



High Availability 2-Node XenServer Pool Reference Design
For use with XenServer 6.x



Reference Design and How-To
For

High Availability 2-Node XenServer Pool
Provides Full Functionality with Live Migration
Without External Shared Storage
for HA-Lizard and XenServer 6.x

Version 2.1

IMPORTANT: For XenServer 7.x – refer to XenServer 7.x specific reference design



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

The information in this document and any product or service specifications referred to herein are subject to change without notice.

XenServer, XenCenter, Xen Cloud Platform and XCP are registered trademarks or trademarks of Citrix System, Inc and Xen.org
DRBD and Distributed Replicated Block Device are registered trademarks or trademarks of LINBIT HA-Solutions GmbH

No part of this document may be reproduced, copied, altered or transmitted in any form or by any means, electronic, mechanical or otherwise for any purpose whatsoever, without the express written permission of the Copyright owner.

The information provided in this document is intended as a guide only and is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

No support is provided as part of the information in this document or any related software. Contact the project sponsor, Pulse Supply (www.pulsesupply.com), for details on support offerings.

**Copyright © 2016 Salvatore Costantino
All rights reserved.**

All rights reserved. No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law.

IMPORTANT

#####

**!!! iSCSI-HA and HA-Lizard are free software: you can redistribute it and/or modify
!!! it under the terms of the GNU General Public License as published by
!!! the Free Software Foundation, either version 3 of the License, or
!!! (at your option) any later version.**

!!!

**!!! iSCSI-HA and HA-Lizard are distributed in the hope that it will be useful,
!!! but WITHOUT ANY WARRANTY; without even the implied warranty of
!!! MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
!!! GNU General Public License for more details.**

!!!

**!!! You should have received a copy of the GNU General Public License
!!! along with iSCSI-HA and HA-Lizard. If not, see <<http://www.gnu.org/licenses/>>.**

#####



Table of Contents

1. iSCSI-HA Add-on for XCP and XenServer..... 6
• Purpose 6
• Requirements 7
2. Create a 2-Node Highly Available Cluster..... 8
• Assumptions 8
Server Hardware 8
Ethernet Switch 8
Required Software..... 8
• IP Addresses and Host Names 9
• Step 1 – Install XCP/XenServer 10
• Step 2 – Create a Bonded Interface and Setup Pool Networking 10
• Step 3 – Install DRBD 13
Installing With Included RPMs 13
Installing from Source (skip this step if using the included DRBD RPMs) See appendix for version specific instruction on installing from source..... 14
• Step 4 - Install HA-Lizard 14
• Step 5 – Install iSCSI-HA 15
• Step 6 – Install iSCSI Target..... 15
• Step 7 – Initialize Packages 15
Identify and Export iSCSI Backing Store..... 16
Configure DRBD..... 16
Update LVM filters 17
Configure iSCSI-HA..... 18
Configure HA-Lizard 18



• Step 8 – Start Services 20
• Step 9 – Create a new SR 20
3. Managing the 2-Node Highly Available Cluster 21
• Performance and System Capacity 21
Resources and Capacity 21
Performance 22
• System Behavior 22
• iscsi-cfg CLI Tool 23
System Logging 23
Viewing Configuration Parameters 23
Viewing iSCSI-HA Status 23
• Operating in Manual Mode 26
Enabling Manual Mode 26
Disabling Manual Mode 26
Becoming the Storage Primary Role 26
Becoming the Storage Secondary Role 26
• Manual Mode Best Practices 27
Existing Manual Mode Cleanly 27
Entering Manual Mode 27
• Dealing with DRBD Split Brain 28
Miscellaneous 29
• Managing Services 29
DRBD 29
TGTD 29



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

- Dependencies and Compatibility 29
- Important Considerations 30
- Security and Ports..... 30
- Support..... 30
- 4. Appendix A – Installing DRBD from Source 31
 - XCP 1.6 and XenServer 6.1 31
 - XenServer 6.2..... 32
- 5. Appendix B - Example Maintenance Operations..... 33

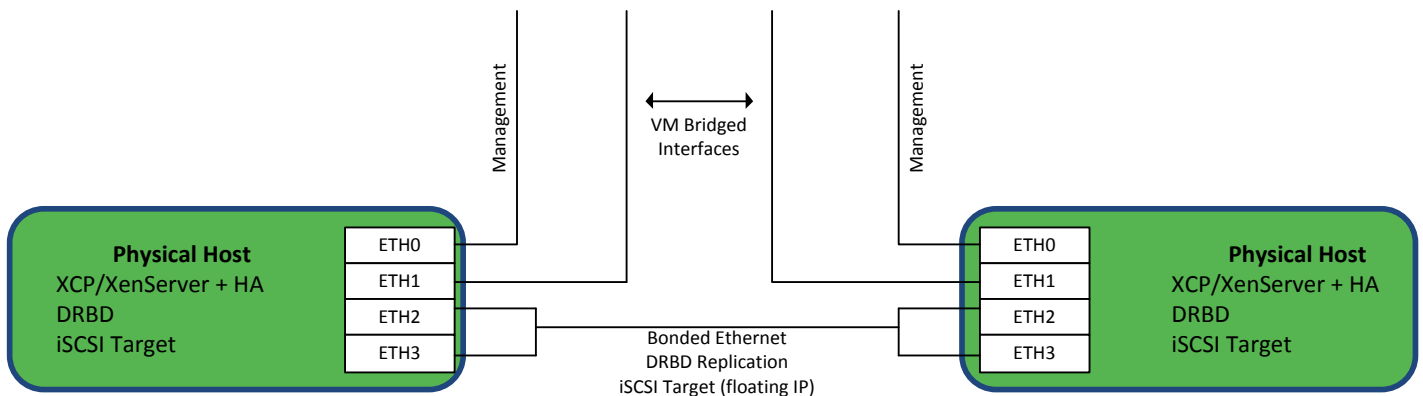
1. iSCSI-HA Add-on for XCP and XenServer

- Purpose**

iSCSI-HA is an add-on module for two node pools utilizing Xen Cloud Platform (XCP) or XenServer virtualization environments. It is intended to build highly available two-node clusters with local storage without limiting pool advanced functionality such as live migration.

Generally, this is achieved with separate iSCSI/SAN and Xen clusters which require a minimum of four physical servers and a pair of redundant Ethernet switches to reach an adequate level of fault tolerance. This may not be the most efficient use of hardware for small cluster applications. The goal of iSCSI-HA is to provide a simple framework for building compact, highly available pools utilizing XenServer or Xen Cloud Platform with just two physical hosts.

iSCSI-HA requires DRBD for block replication of storage, an iSCSI target framework such as TGT and a pool/VM High Availability component capable of providing HA for a 2-node Pool (HA-Lizard HA component). A sample pool design/diagram is shown below for a highly available two node pool.



In this example, the DRBD and iSCSI interface is provided via direct attachment (no Ethernet switches) on a bonded Ethernet link. This approach greatly eliminates the possibility of a split brain scenario since there are no networking devices interconnecting the hosts in any way. Additionally, utilizing a bonded Ethernet link further eliminates the possibility of communication interruption between the hosts.

The iSCSI-HA add-on does not make any decisions or employ any logic relating to cluster management and the roles of the hosts. It relies on an external HA tool such as open source HA-Lizard which supports HA in a 2 node environment with fast switching of roles. The iSCSI-HA add-on relies on the external HA logic to ensure that a pool Master is always available. Based on this, iSCSI-HA will assign a single shared/floating IP address to the Master and promote DRBD resources to follow the floating IP. The slave host will be in a demoted state



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

at all times. In the event of a change of pool roles, iSCSI-HA will automatically detect the new roles and promote the new Master as the iSCSI target and demote the former Master to slave/standby mode. As of version 1.4, a manual mode is also supported which allows for simple pool maintenance with no VM downtime.

- **Requirements**

- XCP version 1.6 or XenServer version 6.1/6.2/6.5
- DRBD Version 8.3 or 8.4 (note: Version 8.4 requires that openvswitch be disabled in dom0)
- iSCSI target – TGT
- Pool HA – Open Source HA-Lizard

iSCSI HA features provided:

- VMs are free to run on the Master or Slave host
- Live Migration support
- Support for shared/floating virtual IP address to export iSCSI LUNs
- Automatic promoting of DRBD resources on pool Master
- Automatic demoting of DRBD resources on pool Slave
- Manual mode allows for host upgrade and reboots with no storage downtime.
- Management of iSCSI service
- Extensive Logging capabilities to system log file
- Email alerting
- Dynamic iSCSI target selection auto-selects roles
- No changes to existing pool configuration required. All logic is external.
- Auto-Plug XenServer SRs that fail to connect on host boot
- Auto-Replug XenServer SRs that are not properly connected
- **Minimal dependencies** – does not compromise pool stability or introduce complex SW packages. Designed to work with the resident packages on a standard XCP/XenServer host with the addition of DRBD and TGT.

Development is well tested and based on:

- Xen Cloud Platform (XCP) version 1.6
- XenServer version 6.1
- XenServer version 6.2
- XenServer version 6.5
- HA-Lizard version 1.6.41.4 or newer
- DRBD 8.3 and DRBD 8.4
- TGT iSCSI Target



2. Create a 2-Node Highly Available Cluster

- **Assumptions**

Server Hardware

Start with two identical servers with 4 LAN interfaces and two disk partitions. This How-To is based on HP DL-360 servers with HW RAID 1+0 and four disks.

- Disks 1+2 create the first RAID 1+0 array and will be used to install XCP/XenServer
- Disks 3+4 create the second RAID 1+0 array and will be used as the iSCSI backing store

Ethernet Switch

A managed Ethernet switch is used to connect the server management interfaces. The switch management IP can be used for HA-Lizard IP Heuristics. If a managed switch is not available, any IP address reachable by the management interface of the XenServer dom0s can be used as long as the switch is traversed to reach the outside IP.

Required Software

- iSCSI-HA version 1.2x (or later)
- XenServer 6.1/6.2/6.5 or Xen Cloud Platform (XCP) 1.6
- Logic to ensure there is always a pool master.
HA-Lizard 1.6.41.4 or newer
- DRBD 8.3 or 8.4 for data replication
- TGT iSCSI target

IMPORTANT – Unless otherwise specified – all steps should be performed on both hosts



- **IP Addresses and Host Names**

The following IP addresses, host names and paths are used in this How-To.
Adapt the settings presented to match your environment.

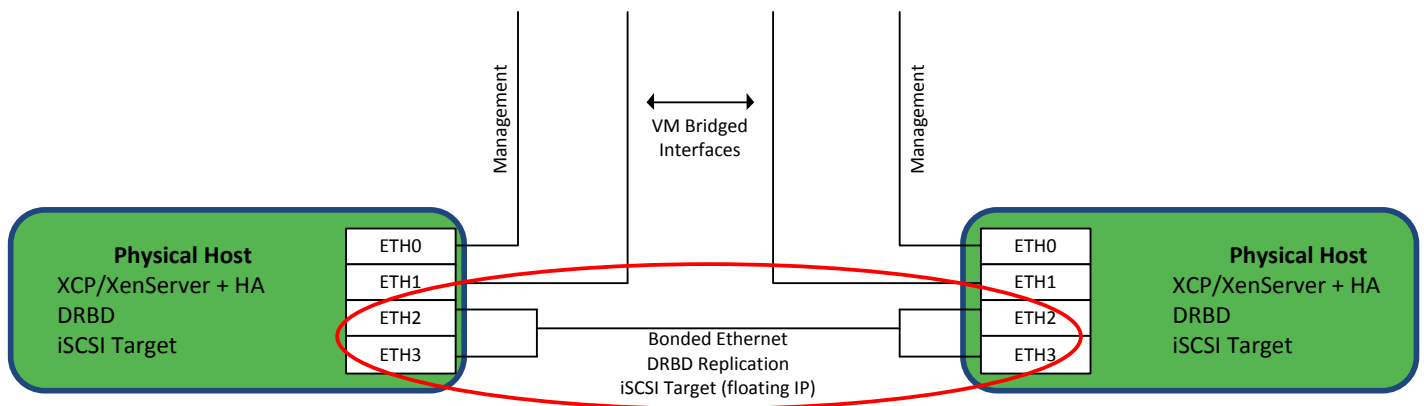
Node 1 hostname	XCP1
Node 2 hostname	XCP2
Node 1 XenServer Management IP	192.168.1.241
Node 2 XenServer Management IP	192.168.1.242
Node 1 DRBD/iSCSI IP Address	10.10.10.1
Node 2 DRBD/iSCSI IP Address	10.10.10.2
Shared (floating) iSCSI Address	10.10.10.3
IP Address of Ethernet Switch on Management Network or some other reliable IP that is accessed by traversing the management network	192.168.1.253
iSCSI/DRBD Backing Device	/dev/cciss/c0d1
DRBD Resource Name	iscsi1
DRBD Local Resource	/dev/drbd1

- **Step 1 – Install XCP/XenServer**

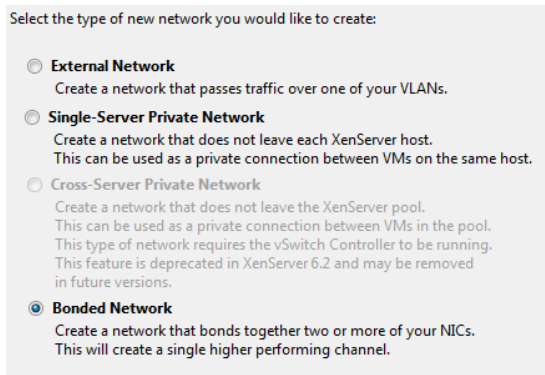
- Install XCP or XenServer on two identical hosts with a minimum four Ethernet interfaces and 2 Disks or Disk partitions. Install XenServer/XCP on the first partition - leave the second disk/partition to be used later as iSCSI storage.
- When installing – select Ethernet 0 as the management interface for each host.
- Connect to one of the hosts with XenCenter and create a new pool with the two hosts.

- **Step 2 – Create a Bonded Interface and Setup Pool Networking**

The third and fourth network interfaces (“NIC2 and NIC3”) will be used to create a bonded network as depicted below. This will serve as both the replication link and the iSCSI interface.



- From XenCenter – Select the “Networking” tab for the pool and “Add Network”
- Select “Bonded Network”





High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

- Select NIC 2 and NIC 3 and then “Finish”

Select the NICs you would like to use in this bond and the bond settings, and confirm whether this network should be added to new VMs.

NIC	MAC	Link Status	Speed	Duplex	Vendor	Device	
<input type="checkbox"/>	NIC 1	00:1b:78:e2:39:50	Connected	100 Mbit/s	Full	Broadcom Corporation	NetXtreme II
<input checked="" type="checkbox"/>	NIC 2	00:24:81:80:0a:5a	Connected	1000 Mbit/s	Full	Intel Corporation	82571EB Gig
<input checked="" type="checkbox"/>	NIC 3	00:24:81:80:0a:5b	Connected	1000 Mbit/s	Full	Intel Corporation	82571EB Gig

Bond mode

Active-active

Active-passive

LACP with load balancing based on IP and port of source and destination

LACP with load balancing based on source MAC address

MTU:

Automatically add this network to new virtual machines

- From within XenCenter – assign an IP address for the bond on each host. In this example we use 10.10.10.1 and 10.10.10.2. Since the replication/iSCSI network is completely closed, it is not necessary to configure a default gateway for the interface.

Storage 1

Name:

Network:

IP address settings:

Automatically obtain settings using DHCP

Use these settings:

IP address:

Subnet mask:

Gateway:



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

- The final pool network configuration should look something like this.

Pool Networks

Networks

Name	Description	NIC	VLAN	Auto	Link Status	MAC	MTU
Bond 2+3		Bond 2+3	-	No	Connected	00:24:81:80:0a:5a	1500
Network 0		NIC 0	-	Yes	Connected	00:1b:78:e3:ce:7a	1500
Network 1		NIC 1	-	Yes	Connected	00:1b:78:e3:ce:92	1500
Network 2 (Slave)		NIC 2	-	Yes	Connected	00:24:81:80:0a:5a	1500
Network 3 (Slave)		NIC 3	-	Yes	Connected	00:24:81:80:0a:5b	1500

Add Network... Properties Remove

IP Address Configuration

Server	Interface	Network	NIC	IP Setup	IP Address	Subnet mask	Gateway	DNS
XCP1	Management	Network 0	NIC 0	Static	192.168.1.241	255.255.255.0	192.168.1.1	8.8.8.8
XCP1	DRBD/iSCSI	Bond 2+3	Bond 2+3	Static	10.10.10.1	255.255.255.0		
XCP2	Management	Network 0	NIC 0	Static	192.168.1.242	255.255.255.0	192.168.1.1	8.8.8.8
XCP2	DRBD/iSCSI	Bond 2+3	Bond 2+3	Static	10.10.10.2	255.255.255.0		

- Check each of the network properties and ensure that “Automatically add this network to new virtual machines” is only selected for Network 1 (assuming you will not use the management interface for VM interfaces).
- Update firewall files to allow DRBD and iSCSI network traffic. The following line can be added to the iptables firewall script just above the “REJECT” line.
“-A RH-Firewall-1-INPUT -s 10.10.10.0/24 -j ACCEPT”

vi /etc/sysconfig/iptables

Insert “-A RH-Firewall-1-INPUT -s 10.10.10.0/24 -j ACCEPT” save/exit – then restart FW



service iptables restart

```
# Firewall configuration written by system-config-securitylevel
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p 50 -j ACCEPT
-A RH-Firewall-1-INPUT -p 51 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp --dport 5353 -d 224.0.0.251 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 67 --in-interface xenapi -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 694 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
-A RH-Firewall-1-INPUT -s 10.10.10.0/24 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

- **Step 3 – Install DRBD**

- DRBD RPMs are included in the iscsi-ha source package. These can be used only if the kernel version of your XCP/XenServer installation exactly matches the version used to build the RPMs. If using a different kernel version, it will be necessary to build RPMs from source for your kernel. The included RPMs will work with XCP 1.6, XenServer 6.1 and XenServer 6.2

Installing With Included RPMs

- Extract the iscsi-ha source package in a temporary location
tar -zxvf iscsi-ha*.tgz
- Move into the RPM directory
cd /iscsi-ha*/RPM
- Make a note of the kernel version used to prepare the provided RPMs
ls -l
- Check Your Kernel Version
uname -a
- Move into the directory matching your kernel
cd <kernel version>



- For **XenServer 6.2** or below, Install the required RPMs for the matching kernel (adapt below to match your version)

```
rpm -ivh drbd-utils-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-km-2.6.32*.rpm
```

```
rpm -ivh drbd-bash-completion-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-heartbeat-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-pacemaker-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-udev-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-xen-8.3.15-1.i386.rpm
```

```
rpm -ivh drbd-8.3.15-1.i386.rpm
```

- For **XenServer 6.5** or newer, Install the required RPMs for the matching kernel (adapt below to match your version)

```
rpm -ivh drbd-utils-8.4.3-2.x86_64.rpm
```

Important: As of iSCSI-HA version 2.0, DRBD RPMs are no longer packed with the SW release. If you are in need of the DRBD packaged, use iSCSI-HA version 1.9.1.

- Ensure that DRBD does not start automatically on boot. iSCSI-HA will be responsible for starting the service.

```
chkconfig drbd off
```

Installing from Source (skip this step if using the included DRBD RPMs)
See appendix for version specific instruction on installing from source.

- **Step 4 - Install HA-Lizard**

- Copy the source tarball into a temporary location (ex. /tmp/)

- Extract its contents and move into the extracted folder

```
tar -zxvf ha-lizard-1.6.4*.tgz
```

- Move into the “scripts” folder

```
cd ha-lizard-1.6.4*/scripts
```

- Run the installer

```
./install --nostart
```



The installer will check if sendmail packages are installed on the server. These are only required for email alerts. Skip the installation of these packages if email alerting is not required.

The installer will install the default pool parameter set in the XAPI database. This step is only required on a single host.

Once the installer is completed, HA and watchdog services will be started. Although these services are running, HA is disabled for the pool by default. HA can be enabled via the command line tool `<ha-cfg>` once installation of additional packages is completed.

- **Step 5 – Install iSCSI-HA**

- Copy the source tarball into a temporary location (ex. /tmp/)
- Extract its contents and move into the extracted folder
`tar -zxvf iscsi-ha-<version>.tgz`
- Move into the “scripts” folder
`cd iscsi-ha-<version>/scripts`
- Run the installer
`./install --nostart`
- Temporarily stop the service (only if your version does not support the `--nostart` argument)
`service iscsi-ha stop -w`

The installer will check if sendmail packages are installed on the server. These are only required for email alerts. Skip the installation of these packages if email alerting is not desired.

- **Step 6 – Install iSCSI Target**

- Install the scsi-target-utils package
`yum --enablerepo=base install scsi-target-utils`
- Ensure that TGT does not start automatically on boot. iSCSI-HA will be responsible for starting the service.
`chkconfig tgtd off`

- **Step 7 – Initialize Packages**



Identify and Export iSCSI Backing Store

- Use fdisk to find the name of the disk partition to be used for the iSCSI backing store. In this example, the device path is “/dev/cciss/c0d1”

fdisk -l

```
[root@XCP2 tmp]# fdisk -l
WARNING: GPT (GUID Partition Table) detected on '/dev/cciss/c0d0': The util fdisk doesn't support GPT. Use GNU Parted.

Disk /dev/cciss/c0d0: 146.7 GB, 146778685440 bytes
256 heads, 63 sectors/track, 17775 cylinders
Units = cylinders of 16128 * 512 = 8257536 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/cciss/c0d0p1  *           1         17776     143338559+  ee  EFI GPT

Disk /dev/cciss/c0d1: 450.0 GB, 450064605184 bytes
256 heads, 32 sectors/track, 107724 cylinders
Units = cylinders of 8160 * 512 = 4177920 bytes

Disk /dev/cciss/c0d1 doesn't contain a valid partition table
```

- DRBD will create a resource replicating this block device (/dev/cciss/c0d1). We will use DRBD resource name “/dev/drbd1” in the iSCSI configuration. Modify the iSCSI target configuration file to export “/dev/drbd1”. (if using multiple partitions, this can be adapted to suit your environment). Set the scsi_id and scsi_sn to suit your needs. These can be omitted if desired.

vi /etc/tgt/targets.conf (or use your preferred editor)

- Add this section to the configuration file (iqn, scsi_id, scsi_sn should be the same on both hosts)
<target iqn.2013-05.com.yourdomain:yourhost>
 backing-store /dev/drbd1
 scsi_id 0000000000
 scsi_sn 0000000001
 lun 10
</target>

Configure DRBD

- Backup current DRBD configuration file in case you need it.
mv /etc/drbd.conf /etc/drbd.conf.backup
- Create/Edit new /etc/drbd.conf with the settings below (adapt hostname, disk and IP addresses to your environment)
vi /etc/drbd.conf
- Insert the following configuration parameters

```
global { usage-count no; }
common { syncer { rate 100M; } }
resource iscsi1 {
    protocol C;
    net {
```




```
    after-sb-0pri discard-zero-changes;
    after-sb-1pri consensus;
}
cram-hmac-alg sha1;
    shared-secret "PUTyourSECRETHere";
}
on XCP1 {
    device /dev/drbd1;
    disk /dev/cciss/c0d1;
    address 10.10.10.1:7789;
    meta-disk internal;
}
on XCP2 {
    device /dev/drbd1;
    disk /dev/cciss/c0d1;
    address 10.10.10.2:7789;
    meta-disk internal;
}
}
```

- Initialize the Disks

```
dd if=/dev/zero bs=1M count=1 of=/dev/cciss/c0d1
drbdadm create-md iscsi1
```

- Start the DRBD service

```
service drbd start
```

```
drbdadm syncer iscsi1 (only required for DRBD version 8.3)
```

**** ON PRIMARY DATA SOURCE ONLY ** :**

```
drbdadm -- --overwrite-data-of-peer primary iscsi1
```

Update LVM filters

LVM filters must be updated to prevent VG/LV metadata from being read from both the backing block device and the DRBD device. VG/LV data must ONLY be read from /dev/iscsi. This step is mandatory for proper operation.

- Edit /etc/lvm/lvm.conf and update filter to look something like this to reject reading LVM headers locally.

```
vi /etc/lvm/lvm.conf
```



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

- Update filter to (restrict local backing device and drbd device – adjust to your environment)
** Important – LVM Headers for iSCSI-HA storage must only be ready from /dev/iscsi **
“filter = ["r|/dev/xvd.|", "r|/dev/VG_Xen.*/*|", "r|/dev/cciss/c0d1|", "r|/dev/drbd.*|"]”
- Set -> “write_cache_state=0”
- When done – erase the LVM cache to ensure cached data is not read by LVM.
rm -f /etc/lvm/cache/.cache && vgscan

Configure iSCSI-HA

- Edit /etc/iscsi-ha/iscsi-ha.conf
vi /etc/iscsi-ha/iscsi-ha.conf
- Make the following configuration changes and save.

```
DRBD_RESOURCES=iscsi1
ISCSI_TARGET_SERVICE=/etc/init.d/tgtd
DRBD_VIRTUAL_IP=10.10.10.3
DRBD_VIRTUAL_MASK=255.255.255.0
DRBD_INTERFACE=xapi0
(if unsure of the DRBD interface try “ip addr show | grep -B 2 10.10.10”
where 10.10.10 are the first 3 octets of the bond IP)
MONITOR_MAX_STARTS=5
MONITOR_DELAY=10
MONITOR_KILLALL=1
MONITOR_SCANRATE=5
ENABLE_LOGGING=1
MAIL_ON=1
MAIL_SUBJECT="SYSTEM ALERT - FROM HOST: $HOSTNAME"
MAIL_FROM="root@localhost"
MAIL_TO='YOUR EMAIL ADDRESS HERE'
```

Configure HA-Lizard

HA-Lizard can be completely configured from the command line. This can be done on either of the two hosts as changes are globally set for all hosts within the pool. The following settings are ideal for use with iSCSI-HA which requires fast detection of host failures and switching of roles. Fencing should be used. The configuration below uses POOL fencing which removes a failed host from the pool, but will not power-off an unresponsive host. ILO or custom fencing can be used if required. Since this design does not allow primary/primary support for DRBD, there is a low likelihood of data corruption should the pool become split. Additionally, the DRBD/iSCSI link is a directly connected Ethernet bond between the two hosts with no switches in between. iSCSI-HA logic utilizes this link to determine which host should act as the iSCSI storage, further reducing the



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

possibility of a split pool. (the below settings assume you are starting with default HA-Lizard settings.. only the following changes from default are required).

```
ha-cfg set FENCE_ENABLED 1
ha-cfg set FENCE_HEURISTICS_IPS 192.168.1.253
ha-cfg set FENCE_MIN_HOSTS 2
ha-cfg set FENCE_QUORUM_REQUIRED 1
ha-cfg set FENCE_USE_IP_HEURISTICS 1
ha-cfg set MAIL_TO <your alert email address>
ha-cfg set MAIL_FROM <your from email address>
ha-cfg set MONITOR_DELAY 15
ha-cfg set MONITOR_MAX_STARTS 20
ha-cfg set XAPI_COUNT 2
ha-cfg set XAPI_DELAY 10
```

The final HA-Lizard configuration should look like the example below. Use “ha-cfg get” to view the configuration.

```
DISABLED_VAPPS=()
ENABLE_LOGGING=1
FENCE_ACTION=stop
FENCE_ENABLED=1
FENCE_FILE_LOC=/etc/ha-lizard/fence
FENCE_HA_ONFAIL=1
FENCE_HEURISTICS_IPS=192.168.1.253
FENCE_HOST_FORGET=0
FENCE_IPADDRESS=
FENCE_METHOD=POOL
FENCE_MIN_HOSTS=2
FENCE_PASSWD=
FENCE_QUORUM_REQUIRED=1
FENCE_REBOOT_LONE_HOST=0
FENCE_USE_IP_HEURISTICS=1
GLOBAL_VM_HA=1
MAIL_FROM="root@localhost"
MAIL_ON=1
MAIL_SUBJECT="SYSTEM_ALERT-FROM_HOST:$HOSTNAME"
MAIL_TO=yourmail@somedomain.com
MONITOR_DELAY=15
MONITOR_KILLALL=1
MONITOR_MAX_STARTS=20
MONITOR_SCANRATE=10
OP_MODE=2
PROMOTE_SLAVE=1
SLAVE_HA=1
SLAVE_VM_STAT=0
XAPI_COUNT=2
XAPI_DELAY=10
XC_FIELD_NAME='ha-lizard-enabled'
XE_TIMEOUT=10
```

- **Step 8 – Start Services**

- HA-Lizard should already be running – check with “service ha-lizard-status -w”. Start the service if it is not running

```
service ha-lizard start -w
```

- Enable HA from either of the hosts in the pool

```
ha-cfg status
```

 (type “yes” when prompted)

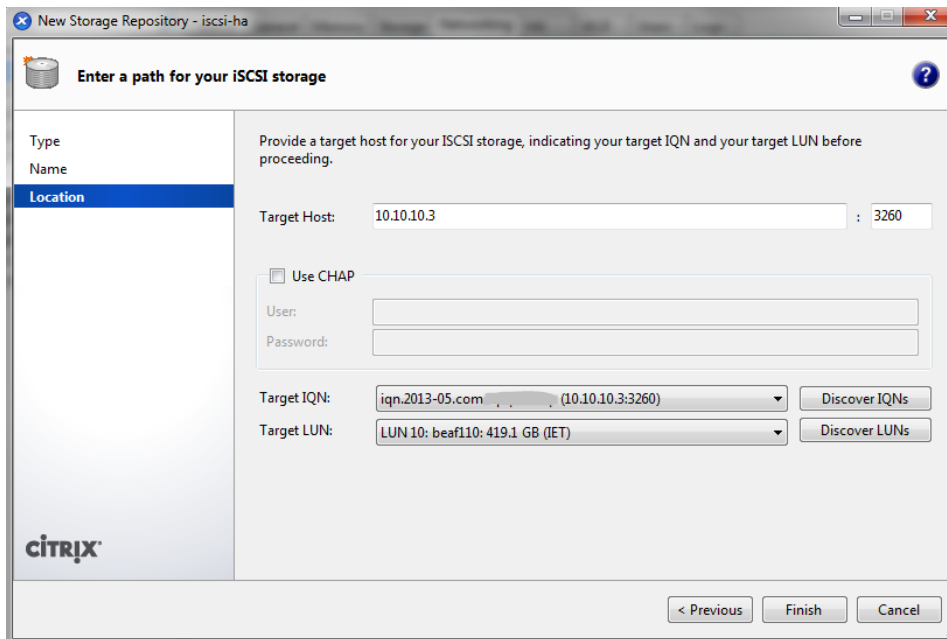
- Start iSCSI-HA

```
chkconfig iscsi-ha on
```

```
service iscsi-ha start -w
```

- **Step 9 – Create a new SR**

From within XenCenter, create a new storage repository of type – iSCSI with a target of the shared/floating IP of 10.10.10.3





3. Managing the 2-Node Highly Available Cluster

With the installation steps completed, the cluster can now be used to create new VMs. All disk writes will be duplicated across the bonded Ethernet link from the master to the slave, thus each logical volume created by XenServer will be duplicated across the hosts. Read/Write access to the storage will only be on a single IP address which is managed by iSCSI-HA, regardless of which host is the current pool master. The overall design of the compact 2-node cluster is intended to be more resilient than traditional architectures with dual RAID levels and fewer network elements.

- Each host employs HW RAID (if configured to do so)
- Storage is duplicated with SW RAID (DRBD)
- Recovery and failover is simplified in a two node architecture
- Low likelihood of a split pool with mechanisms to prevent the possibility
- Modern server architecture can be very dense allowing for a robust, high capacity virtualization environment with minimal HW components.

- **Performance and System Capacity**

Resources and Capacity

Ideally, the completed cluster should be engineered to handle the full load of all VMs on a single host in the event of a failure. This requires that adequate CPU resources be made available to run the entire VM load on a single host, and, more importantly that enough RAM is available on each of the hosts to run all of the VMs. If the cluster is unable to provide enough resources to run the full VM load on a single host then it will be necessary to omit some of the VMs from HA. There are several ways of achieving this which are well documented in the HA-Lizard administration manual. Below is a short summary of considerations:

- Select the least critical VMs to be omitted from HA and assign to a specific host.
- Use HA-Lizard OP-MODE 1 and assign non-HA VMs to an excluded appliance.
- Use HA-Lizard OP-MODE 2 and set GLOBAL_VM_HA=0. The use the CLI tool to set/unset HA for each VM within the pool.
- If the pool has a considerable number of VMs, consider setting HA-Lizard OP-MODE to 1 which manages appliances rather than directly managing VMs. This give the added benefit of configuring a delay between VM starts.



Performance

System performance is dependent on a variety of factors, including:

- Server I/O subsystem performance.
- Disk Speeds
- Replication link speed

VM performance on either the Master or Slave host should be the same with the exception of storage read/write speeds. The performance on the Master should be roughly the same as Direct Attached Storage (DAS). Storage read/write performance from the Slave can be roughly ½ of the Master's performance. This is due to VMs on a Slave host communicating with storage on the Master while the Master simultaneously writes any changes to the Slave's replicated storage. In this scenario, the replication link is roughly split in half between the VM accessing the storage and the storage replication.

The following write performance was measured on a system built with the default settings presented in this document:

Write Speed – VM on Master: 138 MB/s

Write Speed – VM on Slave: 62.6 MB/s

Write Speed – VM on XenServer using local 10K disk (for comparison): 119 MB/s

In summation, with proper capacity and load planning, the entire load can be run on the Master host while achieving DAS performance.

• System Behavior

The completed pool/cluster provides a high degree of automated HA protection for both the hosts and all VMs within the pool. Given that only 2 hosts are used, the following failure/recovery scenarios are possible:

- 1) Master Failure: A master host failure will be detected in 15 seconds or less and recovered (slave promotes itself to master) in less than one minute. Any VMs that were running on the slave will continue to operate. They will experience a short disruption in backing disk connectivity (~ 60 seconds) and will otherwise be unaffected. Any VMs that were running on the failed master will be restarted on the new master (former slave) about 60 seconds after the failure.
- 2) Slave Failure: A slave failure will be detected in 15 seconds or less and will be removed from the pool in less than one minute. Any VMs that were running on the Master will be completely unaffected. VMs that were running on the failed slave will be restarted on the master roughly ~60 seconds after the failure.



The timing of the recovery logic can be customized to be more aggressive, offering significantly faster recovery or more conservative which would require more time to recover services. The settings used in this how-to provide full recovery in about one minute.

- **iscsi-cfg CLI Tool**

A command line tool is provided as part of the iscsi-ha package. The tool can be called with:

iscsi-cfg

A sample output with command line monitoring arguments is shown below:

```
iscsi-HA Monitoring Tool: Add-on for HA-Lizard: XenServer/XCP High Availability
Usage: iscsi-cfg <action>

Available actions:
<log>:          Watch iSCSI-HA log file output in real time
<get>:          Lists all iSCSI-HA configuration parameters
<manual-mode-enable>: Enter manual mode - required to manually select roles
                  Allows for manually moving iSCSI target to desired host
                  Used to manage rolling updates and server reboots
                  with no VM downtime.
<manual-mode-disable>: Exit manual mode - automatic selection of roles enabled
                  Operation returns to normal - iSCSI-HA manages roles
<become-primary>:  Manually promotes host to primary role regardless of
                  role of the host in the pool master/slave. Only works
                  when operating in manual mode.
<become-secondary>: Manually demotes host to secondary role regardless of
                  role of the host in the pool master/slave. Only works
                  when operating in manual mode.
<status>:         Displays the iSCSI-HA operational status
```

System Logging

A live view of the system logs generated by iSCSI-HA is available by invoking:

iscsi-ha log

Viewing Configuration Parameters

A listing of configuration parameters for the local host is available by invoking:

iscsi-cfg get

Viewing iSCSI-HA Status

The iSCSI-HA service is responsible for managing:

- DRBD Running State
- DRBD Resource State (primary/secondary)



High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

- iSCSI Target (TGT) Running State
- Floating IP Address

The status of each of these can be viewed in real time by invoking:

iscsi-cfg status

The information displayed will be relative to the role of the host within the XenServer pool. Below examples show the output from both the pool Master and Slave nodes.

```
-----
| iSCSI-HA Status |
| Tue May 26 17:34:07 EDT 2015 |
-----

| iSCSI-HA Status: Version: IHA 1.5.4 S99iscsi-ha (pid 4020 4005) is running... |
| Last Updated: Tue May 26 17:33:59 EDT 2015 |
| HOST ROLE: MASTER |
| DRBD ROLE: iscsi1=Primary |
| DRBD CONNECTION: iscsi1 in Connected state |
| iSCSI TARGET: tgttd (pid 4503 4502) is running... [expected running] |
| VIRTUAL IP: 10.10.10.3 is local |
-----

Control + C to exit

-----
| DRBD Status |
-----

| version: 8.4.3 (api:1/proto:86-101) |
| srcversion: 19422058F8A2D4AC0C8EF09 |
| 1: cs:Connected ro:Primary/Secondary ds:UpToDate/UpToDate C r----- |
| ns:67751516 nr:0 dw:62736852 dr:7363328 al:594 bm:419 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:0 |
-----
```

```
-----
| iSCSI-HA Status |
| Tue May 26 17:33:07 EDT 2015 |
-----

| iSCSI-HA Status: Version: IHA 1.5.4 iscsi-ha (pid 15325 15319) is running... |
| Last Updated: Tue May 26 17:33:01 EDT 2015 |
| HOST ROLE: SLAVE |
| VIRTUAL IP: 10.10.10.3 is not local |
| iSCSI TARGET: tgttd is stopped [expected stopped] |
| DRBD ROLE: iscsi1=Secondary |
| DRBD CONNECTION: iscsi1 in Connected state |
-----

Control + C to exit

-----
| DRBD Status |
-----

| version: 8.4.3 (api:1/proto:86-101) |
| srcversion: 19422058F8A2D4AC0C8EF09 |
| 1: cs:Connected ro:Secondary/Primary ds:UpToDate/UpToDate C r----- |
| ns:0 nr:152780 dw:152780 dr:0 al:0 bm:36 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:0 |
-----
```




High Availability 2-Node XenServer Pool Reference Design For use with XenServer 6.x

The status will not be displayed if the iSCSI-HA service is stopped or a system failure is preventing it from running correctly. In this case, the status output will look like the below example.

```
Sun Jul 14 20:59:30 EDT 2013
#####
## iSCSI-ha Status cache is more than 33 seconds old
## Tolerance = 15 seconds. Check status
#####

Possible causes: 1) iSCSI-HA is actively failing over to standby host
                   this can take up to 60 seconds or more depending on settings

                   2) Service is not running. try 'service iscsi-ha status -w' to check running status
                   If the service is not running, try 'service iscsi-ha start -w' to start the service

Control + C to exit
```



- **Operating in Manual Mode**

Under normal operating conditions, iSCSI-HA operates in a completely automatic and dynamic fashion such that the system can automatically recover from most failures with no user intervention. There are cases, however, where automation must be disabled so that hosts can be safely rebooted or upgraded with zero VM downtime. This is important since under normal operating conditions the storage will *always* be exposed on the pool master. Manual mode allows administrators to disable the iSCSI-HA logic so that storage can be moved between hosts as needed.

For example: Assuming a pool Master must be rebooted as part of an upgrade, an administrator would enable manual mode on both hosts and then safely move the VMs and the storage to the pool slave with no downtime. ***Important Note: High Availability must also be disabled in this case before rebooting a host.***

Enabling Manual Mode

The following command is used to enable manual mode on a host:

```
iscsi-cfg manual-mode-enable
```

Important Note: This should be applied to both hosts in the pool to avoid any possible contention

Disabling Manual Mode

The following command is used to disable manual mode and return the pool to automatic selection of roles:

```
iscsi-cfg manual-mode-disable
```

Important Note: Remember to re-enable HA if it was disabled

Becoming the Storage Primary Role

Storage can be manually promoted to master mode while iSCSI-HA is in manual mode. This can only be done on a single host (iSCSI-HA does not support dual primary mode). A warning will be displayed if an attempt to promote a host is made while the peer host is already in the primary storage role. The following command is used to promote a host to primary mode:

```
iscsi-cfg become-primary
```

CAUTION: REBOOTING A HOST THAT IS IN THE PRIMARY ROLE WHILE IN MANUAL MODE WILL CAUSE VMs TO LOSE CONNECTIVITY TO THE STORAGE REPOSITORY. USE CAUTION AND ONLY RESTART A HOST THAT IS IN SECONDARY MODE MAKING SURE THAT THE PEER IS PRIMARY

Becoming the Storage Secondary Role

Storage can be manually demoted to the secondary role while iSCSI-HA is in manual mode. This can be done on any host in the pool with no restrictions. When swapping primary/secondary roles, it is first necessary to put both hosts into the secondary role and then promote the desired host to the primary role. The following



command is used to demote a host to the secondary role:

```
iscsi-cfg become-secondary
```

- **Manual Mode Best Practices**

Manual mode is intended to provide a means for managing a 2-node virtualization/storage cluster manually. This is especially important for environments and applications which require near 100% uptime. When utilizing manual mode, the storage can be exposed on either host as needed. Data replication continues to operate as usual when moving storage between hosts. All iSCSI-HA operations (eg. promoting/demoting storage hosts) are completely transparent to XenServer and any underlying VMs.

iSCSI-HA employs several conditional checks while operating in manual mode. These checks ensure that a user cannot destabilize a system or assert a role which is disallowed for a particular host.

Existing Manual Mode Cleanly

Manual mode cannot be disabled unless both hosts are first restored to their native storage roles. This means that the XenServer pool master must be promoted to the storage master (`iscsi-cfg become-master`) and the pool slave must be demoted to the storage secondary role (`iscsi-cfg become-secondary`). With both hosts in their native roles manual mode should be disabled on both hosts with `iscsi-cfg manual-mode-disable`. This behavior is intentional and ensures that there is no storage downtime when returning the pool to normal (automated) operation.

Entering Manual Mode

Manual mode is generally used to perform maintenance operations on hosts within a given pool. Although there are no restrictions set when entering manual mode, one should ensure that any pool HA functionality is disabled while operating in manual mode if hosts require a shutdown or restart.



- **Dealing with DRBD Split Brain**

Certain reboot scenarios and unexpected host restarts can cause DRBD to detect split brain which will prevent the DRBD resources from synchronizing. This scenario will likely not impact the operation of the pool since the Master node will still manage DRBD locally and ensure it is in the Primary state. Recovery from split brain should be handled cautiously as storage for one of the nodes will need to be overwritten. The following steps offer a general guideline for recovery.

- First, ensure that the XenServer pool is not split. Although highly unlikely with the 2-node architecture, it is possible that both hosts have entered into the Master role. If this has occurred and VMs on both hosts are not running in duplicate, then it will be necessary to manually merge the known good Logical Volumes from both hosts onto the new Master. Once done, the Master nodes storage can be re-synchronized with the Slave. All HA processes should be stopped during this operation.
- If it is clear which node holds the current data, the following steps should clear up the DRBD split brain.
 - o On the node that is to lose its data by synchronizing with the good node (where iscsi1 is the DRBD resource name):

```
drbdadm secondary iscsi1
```

```
drbdadm -- --discard-my-data connect iscsi1
```
 - o On the host that is the survivor with known good data:

```
drbdadm connect iscsi1
```



Miscellaneous

- **Managing Services**

DRBD

It is best to ensure that the DRBD service is **NOT**-automatically started on each host when the system boots as this can prevent a host from fully booting in the event that both hosts are rebooted and only a single host returns to operating status. This is due to DRBD waiting for its peer to connect indefinitely which can prevent services from starting during system boot.

```
chkconfig drbd off
```

iSCSI-HA will automatically start DRBD after all system services have safely started thus eliminating the possibility of a hang during boot time. The iSCSI-HA process will always acts as a watchdog for DRBD ensuring that the service is always on.

TGTD

The TGTD iSCSI target should be managed by iSCSI-HA as the service should only be running on one of the hosts. For proper operation it is necessary to instruct the host not to automatically start TGTD. This can be done with:

```
chkconfig tgt off
```

- **Dependencies and Compatibility**

When installing iSCSI-HA onto a default Centos based DomO (XCP or XenServer), all the required tools needed to run iSCSI-HA are resident on the system with the exception of:

- DRBD – version 8.3 required. RPMs are provided in /etc/iscsi-ha/RPM/
- TGT iSCSI Target (can be installed with “yum –enablerepo=base install scsi-target-utils”)

Package is compatible with XCP version 1.6 XenServer version 6.1 and XenServer version 6.2. Prior releases may work but have not been tested.

For custom DomO installations, ensure the following tools are available:

```
xapi and xe toolstack  
/bin/cat  
/bin/awk  
/bin/echo  
/sbin/drbdadm
```



/bin/logger
/sbin/ifconfig
hostname
/bin/mail
/sbin/ip
/sbin/arping

- **Important Considerations**

- iSCSI-HA requires that a node within the 2-node pool **Always** is the pool master. If a pool failure results in a situation with no master, the iSCSI target will be unavailable and VMs cannot operate. To ensure that a master is always available HA logic should be employed in the pool.
- In the event that there is no pool master, manual intervention is required to expose the iSCSI target.
- iSCSI-HA only supports 2-node pools. It can be adapted to larger pools with some work.
- AVOID Upgrading XenServer after completing this how-to – installed packages and configurations will likely be lost as part of the upgrade process (eg. Upgrading from version 6.1 to 6.2).

- **Security and Ports**

- iscsi port 3260 used as the listen port for the iscsi target
- ICMP (ping) is used to check whether the virtual IP is live
- DRBD – ensure that the port numbers specified in drbd.conf are open

- **Support**

- Post a question on the support forum
<http://www.halizard.com/index.php/forum>
- Contact the project sponsor for paid support options
<http://www.pulsesupply.com>



4. Appendix A – Installing DRBD from Source

- **XCP 1.6 and XenServer 6.1**

- Install the required packages for building DRBD from source
`yum --enablerepo=base install gcc flex rpm-build redhat-rpm-config make libxslt -y`
- Get the kernel-xen-devel rpm from XCP (or XenServer) binpkg.iso image and install
`rpm -ivh kernel-xen-devel-2.6.32.43-0.4.1.xs1.6.10.734.170748.i686.rpm`
- Create RPMs
`mkdir /drbd/
cd /drbd/
wget http://oss.linbit.com/drbd/8.3/drbd-8.3.15.tar.gz
tar zxvf drbd-8.3.15-1.tar.gz
cd drbd-8.3.15-1
./configure --prefix=/usr --localstatedir=/var --sysconfdir=/etc --with-km
make tgz drbd.spec drbd-km.spec
cp drbd*.tar.gz `rpm -E %_sourcedir`
rpmbuild -bb drbd.spec
rpmbuild -bb drbd-km.spec
cd /usr/src/redhat/RPMS/i386/`
- Install the required RPMs
`rpm -ivh drbd-utils-8.3.15-1.i386.rpm
rpm -ivh drbd-km-2.6.32.43*.rpm
rpm -ivh drbd-bash-completion-8.3.15-1.i386.rpm
rpm -ivh drbd-heartbeat-8.3.15-1.i386.rpm
rpm -ivh drbd-pacemaker-8.3.15-1.i386.rpm
rpm -ivh drbd-udev-8.3.15-1.i386.rpm
rpm -ivh drbd-xen-8.3.15-1.i386.rpm
rpm -ivh drbd-8.3.15-1.i386.rpm`



- **XenServer 6.2**

- Mount the binpkg.iso image and move into the `cd /mnt/domain0/RPMS/i386/` folder. Install the following RPMs.

```
rpm -ivh kernel-xen-devel-2.6.32.43-0.4.1.xs1.8.0.835.170778.i686.rpm
rpm -ivh kernel-headers-2.6.32.43-0.4.1.xs1.8.0.835.170778.i686.rpm
rpm -ivh glibc-headers-2.5-107.el5_9.1.i386.rpm
```

- Install the required packages for building DRBD from source
`yum --enablerepo=base install gcc flex rpm-build redhat-rpm-config make libxslt -y`

- Create RPMs

```
mkdir /drbd/
cd /drbd/
wget http://oss.linbit.com/drbd/8.3/drbd-8.3.15.tar.gz
tar zxvf drbd-8.3.15.tar.gz
cd drbd-8.3.15
./configure --prefix=/usr --localstatedir=/var --sysconfdir=/etc --with-km
make tgz drbd.spec drbd-km.spec
cp drbd*.tar.gz `rpm -E %_sourcedir`
rpmbuild -bb drbd.spec
rpmbuild -bb drbd-km.spec
cd /usr/src/redhat/RPMS/i386/
```

- Install the required RPMs

```
rpm -ivh drbd-utils-8.3.15-1.i386.rpm
rpm -ivh drbd-km-2.6.32.43*.rpm
rpm -ivh drbd-bash-completion-8.3.15-1.i386.rpm
rpm -ivh drbd-heartbeat-8.3.15-1.i386.rpm
rpm -ivh drbd-pacemaker-8.3.15-1.i386.rpm
rpm -ivh drbd-udev-8.3.15-1.i386.rpm
rpm -ivh drbd-xen-8.3.15-1.i386.rpm
rpm -ivh drbd-8.3.15-1.i386.rpm
```




5. Appendix B - Example Maintenance Operations

The below example illustrates the steps necessary to perform an update to both the pool master and slave hosts which requires that they be rebooted. The following procedure ensures that there is no downtime on any VMs. For this example it is assumed that the pool master will be rebooted first and then the pool slave.

- 1) Disable HA for the pool (for HA-Lizard 'ha-cfg' can be used to disable HA)
- 2) Enter manual mode on each host **iscsi-cfg manual-mode-enable**
- 3) Migrate all VMs to the pool slave
- 4) Demote the pool master's storage role **iscsi-cfg become-secondary**
- 5) Promote the pool slave's storage role **iscsi-cfg become-primary**
Important Note: there should be minimal delay between the demote/promote actions to ensure that the VMs experience little storage downtime. Typical delay when manually switching roles is 3 seconds.
- 6) The master host can now be safely worked on (enter maintenance mode, shutdown, rebooted, etc...)
- 7) Rejoin the master host to the pool
- 8) Migrate all VMs from the slave to the pool master
- 9) Demote the pool slave's storage role **iscsi-cfg become-secondary**
- 10) Promote the pool master's storage role **iscsi-cfg become-primary**
Important Note: there should be minimal delay between the demote/promote actions to ensure that the VMs experience little storage downtime. Typical delay when manually switching roles is 3 seconds.
- 11) The slave host can now be safely worked on (enter maintenance mode, shutdown, rebooted, etc...)
- 12) Rejoin the slave host to the pool
- 13) Exit manual mode on both hosts **iscsi-cfg manual-mode-disable**
- 14) Enable HA for the pool (for HA-Lizard 'ha-cfg' can be used to enable HA)