Linuxha.net
Step-by-step Guide:
Samba

Introduction

The purpose of this document is to suggest procedures for creating a clustered Samba configuration with Linuxha.net. These procedures were tested on the following configuration:

- Fedora Core 5
- DRBD version 0.7.20
- Linuxha.net version 1.2.2
- Samba version 3.0.23b

The following conventions are used throughout this document:

`# ls`	Single-line command entered as root user.

`# useradd \`
`> --home-dir /home2/xyz \`
`> --gid abc \`
`> xyz`
	Multi-line command entered as root user.

`$ ls`	Single-line command entered as non-root user

`$ cut \`
`> --delimiter=":" \`
`> --fields=1,5 \`
`/ etc/passwd`
	Multi-line command entered as non-root user.

`127.0.0.1 localhost`
`192.168.1.32 fc5s1`
`192.168.1.33 fc5s2`
`192.168.100.32 fc5s1b`
`192.168.100.33 fc5s2b`
	Command output or file contents

Installation

This section provides basic instructions for downloading and installing Samba version 3.0 from i386 binary RPM. Instructions for installing or upgrading using other package formats or from source are beyond the scope of this guide.

All commands are executed on both nodes as ***root***, unless otherwise indicated.

1. Check whether Samba is installed, by executing

   ```
   # find /usr -name "smb[cd]*" -path "*bin*" -print
   ```

   If the result contains the following entries, then both the Samba server and client are installed and you may proceed to step 4. Otherwise, continue to step 2.

   ```
   /usr/sbin/smbd
   /usr/bin/smbclient
   ```

2. Download Samba RPM's as follows (these are all single-line commands):

   ```
   # wget http://us1.samba.org/samba/ftp/Binary_Packages/Fedora/RPMS/i386/core/5/samba-common-3.0.23b-1.i386.rpm
   # wget http://us1.samba.org/samba/ftp/Binary_Packages/Fedora/RPMS/i386/core/5/samba-3.0.23b-1.i386.rpm
   # wget http://us1.samba.org/samba/ftp/Binary_Packages/Fedora/RPMS/i386/core/5/samba-client-3.0.23b-1.i386.rpm
   ```

3. Install Samba.

```
# rpm --hash --install samba-common-3.0.23b-1.i386.rpm
# rpm --hash --install samba-3.0.23b-1.i386.rpm
# rpm --hash --install samba-client-3.0.23b-1.i386.rpm
```

4. Delete Samba startup and shutdown scripts.

```
# rm /etc/rc3.d/S??smb /etc/rc0.d/K??smb
```

5. Configure firewall

   If the firewall is running, then it should be configured to accept connections on ports 137, 138, 139 and 445.

   Insert the following lines into **/etc/sysconfig/iptables:**

```
-A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 137 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 138 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 139 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 445 -j ACCEPT
```

   Restart the firewall

```
# /etc/init.d/iptables/restart
```

Files / Directories for Replication

The contents of the Samba configuration directory and all shared directories should be replicated. This guide will describe how to set up the Samba clustered application with the following replicated directories.

Directory	Comment
/etc/samba	Location of Samba configuration scripts
public	Globally readable public directory
home	Location of Samba users' home directories.

Create Replicated File System

The commands in this section are executed on both nodes as ***root***.

1. Use **fdisk** to create a partition (*/dev/sda6*) large enough to accommodate all directories shared by Samba.

2. Initialize the partition.

```
# pvcreate /dev/sda6
```

3. Create application volume group (*sambavg*)

```
# vgcreate sambavg /dev/sda6
```

4. Create mount point for replicated file system

```
# mkdir /samba
```

5. Create logical volume (*shareslv*).

```
# lvcreate --size 512M --name shareslv sambavg
```

6. Create file system on *shareslv*.

```
# mkfs -t ext3 /dev/sambavg/shareslv
```

Populate Replicated File System

All commands are executed as ***root*** on both nodes, unless indicated otherwise.

1. Shut down Samba server (both nodes)

```
# /etc/init.d/smb stop
```

2. Mount logical volume

```
# mount -t ext3 /dev/sambavg/shareslv /samba
```

3. Create directory structure

```
# mkdir /samba/conf
# mkdir /samba/scripts
# touch /samba/scripts/startapp /samba/scripts/stopapp
# chmod u+x /samba/scripts/*
# mkdir /samba/home
# mkdir /samba/public
```

4. Copy Samba configuration files to logical volume

```
# cd /etc/samba
# find . -print | cpio --pass-through /samba/conf
```

Configuration

1. Insert the following into **/samba/scripts/startapp**:

```
ISNETBIOSDISABLED=$(testparm -s /samba/conf/smb.conf 2>/dev/null | \
        sed -n '/\[global\]/,/^$/p' | \
        grep "disable netbios = Yes" | \
        awk 'BEGIN{FS=" = "}{print $2}')

smbd -D -s /samba/conf/smb.conf -l /var/log/samba

if [ $? -eq 0 -a "x$ISNETBIOSDISABLED"!="x" ]
then nmbd -D -s /samba/conf/smb.conf -l /var/log/samba
fi
```

2. Insert the following into **/samba/scripts/stopapp**:

```
if [ -f /var/run/nmbd.pid ]
then
    kill `cat /var/run/nmbd.pid`
    rm -f /var/run/nmbd.pid
fi

if [ -f /var/run/smbd.pid ]
then
```

```
   kill `cat /var/run/smbd.pid`
   rm -f /var/run/smbd.pid
fi
```

3. Add the following entries to the *global* section of **/samba/conf/smb.conf**.

```
   interfaces = 192.168.1.55/24
   smb passwd file = /samba/conf/smbpasswd
```

The value of *interfaces* should be set to the clustered Samba application virtual IP address.

4. (Optional) Add the following share definitions to **/samba/conf/smb.conf**

```
[homes]
   comment = Home Directories
   browseable = no
   writable = yes

[public]
   comment = Public Stuff
   path = /samba/public
   public = yes
   writable = no
   printable = no
```

5. (Optional) Create a test file in **/samba/public**

```
# echo "public data" > /samba/public/test1
```

User Account Creation

Follow the instructions in this section in order to replicate shared home directories. Users whose home directories are to be shared MUST have the same user and group ids on both nodes.

The commands in this section are executed as ***root*** on both nodes, unless indicated otherwise.

1. Identify the next available group id.

```
# expr `cut --delimiter=":" --fields=3 /etc/group | \
> grep --regexp="^[5-9][0-9][0-9]$" | sort --numeric-sort|tail -1` + 1
```

2. Create ***samba*** group and assign the larger of the group ids identified in step 1, e.g. 503.

```
# groupadd -g 503 samba
```

3. Identify the next available user id

```
# expr `cut --delimiter=":" --fields=3 /etc/passwd | \
> grep --regexp="^[5-9][0-9][0-9]$" | sort --numeric-sort|tail -1` + 1
```

4. Create ***samba1*** user and assign the larger of the user ids identified in step 3, e.g. 503. On the secondary node use the **-M** option so that **useradd** will not attempt to create the home directory.

```
# useradd --uid 503 --home-dir /samba/home/samba1 -gid samba samba1
```

5. On the primary node, create a Samba password for the new user.

```
# echo -e "qwerty\nqwerty" | \
> smbpasswd -c /samba/conf/smb.conf -a -s samba1
```

Build Application

The commands in this section are to be executed as ***root*** on the primary node.

1. Create directory **/etc/cluster/samba**.

```
# mkdir /etc/cluster/samba
```

2. Create **/etc/cluster/samba/appconf.xml** as shown:

```
<?xml version="1.0"?>
<appconf>
   <global>
      <version>0.1</version>
      <name>samba</name>
      <takeover>normal</takeover>
      <syncrate>2000</syncrate>
      <preferred_node>fc5s1</preferred_node>
   </global>

   <networks>
      <network net="main" ip="192.168.1.55" netmask="255.255.255.0" />
   </networks>

   <vg>
      <name>sambavg</name>
      <type>filesystems</type>
   </vg>

   <application>
      <startscript>/samba/scripts/startapp</startscript>
      <stopscript>/samba/scripts/stopapp</stopscript>
      <maxstoptime>10</maxstoptime>
      <maxstarttime>20</maxstarttime>
   </application>
</appconf>
```

The bolded values are configuration-specific, as described in the following table:

Entry	*Value*
global/preferred_node	Host name of one of the nodes in the cluster, or LEAST_CPU_LOAD
networks/network.net	Same as one of *node/network.name* in **/etc/cluster/clconf.xml**
networks/network.ip	Virtual IP address of application
networks/network.netmask	Netmask corresponding to virtual IP address
vg/name	Name of application volume group

3. Create **/etc/cluster/samba/lems.local.xml** as shown:

```
<?xml version="1.0"?>
<lems_config>
   <globals modules="/sbin/cluster/lems/modules"
      programs="/sbin/cluster/lems/programs"
```

```
      logs="/var/log/cluster/lems"
   />

   <check>
      <name>flag_check</name>
      <type>internal</type>
      <module>flag_check samba</module>
      <interval>5</interval>
      <action_list>
         <action rc="0" action="NOP"/>
         <action rc="1" action="%RCDATA%"/>
         <action rc="2" action="ABORT"/>
      </action_list>
   </check>
   <check>
      <name>smbd</name>
      <type>internal</type>
      <module>procmon /etc/cluster/samba/smbd.xml</module>
      <interval>10</interval>
      <action_list>
         <action rc="0" action="NOP"/>
         <action rc="1" action="STOP"/>
         <action rc="2" action="FAILOVER"/>
      </action_list>
   </check>
   <check>
      <name>ip</name>
      <type>internal</type>
      <module>ip_module samba</module>
      <interval>10</interval>
      <action_list>
         <action rc="0" action="NOP"/>
         <action rc="1" action="RUN move_ip"/>
         <action rc="2" action="STOP"/>
      </action_list>
   </check>
   <check>
      <name>fsmonitor</name>
      <type>internal</type>
      <module>fsmon samba</module>
      <interval>10</interval>
      <action_list>
         <action rc="0" action="NOP"/>
         <action rc="1" action="PAUSE 30"/>
         <action rc="2" action="STOP"/>
         <action rc="3" action="FAILOVER"/>
         <action rc="10" action="PAUSE 60"/>
      </action_list>
   </check>
</lems_config>
```

4. Create **/etc/cluster/samba/smbd.xml** as shown:

```
<?xml version="1.0"?>
<procmon>
   <global>
      <logdir>/var/log/cluster</logdir>
      <restarts>5</restarts>
      <resetwindow>3600</resetwindow>
      <restartcmd>
```

```
         /samba/scripts/stopapp ; /samba/scripts/startapp
      </restartcmd>
   </global>


   <process>
      <label>Samba Processes</label>
      <user>root</user>
      <process_string>
         (s|n)mbd -D -s /samba/conf/smb.conf*
      </process_string>
      <min_count>2</min_count>
      <max_count>10</max_count>
   </process>
</procmon>
```

5. Verify LEMS configuration by executing:

```
# lems.pl --config /etc/cluster/samba/lems.local.xml \
> --application samba --verbose --check --file /dev/tty
```

If successful, a result similar to the following will be displayed.

```
INFO  26/08/2006 19:54:06 Using modules from  : /sbin/cluster/lems/modules
INFO  26/08/2006 19:54:06 Using programs from : /sbin/cluster/lems/programs
INFO  26/08/2006 19:54:06 Writing logs to     : /var/log/cluster/lems
INFO  26/08/2006 19:54:06 Listening on port   : 9904
INFO  26/08/2006 19:54:06 Global initialisation complete.
INFO  26/08/2006 19:54:06 Started local server on port 9904
INFO  26/08/2006 19:54:06 Validating monitor entry ip...
INFO  26/08/2006 19:54:06 Validated monitor entry ip successfully.
INFO  26/08/2006 19:54:06 Validating monitor entry fsmonitor...
INFO  26/08/2006 19:54:06 Validated monitor entry fsmonitor successfully.
INFO  26/08/2006 19:54:06 Validating monitor entry smbd...
INFO  26/08/2006 19:54:06 Validated monitor entry smbd successfully.
INFO  26/08/2006 19:54:06 Validating monitor entry flag_check...
INFO  26/08/2006 19:54:06 Validated monitor entry flag_check successfully.
INFO  26/08/2006 19:54:06 Check mode - transferring validated config to remote node.
INFO  26/08/2006 19:54:06 Configuration transferred successfully.
INFO  26/08/2006 19:54:06 Calculated a check interval of 2.5 seconds.
```

6. Build **samba**.

```
# clbuildapp --application samba --sync
```

The output of a successful build is shown below:

```
INFO  26/08/2006 15:56:04 Backups directory defaulted to /clbackup
INFO  26/08/2006 15:56:04
INFO  26/08/2006 15:56:04 Validation of Application 'samba' started.
INFO  26/08/2006 15:56:04   ['/var/log/cluster/build/samba-check-300608261556.log']
INFO  26/08/2006 15:56:07 Initial Validation of Application successful.
INFO  26/08/2006 15:56:07
INFO  26/08/2006 15:56:07 NOTE: Build of new application is being performed.
INFO  26/08/2006 15:56:07
INFO  26/08/2006 15:56:07 Host Environment Validation started.
INFO  26/08/2006 15:56:07 ['/var/log/cluster/build/samba-envcheck-300608261556.log']
INFO  26/08/2006 15:56:09 Host Environment Validation successful.
INFO  26/08/2006 15:56:09
INFO  26/08/2006 15:56:09 Cluster state    : RUNNING
INFO  26/08/2006 15:56:09 Application state: UNDEFINED
INFO  26/08/2006 15:56:09
INFO  26/08/2006 15:56:09 Volume Group Configuration started.
INFO  26/08/2006 15:56:09   ['/var/log/cluster/build/samba-lvm-300608261556.log']
INFO  26/08/2006 15:56:15 Volume Group Configuration successful.
INFO  26/08/2006 15:56:15
```

```
INFO  26/08/2006 15:56:15 Application Resource Allocation started.
INFO  26/08/2006 15:56:15   ['/var/log/cluster/build/samba-build-300608261556.log']
INFO  26/08/2006 15:56:21 Application Resource Allocation successful.
INFO  26/08/2006 15:56:21
INFO  26/08/2006 15:56:21 Application Data Synchronisation started.
INFO  26/08/2006 15:56:21 ['/var/log/cluster/build/samba-syncdata-300608261556.log']
Storage Syncing:        256Mb/         1Mb [0.6 % Complete]
Storage Syncing:          0Mb/         0Mb [100 % Complete]
INFO  26/08/2006 15:58:42 Application Data Synchronisation successful.
INFO  26/08/2006 15:58:42
```

Run Application

The commands in this section are executed as ***root*** on either node except where indicated.

1. If necessary, form the cluster

```
# clform
```

2. Start **samba** on the primary node (*fc5s1*)

```
# clrunapp --application samba --node fc5s1
```

3. Check state of **samba**.

```
# clstat --application samba
```

A result similar to the following should be returned:

```
Cluster: cluster1 - UP

 Application       Node      State  Runnnig  Monitor  Stale  Fail-over?
       samba      fc5s1    STARTED  0:00:00  Running      0         Yes

 File Systems

 Mount Point              Valid   Type      State   % Complete  Completion
 /samba                    both   drbd       Sync

 Process Monitors

        Name    Status  Restarts   Current          Reset at
        smbd   Running         5         0               N/A

 General Monitors

             Type          Name    Status
      Flag Check    flag_check   Running
      FS Monitor     fsmonitor   Running
      IP Monitor            ip   Running
```

4. On the node where the application is running, list the Samba processes.

```
# ps -ef | grep "[sn]mbd" | grep -v grep
```

A result similar to the following should be returned:

```
root 7839    1 0 14:04 ? 00:00:00 smbd -D -s /samba/conf/smb.conf -l /var/log/samba
root 7840 7839 0 14:04 ? 00:00:00 smbd -D -s /samba/conf/smb.conf -l /var/log/samba
root 7842    1 0 14:04 ? 00:00:00 nmbd -D -s /samba/conf/smb.conf -l /var/log/samba
-H /samba/conf/lmhosts
```

5. Connect to the public share as the ***samba1*** user and display the contents of the test file.

```
# smbclient --user=samba1 --command="more test1;exit" \
> //192.168.1.55/public qwerty
```

A result similar to the following should be returned:

```
Domain=[FC5S1] OS=[Unix] Server=[Samba 3.0.23b]
getting file \test1 of size 12 as /tmp/smbmore.rxwuOE (11.7 kb/s) (average
11.7 kb/s)
public data
```

6. Connect to the private share as the ***samba1*** user.

```
# smbclient --user=samba1 --command="pwd;exit" \
> //192.168.1.55/samba1 qwerty
```

A result similar to the following should be returned:

```
Domain=[FC5S1] OS=[Unix] Server=[Samba 3.0.23b]
Current directory is \\192.168.1.55\samba1\
```

7. On the node where it is running, shut down **samba.**

```
# clhaltapp --application samba
```

8. Start **samba** on the secondary node (*fc5s2*).

```
# clrunapp --application samba --node fc5s2
```

9. Repeat steps 3 to 7 to test **samba** on the secondary node.