# [Raid Managment](http://shabbathster.blogspot.com/2013/07/raid-managment.html)

# RAID Managment

# RAID 0, RAID 1, RAID 5, RAID 10

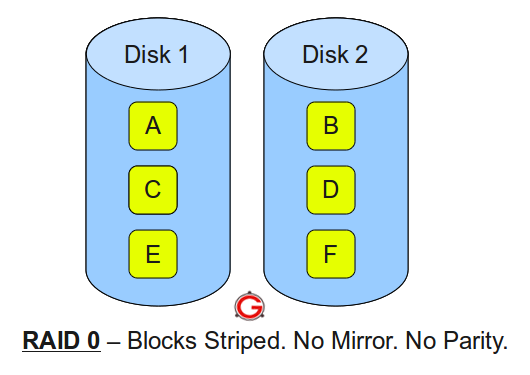
RAID stands for Redundant Array of Inexpensive (Independent) Disks.  
On most situations you will be using one of the following four levels of RAIDs.

* RAID 0
* RAID 1
* RAID 5
* RAID 10 (also known as RAID 1+0)

This article explains the main difference between these raid levels along with an easy to understand diagram.  
  
In all the diagrams mentioned below:

* A, B, C, D, E and F – represents blocks
* p1, p2, and p3 – represents parity

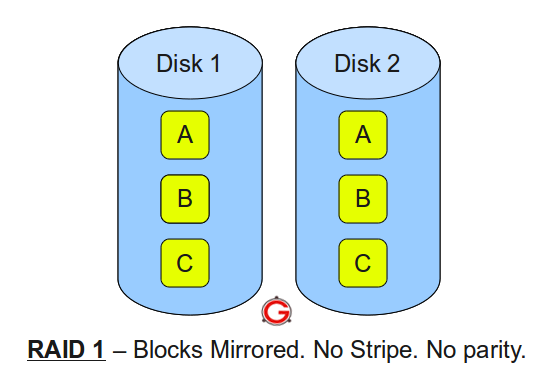
### RAID LEVEL 0



Following are the key points to remember for RAID level 0.

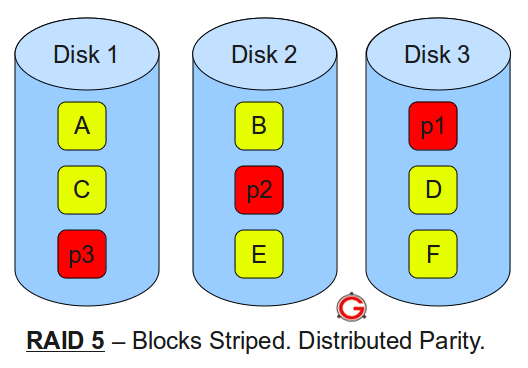
* Minimum 2 disks.
* Excellent performance ( as blocks are striped ).
* No redundancy ( no mirror, no parity ).
* Don’t use this for any critical system.

### RAID LEVEL 1

Following are the key points to remember for RAID level 1.

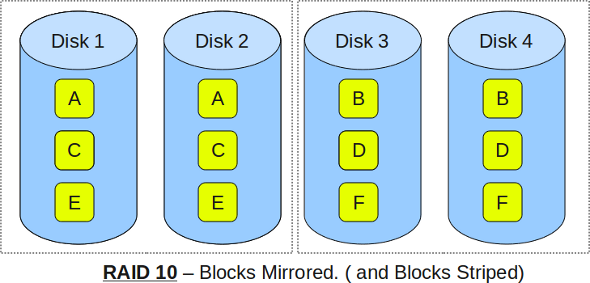
* Minimum 2 disks.
* Good performance ( no striping. no parity ).
* Excellent redundancy ( as blocks are mirrored ).

### RAID LEVEL 5

  
Following are the key points to remember for RAID level 5.

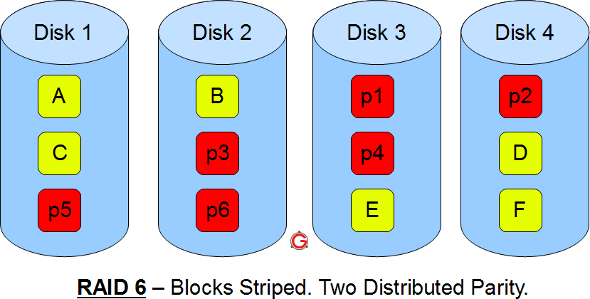
* Minimum 3 disks.
* Good performance ( as blocks are striped ).
* Good redundancy ( distributed parity ).
* Best cost effective option providing both performance and redundancy. Use this for DB that is heavily read oriented. Write operations will be slow

### RAID LEVEL 10

Following are the key points to remember for RAID level 10.

* Minimum 4 disks.
* This is also called as “stripe of mirrors”
* Excellent redundancy ( as blocks are mirrored )
* Excellent performance ( as blocks are striped )
* If you can afford the dollar, this is the BEST option for any mission critical applications (especially databases).

### RAID 6

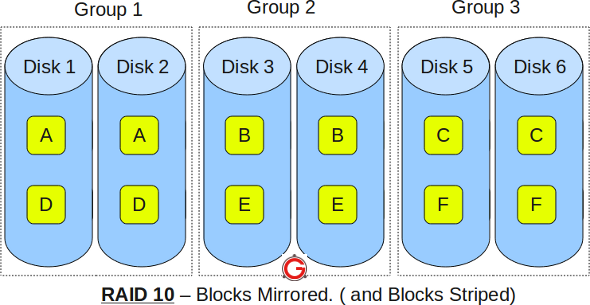


* Just like RAID 5, this does block level striping. However, it uses dual parity.
* In the above diagram A, B, C are blocks. p1, p2, p3 are parities.
* This creates two parity blocks for each data block.
* Can handle two disk failure
* This RAID configuration is complex to implement in a RAID controller, as it has to calculate two parity data for each data block.

# RAID 10 Vs RAID 01 (RAID 1+0 Vs RAID 0+1)

RAID 10 is not the same as RAID 01.  
This article explains the difference between the two with a simple diagram.  
I’m going to keep this explanation very simple for you to understand the basic concepts well. In the following diagrams A, B, C, D, E and F represents blocks.

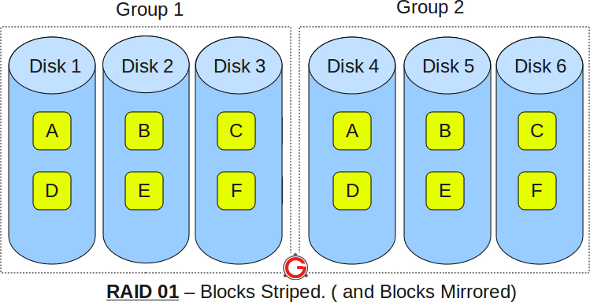
### RAID 10



* RAID 10 is also called as RAID 1+0
* It is also called as “stripe of mirrors”
* It requires minimum of 4 disks
* To understand this better, group the disks in pair of two (for mirror). For example, if you have a total of 6 disks in RAID 10, there will be three groups–Group 1, Group 2, Group 3 as shown in the above diagram.
* Within the group, the data is mirrored. In the above example, Disk 1 and Disk 2 belongs to Group 1. The data on Disk 1 will be exactly same as the data on Disk 2. So, block A written on Disk 1 will be mirroed on Disk 2. Block B written on Disk 3 will be mirrored on Disk 4.
* Across the group, the data is striped. i.e Block A is written to Group 1, Block B is written to Group 2, Block C is written to Group 3.
* This is why it is called “stripe of mirrors”. i.e the disks within the group are mirrored. But, the groups themselves are striped.

If you are new to this, make sure you understand how [RAID 0, RAID 1 and RAID 5](http://www.thegeekstuff.com/2010/08/raid-levels-tutorial/) and [RAID 2, RAID 3, RAID 4, RAID 6](http://www.thegeekstuff.com/2011/11/raid2-raid3-raid4-raid6/)works.

### RAID 01

* RAID 01 is also called as RAID 0+1
* It is also called as “mirror of stripes”
* It requires minimum of 3 disks. But in most cases this will be implemented as minimum of 4 disks.
* To understand this better, create two groups. For example, if you have total of 6 disks, create two groups with 3 disks each as shown below. In the above example, Group 1 has 3 disks and Group 2 has 3 disks.
* Within the group, the data is striped. i.e In the Group 1 which contains three disks, the 1st block will be written to 1st disk, 2nd block to 2nd disk, and the 3rd block to 3rd disk. So, block A is written to Disk 1, block B to Disk 2, block C to Disk 3.
* Across the group, the data is mirrored. i.e The Group 1 and Group 2 will look exactly the same. i.e Disk 1 is mirrored to Disk 4, Disk 2 to Disk 5, Disk 3 to Disk 6.
* This is why it is called “mirror of stripes”. i.e the disks within the groups are striped. But, the groups are mirrored.

### Main difference between RAID 10 vs RAID 01

* Performance on both RAID 10 and RAID 01 will be the same.
* The storage capacity on these will be the same.
* The main difference is the fault tolerance level. On most implememntations of RAID controllers, RAID 01 fault tolerance is less. On RAID 01, since we have only two groups of RAID 0, if two drives (one in each group) fails, the entire RAID 01 will fail. In the above RAID 01 diagram, if Disk 1 and Disk 4 fails, both the groups will be down. So, the whole RAID 01 will fail.
* RAID 10 fault tolerance is more. On RAID 10, since there are many groups (as the individual group is only two disks), even if three disks fails (one in each group), the RAID 10 is still functional. In the above RAID 10 example, even if Disk 1, Disk 3, Disk 5 fails, the RAID 10 will still be functional.
* So, given a choice between RAID 10 and RAID 01, always choose RAID 10.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
**Raid Configuration Command**  
pv /dev/sda2 > /dev/null  
Removing the failed partition(s) and disk:  
# mdadm –manage /dev/md0 –fail /dev/sdb1  
# mdadm –manage /dev/md1 –remove /dev/sdb2  
shutdown -h now  
Copy partition table from good OS drive to new drive  
dd if=/dev/sd[letter of current drive] of=/dev/sd[letter of new drive] bs=512 count=1  
Adding the new disk to the RAID Array  
# mdadm –manage /dev/md0 –add /dev/sdb1  
#cat /proc/mdstat  
#/grub-install /dev/sd2   
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
If the device contains a valid md superblock, the block is overwritten with zeros:  
# mdadm –zero-superblock /dev/sdc /dev/sdd  
Create RAID1 using /dev/sdc1 and /dev/sdd1  
# mdadm –create /dev/md0 –level=1 –raid-devices=2 /dev/sdc1 /dev/sdd1  
#mkfs.ext3 /dev/md0  
#mkdir /raid1  
# mount /dev/md0 /raid1  
# df -H  
Edit /etc/fstab  
Make sure RAID1 get mounted automatically. Edit /etc/fstab and append the following line:  
/dev/md0 /raid1 ext3 noatime,rw 0 0\  
See RAID Array Building Progress and Current Status  
# watch -n 2 cat /proc/mdstat or # tail -f /proc/mdstat  
Get Information On Existing Array  
# mdadm –query /dev/md0  
Update /etc/mdadm.conf File  
ARRAY /dev/md0 devices=/dev/sdc1,/dev/sdd1 level=1 num-devices=2 auto=yes  
###############################################################################  
  
         RAID(REDUNDANT ARRAY OF INDEPENDENT DISK)  
        --------------------------------------------------  
  
Used For:- Fault tolerence and load balancing  
---------  
  
Fault Tplerence:- Not loose the data for that take a precation so for that fault tolerence is used  
  
Level:--  
--------------  
  
Raid0 :- stripped without parity  
  
Raid1 :- Mirroring  
  
Raid5 :- Stripped with parity  
  
Uses Of levels:-  
---------------------  
  
Raid0:- Faster read and write access but dosnt provide fault tolerence  
  
Raid1 , Raid5 :- Fault Tolerence  
  
  
  
Commands to create raid  
-----------------------------  
  
Raid levels 0 -->  
  
--> Create two partitions and change their system id to fd.  
  
#mdadm -C /dev/md0 --level=0 --raid-devices=2 /dev/hda12 /dev/hda13  
  
#mke2fs -j /dev/md0  
  
#mount /dev/md0 /mnt  
  
#mdadm --detail /dev/md0   --> show details of /dev/md0  
  
#cat /proc/mdstat    --> ^  
  
#mdadm -f /dev/md0 /dev/hda12  --> make one disk faulty from raid device.  
  
#mdadm -r /dev/md0 /dev/hda12   --> remove faulty disk from raid device  
  
--> Create a new partition (eg. /dev/hda14) and change it's system id to fd.  
  
#mdadm -a /dev/md0 /dev/hda14   --> add new disk in raid device.  
  
#mdadm --grow /dev/md0 raid-devices=3  
  
Note:- All MAjor and Minor number are present in this file /usr/share/doc/MAKE-3.23/devices-2.6+.txt

888888888888888888888888888888888888888888

[Raid Partition how to](http://shabbathster.blogspot.com/2014/07/raid-partition-how-to.html)

RAID (Redundant Array of Inexpensive Disks)  
  
Create 3 partitions for implementing RAID using fdisk command.  
  
e.g. #fdisk /dev/hda  
  
Press n to create the 3 new partitions each of 100Mb in size.  
  
Press p to see the partition table.  
  
Press t to change the partition id of all the three partitions created by you to fd  
  
(linux raid auto).  
  
Press wq to save and exit from fdisk utility in linux.  
  
#partprobe  
  
Use fdisk -l to list the partition table.  
  
Creating RAID  
  
# mdadm --create /dev/md0 --level=5 --raid-devices=3 /dev/hda6 /dev/hda7 /dev/  
  
hda8  
  
Press y to create the arrays.  
  
To see the details of raid use the following command: -  
  
# cat /proc/mdstat  
  
# mdadm --detail /dev/md0  
  
Creating the file system for your RAID devices  
  
#mkfs.ext3 /dev/md0  
  
Mounting the RAID partition  
  
#mkdir data  
  
# mount /dev/md0 data  
  
#df -h /root/data (Command is used to see the space allocation).  
  
Crashing the raid devices  
  
# mdadm --manage /dev/md0 --fail /dev/hda8  
  
Removing raid devices  
  
# mdadm --manage /dev/md0 --remove /dev/hda8  
  
Adding raid devices  
  
# mdadm --manage /dev/md0 --add /dev/hda8  
  
View failed and working raid devices  
  
# cat /proc/mdstat  
  
# mdadm --detail /dev/md0  
  
# tail /var/log/messages  
  
To remove the RAID follow these steps: -  
  
1) unmount the mounted directory where raid is mounted.  
  
 e.g. umount data  
  
2) Stop the device  
  
 e.g. mdadm --stop /dev/md0  
  
3) View the details of your raid level using following command: -  
  
 #cat /proc/mdstat  
  
 #mdadm --detail /dev/md0