i eth0 -n tg3,bnx2,sunrpc,lockd,nfs,nfs_acl -o fedora8 -p compute
packimage -o fedora8 -p compute -a x86_64
nodeset rra001a netboot
rpower rra001a boot
```

11.2.4 Test LDAP Client

```
ssh to rra001a
```

Run:

```
ldapsearch -x -v -D  "cn=root,dc=cluster,dc=net" -w cluster  -b
   "ou=People,dc=cluster,dc=net" "uid=ibm"
```

Check to see if you get output from the LDAP server as in Migrate Users.

Now authenticate the ibm users from LDAP by changing it's password and su to ibm.

```
passwd ibm
su - ibm
```

12.0 Setup Hierarchical LDAP

TBD

13.0 Install Torque

13.1 Setup Torque Server

```
cd /tmp
wget http://www.clusterresources.com/downloads/torque/torque-2.3.0.tar.gz
tar zxvf torque-2.3.0.tar.gz
cd torque-2.3.0
CFLAGS=-D__TRR ./configure \
        --prefix=/opt/torque \
        --exec-prefix=/opt/torque/x86_64 \
        --enable-docs \
        --disable-gui \
        --with-server-home=/var/spool/pbs \
        --enable-syslog \
        --with-scp \
        --disable-rpp \
        --disable-spool
make
make install
```

13.2 Configure Torque

```
cd /opt/torque/x86_64/lib
ln -s libtorque.so.2.0.0 libtorque.so.0
echo "/opt/torque/x86_64/lib" >>/etc/ld.so.conf.d/torque.conf
ldconfig
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/xpbsnodes /opt/torque/x86_64/bin/
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbsnodestat
   /opt/torque/x86_64/bin/
```

Create /etc/profile.d/torque.sh:

```
export PBS_DEFAULT=mn20
export PATH=/opt/torque/x86_64/bin:$PATH
chmod 755 /etc/profile.d/torque.sh
source /etc/profile.d/torque.sh
```

13.3 Define Nodes

```
cd /var/spool/pbs/server_priv
nodels '/rr.*a' groups | sed 's/: groups://' | sed 's/,/ /g' | sed 's/$/ np=4/'
   >nodes
```

13.4 Setup and Start Service

```
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbs /etc/init.d/
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbs_mom /etc/init.d/
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbs_sched /etc/init.d/
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbs_server /etc/init.d/
chkconfig --del pbs
chkconfig --del pbs_mom
chkconfig --del pbs_sched
chkconfig --level 345 pbs_server on
service pbs_server start
```

13.5 Install pbstop

```
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/pbstop /opt/torque/x86_64/bin/
chmod 755 /opt/torque/x86_64/bin/pbstop
```

13.6 Install Perl Curses for pbstop

```
yum install perl-Curses
```

13.7 Create a Torque Default Queue

```
echo "create queue dque
set queue dque queue_type = Execution
set queue dque enabled = True
set queue dque started = True
set server scheduling = True
set server default_queue = dque
set server log_events = 127
set server mail_from = adm
set server query_other_jobs = True
set server resources_default.walltime = 00:01:00
set server scheduler_iteration = 60
set server node_pack = False
set server keep_completed=300" | qmgr
```

13.8 Setup Torque Client (x86_64 only)

13.8.1 Install Torque

```
cd /opt/xcat/share/xcat/netboot/add-on/torque
./add_torque /install/netboot/fedora8/x86_64/compute/rootimg mn20 /opt/torque
   x86_64 local
```

13.8.2 Configure Torque

13.8.2.1 Set Up Access

```
cd /install/netboot/fedora8/x86_64/compute/rootimg/etc/security
echo "-:ALL EXCEPT root:ALL" >>access.conf
cp access.conf access.conf.BOOT
cd /install/netboot/fedora8/x86_64/compute/rootimg/etc/pam.d
```

Edit system-auth and replace:

```
account     sufficient    pam_ldap.so
account     required      pam_unix.so
```

with:

```
account     required      pam_access.so
account     sufficient    pam_ldap.so
account     required      pam_unix.so
```

13.8.2.2 Set Up Node to Node ssh for Root

This is needed for cleanup:

```
cp /root/.ssh/* /install/netboot/fedora8/x86_64/compute/rootimg/root/.ssh/
cd /install/netboot/fedora8/x86_64/compute/rootimg/root/.ssh/
rm known_hosts
```

Setup the config file:

```
echo "StrictHostKeyChecking no
FallBackToRsh no
BatchMode yes
```

```
ConnectionAttempts 5
UsePrivilegedPort no
Compression no
Cipher blowfish
CheckHostIP no" >config
```

13.8.3 Pack and Install image

```
packimage -o fedora8 -p compute -a x86_64
nodeset  opteron netboot
rpower  opteron boot
```

14.0 Setup Moab

14.1 Install Moab

```
cd /tmp
wget http://www.clusterresources.com/downloads/mwm/moab-5.2.1-linux-x86_64-
   torque.tar.gz
tar zxvf /tmp/moab-5.2.1-linux-x86_64-torque.tar.gz
cd moab-5.2.1
./configure --prefix=/opt/moab
make install
```

14.2 Configure Moab

```
mkdir -p /var/spool/moab/log
mkdir -p /var/spool/moab/stats
```

Create /etc/profile.d/moab.sh:

```
export PATH=/opt/moab/bin:$PATH
```

```
chmod 755 /etc/profile.d/moab.sh
source /etc/profile.d/moab.sh
```

Edit moab.cfg and **change:**

```
RMCFG[mn20]      TYPE=NONE
```

to:

```
RMCFG[mn20]      TYPE=pbs
```

Append to moab.cfg :

```
NODEAVAILABILITYPOLICY      DEDICATED:SWAP
JOBNODEMATCHPOLICY          EXACTNODE
NODEACCESSPOLICY            SINGLEJOB
NODEMAXLOAD                 .5
```

```
JOBMAXSTARTTIME              00:05:00
DEFERTIME                    0
JOBMAXOVERRUN                0
LOGDIR                       /var/spool/moab/log
LOGFILEMAXSIZE               10000000
LOGFILEROLLDEPTH             10
STATDIR                      /var/spool/moab/stats
```

14.2.1 Start Moab

```
cp -f /opt/xcat/share/xcat/netboot/add-on/torque/moab /etc/init.d/
chkconfig --level 345 moab on
service moab start
```

15.0 References

- XCAT2.0 Beta Cookbook - http://xcat.svn.sourceforge.net/svnroot/xcat/xcat-core/trunk/xCAT-client-2.0/share/doc/xCAT2.0.pdf