LVM2 and software RAID

TD3 – Jérémy Briffaut

First part : howto create and use LVM2
Second part : howto create RAID1 arrays and then setup LVM2

1 Preparation

1.1 Choose and modify VMWare image

Choose GENTOO-OK (or GENTOO if your last image works).

Add 2 hard-drive disk (scsi) with 8GO. You must have 3 disks.

1.2 Recompil the kernel

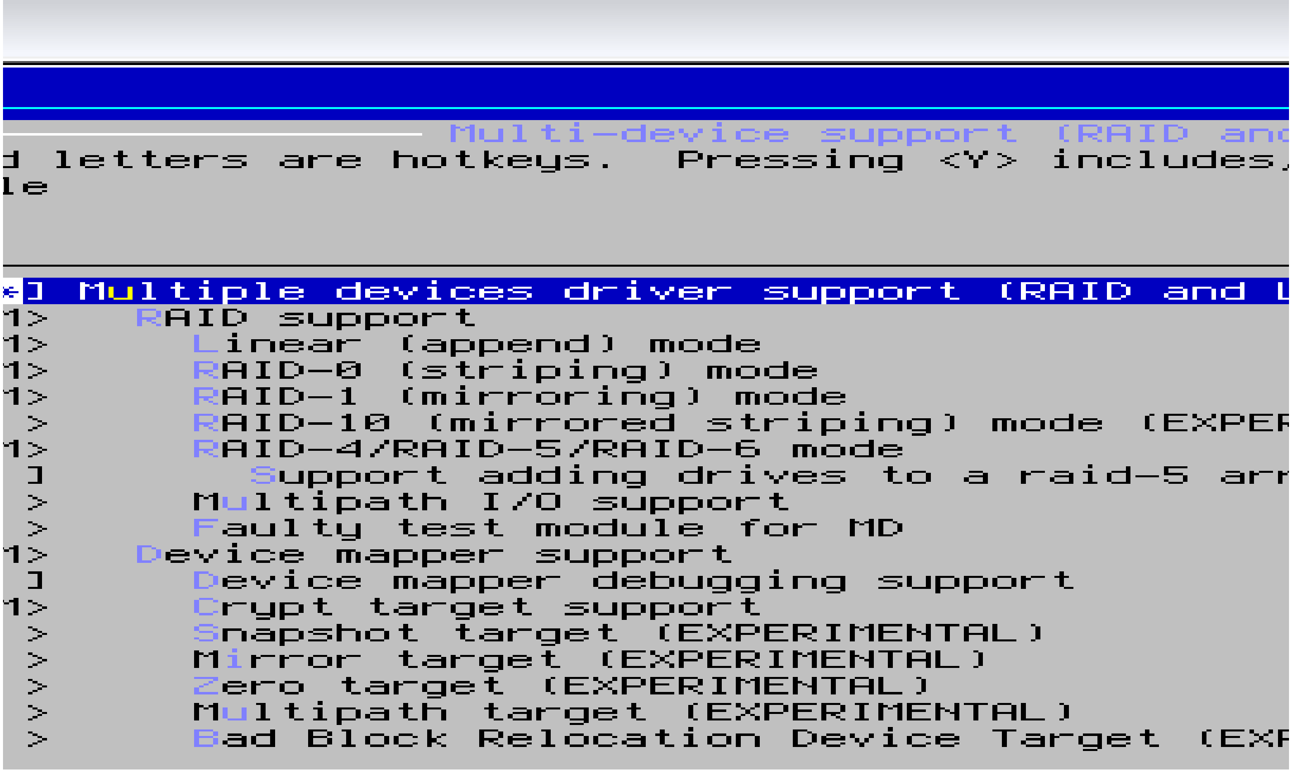
cd /usr/src/linux

make menuconfig

-> Device Driver

-> Multi-devices support

--> Multi-devices driver support (RAID + LVM)



```
                     Multi-device support (RAID and
d letters are hotkeys.  Pressing <Y> includes,
le

*] Multiple devices driver support (RAID and L
M>   RAID support
M>     Linear (append) mode
M>     RAID-0 (striping) mode
M>     RAID-1 (mirroring) mode
 >     RAID-10 (mirrored striping) mode (EXPER
M>     RAID-4/RAID-5/RAID-6 mode
 ]       Support adding drives to a raid-5 arr
 >     Multipath I/O support
 >     Faulty test module for MD
M>   Device mapper support
 ]     Device mapper debugging support
M>     Crypt target support
 >     Snapshot target (EXPERIMENTAL)
 >     Mirror target (EXPERIMENTAL)
 >     Zero target (EXPERIMENTAL)
 >     Multipath target (EXPERIMENTAL)
 >     Bad Block Relocation Device Target (EXP
```

Compil and install the system :

```
cd /usr/src/linux
mount /boot/
make bzImage modules modules_install
cp System.map /boot
cp arch/i386/boot/bzImage /boot/kernel
grub-install /dev/sda
reboot
```

1.3 Install precompiled packages

In your host :

download **packages-td3.bz2** from Enseignement > 2ASTIADMINSYS

```
scp     packages-td3.bz2     root@VMWAREHOSTIP:/usr/portage/
```

In the VMWare host :

```
cd /usr/portage/
tar jxfv packages-td3.bz2
```

2 Part 1 : LVM2

2.1 Preparation

Load the LVM2 dm-mod module.

Loading the LVM2 module

```
# modprobe dm-mod
```

Install LVM2 tools

Compil and Install lvm2

```
emerge -K lvm2
```

Create your partitions on the disks /dev/sdb and /dev/sdc with cfdisk.

Create two partitions on sdb : /dev/sdb1 (4GO) and /dev/sdb2 (4GO)

Create three partitions on sdc : /dev/sdc1 (3GO), /dev/sdc2 (3GO) and /dev/sdc3 (2GO)

2.2 Preparing the Partitions

Add the physical partitions that you want use in your LVM :

Preparing the partitions

```
# pvcreate /dev/sdb1 /dev/sdb2
  Physical volume "/dev/sdb1" successfully created
  Physical volume "/dev/sdb2" successfully created

# pvcreate /dev/sdc1 /dev/sdc2 /dev/sdc3
  Physical volume "/dev/sdc1" successfully created
  Physical volume "/dev/sdc2" successfully created
  Physical volume "/dev/sdc3" successfully created

# pvdisplay
(verify that all is ok)
```

2.3 Volume Group

Setup a volume group. A volume group is the result of combining several physical units into a single logical device.

In our example, /dev/sdb1, /dev/sdc1 and /dev/sdc3 are the boot, swap and root partitions so we need to combine them. It can be done with a single command, but, as an example, we will create our volume group and extend it.

Creating and extending a volume group

```
(Create a volume group named vg1)
# vgcreate vg1 /dev/sdb1
(Extending an existing volume group)
# vgextend vg1 /dev/sdc1
# vgextend vg1 /dev/sdc3
(diplay and verify)
# vgdiplay
```

We will create a second volume group for /var and /tmp

Creating and extending a volume group

```
(Create a volume group named vg2)
# vgcreate vg2 /dev/sdb2
(Extending an existing volume group)
# vgextend vg2 /dev/sdc2
(diplay and verify)
                                           # vgdisplay
```

2.4 Logical Volumes

Create the logical volumes. Logical volumes are the equivalent of partitions you would create using fdisk in a non LVM2 environment. In our example, we create the following partitions:

Directory	**Size**
/usr	2 GB
/home	2 GB
/opt	2 GB
/var	3 GB
/tmp	1 GB

Since we are going to use LVM2, we should not worry too much about partition sizes because they can always be expanded as needed.

NOTE : It is easier to increase the size of a partition then to shrink it. You might therefore want to start with smaller partitions and increase their size as needed.

Creating and extending logical volumes

```
# lvcreate -L2G  -nusr  vg1
# lvcreate -L2G  -nhome vg1
# lvcreate -L2G  -nopt  vg1
# lvcreate -L3G  -nvar  vg2
# lvcreate -L1G  -ntmp  vg2
(As an example, let's extend a logical volume with 1 extra Gbytes)
# lvextend -L+1G /dev/vg1/home
```

2.5 Create the Partitions

Create filesystems on the logical volumes the same way you would on a regular partition. We use ext3 on the logical volumes but any filesystem of your choice will work:

Code Listing 2.6: Creating the filesystems

```
# mke2fs -j /dev/vg1/usr
# mke2fs -j /dev/vg1/home
# mke2fs -j /dev/vg1/opt
# mke2fs -j /dev/vg2/var
# mke2fs -j /dev/vg2/tmp
```

Mount your partitions as described in the handbook and mount your LVM2 logical volumes as if they were partitions. Replace the usual /dev/hdxx with /dev/vg/logical_volumename.

Code Listing 2.7: Mounting your logical volumes

```
(Make sure you have mounted your root partition as described in the handbook
first)
# mkdir -p /mnt/gentoo/usr
# mount /dev/vg1/usr /mnt/gentoo/usr
# mkdir /mnt/gentoo/home
# mount /dev/vg1/home /mnt/gentoo/home
# mkdir /mnt/gentoo/opt
# mount /dev/vg1/opt /mnt/gentoo/opt
# mkdir /mnt/gentoo/var
# mount /dev/vg2/var /mnt/gentoo/var
# mkdir /mnt/gentoo/tmp
# mount /dev/vg2/tmp /mnt/gentoo/tmp
```

2.6 Clean the installation

In order to do the PART 2, you must delete all your LVM installation :

umount all partitions :

umount /mnt/gentoo/usr /mnt/gentoo/home /mnt/gentoo/opt /mnt/gentoo/var /mnt/gentoo/tmp

remove Volume Groups :

lvremove vg1

lvremove vg2

remove physical volume :

pvremove /dev/sdb1 -ff

pvremove /dev/sdb2 -ff

pvremove /dev/sdc1 -ff

pvremove /dev/sdc2 -ff

pvremove /dev/sdc3 -ff

3 Part 2 : RAID 1 + LVM2

We will setup 2 disks in RAID 1 (mirroring mode). We will create 2 partitions on each disk and then use LVM to manage the filesystems.

Warning: The performance will drop by about 30%! (Some people even recognized a drop of 50%!) So do not use LVM2 with a system tuned for speed! It was tested with a 10GB real-partition and the same partition as LVM2. Both with ext2 and benchmarked with a 9GB file created by dd with blocksize of 1MB. Of course, using a RAID0 can speed things up alot, too.

3.1 Preparation

- Load softraid module for raid1:

```
modprobe raid1
```

- Partition your harddisks and set partition types to Linux raid autodetect

```
cfdisk /dev/sdb
cfdisk /dev/sdb
```

- Create two partitionc of 4GO on each disk :

```
/dev/sdb1 (4GO), /dev/sdb2 (4GO)

/dev/sdc1 (4GO), /dev/sdc2 (4GO)
```

- We need to create raid device nodes before starting raid

```
mknod /dev/md1 b 9 1
mknod /dev/md2 b 9 2
```

3.2 Install RAID tools

In order to use mdadm command, install the package mdadm :

```
emerge -K mdadm
```

3.3 Creating a RAID Arrays for partition mirroring (RAID 1)

NOTE : Linux has to have at least 1 non-raid partition to boot from, this can be a RAID1 array since RAID1 makes an exact copy of the disk. With RAID1 Disk1 and Disk2 will be exactly the same and therefore the disks will be separately mountable like normal. The reason why linux can't be booted from software Raid0 arrays is simple, the kernel has the drivers that are needed to read the raid array so the boot loader will not be able to access the kernel. It's the chicken or the egg problem. A RAID0 array or any RAID array that utilizes striping cannot be mounted without the raid software since all data spans multiple drives, without the software it's unreadable. This is not an issue with RAID1 (Mirrored) arrays since the disks themselves contain all the information for every partition and can be addressed individually just like any single partitioned disk. Because of this feature and to make sure in the event of a failure we can get back up and running quickly we will create a RAID1 array for /boot to allow us to have a backup copy of our kernels and other important files.

Create two mirrors :

```
mdadm --create --verbose /dev/md1 --level=1 --raid-devices=2 /dev/sdb1
/dev/sdc1
mdadm --create --verbose /dev/md2 --level=1 --raid-devices=2 /dev/sdb2
/dev/sdc2
```

Wait for the end of arrays creation :

```
cat /proc/mdstat
```

(wait for unused devices ...)

Setup the file /etc/mdadm.conf :

File: /etc/mdadm.conf

```
# paste this inside
DEVICE          /dev/sdb*
DEVICE          /dev/sdc*
ARRAY           /dev/md1 devices=/dev/sdb1,/dev/sdc1
ARRAY           /dev/md2 devices=/dev/sdb2,/dev/sdc2
MAILADDR        root@localhost
```

3.4 Filesystems setup

- Create the first partition filesystem

  ```
  mke2fs -j /dev/md1
  ```

- Create the second partition filesystem

  ```
  mke2fs -j /dev/md2
  ```

3.5 LVM2 over RAID1 :

Question : you must setup LVM2 over the 2 RAID arrays:

1) Add the two physical partitions into the physical volume
2) Create a Volume Group **vg** with these two physical volume
3) Create one logical volume of 7GO for /usr
4) Mount it into /mnt/usr

5) Modify /etc/fstab to automount this logical volume into /mnt/usr
6) reboot
7) What is the problem? Howto to correct it?
8) Modify, Reboot and Test