Re: Building a file server − advice please

Source: http://linux.derkeiler.com/Newsgroups/comp.os.linux.setup/2006−01/msg00120.html

- *From*: Alan Adams <alan.adams@xxxxxxxxxxxxxxxxxxxxxxxxxxx>
- *Date*: Thu, 05 Jan 2006 19:19:48 GMT

In message <11rqaf3lqdgm4ba@xxxxxxxxxxxxxxxxxx>
Jean−David Beyer <jdbeyer@xxxxxxxxxxx> wrote:

> Nico Kadel−Garcia wrote (in part):
>
> > That's what RAID5 does. That way, if one drive fails, you can keep going for
> > a while until you can replace the drive. And with 4 drives, your odds of a
> > single drive failing go way, way, way up.
> >
> Do I misunderstand you, or did you say it ambiguously?
>
> With 5 drives, the odds of a single drive failing _remain the same_; i.e.,
> connecting up 5 drives in a RAID5 system does not affect the Mean Time To
> Failure of a drive in any way (unless the box is improperly cooled or the
> power supply overloaded). The perceived failure rate of a single drive may
> go up slightly because of the failure rate of the RAID controller, but that
> should be very small since the controller has no moving parts, so the
> failure rate of the connectors and cables probably dominates this.
>
> The odds of the entire system failing presumably go down, which is an
> important reason to use a RAID system. I do not know the details of one
> level of RAID from another, but if it is true that RAID 5 can tolerate one
> drive failing and still work, the odds of a system failure with 5 drives in
> there would be 1/5 those of using a single drive.
>

As originally designed, and as currently implemented in Hardware
controllers, Raid allows for redundancy (Except Raid 0 which is simply
adding disks together to get a big one − that INCREASES the risk of failure,
as a failure on either disk trashes the data on both.)

The concept is to "duplicate" part or all of the data on additional disks.
With Raid 3 and 4, it's done by calculating a parity value and storing that.
With Raid 5 the entire disc is duplicated. Raid 5 is also called mirroring.

Hardware controllers generally can have an additional spare disk configured
so that when a disk in any of the raid arrays fails, the spare can be
automagically used to replace it. This means you still have your original
degree of redundancy, and now have to replace an offline disk when

convenient.

Software raid emulates some of these functions, but doesn't give such a good degree of control.

Hardware raid presents each raid array to the host as one disk, or in some cases, can partition the array and present that as a disk. In either case the host operating system is ignorant of the raid system, and thus needs no knowledge of how to manage it. (There will usually be a utility program to do the management, but that is independent of the operating system, and may run on separate hardware from the host.)

A typical Raid 3/4 array will consist of 6 disks − five data and one parity. This gives a 20% overhead, i.e. 5 disks worth of space for the cost of 6 disks. In one of these, I forget which, it really does use a parity disk, in the other the data and parity are striped across all disks. It makes a performance difference − one works better with big files like databases, the other is better with small files, but generally it doesn't matter which you use.

Raid 0 is sometimes called striping. In that implementation, every file is likely to be spread across all the disks in the array. Not good if a disk goes down.

High performance large raid systems are called SAN or NAS and usually connect to the host via Fibre Channel. The concept was in use before NAS and SAN became buzzwords. They have the advantage that they can be accessed by more than one host computer, so you can get host system redundancy as well.

All you need for those is a six−figure budget.

========

Going back on track for your storage needs, a raid array of 6 250GB disks will give you 1250GB of space. Since your host system may have trouble accessing a virtual disk that big, it might be necessary to partition it on the controller, and present it to the host in, say 250GB chunks. The host thinks it has 5 x 250GB disks, but all 5 of them benefit from the one redundant parity disk.

I don't know the limits of an ext3 filesystem, but if you're serving it to Windows, you need to consider the limits in Windows. I think there may be issues at 256GB, but don't quote me on that. If you're serving it to RISC OS you'll need to check the limits in LanMan98 or Sunfish. (My limited testing has worked better with LanMan98, but that may be because I know a bit more about that area then NFS.) I suspect you'll hit problems at 128GB.

−−
Alan Adams, from Northamptonshire
alan.adams@xxxxxxxxxxxxxxxxxxxxxxxxxxx
http://www.nckc.org.uk/

.

- ***Follow−Ups***:
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* Dave Stratford
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* Michael Heiming

- ***References***:
  - ♦ ***Building a file server − advice please***
    - ◊ *From:* daves
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* Michael Heiming
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* daves
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* Nico Kadel−Garcia
  - ♦ ***Re: Building a file server − advice please***
    - ◊ *From:* Jean−David Beyer

- Prev by Date: ***Re: grub fallback mechanism***
- Next by Date: ***Re: Building a file server − advice please***
- Previous by thread: ***Re: Building a file server − advice please***
- Next by thread: ***Re: Building a file server − advice please***
- Index(es):
  - ♦ ***Date***
  - ♦ ***Thread***