

## Contents

- 1 Initial Setup
- 2 Reboot into new Install
- 3 Install Xen
  - ◆ 3.1 Check and Update Debian Package Sources
  - ◆ 3.2 Main Install of Xen
    - ◇ 3.2.1 Debian Etch AMD64
    - ◇ 3.2.2 Debian 32bit Etch
    - ◇ 3.2.3 dom0 mem Boot Setting
      - 3.2.3.1 dom0 mem Automatic setting
      - 3.2.3.2 Why dom0 mem
    - ◇ 3.2.4 Xen Network Settings
    - ◇ 3.2.5 Reboot and Test
- 4 Setup and Config of Xen-Tools
- 5 Automatically create a domU using Xen-Tools
  - ◆ 5.1 Automatically create a domU on a Private Subnet
- 6 Migrate a domU from one Server to another
  - ◆ 6.1 Migrate from LVM based to LVM based Partition
    - ◇ 6.1.1 On the New Xen Server (dom0)
    - ◇ 6.1.2 On the Old Xen Server (dom0)
  - ◆ 6.2 Migrate from File Based to LVM based Partition
    - ◇ 6.2.1 On the New Xen Server (dom0)
    - ◇ 6.2.2 On the Old Xen Server (dom0)
  - ◆ 6.3 Migrate a Live LVM System to a LVM based Partition
    - ◇ 6.3.1 On the New Xen Server (dom0)
    - ◇ 6.3.2 On the Old Xen Server (dom0)
- 7 Remove VMs from dom0
  - ◆ 7.1 Automatically with Xen-Tools

## Initial Setup

Reboot into Hetzner's Rescue System. I was going to use Debian's AMD64 (which is not related to AMD processors!) Anyways - 32bit Etch. Configure for raid1, /boot for ext2 and / for dom0-root.

```
PART 1 /boot ext2 256M
PART 2 lvm vg0 all
LV vg0 root / ext3 30G
```

Note: grub is required for Xen. Lilo will not work for Xen.

## Reboot into new Install

We need to add in a swap space using a neat lv (as opposed to a physical partition via the hetzner installer)

```
lvcreate -n dom0-swap --size 256M vg0
mkswap /dev/vg0/dom0-swap
vi /etc/fstab
    #Add in the following:
    /dev/vg0/dom0-swap none swap sw 0 0

swapon /dev/vg0/dom0-swap
```

## Install Xen

### Check and Update Debian Package Sources

Check apt's source.list:

```
vi /etc/apt/sources.list
deb http://ftp.de.debian.org/debian/          stable main
deb-src http://ftp.de.debian.org/debian/        stable main

deb http://security.debian.org/ stable/updates main
#The above is a minimal config for a server located in Germany.
```

## Main Install of Xen

### Debian Etch AMD64

Note: AMD64 has nothing to do with AMD processors!! Its the name for the 64bit OS.

```
apt-get install linux-image-2.6.18-6-xen-amd64 xen-hypervisor-3.0.3-1-amd64 xen-tools bridge-utils s
#note libc6-xen is not required for AMD64. Note: Etch AMD64 is not just for AMD Processors! It is th
```

### Debian 32bit Etch

```
apt-get install linux-image-2.6-xen-686 xen-hypervisor-3.0.3-1-i386-pae xen-tools bridge-utils libc6
```

Check the grub boot:

```
vi /boot/grub/menu.lst
#the xen kernel should have been inserted first.
```

### dom0\_mem Boot Setting

This step is optional. If you only want to assign dom0 only 128MB of memory as opposed to allowing it to balloon out and use all memory (until a domU is created which would take the memory from dom0), you can do the following:

```
vi /boot/grub/menu.lst
#check the xen grub entry and add in the dom0_mem on the end of the kernel line:
title Xen 3.0.3-1-i386-pae / Debian GNU/Linux, kernel 2.6.18-6-xen-686
root (hd0,0)
kernel /xen-3.0.3-1-i386-pae.gz dom0_mem=128000
module /vmlinuz-2.6.18-6-xen-686 root=/dev/mapper/vg0-dom0--root ro console=tty0
module /initrd.img-2.6.18-6-xen-686
savedefault
```

### dom0\_mem Automatic setting

The previous step works ok. However after an apt-get upgrade for a new kernel etc. it will disappear. To have the dom0\_mem always added to the grub config:

```
vi /boot/grub/menu.lst
#look for the following line:
# xenhopt=
change it to:
# xenhopt=dom0_mem=128000
(Leave the # there.)

update-grub
#update-grub scans for installed kernels and creates a new menu.lst
```

Thanks to: <http://cmrg.fifthhorseman.net/wiki/xen>

### Why dom0\_mem

dom0\_mem=128000 is needed to dedicate memory to dom0. Otherwise the following errors may be seen in /var/log/kern.log. This is with Debian Etch. It has also been seen that this squeeze error can appear after a few weeks of dom0 uptime. I highly recommend using this kernel option.

```
Nov 25 17:21:39 flexo kernel: printk: 4 messages suppressed.
Nov 25 17:21:39 flexo kernel: xen_net: Memory squeeze in netback driver.
Nov 25 17:21:44 flexo kernel: printk: 4 messages suppressed.
Nov 25 17:21:44 flexo kernel: xen_net: Memory squeeze in netback driver.
```

### Xen Network Settings

The network setup will be outlined and described later. Until then this section may be confusing to some.

```
vi /etc/xen/xend-config.sxp
#comment out the network-dummy etc. and have the following:
```

## Install\_and\_Config\_Base\_OS

```
(network-script myscript)
(vif-script      vif-route)

vi /etc/xen/scripts/myscript
#!/bin/sh
dir=$(dirname "$0")
"$dir/network-nat2" start
"$dir/network-route" start
chmod 755 /etc/xen/scripts/myscript

cp /etc/xen/scripts/network-nat /etc/xen/scripts/network-nat2
vi /etc/xen/scripts/network-nat2
#op_start() {
#    echo 1 >/proc/sys/net/ipv4/ip_forward
#    iptables -t nat -A POSTROUTING -o ${netdev} -j MASQUERADE
#    [ "$dhcp" != 'no' ] && dhcp_start
#}
op_start() {
    echo 1 >/proc/sys/net/ipv4/ip_forward
    iptables -t nat -A POSTROUTING -o ${netdev} -s 10.0.0.0/24 -j MASQUERADE
    iptables -t nat -A POSTROUTING -o ${netdev} -s 192.168.1.0/24 -j MASQUERADE
    [ "$dhcp" != 'no' ] && dhcp_start
}
#I realise that I technically should do the same above for the op_stop function.

chmod 755 /etc/xen/scripts/network-nat2

vi /etc/network/interfaces
#add the following for the public ip address range
auto dummy0
    iface dummy0 inet static
        address 78.1.1.20
        netmask 255.255.255.248
        broadcast 78.1.1.27
#This gives the first routable Public IP to dom0 so it can act as a gateway for the domU's.
#Note the above is the quick and simple way to get networking on Xen. There is a pointpoint method
```

## Reboot and Test

```
uname -a
Linux root2 2.6.18-6-xen-amd64 #1 SMP Fri Jun 6 06:38:05 UTC 2008 x86_64 GNU/Linux

xm list
xm top
free
ifconfig
iptables -t nat -L
ping 78.1.1.20
```

## Setup and Config of Xen-Tools

```
vi /etc/xen-tools/xen-tools.conf
#below are the lines only which are uncommented:
lvm = vg0
debootstrap = 1
size = 4Gb      # Disk image size.
memory = 128Mb  # Memory size
```

## Install\_and\_Config\_Base\_OS

```
swap = 128Mb # Swap size
fs = ext3 # use the EXT3 filesystem for the disk image.
dist = etch # Default distribution to install.
image = sparse # Specify sparse vs. full disk images.
#The network will be done manually later via xm console
kernel = /boot/vmlinuz-2.6.18-6-xen-amd64
initrd = /boot/initrd.img-2.6.18-6-xen-amd64
mirror = http://ftp.de.debian.org/debian/
```

## Automatically create a domU using Xen-Tools

```
xen-create-image --hostname vm01-burkesys --ip 78.1.1.21
lvscan # to see if a disk and swap lvm were created
vi /etc/xen/vm01-burkesys.cfg #Make sure all is in order
xm create vm01-burkesys.cfg -c
#log in as root with no password
passwd #change the root password
vi /etc/network/interfaces
gateway 78.1.1.20
netmask 255.255.255.0
ifup eth0
ping www.debian.org
apt-get update
apt-get upgrade
```

## Automatically create a domU on a Private Subnet

All that changes, is that the --ip is given a 10.0.0.0 range. The config for this is done previously in the Xen Network Settings.

```
xen-create-image --hostname vm02-priv-burke --ip 10.0.0.1
xm create vm02-priv-burke -c
#login as root with no password
passwd
vi /etc/network/interfaces
gateway 10.0.0.1
netmask 255.255.255.0
ifup eth0
ping www.debian.org
#Enjoy NAT'ed traffic out.
```

## Migrate a domU from one Server to another

### Migrate from LVM based to LVM based Partition

#### On the New Xen Server (dom0)

```
lvcreate -n vm03-twister-disk --size 3g vg0
lvcreate -n vm03-twister-swap --size 256m vg0
```

Migrate a domU from one Server to another

## Install\_and\_Config\_Base\_OS

```
mkfs.ext3 /dev/vg0/vm03-twister-disk
mkswap /dev/vg0/vm03-twister-swap
nc -l -p 7000 | gzip -dfc | dd of=/dev/vg0/vm03-twister-disk
#if you want it faster and dont mind about bandwidth:
#nc -l -p 7000 | dd of=/dev/vg0/vm03-twister-disk
```

### On the Old Xen Server (dom0)

```
dd if=/dev/main-vol2/twister | gzip -cf | nc newserver.ip.address 7000 -q 10
#if you want it faster and dont mind about bandwidth:
#dd if=/dev/main-vol2/twister | nc newserver.ip.address 7000 -q 10
scp /etc/xen/twister root@newserver:/etc/xen/
```

Thats it. Simply do a xm create on the new Xen server for the VM. Note: if the new Xen Server has a different kernel, then there may be issues. If the kernel of the VM is different on the new Server, then the /lib/modules/2.6.kernel will have to be copied onto the VM. Note: Make sure that the twister VM xen config is correct.

## Migrate from File Based to LVM based Partition

In this particular case, a 2GB File based image is been copied (using dd) to a 3GB LVM Partition on a New Server. If it was only a 2GB LVM Partition, there would be nothing extra required. If the LVM partition was too small, the "dd" command would fail once the LVM filled up.

### On the New Xen Server (dom0)

```
lvcreate -n vm04-volcano-disk --size 3g vg0
lvcreate -n vm04-volcano-swap --size 256m vg0
mkfs.ext3 /dev/vg0/vm04-volcano-disk
mkswap /dev/vg0/vm04-volcano-swap
nc -l -p 7000 | gzip -dfc | dd of=/dev/vg0/vm04-volcano-disk
```

### On the Old Xen Server (dom0)

```
dd if=/xens/volcano/centos.5-0.img | gzip -cf | nc newserver.ip.address 7000 -q 10
#Extra steps required because the 3GB LVM is larger than the 2GB file:
e2fsck -f /dev/vg0/vm04-volcano-disk
resize2fs /dev/vg0/vm04-volcano-disk
```

## Migrate a Live LVM System to a LVM based Partition

With LVM, it is possible to take snapshots of a LV (Logical Volume) which is Live, i.e. currently in use. This is extremely useful for taking backups of a live system without having any downtime. It can also be used for a barebones recovery as we are going to copy/dd all of the LV.

## On the New Xen Server (dom0)

```
lvcreate -n vm05-zachome-disk --size 90g vg0
lvcreate -n vm05-zachome-swap --size 1g vg0
mkfs.ext3 /dev/vg0/vm05-zachome-disk
mkswap /dev/vg0/vm05-zachome-swap
nc -l -p 7000 | gzip -dfc | dd of=/dev/vg0/vm05-zachome-disk
```

## On the Old Xen Server (dom0)

```
lvcreate -L 1G -s -n zachomebackup /dev/main-vol2/zachome
lvscan
dd if=/dev/main-vol2/zachomebackup | gzip -cf | nc root2.burkesys.com 7000 -q 10
```

Note: I hope to write a full article on LVM, how it can be used for creating snapshots etc. etc.

# Remove VMs from dom0

## Automatically with Xen-Tools

Typically when `xen-create-image` is used to create a Xen VM, it creates the LVM partitions etc. and also a key config file:

```
/etc/xen/nameofvm.cfg
```

Its by the file extension `.cfg` that it will use to produce a list when `xen-list-images` is called.

```
#xen-list-images
#^ by default lists vms which were created with xen-tools
xen-delete-image nameofvm
```

```
#if however you created the lvm and xen config manually, you can do the following to add a .cfg exte
mv /etc/xen/nameofvm02 /etc/xen/nameofvm02.cfg
xen-list-images
xen-delete-image nameofvm02.cfg
```

So `xen-list-images` parses all `.cfg` files in `/etc/xen/`. Whats nice is that when `xen-delete-image` is called, it also parses `.cfg` files in `/etc/xen/` picks up the name of the lvm's they are using and removes them cleanly. It just saves from having to do a manual `lvmremove` for vms created manually.