xCAT Statelite

vallard@us.ibm.com
01/27/10, 18:05:48 A1/P1
Updated on January 27, 2010 by minjun.xi@cn.ibm.com

1. Introduction

This document details a design for native xCAT NFS Root. We call the design statelite because in addition to having an NFS Root implementation, some files can be stored persistently and maintain state through reboots.

Statelite offers the following advantages over xCAT's stateless implementation:

1. Some files can be made persistent over reboot. This is useful for license files or database servers where some state is needed. However, you still get the advantage of only having to manage a single image. Statelite offers the best of both worlds of stateless and stateful.

2. Changes to hundreds of machines can take place instantly and automatically by updating one main image. In most cases, machines do not need to reboot for these changes to take affect.

3. Ease of administration by being able to lock down an image. Many parts of the image can be read only so no modifications can transpire without updating the main image.

4. Files can be managed in a hierarchical manner. For example: Suppose you have a machine that is in one lab in Tokyo and another in London. You could set table values for those machines in the xCAT database to allow machines to sync from different places based on their attributes. This allows you to have one base image with multiple sources of file layover.

5. Ideal for virtualization – In a virtual environment you may not want a disk image (neither stateless nor stateful) on every node as it consumes memory and disk. Virtualizing with the statelite approach allows for images to be smaller, easier to manage, use less disk, less memory, and more agile.

Disadvantages
1. NFS Root requires more network traffic to run as the majority of the disk image runs over NFS. This may depend on your workload, but can be made very light.
2. NFS Root can be complex to set up as well. As more files are created in different

places there are greater chances for failures. We think this flexibility is also one of the great virtues of Statelite. The image can work in nearly any environment.

1.2 Limitations

At the beginning, Statelite is limited to Red Hat and Red Hat Clone environments on x86_64 platforms; and then, Statelite is implemented for SLES11 (SuSE Linux Enterprise Server 11) environment on POWER systems, which should also work on other hardware platforms.

2. Usage

Getting started with Statelite provisioning requires that you have xCAT set up and running. You should be able to provision stateless images if you are familiar with that approach. We have put a great deal of thought into error detection and debugging(See section 5 on debugging techniques), but there are always cases that were not considered. Please report any errors to the xCAT mailing list.

2.1 Lock Down main image

Any statelite image will need some files to be read/writable for individual nodes. You could potentially make the entire image read/writeable, but you will have big problems when multiple nodes access the same root tree and start changing things!

We encourage you to first lock down the main image. The image is placed in /install/netboot/<os>/<arch>/<profile>/rootimg.

xCAT, by default, during installation will create on the management node an /etc/exports entry that has this directory read/writeable. Please change it to read-only, so it looks as follows:

```
/install *(ro,no_root_squash,sync)
```

Next, restart the NFS server:

```
service nfs restart
```

2.2 litefile

Files that need to be read/write for individual nodes will by default be stored in the memory of the running node. By default these files will not be persistent but are generated at boot time by being copied from the main image.

The list of files is stored in the xCAT *litefile* table. (e.g: tabedit litefile)

There are several flags that can direct the behavior of the file list to change it to be read only, persistent, or concatenated through.

The column headings in the litefile table appears as follows:

```
#image,file,options,comments,disable
```

image
The first entry can be the image name, as specified in the osimage table. This can be left empty, or specified with "ALL" to signify that the file should be unique for all images.
file
The second entry is the file name. This should the full path of the file. If the file is a directory, then it should be terminated with a '/'.
options
The third entry contains the options of a file. xCAT supports the following:

- *Blank or 'ALL'* – file will be placed in tmpfs on the booted node with read/write mode. When searching for the file, the first one to be found in the litetree hierarchy will be used.
- *tmpfs* (same as default)
- *persistent* (requires statelite table to be filled out with a spot for persistent storage). This means that the file will be persistent across reboots. If the file does not exist at first, it will be created during initialization. Every time there after the file will be left alone if it exists.
- *ro* – file will be read only. Generally this means that it will be linked to some place in the directory hierarchy.
- *con* – The contents of the pathname are concatenated onto the contents of the existing file. For this directive the searching in the hierarchy does not stop when the first match is found. Con is similar to tmpfs, but all files found in the hierarchy will be concatenated to the file when found.

Sample Data for RedHat
We encourage you to use the following list to start out your litefile table entry in your RedHat and RedHat clone environments. Notice that all files are in tmpfs. This gives you an NFS root solution with no persistent storage.

```
image,file,options,comments,disable
"ALL","/etc/adjtime",,,
"ALL","/etc/fstab",,,
"ALL","/etc/inittab",,,
"ALL","/etc/lvm/.cache",,,
"ALL","/etc/mtab",,,
"ALL","/etc/ntp.conf",,,
"ALL","/etc/ntp.conf.predhclient",,,
"ALL","/etc/resolv.conf",,,
"ALL","/etc/resolv.conf.predhclient",,,
"ALL","/etc/ssh/",,,
```

```
"ALL","/tmp/",,,
"ALL","/var/account/",,,
"ALL","/var/arpwatch",,,
"ALL","/var/cache/alchemist",,,
"ALL","/var/cache/foomatic/",,,
"ALL","/var/cache/logwatch/",,,
"ALL","/var/cache/man/",,,
"ALL","/var/cache/mod_ssl/",,,
"ALL","/var/cache/mod_proxy/",,,
"ALL","/var/cache/php-pear/",,,
"ALL","/var/cache/systemtap/",,,
"ALL","/var/empty/",,,
"ALL","/var/db/nscd/",,,
"ALL","/var/gdm/",,,
"ALL","/var/lib/dav/",,,
"ALL","/var/lib/dhcp/",,,
"ALL","/var/lib/dhclient/",,,
"ALL","/var/lib/php/",,,
"ALL","/var/lib/scsi/",,,
"ALL","/var/lib/ups/",,,
"ALL","/var/lib/random-seed",,,
"ALL","/var/lib/iscsi",,,
"ALL","/var/lib/logrotate.status",,,
"ALL","/var/lib/ntp/",,,
"ALL","/var/lib/xen/ntp",,,
"ALL","/var/lock/",,,
"ALL","/var/log/",,,
"ALL","/var/run/",,,
"ALL","/var/tmp/",,,
"ALL","/var/tux/",,,
```

Sample Data for SLES11

As we know, SuSE uses many different packages, so the sample data for SLES11 are different from the sample data for RedHat. We also recommend you to use the following list to start your litefile table entry for SLES11.

```
#image,file,options,comments,disable
"ALL","/etc/inittab",,,
"ALL","/etc/lvm/.cache",,,
"ALL","/etc/mtab",,,
"ALL","/etc/ntp.conf",,,
"ALL","/etc/resolv.conf",,,
"ALL","/etc/ssh/",,,
"ALL","/etc/sysconfig/",,,
"ALL","/etc/syslog-ng/",,,
"ALL","/tmp/",,,
"ALL","/var/tmp/",,,
"ALL","/var/run/",,,
"ALL","/etc/yp.conf",,,
"ALL","/var/lib/",,,
"ALL","/var/empty/",,,
"ALL","/var/spool/",,,
"ALL","/var/lock/",,,
"ALL","/var/log/",,,
"ALL","/var/cache/",,,
```

```
"ALL","/etc/fstab",,,
"ALL","/var/adm/",,,
"ALL","/root/.viminfo",,,
"ALL","/root/.ssh/",,,
"ALL","/root/.bash_history",,,
"ALL","/opt/xcat/",,,
"ALL","/xcatpost/",,,
```

2.3 litetree

When a node boots up in statelite mode, it will by default copy all of its tmpfs files from the root image in the /.default directory. You may decide that you want files pulled from different locations that are different per node.

For example, a user may have two directories with a different */etc/motd* for each location in a different language:

```
10.0.0.1:/syncdirs/newyork-590Madison/rhels5.4/x86_64/compute/etc/motd
10.0.0.1:/syncdirs/shanghai-11foo/rhels5.4/x86_64/compute/etc/motd
```

You can consolidate these into one directory in the litetree table:
`1,,10.0.0.1:/syncdirs/$nodepos.room/$nodetype.os/$nodetype.arch/$nodetype.profile`

You may also want to look by default into directories containing the node name first:
`$noderes.nfsserver:/syncdirs/$node`

The litetree prioritizes where node files are created. The first field is the priority. The second field is the image name (ALL for all images) and the final field is the mount point.

Our example is as follows:
```
1,,$noderes.nfsserver:/statelite/$node
2,,cnfs:/gpfs/dallas/
```

The two directories /statelite/$node on the node's $noderes.nfsserver and the /gpfs/dallas on the node cnfs contain root tree structures that are sparsely populated with files that we want to place in those nodes. If files are not found first in one directory, it goes to the next directory. If none of the files can be found in the litetree hierarchy, then they are searched for in /.default on the local image.

2.4 Determine where state will be held

You may want some files to be stored permanently for the image to survive reboots. This is done by entering the information into the statelite table.

The headings are as follows for this table:

```
#node,image,statemnt,comments,disable
```

An example would be:
```
“japan”,,”cnfs:/gpfs/state”,,,
```

All nodes in the japan node group will have their state stored in the /gpfs/state directory on the machine known as cnfs. This is true for all images, though we could specify that some images be stored in different places.

When the node boots up, then the value of statemnt will be mounted to /.statelite/persistent.
The code will then create the following subdirectory /.snapshot/persistent/<nodename>
This directory will be the root of the image for persistent files.

NOTE: Do not name your persistent storage directory after the node name, as this will be placed in the directory automatically. If you do, then a directory named /state/n01 will have its state tree inside /state/n01/n01.

2.5 Policy

Ensure policies are set up correctly. When a node boots up, it queries the xCAT database to get the lite-files and the lite-tree. In order for this to work, the command must be set in the policy table to allow nodes to request it.

```
chtab priority=4.7 policy.commands=litetree
chtab priority=4.8 policy.commands=litefile
```
This should happen automatically when xCAT is installed, but you may want to verify it if you experience problems booting.

2.6 Create Statelite Image

After all our tables are set up, it is time to create a base image. You will first need to determine the Operating System and the name of the image.

Let’s suppose we want our operating system to be RedHat 5.3. We will name our image “test1”.

We must first create a list of packages to be installed for test1. You should start with the base packages in the compute template. These are required.

```
cd /opt.xcat/share/xcat/netboot/rh
cp compute.pkglist test1.pkglist
```

You can then add more packages to the test1.pkglist. Once you have your template in place, run the genimage command in /opt/xcat/bin (this should be in your path)

```
genimage
```
The command will prompt you for the necessary inputs (OS, profile name, etc)

Or, run:
```
cd /opt/xcat/share/xcat/netboot/rh
./genimage –o rhels5.3 –p test1 –i eth0 –n mlx4_core,mlx4_en,igb,bnx2 –
m statelite
```

The genimage command will do several things:
1. It will create a an image in
`/install/netboot/<os>/<arch>/<profile>/rootimg`
2. It will create a few statelite directories inside the image;
/.statelite
/.default
/etc/init.d/statelite
3. It will create a ramdisk and kernel that can be used to boot the initial node.

This image that you have created can be used for stateless or statelite booting.

2.6 Modify statelite image (For RedHat)

Note: This section only works for RedHat and RedHat clone environments; for SLES, the commands (including "add_passwd" and "add_ssh") have been executed in the "genimage" command.

Since the files that were now just created will be the default, you can edit the image directly by visiting the root tree in:

```
/install/netboot/<os>/<arch>/<profile>/rootimg
```

You can do chroot to make changes or perform additional yum updates/install using the –installroot flag.

For general xCAT settings, and a base working image, all that is necessary is to copy some passwd files and ssh settings from xCAT into the image:

```
cd /opt/xcat/share/xcat/netboot/add-on/statelite/
./add_passwd /install/netboot/<os>/<arch>/<profile>/rootimg
./add_ssh /install/netboot/<os>/<arch>/<profile>/rootimg
```

2.7 run liteimg <os>-<arch>-<profile>

The liteimg command will modify your statelite image (or any image) by creating a series of links. Once you are satisfied your image is in place, run:

```
liteimg <os>-<arch>-<profile>
e.g: liteimg rhels5.3-x86_64-test1
liteimg –o rhels5.3 –a x86_64 –p test1
```

This creates 2 levels of indirection so that files can be modified while in their image state as well as during runtime. For example, a file like "<$imgroot>/etc/ntp.conf" will have the following operations done to it:

```
mkdir –p $imgroot/.default/etc
mkdir –p $imgroot/.statelite/tmpfs/etc
mv $imgroot/etc/ntp.conf $imgroot/.default/etc
cd $imgroot/.statelite/tmpfs/etc
ln –sf ../../../.default/etc/resolv.conf .
cd $imgroot/etc
ln –sf ../.statelite/tmpfs/etc/resov.conf .
```

When finished, the original file will reside in $imgroot/.default/etc/ntp.conf. $imgroot/etc/ntp.conf will link to $imgroot/.statelite/tmpfs/etc/ntp.conf which will in turn link to $imgroot/.default/etc/ntp.conf

Note: If you make any changes to your litefile table after running liteimg then you will need to rerun liteimg again. This is because files and directories need to have the two levels of redirects created.

2.8 Set the boot state to "statelite"

You can now install the node:

```
nodeset <noderange> statelite=centos5.3-x86_64-test1
```

Or just run:

```
nodeset <noderange> statelite
```

This will create the necessary files in /tftpboot for the node to boot correctly.

2.9 Install the Node:

Finally, reboot the node so that it installs.

For Redhat on x86_64 platform:

```
rpower <noderange> boot
```

Nodeset will have generated the appropriate PXE file so that the node boots off the nfsroot image. This file will look similar to the following:

```
#statelite centos5.3-x86_64-all
DEFAULT xCAT
LABEL xCAT
```

```
 KERNEL xcat/netboot/centos5.3/x86_64/all/kernel
 APPEND initrd=xcat/netboot/centos5.3/x86_64/all/initrd.gz
NFSROOT=172.10.0.1:/install/netboot/centos5.3/x86_64/all
STATEMNT=cnfs:/gpfs/state XCAT=172.10.0.1:3001 console=tty0
console=ttyS0,115200n8r
```

For SLES11 on POWER platform:

Run the following command to boot up the nodes into statelite mode:

```
rnetboot <noderange>
```

Nodeset will have generated the appropriate yaboot.conf-MAC-ADDRESS file so that the node boots off the nfsroot image. This file will look similar to the following:

```
#statelite sles11-ppc64-compute
timeout=5
image=xcat/netboot/sles11/ppc64/compute/kernel
        label=xcat
        initrd=xcat/netboot/sles11/ppc64/compute/initrd.gz
        append="NFSROOT=192.168.11.108:/install/netboot/sles11/ppc64/co
mpute STATEMNT= XCAT=192.168.11.108:3001 "
```

You can then use rcons or wcons to watch the node boot up.

3. Code Changes

/opt/xcat/bin/genimage
- Ask user if they want statelite or stateless.
- Get rid of default color

/opt/xcat/lib/perl/xCAT/schema.pm
- Add the tables litefile, litetree, and statelite

/opt/xcat/lib/perl/xCAT_plugin/anaconda.pm
- modify the mknetboot to have options for statelite subtilties.
- Check the statelite table for the snapshot directory.

/opt/xcat/lib/perl/xCAT_plugin/destiny.pm
- help messages and nodeset code.

/opt/xcat/lib/perl/xCAT_plugin/litetree.pm
- code for ilitetree, litetree, and iliteimg

/opt/xcat/lib/perl/xCAT_plugin/statelite.pm
- code for liteimg

/opt/xcat/lib/perl/xCAT_plugin/pxe.pm
/opt/xcat/lib/perl/xCAT_plugin/xnba.pm
/opt/xcat/sbin/xcatconfig
- Added litetree and litefile to the policy table.

/opt/xcat/share/xcat/netboot/[centos|rh]/genimage
- -m flag adds the NFS kernel modules necessary for NFS mounting to the initrdfs

- Creates script to boot up initrd.

/opt/xcat/share/xcat/add-on/statelite/add_passwd
/opt/xcat/share/xcat/add-on/statelite/add_ssh
/opt/xcat/share/xcat/add-on/statelite/rc.statelite
xCAT-client.spec

- Add symbolic links for litefile, litetree, ilitefile, liteimg

/install/postscripts/xcatdsklspost

- "xcatdsklspost 4" means statelite mode is enabled during the node boots up.

/opt/xcat/share/xcat/netboot/sles/genimage

- Following the implementation of "genimage" for RedHat, some code changes are added to provide statelite mode for SLES.

3.1 New Commands

The following commands are in /opt/xcat/bin:

```
ln –s xcatclient litefile
ln –s xcatclient litetree
ln –s xcatclientnnr ilitefile
ln –s xcatclientnnr liteimg
```

litefile <nodename>
Shows all the statelite files that are not to be taken from the base of the image.

litetree <nodename>
Shows the NFS mount points for a node.

Ilitefile <image name>
Shows the stateless files that will be used for a node image.

liteimg <image name>
Creates a series of symbolic links in an image that is compatible with statelite booting.

4. Statelite Directory Structure

Each statelite image will have the following directories:

```
/.statelite/tmpfs/
/.statelite/persistent/<nodename>
/.statelite/mnt  # where directory tree is mounted from.
/.default/
/etc/init.d/statelite
```

All files that are symbolic links, will link to /.statelite/tmpfs.

tmpfs files that are persistent link to /.statelite/persistent/<nodename>/
/.statelite/persistent/<nodename> is the directory where the node's individual storage will be mounted to.

/.default is where default files will be copied to from the image to tmpfs if the files are not found in the litetree hierarchy.

4.5. Options filling out tables

noderes.nfsserver can be filled out for the NFSroot server. If this is not filled out then it uses the management server.

noderes.nfsdir – can be filled out: /vol/xCAT/install. At that point it assumes that the directory structure is the same as the install directory.

5. Debugging techniques

1. When a node boots up in staelite mode, there is a script run called statelite that is in the root directory of $imgroot/etc/init.d/statelite. This script is not run as part of the rc scripts, but as part of the pre switch root environment. Thus, all the linking is done in this script. There is a "set –x" near the top of the file. You can uncomment it and see what the script runs. You will then see lots of mkdirs and links on the console.
2. You can also set the machine to shell. Just add the word "shell" on the end of the pxeboot file of the node in the append line. This will make the init script in the initramfs pause 3 times before doing a switch_root.
3. When all the files are linked they are logged in /.statelite/statelite.log on the node. You can get into the node after it has booted and look in the /.statelite directory.