Clustered Samba in SLES 11
7.05.2010

Rolf Schmidt
Support Engineer
Novell, Inc.
Rolf.Schmidt@novell.com

May 27, 2010

Agenda

Introduction

Prerequisites

Limitations

Planning

Cluster Setup

CTDB integration

Debugging

Introduction

Broken Dreams

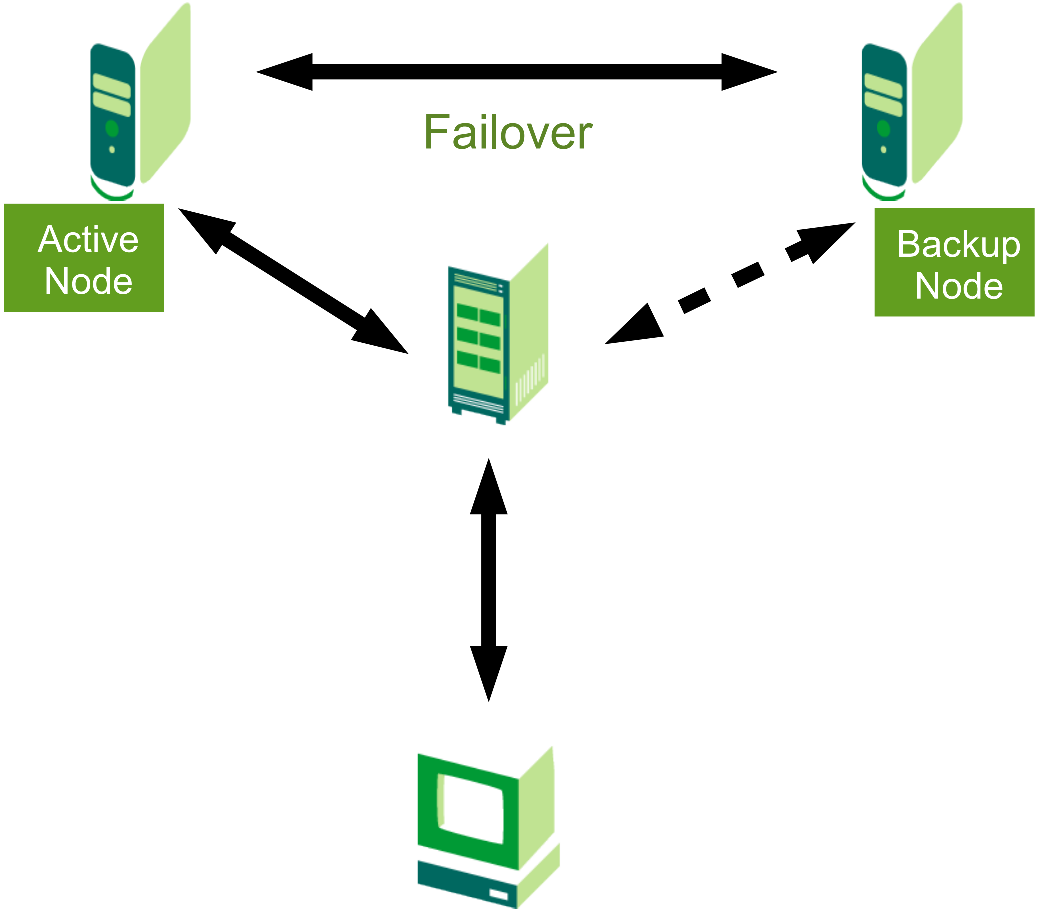
- Redundant but powerful File servers are extremely needed
- The “classic” way is to setup a High Availability solution like Heartbeat 1
- One node is active and one other in standby
- In case of a failure the service is stopped and started on the other node
- Advantage is that there is no data loss as the clients is notified of the interruption
- Drawback is the waste of resources, one node idles all the time

Heartbeat 1 Cluster

Real Servers

Virtual Server

Client



Broken Dreams

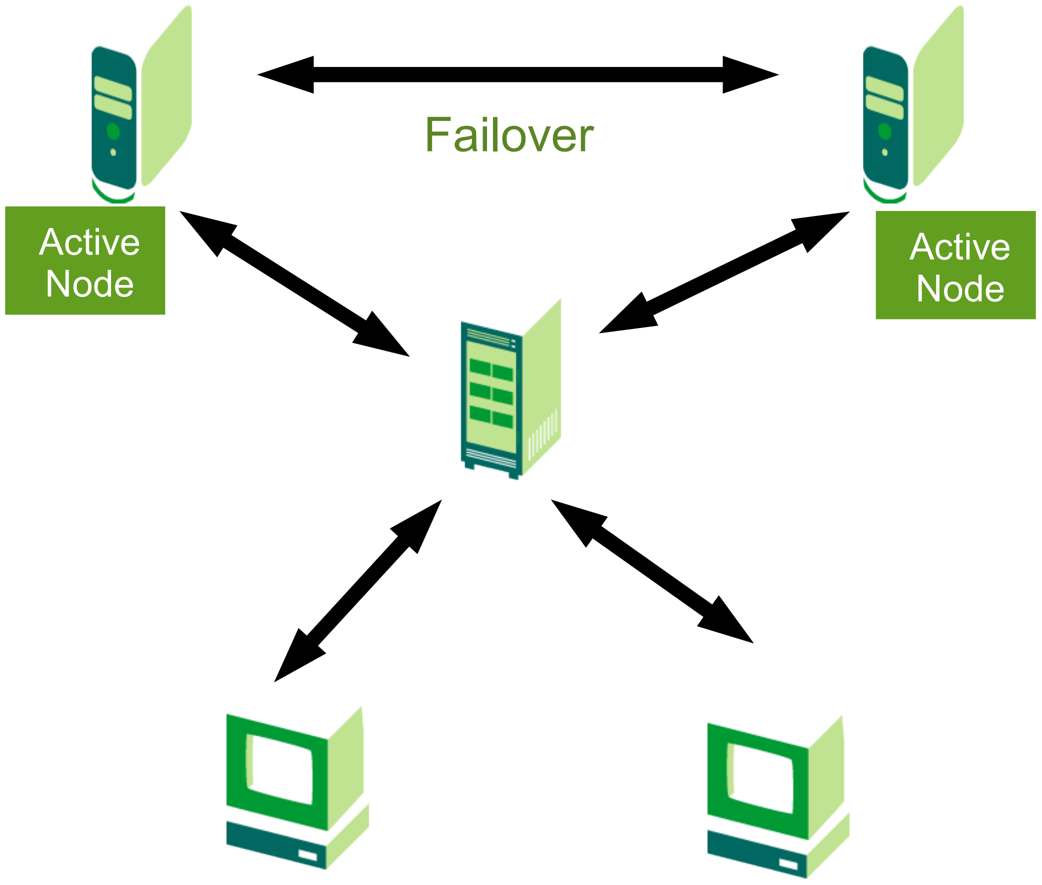
- The aim would be to have a cluster that
  - Provides on every node a service
  - Can provide the the service of the failed node
  - Would be as transparent as possible for the client
- And when the Samba Team showed CTDB I though my dreams would come true
- The dream Samba Cluster with Novell Support on SLES

Dream Cluster

Real Servers

Virtual Server

Client



Broken Dreams

- But that is much more difficult than it looks
-

Prerequisites

From Samba

- Version > 3.2
  - That rules out SLES 10 already as we are stuck with 3.0.x there.
- Cluster Filesystem with POSIX fcntl:
  - That rules out ocfs2 up to and including SLES11 GA (General Availbility)
- Solution:
  - SLES 11 SP 1 provides all that, Samba Version 3.4 and ocfs2 with locking
  -

From Novell

- Cluster control for the Cluster Filesystem
  - Running ocfs2 without openais/corosync on SLES11 is not supported. Only if it is Oracle RAC. Otherwise no.
- Cluster Contol for the IP Addresses:
  - Only openais/corosync can manage the IP Addresses . Not ctdb itself
- For a shared storage a STONITH device has to be in place
  - another piece of hardware has to be implemented
-

Conclusion

- SLES 11 SP1
  - We do not even have to try anything before that. Satisfies Clustered File system and Samba requirements from ctdb
- STONITH Device:
  - We need one otherwise no support
- SLES 11 SP1 HAE:
  - With SLES 11 it was decided to make High Availability an Ad don. Ctdb and ocfs2 are not shipped with the “normal” SLES 11
  -

Limitations

Losses

- Builtin Quorum determination from ctdb
  - Albeit it provides this, we cannot use it because we need openais/corosync as cluster
- IP Address control from ctdb:
  - Albeit it provides it, we cannot use it because we need openais/corosync as cluster
- Relative ease of setup
  - Instead of Samba, ctdb and clustered filesystem we have to setup Samba, ctdb, clustered filesystem and openais/corosync
-

Planning

My plan

- 2 node cluster
  - SLES 11 SP1 & HAE
- Windows Domain Controller
  - Windows 2003 R2
- Windows XP client
  - XP SP3
- Shared storage
  - iSCSI
  - This should be a small setup that still explains some of the pitfalls
-

Cluster Setup

Software

- SLES 11 SP1 Repository
- SLES 11 SP1 HAE Repository
  - selected patterns should be High Availability. Additionally it is necessary either to select File Server Pattern or manually samba and winbind. Ctdb has to be selected manually in any case, but is in SLES 11 SP1 HAE
-

Hardware

- 4 machines as 2 nodes and 1 server and 1 client
- One shared storage

  Hold on. Something missing. Did I not tell you that we do not support ocfs2 without STONITH device in place? We can have our STONITH and do NOT need special hardware.

SBD

The solution is the shared storage. It is called sbd device. Sbd is short for "split brain detection" and was and is used on NCS (Novell cluster Services).

- The idea is that a shared storage is used and one partition on this storage is reserved for sbd. This sbd writes onto this partition in a slot and reads from it. It is used to place a "Poison pill" iIf the cluster determines that one node has to die. Options:
  - either the node to be killed can access the shared storage, then it will die because of the poison pill
  - or the node to be killed can not access the shared storage, then it does not matter
- So we need no feedback from the STONITH apart from a successful write.

SBD

- Documentation
  - http://www.linux-ha.org/SBD_Fencing
- Configuration:
  - /etc/sysconfig/sdb
  - Which should read something like
- SBD_DEVICE="/dev/disk/by-id/scsi-SOMENAME-part1"
- SBD_OPTS="-W"
-

Shared Storage - iSCSI

- Setup
  - With the yast modul iscsi-initiator
- Problems:
  - SuSEfirewall2l
  - The SuSEfirewall2 comprises of two scripts, SuSEfirewall2_init and SuSEfirewall2_setup
  - SuSEfirewall2_init is started before the network and closes all ports. SuSEfirewall2_setup opens the selected ones afterwards
- Solution:
  - insserv -f -r SuSEfirewall2_init

openais/corosync Cluster

- 3 ways to configure
  - With xml snippets via cibadmin command from commandline
    - Can be automated, is extremly dangerous for typos
  - With crm shell from commandline
    - Fast but still requires to know all the syntax
  - With hb_gui
    - Slow but has a syntax checker, explanation and shows configuration options
    - Caveat, hb_gui is started as user hacluster that has no valid password set initially

Quorum woes

- Quorum
  - The cluster decides according to the quorum, meaning majority
  - The idea is, that if from 3 nodes 2 have one result and one has another then the two must be right
  - In a 2 node cluster there is no majority
  - In SLES 10 there was a tiebreaker plug in
  - In SLES 11 there is none anymore and the quorum setting has a meaning for 2 node clusters
  - In SLES 11 set it to “ignore” on a 2 node cluster

Corosync.conf

- Main Cluster infrastructure
  - The config file has changed. It is not as in SLES10 /etc/heartbeat/ha.cf anymore. Nor is it like SLES11 /etc/ais/openais.conf
  - It is now in /etc/corosync/corosync.conf
  - Can be configured via a yast module
  - Corosync supports more than one communication ring

Corosync Detour

- The changes from heartbeat2 (SLES10) to openais (SLES11) to corosync (SLES11 SP1) might seem strange
- But they actually show an advance in the product

http://www.openais.org/doku.php

http://www.corosync.org/doku.php

Dependencies

- The necessary elements to setup are, in this order
  - DLM Clone
  - O2cb Clone
  - Mount clone
- Order and Colocation:
  - The Clones should all have a Colocation INFINITY and an Order 0

```
colocation c1 inf: o2cb-base dlm-base
colocation c2 inf: mount-base o2cb-base
order o1 0: dlm-base o2cb-base
order o2 0: o2cb-base mount-base
```

Ctdb Integration

Runlevel Dangers

- The runlevel scripts for all resources controlled by the cluster have to be disabled
  - Nmb
  - Smb
  - Winbind
  - Ctdb

Ctdb configuration

- Apart from /etc/ctdb/nodes ctdb does not need to be configured
  - The cluster resource agent will modify the configuration file and overwrite all settings
  - Necessary configuration can be done
    - via the cluster resource agent
    - Via smb.conf

smb.conf

- smb.conf will be partially modified by the ctdb resource agent. Changes will be

```
# CTDB-RA: Begin auto-generated section (do not change below)
passdb backend = tdbsam
clustering = yes
idmap backend = tdb2
private dir = /var/lib/private
ctdbd socket = /var/lib/ctdb/ctdb.socket
# CTDB-RA: End auto-generated section (do not change above)
```

smb.conf of testcluster

- smb.conf will be changed by the yast module to join the windows domain, so most parameters will be set by this. Also nsswitch, pam, package dependencies and krb5.conf will be dealt with by this module.
- Other than that there is set among others

```
netbios name = appenzeller
winbind use default domain = yes
Interfaces = 149.44.174.137
password server = 149.44.174.140
fileid:algorithm = fsid
vfs objects = fileid
```

Debugging

Understand your enemy

The Cluster

- Keep in mind that all logs from the time of the error and before can be important.
- The cluster might have reacted and documented this reaction on one node
- But the actual cause for this might be on another node

Understand your enemy

The Stack

- Samba is controlled by ctdb
- Ctdb is run by the cluster
- The cluster is in his environment

- One approach is top to bottom
  - A cluster resource fails
    - Look in /var/log/messages for anything like error or timeout, the most common problems
      - Identify the resource
      -

Understand your enemy - closer

- Resource belongs to runlevel/system
    - File system
    - Network
    - Look in dmesg and check system information like ifconfig for dropped packages
- Resource belongs to Samba
    - Check /var/log/ctdb/log.ctdb
        - Identify element of Samba Suite that failed
            - Check corresponding log file in /var/log/samba/
            - Once the error is clear search for the system error that could have caused it.

Support

If anything does not work as expected then there is the support. Be it a Bug or be it misconfiguration

- For Novell Customer Support to work we need data
- The best tools/sources on SLES for this are
- Supportconfig
  - http://www.novell.com/coolsolutions/tools/16106.html
- hb_report
- /var/log/messages
- /var/log/ctdb/log.ctdb
  - (until integrated in supportconfig)

Common culprits

Time

- Ntp time synchronization is not only necessary for samba but also for the cluster software. Every Windows Server is also an NTP Server

- Paths
  - Not using /dev/disk/by.id/ in a cluster environment is asking for trouble because most paths have to be consistently the same on every node
- Cluster configuration
  - The complexity of a cluster configuration should not be underestimated

Common Misconceptions

Documentation

- There is documentation available from Novell

http://www.novell.com/documentation/

http://www.novell.com/documentation/sle_ha/

http://www.novell.com/documentation/sles11/

- And there is new documentation just waiting for SLES11 SP1 to be released already.

-

Configuration files 2 node Member in AD

smb.conf

```
workgroup = MIRACLE-WORKERS
realm = MIRACLE-WORKERS.DE
netbios name = appenzeller
printing = cups
log level = 1
password server = IP_OF_KDC
printcap name = cups
printcap cache time = 750
cups options = raw
map to guest = Bad User
include = /etc/samba/dhcp.conf
logon path = \\%L\profiles\.msprofile
logon home = \\%L\%U\.9xprofile
logon drive = P:
wins server = IP_OF_KDC
usershare allow guests = No
interfaces = 149.44.174.137
winbind use default domain = yes
security = ADS
idmap uid = 1000000-20000000
idmap gid = 1000000-20000000
fileid:algorithm = fsid
vfs objects = fileid
template homedir = /home/%D/%U
template shell = /bin/bash
winbind refresh tickets = yes
```

cluster

```
node appenzeller \
        attributes standby="off"
node fortuna \
        attributes standby="off"
primitive clvm-base ocf:lvm2:clvmd \
        operations $id="clvm-base-operations" \
        op monitor interval="10" timeout="20"
primitive ctdb-base ocf:heartbeat:CTDB \
        operations $id="ctdb-base-operations" \
        op monitor interval="10" timeout="20" \
        params ctdb_recovery_lock="/mnt/private/ctdb.lock" smb_private_dir="/var/lib/private"
primitive dlm-base ocf:pacemaker:controld \
        operations $id="dlm-base-operations" \
        op monitor interval="10" timeout="20" start-delay="0"
primitive fs-base ocf:heartbeat:Filesystem \
        operations $id="fs-base-operations" \
        op monitor interval="20" timeout="40" \
        params device="/dev/test/muell" directory="/mnt" fstype="ocfs2" options="acl,user_xattr"
primitive ip-base ocf:heartbeat:IPaddr2 \
        operations $id="ip-base-operations" \
        op monitor interval="10s" timeout="20s" \
        params ip="149.44.174.137" clusterip_hash="sourceip-sourceport" cidr_netmask="24"
primitive killer stonith:external/sbd \
        operations $id="killer-operations" \
        params sbd_device="/dev/disk/by-id/scsi-3600a0b800017b715000001054ab797aa-part1"
primitive o2cb-base ocf:ocfs2:o2cb \
        operations $id="o2cb-base-operations" \
        op monitor interval="10" timeout="20"
```

cluster - continued

```
primitive vg-base ocf:heartbeat:LVM \
        operations $id="vg-base-operations" \
        op monitor interval="10" timeout="30" \
        params volgrpname="test"
clone clvm-clone clvm-base \
        meta interleave="true" target-role="started"
clone ctdb-clone ctdb-base \
        meta interleave="true" target-role="started"
clone dlm-clone dlm-base \
        meta interleave="true" target-role="started"
clone fs-clone fs-base \
        meta interleave="true" target-role="started"
clone ip-clone ip-base \
        meta interleave="true" target-role="started" globally-unique="true"
clone moerder killer \
        meta interleave="true" target-role="started"
clone o2cb-clone o2cb-base \
        meta interleave="true" target-role="started"
clone vg-clone vg-base \
        meta interleave="true" target-role="started"
colocation c0 inf: dlm-clone moerder
colocation c1 inf: clvm-clone dlm-clone
colocation c2 inf: vg-clone clvm-clone
colocation c3 inf: o2cb-clone vg-clone
colocation c4 inf: fs-clone o2cb-clone
colocation c5 inf: ip-clone fs-clone
colocation c6 inf: ctdb-clone ip-clone
```

cluster - continued

```
order o0 0: moerder dlm-clone
order o1 0: dlm-clone clvm-clone
order o2 0: clvm-clone vg-clone
order o3 0: vg-clone o2cb-clone
order o4 0: o2cb-clone fs-clone
order o5 0: fs-clone ip-clone
order o6 0: ip-clone ctdb-clone
property $id="cib-bootstrap-options" \
        dc-version="1.1.1-536bf0b9d3ba6d412c67b27f89682ae9380b28ff" \
        cluster-infrastructure="openais" \
        expected-quorum-votes="2" \
        no-quorum-policy="ignore" \
        last-lrm-refresh="1274366814"
```

Thank you for your attention and patience

Novell®