

**Definitions**

HASF #High Availability Storage Foundation
Hypervisor #Domain 0, host, microkernel, VM Monitor
 #Privileged domain
Virtual Machine (vm) #DomU, guest, unprivileged domain
Migration #Relocation of vm; about 100ms downtime
STONITH #Shoot The Other Node In The Head

Xen restrictions

#[novell.com/documentation sles10 guide](http://novell.com/documentation/sles10/guide):
 ISA or PCMCIA devices tend not to work. ISA DMA is not supported.
 ACPI support in Xen is improving, but may not be up to date with respect to Linux. Some large-memory machines may not boot with ACPI enabled.
 Power management features, such as the suspend-to-disk feature, are not supported and should be disabled on both Domain 0 and any VM.
 SLES9 SP3 as VM unsupported #Possible SLES10 SPI feature

See SLES10 GA Release Notes, e.g:

No Full Virtualization guarantee #Intel VT or AMD Virtualization cpu works
http://wiki.xensource.com/xenwiki/HVM_Compatible_Processors
 Migration on x86-64 not stable #Will be available via update (release note)
 Hot-add-memory not yet #Will be available via update (release note)
 Max 16GB memory is stable #PAE kernel for memory > 4GB

Xen VM Migration setup

<http://www.novell.com/documentation/vmserver/index.html>

Install sles10 on 3 machines (2 nodes)

#Reserve >=4 GB unpart. space #Use bare part./LVM/EVMS/or sparse file
 #Include High Availability & XEN pattern, and yast2-vm, yast2-heartbeat
 #Do not use Network Manager
 #Enable ip_forward #By using yast or systemctl.conf
Packages iSCSI #iSCSI for demo purposes
 open-iscsi & iscsitarget #Do not plan initiator & target on 1 node
Packages XEN #Physical Address Extension: 32bit & 4G
 kernel-xen[pa]e
 xen, xen-libs, xen-tools, xen-tools-ioemu

Survive a reboot

/etc/init.d/powersaved stop #Prevent power management conflicts
 chkconfig powersaved off #Do not survive a reboot
 /etc/init.d/xend start #Start XEN daemon & survive reboot:
 checkconfig --level 35 xend on #Possibly troubleshoot network aft reboot
 #If required, change eth0 settings in three steps:
 /etc/xen/scripts/network-bridge stop netdev=eth0
 yast2 lan
 /etc/xen/scripts/network-bridge start netdev=eth0

Disable SuSEfirewall

#Or enable xenbr0 forwarding:
 vi /etc/sysconfig/SuSEfirewall2
 FW_FORWARD_ALWAYS_INOUT_DEV="xenbr0"
 /etc/init.d/SuSEfirewall2_setup restart

Default bootmenu

#default title nr starts with 0
 vi /boot/grub/menu.lst
 title SUSE Linux Enterprise XEN Server 10
 root (hd0,4)
 kernel /boot/xen-pae.gz
 module /boot/vmlinuz-xenpae root=/dev/sda5 vga=0x31a splash=silent
 showopts
 module /boot/initrd-xenpae

http://en.opensuse.org/Installing_Xen3

brctl show #Output after a reboot, e.g.:
 bridge name bridge id STP enabled interfaces
 xenbr0 8000.fffffffff no vif0.0 peth0

Setup a Virtual Machine

#Prepare XEN installation source, e.g.
 #Copy SLES10 ISO file to host
 #Show unpartitioned space
 fdisk -l
 #No cluster support for SLE10 and LVM vg. Use EVMS container(s) instead.
 vgdisplay #Show free space for LVM
 lvdisplay #Show Logical Volumes
 mount #Show mounted volumes
 evms_activate #If EVMS is not used during installation
 evmsgui #Show free space for EVMS
 dmsetup ls #Map Volume Name to mapper device
 system-vm2(253,3) #Maps to: /dev/dm-3
 vi /etc/xen/scripts/block #Remove -y for fast unsecure sparse file
 do_or_die losetup -y "\$sloopdev" "\$file"

Yast, System, Virtual Machine Management (Xen)

xm create /etc/xen/vm/vm1 #Start VM. Create domain in memory.
 xm list #Get <DomId>
 xm console 2 #Open screen <DomId>
 #Disconnect: <Ctrl-]>
 xm top #Monitor domains. Base info: xm info
 xm shutdown 2 #Shutdown VM <DomId>
 xm reboot 2 #Restart VM <DomId>
 xm destroy 2 #Kill VM <DomId>
 xm migrate 2 host #Migrate <DomId> hostname
 xm pause 2 #Standby VM <DomId>
 xm unpause 2 #Start VM <DomId> from standby
 xm save 2 file #Suspend VM <DomId> to filename
 xm restore file #Resume VM from filename
 xm mem-set 0 512 #Set memory of domain 0 to 512MB

Heartbeat 2 Setup

#At least 2 ethernet cards, 2 nodes
 #Recommended redundant NICs #Software RAID 1 is not supported
 #Optional STONITH Agent for power supply
http://www.novell.com/documentation/sles10/hb2/data/hb2_config.html

Heartbeat 2 definitions

#Four Layers
 Messaging/Infrastructure Layer #"I am alive" "Heartbeat" Layer
 Membership Layer #Cluster Consensus Membership service
 Resource Allocation Layer #Resource Administration Services
 Resource Layer # (RA) Resource Agents with scripts
 Standby node #Node with ability to run a resource
 Cluster Resource Manager # (CRM) Master of Ceremony. One is DC:
 Designated Coordinator # (DC) Own/react on master CIB changes
 Cluster Information Base # (CIB) XML cluster setup/view
 Policy Engine # (PE) Step ordering
 Transition Engine # (TE) Step execution
 Local Resource Manger # (LRM) Call Resource Agents

Setup name resolution

#Make all hostnames resolvable on nodes
 #Or setup DNS. Check via: ping nodename
 vi /etc/hosts #Initial time set: ntpdate ntp.srv.com, or:

Setup time sync

ssh node1 date \$(date +%m%d%H%M)
 ssh node2 date \$(date +%m%d%H%M)
 vi /etc/ntp.conf #Time syncs only with max 1000 sec delta
 server ntp.srv.com #Comment out server and fudge lines
 /etc/init.d/ntp start #Start time sync and wait 5 minutes.
 ntpq -p #Check time sync. (or ntptrace)

**Initial heartbeat setup**

`yast2 heartbeat`
Add node(s), Next
Select authentication key, Next
On and Survive reboot
`/usr/lib/heartbeat/ha__propagate`
`cat /etc/ha.d/authkeys`
`cat /etc/ha.d/ha.cf`
`cat /var/lib/heartbeat/crm/cib.xml`
`/etc/init.d/heartbeat start`
`chkconfig --level 35 heartbeat on`
`passwd hacluster`
`hb_gui`
+ (add new item), native type, OK,
Resource ID: `test-ip`, IPaddr (OCF RA) as Type, Param. Value: `ip:172.17.0.170`,
Optional: Add Parameter, Name: `nic`, Value: `eth0`, OK,
Start Resource (MB2)

Heartbeat administrative tools

`hb_gui`
`crmadmin`
`cibadmin`
`crm_verify`
`crm_mon`
`crm_resource`
`crm_standby`
`cl_status`
<http://linux-ha.org/v2>

iSCSI Setup

Initiator
Target
Add a new partition on target
`yast disk`
Do not mix LVM & EVMS
No cluster support for SLE10 and LVM vg. Use EVMS container(s) instead,
but iSCSI provides a block device not an LVM volume group
`dmsetup ls`
`system-vm2(253,3)`

Configure iSCSI target

`yast iscsi-server`
When booting, (Open Firewall)
Delete demo target
Add target
No authentication in demo

Configure iSCSI initiator

`yast iscsi-client`
When booting
Discovery, Fill in IP of target
Login, no authentication
Toggle startup, Finish
`yast disk`
Create part. for VM image
Create part. for VM config. files
Create part. for VM data storage

Configure iSCSI initiator

#Add nodes & propagate configuration
#On one of the nodes
#Unlimited. Tested to 16
#Same on all nodes (none, md5, or sha1)
#`chkconfig --level 35 heartbeat on`
#Replicate configuration to nodes
#View configuration file
#View configuration file
#Is replicated aft heartbeat start
#On the other node(s)
#On the other node(s)
#Give user(s) `hacluster` a password for:
#Add a resource (from any node), e.g.:

#Used as cheap SAN for Image Store
#User of the block level iSCSI device
#Host sharing the block device
#e.g. `/dev/hdab`, `/dev/vg/lv`,
#`/dev/evms/lvm2/cont/lv`
#Best practices
#Map Volume Name to mapper device
#Maps to: `/dev/dm-3`
#File, block device, RAID or LVM device
#Provide `/dev/sd` device for client
#In Service tab
#In Targets tab
#e.g. `/dev/vg-usb/lv-xeni`
#Next, OK, Finish, Restart, Yes
#From first node
#Connect IET-VIRTUAL-DISK `/dev/sd`
#In Service tab
#In Discovered Targets tab

#Survive reboot
#Add 3 partitions (IET-VIRTUAL-DISK):
#Leave e.g. 300MB free. No mount point
#Use e.g. 200MB. No mount point
#Use e.g. 100MB. No mount point
#From other node(s) (discovery only)

OCFS2 Setup

Can run in pure OCFS2 Cluster
`ocfs2console`
Initialize the native OCFS2 stack
Cluster, Configure nodes, Close
Add nodes (incl. first), Close
Close, Cluster, Propagate Config.
`/etc/init.d/o2cb configure`
`/etc/init.d/o2cb force-reload`
`cat /sys/o2cb/heartbeat_mode`
`find /sys/kernel/config/cluster`
`mkfs.ocfs2 /dev/sda1`
`mounted.ocfs2 -d /dev/sda1`

OCFS2 in heartbeat cluster

`crm_mon -l`
Test by manually mount ocfs2
Setup ssh keys for root
Enable atd
`pkill heartbeat`

VM as Cluster Resource

Change sync mode of loop device
`vi /etc/xen/scripts/block`
Create VM on node1
Restore -y
Stop VM on node1
`yast xen`
`cibadmin -C -o crm_config -x /vmllocation.xml`
`cibadmin -C -o crm_config -x /vml.xml`
`crm_mon -l`
`cibadmin -C -o crm_config -x /vmlorderconstraints.xml`
`pkill heartbeat`
`crm_mon -l`

Extend maximum loop mounts

`rmmmod loop`
`modprobe loop max_loop=64`
`vi /etc/modprobe.conf`
options loop max_loop=64

#Oracle Cluster File System
#Integration with heartbeat2 (user space)
#GUI for setup and propagation
#Only on ONE node
#Name, IP and port (TTTT)
#Copy via ssh and close ocfs2console
#Add heartbeat and bootconfig (all nodes)
#On first node only (because of failure)
#Check for 'user'
#Interface between kernel & user space
#Create OCFS2 file systems (`sda1` & `sda2`)
#Ask for UUID
#Integrate
#Clonesets can run concurrent and on all nodes
#OCFS2 via clone File System RA on each node
#Notify resource stop/start, node join/leave
#Demo stonith device is ssh reboot:
#Simulate node crash by killing heartbeat and not unplug cable
#Create XML blobs for the CIB
#List cluster resources
#Remount should occur
#For unattended ssh stonith
#For ssh stonith
#Test by node crash emulation
#Undo after installing VM
#Remove -y at `do_or_die`
#Use default Sparse File
#Sync mode
#Inside VM is a not cluster safe fs
#Check availability on node2
#xm list
#Test by node1 crash emulation
#xm list on node2
#Default max loops is 8
#Extend without reboot. Remove module.
#Extend without reboot
#Was max_loop=64 as SLES9 boot par.