

### 3.3. Espelhamento (RAID1)

Neste servidor estamos usando dois HDs idênticos, cada um com 160GB de espaço em disco. Nos passos seguintes é importante existir dois HDs idênticos em tamanho. Caso ocorra a necessidade de utilizar um HD diferente, consulte as notas sobre o assunto nos passos posteriores.

De acordo com a instalação, foram inicialmente criadas 3 partições no primeiro disco. Supondo que este primeiro disco seja /dev/sda e o segundo seja /dev/sdb, as três partições criadas são: /dev/sda1 (swap), /dev/sda2 (/boot), /dev/sda3 (raiz).

O que faremos aqui será criar um dispositivo de RAID1 no sistema com um disco faltando (apenas funcionando com o disco 2), copiaremos todo o conteúdo atual do servidor para este dispositivo, começaremos a usar o sistema com o RAID1 e integraremos o primeiro HD neste dispositivo, que ficará completo.

**Fonte:** Este procedimento foi baseado nas informações encontradas na seguinte URL (existente até a presente data em que este documento foi escrito):

- <http://togami.com/~warren/guides/remoteraidcrazies/>  
Copyright 2004 Warren Togami <wtogami@redhat.com>

#### 3.3.1. Verificações

Antes de começar, verifique os arquivos: /etc/fstab e /boot/grub/menu.lst e veja se não há nenhuma referência às partições via LABEL, ao invés do arquivo-dispositivo (p.ex.: /dev/sda3). Caso tenha algum LABEL configurado nas partições, use o comando “e2label /dev/sda2 “” “ para remover o LABEL no respectivo dispositivo (no exemplo: /dev/sda2). Se as notas de instalação tiverem sido seguidas não haverá problemas nesta verificação.

Verifique se os dois discos tem exatamente o mesmo tamanho, utilize os comandos como no exemplo a seguir:

```
# cat /sys/block/sda/size
312581808
# cat /sys/block/sdb/size
312581808
```

Verifique também se os pacotes dmsetup e mdadm estão instalados. Para verificar, utilize o comando “dpkg -l <pacote>”. Caso os pacotes não tenham sido instalados, utilize o comando apt-get para instalá-los:

```
apt-get install dmsetup mdadm
```

Lembrando que você precisará estar conectado na Internet para fazê-lo, ou ter os pacotes em CD.

#### 3.3.2. Particionamento

Uma vez verificado os tamanhos exatos, mude os tipos das partições do primeiro disco para “Linux raid autodetect”. Este tipo será utilizado para montarmos o dispositivo RAID1 posteriormente. Para mudar o tipo, utilize a ferramenta fdisk:

```
# fdisk /dev/sda
```

Depois mude o tipo das três partições para o tipo **fd**. O comando dentro do `fdisk` para fazer isto é o `t` seguida pelo número da partição. Mudado os três tipos, o comando `p` dentro do `fdisk` irá mostrar algo parecido com isto:

```
Disk /dev/sda: 160.0 GB, 160041885696 bytes
255 heads, 63 sectors/track, 19457 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            1          122     979933+   fd  Linux raid autodetect
/dev/sda2            *        123          134      96390   fd  Linux raid autodetect
/dev/sda3           135       19457    155211997+  fd  Linux raid autodetect
```

Salve com o comando `w` do `fdisk`. Depois reinicie a máquina.

Depois de reiniciada, copie a tabela de partições do HD primário (`sda`) para o HD secundário (`sdb`), assim os dois vão ficar com as partições exatamente na mesma ordem e do mesmo tamanho também. Para esta tarefa, use o comando “`sfdisk`” como no exemplo abaixo:

```
# sfdisk -d /dev/sda > /tmp/partitions.txt
# sfdisk /dev/sdb < /tmp/partitions.txt
```

O primeiro comando acima escreve no arquivo `/tmp/partitions.txt` a tabela exata das partições do primeiro disco. O segundo comando particiona o segundo disco exatamente de acordo com o conteúdo do arquivo `/tmp/partitions.txt`.

**Lembre-se:** O segundo comando destrói toda a tabela de partições no disco `/dev/sdb`. Certifique-se de que o dispositivo seja este mesmo antes de executar o comando para não perder dados.

**Nota:** Caso você não possua HDs idênticos, você terá que fazer o processo de particionamento manualmente. Ao invés de utilizar o comando `sfdisk` para copiar a tabela de partições de um HD para o outro, crie manualmente via `fdisk` as partições na mesma ordem e tamanho do primeiro HD.

### 3.3.3. Dispositivo RAID

Agora é hora de criar o dispositivo RAID, que por enquanto só terá como membro o segundo HD, pois o primeiro está sendo utilizado pelo sistema. Como temos 3 partições que queremos incluir no RAID1, criaremos três dispositivos RAID1:

```
# mdadm --create /dev/md0 --level=1 --raid-devices=2 missing /dev/sdb3
# mdadm --create /dev/md1 --level=1 --raid-devices=2 missing /dev/sdb1
# mdadm --create /dev/md2 --level=1 --raid-devices=2 missing /dev/sdb2
```

Os comandos acima criam três dispositivos: `/dev/md0`, `/dev/md1` e `/dev/md2`. Todos eles são do tipo RAID1 (`--level=1`), contendo duas partições: uma faltando (primeiro disco) e a partição do segundo disco em si.

Para fácil memorização, aqui estão os dispositivos e a equivalência de sua função no sistema:

- `/dev/md0` – / (raiz)
- `/dev/md1` – Memória SWAP
- `/dev/md2` – /boot

Para verificar se os dispositivos foram criados e reconhecidos, tem que haver semelhança com a saída do comando a seguir:

```
# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4]
[raid10]
md1 : active raid1 sdb1[0]
      979840 blocks [2/1] [U_]

md2 : active raid1 sdb2[0]
      96320 blocks [2/1] [U_]

md0 : active raid1 sdb3[0]
      155211904 blocks [2/1] [U_]

unused devices: <none>
```

Caso não esteja de acordo, reinicie a máquina para o kernel reconhecer as novas configurações.

### 3.3.4. Sistemas de Arquivos

Primeiro crie o sistema de swap na partição /dev/md1:

```
# mkswap /dev/md1
```

Depois crie o sistema de arquivos *ext3* na partição /dev/md0 (raiz):

```
# mke2fs -j -m1 /dev/md0
```

Em seguida crie o sistema de arquivos *ext3* na partição /dev/md2 (/boot):

```
#mke3fs -j -m0 /dev/md2
```

### 3.3.5. O sistema no segundo disco

Agora que o segundo disco já está pronto com os dispositivos RAID1 funcionais, é hora de copiar todo o conteúdo do primeiro HD para o segundo HD.

Antes de começar toda a cópia, modifique a configuração do GRUB para ele carregar o sistema no dispositivo RAID ao invés de diretamente do primeiro HD. Para fazer isso, edite o arquivo /boot/grub/menu.lst, deixando a entrada do sistema operacional como no exemplo a seguir:

```
title          Debian GNU/Linux, kernel 2.6.24-1-686
root          (hd1, 1)
kernel        /vmlinuz-2.6.24-1-686 root=/dev/md0 ro
initrd        /initrd.img-2.6.24-1-686
savedefault
```

Em seguida, é hora de fazer toda a cópia. Monte o esqueleto do segundo HD no diretório /mnt e copie via comando “cp -a” os diretórios da raiz do primeiro HD para o esqueleto.

Use o exemplo a seguir como um guia:

```
# cd /mnt
# mkdir novo-boot nova-raiz
```

```
# mount -t ext3 /dev/md2 novo-boot
# mount -t ext3 /dev/md0 nova-raiz

# cp -a /boot/* novo-boot

# cd nova-raiz
# mkdir boot initrd mnt proc sys
# cp -a /bin /cdrom /dev /etc /home /initrd.img /lib /media ./
# cp -a /opt /root /sbin /srv /tmp /usr /var /vmlinuz ./
```

Depois de copiado todos os arquivos (pode demorar um pouco), edite o arquivo `/mnt/nova-raiz/etc/fstab`, que fica no dispositivo RAID, para refletir as configurações da nova localização do sistema. Mude os dispositivos das partições para seus equivalentes como dispositivos RAID. Veja o exemplo a seguir:

<code>proc</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>	
<code>/dev/md0</code>	<code>/</code>	<code>ext3</code>	<code>defaults,errors=remount-ro</code>	<code>0</code>	<code>0</code>	<code>1</code>
<code>/dev/md2</code>	<code>/boot</code>	<code>ext3</code>	<code>defaults</code>	<code>0</code>	<code>2</code>	
<code>/dev/md1</code>	<code>none</code>	<code>swap</code>	<code>sw</code>	<code>0</code>	<code>0</code>	
<code>/dev/hda</code>	<code>/media/cdrom0</code>	<code>udf,iso9660</code>	<code>user,noauto</code>	<code>0</code>	<code>0</code>	

Feito isso, reinicie o sistema. O GRUB deverá carregar o sistema no segundo HD, deixando o primeiro livre. Para verificar se isto foi realmente feito, utilize o comando `mount` e compare com o exemplo a seguir:

```
# mount
/dev/md/0 on / type ext3 (rw,errors=remount-ro)
tmpfs on /lib/init/rw type tmpfs (rw,nosuid,mode=0755)
proc on /proc type proc (rw,noexec,nosuid,nodev)
sysfs on /sys type sysfs (rw,noexec,nosuid,nodev)
procbususb on /proc/bus/usb type usbfs (rw)
udev on /dev type tmpfs (rw,mode=0755)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,noexec,nosuid,gid=5,mode=620)
/dev/md/2 on /boot type ext3 (rw)
```

### 3.3.6. Adicionando o primeiro disco no RAID1

Agora que o sistema está funcionando exclusivamente no dispositivo RAID, o primeiro HD está livre e podemos adicioná-lo ao dispositivo também. Lembrando que esta operação irá destruir todos os dados do primeiro HD.

Para adicionar as três partições do primeiro disco no RAID, utilize os comandos:

```
mdadm /dev/md0 -a /dev/sda3
mdadm /dev/md1 -a /dev/sda1
mdadm /dev/md2 -a /dev/sda2
```

O processo de adição do primeiro disco no RAID pode demorar bastante. Uma vez adicionada as partições, o Linux irá sincronizar todos os dados, copiando todas as informações do segundo disco no primeiro e criando os recursos de redundância (espelhamento). Para aumentar a velocidade deste processo, certifique-se que o sistema não esteja sendo muito usado e execute o comando:

```
echo -n 500000 > /proc/sys/dev/raid/speed_limit_max
```

Isso irá aumentar a velocidade padrão de atividades de sincronismo do RAID. Se você quiser

acompanhar o processo detalhadamente, abra um novo terminal e utilize o seguinte comando:

```
watch --interval 1 'cat /proc/mdstat'
```

Este comando lhe passará o conteúdo do arquivo /proc/mdstat (que contém o status atual dos dispositivos RAID do sistema) em um intervalo de 1 segundo. Ao final do processo, o conteúdo do arquivo /proc/mdstat deverá estar semelhante ao seguinte:

```
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4]
[raid10]
md1 : active raid1 sda1[0] sdb1[1]
      979840 blocks [2/2] [UU]

md2 : active raid1 sda2[0] sdb2[1]
      96320 blocks [2/2] [UU]

md0 : active raid1 sda3[0] sdb3[1]
      155211904 blocks [2/2] [UU]

unused devices: <none>
```

Pronto, agora o espelhamento entre HDs já está funcionando.

### 3.3.7. Instalando o GRUB nos dois discos

Apesar do espelhamento estar funcionando, o antigo GRUB foi destruído e precisamos agora instalá-lo nos dois discos para uma redundância verdadeira. Para fazer isso execute o comando “grub”, que irá lhe colocar em uma linha de comando do próprio grub. Os três comandos a seguir irão instalar o GRUB no primeiro HD:

```
grub> device (hd0) /dev/sda
grub> root (hd0,0)
  Filesystem type is ext2fs, partition type 0xfd
grub> setup (hd0)
  Checking if "/boot/grub/stage1" exists... no
  Checking if "/grub/stage1" exists... yes
  Checking if "/grub/stage2" exists... yes
  Checking if "/grub/e2fs_stage1_5" exists... yes
  Running "embed /grub/e2fs_stage1_5 (hd0)"... 15 sectors are embedded.
  succeeded
  Running "install /grub/stage1 (hd0) (hd0)1+15 p (hd0,0)/grub/stage2
/grub/grub.conf"... succeeded
Done.
```

Agora faça o mesmo para o segundo disco:

```
grub> device (hd0) /dev/sdb
grub> root (hd0,0)
  Filesystem type is ext2fs, partition type 0xfd
grub> setup (hd0)
  Checking if "/boot/grub/stage1" exists... no
  Checking if "/grub/stage1" exists... yes
  Checking if "/grub/stage2" exists... yes
  Checking if "/grub/e2fs_stage1_5" exists... yes
  Running "embed /grub/e2fs_stage1_5 (hd0)"... 15 sectors are embedded.
  succeeded
  Running "install /grub/stage1 (hd0) (hd0)1+15 p (hd0,0)/grub/stage2
/grub/grub.conf"... succeeded
```

Done.

Por último, edite o arquivo de configuração do GRUB para deixar a entrada do sistema da seguinte forma:

```
title          Debian GNU/Linux, kernel 2.6.24-1-686
root          (hd0, 1)
kernel        /vmlinuz-2.6.24-1-686 root=dev/md0 ro
initrd        /initrd.img-2.6.24-1-686
savedefault
```

Com este último passo, todas as configurações do espelhamento estarão feitas e o sistema inteiro (inclusive a memória swap) estará funcionando nos dois discos ao mesmo tempo. Você pode reiniciar a máquina para verificar se tudo está funcionando devidamente, lembrando que pode-se verificar o status dos dispositivos RAID no conteúdo do arquivo `/proc/mdstat`.