

Process of creating a bootable mirrored RAID volume in Tiger

2/15/06 12:47 PM

In this example we have two 80 gig drives in an xserve. We are booted to one drive and the server is live. The second drive has not really been used as of yet and we want to set the boot drive to be a mirrored raid without long downtime or a reformat.

- First we need to format the extra drive. This can be done in Disk Utility or Terminal.
- We can check the disks using the diskutil command.

```
#: diskutil list
```

```
/dev/disk0
```

#:	type name	size	identifier
0:	Apple_partition_scheme	*76.7 GB	disk0
1:	Apple_partition_map	31.5 KB	disk0s1
2:	Apple_HFS Startup	76.6 GB	disk0s3

```
/dev/disk1
```

#:	type name	size	identifier
0:	Apple_partition_scheme	*76.7 GB	disk0
1:	Apple_partition_map	31.5 KB	disk0s1
2:	Apple_HFS Server_HD	76.7 GB	disk0s3

- In this case, we are going to format disk1:

```
diskutil eraseDisk "Journaled HFS+" Mirror disk1
```

- This names the new drive "Mirror"
- We now change it over to a degraded mirror raid set.

```
diskutil enableRAID mirror /Volumes/Mirror
```

- Once this is set, we need to clone the data from the existing startup drive to the new mirror. I prefer to do this with rsync, but a disk utility restore should work as well as other cloning apps. With rsync and other apps ensure that the permissions as being preserved across both volumes via the Get Info window in the Finder.

```
rsync -aHP --exclude /dev/* --exclude /afs/* --exclude /private/tmp/* --exclude /Network/* --exclude /Volumes/* --exclude /automount/* --exclude /private/var/run/* "/" /Volumes/Mirror -P -delete
```

- Now we should have a good clone. Go to System Preferences and select the mirror as the new startup disk and reboot the system.
- Everything should boot up fine with the system not noticing a difference except we are now on a broken raid set. We need to now format the original startup disk drive and add it in to the broken raid.

```
#: diskutil list
```

```
/dev/disk0
```

#:	type name	size	identifier
0:	Apple_partition_scheme	*76.7 GB	disk0
1:	Apple_partition_map	31.5 KB	disk0s1
2:	Apple_HFS Startup	76.6 GB	disk0s3

```
/dev/disk1
```

#:	type name	size	identifier
0:	Apple_partition_scheme	*76.7 GB	disk1
1:	Apple_partition_map	31.5 KB	disk1s1
2:	Apple_Boot	128.0 MB	disk1s2
3:	Apple_RAID	76.6 GB	disk1s3

```
/dev/disk2
```

#:	type name	size	identifier
0:	Apple_HFS Mirror	*76.6 GB	disk2

- The disk we need to format in this case is Startup – disk0.
diskutil eraseDisk "Journaled HFS+" blank disk0
- The disk is now called blank. Now we add it to the raid. We do this with diskutil providing the raid logical volume and the new blank partition.
diskutil repairMirror disk2 disk0s3
- This should work and the rebuild should start. We can check this by typing:
diskutil checkRAID
- I did have one instance where checkRAID showed the two drives but listed one as a "Spare" and didn't rebuild. In this case I changed the drives membership from a spare to a member which kicked off a rebuild.
diskutil addToRAID member disk0s3 disk2