High Availability Storage [closed]

Asked 16 years, 3 months ago Modified 14 years, 10 months ago

Viewed 4k times

▲

4

▼

Closed. This question is off-topic. It is not currently accepting answers.

Want to improve this question? Update the question so it's on-topic for Stack Overflow.

Closed 12 years ago.

Improve this question

I would like to make 2 TB or so available via NFS and CIFS. I am looking for a 2 (or more) server solution for high availability and the ability to load balance across the servers if possible. Any suggestions for clustering or high availability solutions?

This is business use, planning on growing to 5-10 TB over next few years. Our facility is almost 24 hours a day, six days a week. We could have 15-30 minutes of downtime, but we want to minimize data loss. I want to minimize 3 AM calls.

We are currently running one server with ZFS on Solaris and we are looking at AVS for the HA part, but we have had minor issues with Solaris (CIFS implementation doesn't work with Vista, etc) that have held us up.

We have started looking at

- DRDB over GFS (GFS for distributed lock capability)
- Gluster (needs client pieces, no native CIFS support?)
- Windows DFS (doc says only replicates after file closes?)

We are looking for a "black box" that serves up data.

We currently snapshot the data in ZFS and send the snapshot over the net to a remote datacenter for offsite backup.

Our original plan was to have a 2nd machine and rsync every 10 - 15 min. The issue on a failure would be that ongoing production processes would lose 15 minutes of data and be left "in the middle". They would almost be easier to start from the beginning than to figure out where to pickup in the middle. That is what drove us to look at HA solutions.

storage **high-availability**

Share

Improve this question

Follow

edited Sep 18, 2008 at 1:05

asked Sep 17, 2008 at 5:09

petey

81 ● 1 ● 2 ● 5

This is for business use, We are testing DRDB on GFS, looking into Gluster,but that requires client pieces,We've played around with ZFS on Solaris (and we are looking at AVS for availability), but Solaris keeps having small problems to be over come. We have also started looking at windows DFS. – petey Sep 17, 2008 at 11:25

We are planning to scale up over the next few years to 5-10 TB, but nothing huge. It is a production facility, we could spend 15-30 minutes switching over if we had too, but we want to minimize data loss. Our process are almost 24 x 6, so want to minimize the 3 am calls, would prefer HA – petey Sep 17, 2008 at 11:28

10 Answers

Sorted by: Highest score (default)

6

I've recently deployed hanfs using DRBD as the backend, in my situation, I'm running active/standby mode, but I've tested it successfully using OCFS2 in primary/primary mode too. There unfortunately isn't much documentation out there on how best to achieve this, most that exists is barely useful at best. If you do go along the drbd route, I highly recommend joining the drbd mailing list, and reading all of the documentation. Here's my ha/drbd setup and script I wrote to handle ha's failures:

DRBD8 is required - this is provided by drbd8-utils and drbd8-source. Once these are installed (I believe they're provided by backports), you can use module-assistant to install it - m-a a-i drbd8. Either depmod -a or reboot at this point, if you depmod -a, you'll need to modprobe drbd.

You'll require a backend partition to use for drbd, do not make this partition LVM, or you'll hit all sorts of problems. Do not put LVM on the drbd device or you'll hit all sorts of problems.

Hanfs1:

```
/etc/drbd.conf

global {
        usage-count no;
}
common {
        protocol C;
        disk { on-io-error detach; }
}
resource export {
        syncer {
                rate 125M;
        }
        on hanfs2 {
                address         172.20.1.218:7789;
                device          /dev/drbd1;
                disk            /dev/sda3;
                meta-disk       internal;
        }
        on hanfs1 {
                address         172.20.1.219:7789;
                device          /dev/drbd1;
                disk            /dev/sda3;
                meta-disk       internal;
       }
}
```

Hanfs2's /etc/drbd.conf:

```

global {
        usage-count no;
}
common {
        protocol C;
        disk { on-io-error detach; }
}
resource export {
        syncer {
                rate 125M;
        }
        on hanfs2 {
                address         172.20.1.218:7789;
                device          /dev/drbd1;
                disk            /dev/sda3;
                meta-disk       internal;
        }
        on hanfs1 {
                address         172.20.1.219:7789;
                device          /dev/drbd1;
                disk            /dev/sda3;
                meta-disk       internal;
       }
}
```

Once configured, we need to bring up drbd next.

```
drbdadm create-md export
drbdadm attach export
drbdadm connect export
```

We must now perform an initial synchronization of data - obviously, if this is a brand new drbd cluster, it doesn't matter which node you choose.

Once done, you'll need to mkfs.yourchoiceoffilesystem on your drbd device - the device in our config above is /dev/drbd1. http://www.drbd.org/users-guide/p-work.html is a useful document to read while working with drbd.

Heartbeat

Install heartbeat2. (Pretty simple, apt-get install heartbeat2).

/etc/ha.d/ha.cf on each machine should consist of:

hanfs1:

```
logfacility local0
keepalive 2
warntime 10
deadtime 30
initdead 120



ucast eth1 172.20.1.218



auto_failback no



node hanfs1
node hanfs2
```

hanfs2:

```
logfacility local0
keepalive 2
warntime 10
deadtime 30
initdead 120



ucast eth1 172.20.1.219



auto_failback no



node hanfs1
node hanfs2
```

/etc/ha.d/haresources should be the same on both ha boxes:

```
hanfs1 IPaddr::172.20.1.230/24/eth1
hanfs1  HeartBeatWrapper
```

I wrote a wrapper script to deal with the idiosyncracies caused by nfs and drbd in a failover scenario. This script should exist within /etc/ha.d/resources.d/ on each machine.

```
!/bin/bash

heartbeat fails hard.

so this is a wrapper

to get around that
stupidity

I'm just wrapping the
heartbeat scripts,
except for in the
```

With all of the above done, you'll then just want to configure /etc/exports

```
/export
172.20.1.0/255.255.255.0(rw,sync,fsid=1,no_root_squa
```

Then it's just a case of starting up heartbeat on both machines and issuing hb_takeover on one of them. You can test that it's working by making sure the one you issued the takeover on is primary - check /proc/drbd, that the device is mounted correctly, and that you can access nfs.

--

Best of luck man. Setting it up from the ground up was, for me, an extremely painful experience.

Share Improve this answer Follow

edited Sep 17, 2008 at 19:24

answered Sep 17, 2008 at 17:27

Tony Dodd
61 ● 4

There needs to be a sort of badge out there for this sort of post. – Kent Fredric Sep 17, 2008 at 20:04

3

These days 2TB fits in one machine, so you've got options, from simple to complex. These all presume linux servers:

- You can get poor-man's HA by setting up two machines and doing a periodic rsync from the main one to the backup.

- You can use DRBD to mirror one from the other at the block level. This has the disadvantage of being somewhat difficult to expand in the future.
- You can use OCFS2 to cluster the disks instead, for future expandability.

There are also plenty of commercial solutions, but 2TB is a bit small for most of them these days.

You haven't mentioned your application yet, but if hot failover isn't necessary, and all you really want is something that will stand up to losing a disk or two, find a NAS that support RAID-5, at least 4 drives, and hotswap and you should be good to go.

Share Improve this answer Follow

edited Sep 17, 2008 at 5:36

answered Sep 17, 2008 at 5:15

pjz
43k ● 6 ● 53 ● 60

1

I would recommend **NAS Storage. (Network Attached Storage).**

HP has some nice ones you can choose from.

http://h18006.www1.hp.com/storage/aiostorage.html

as well as Clustered versions:

http://h18006.www1.hp.com/storage/software/clusteredfs/index.html?jumpid=reg_R1002_USEN

Share Improve this answer Follow

answered Sep 17, 2008 at 5:13

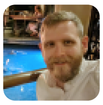
Sev
15.7k ● 11 ● 60 ● 76

0

Are you looking for an "enterprise" solution or a "home" solution? It is hard to tell from your question, because 2TB is very small for an enterprise and a little on the high end for a home user (especially two servers). Could you clarify the need so we can discuss tradeoffs?

Share Improve this answer Follow

answered Sep 17, 2008 at 5:13



David Ackerman
12.7k ● 6 ● 25 ● 19

0

There's two ways to go at this. The first is to just go buy a SAN or a NAS from Dell or HP and throw money at the problem. Modern storage hardware just makes all of this easy to do, saving your expertise for more core problems.

If you want to roll your own, take a look at using Linux with DRBD.

http://www.drbd.org/

DRBD allows you to create networked block devices. Think RAID 1 across two servers instead of just two disks. DRBD deployments are usually done using Heartbeat for failover in case one system dies.

I'm not sure about load balancing, but you might investigate and see if LVS can be used to load balance across your DRBD hosts:

http://www.linuxvirtualserver.org/

To conclude, let me just reiterate that you're probably going to save yourself a lot of time in the long run just forking out the money for a NAS.

Share Improve this answer Follow

answered Sep 17, 2008 at 5:23

bmdhacks
16.3k ● 8 ● 36 ● 55

0

I assume from the body of your question is you're a business user? I purchased a 6TB RAID 5 unit from Silicon Mechanics and have it NAS attached and my engineer installed NFS on our servers. Backups performed via rsync to another large capacity NAS.

Share Improve this answer Follow

answered Sep 17, 2008 at 6:13

user15094

0

Your best bet maybe to work with experts who do this sort of thing for a living. These guys are actually in our office complex...I've had a chance to work with them on a similar project I was lead on.

http://www.deltasquare.com/About

Share Improve this answer Follow

answered Sep 17, 2008 at 17:16

ben
3,136 ● 3 ● 38 ● 51

0

Have a look at Amazon Simple Storage Service (Amazon S3)

http://www.amazon.com/S3-AWS-home-page-Money/b/ref=sc_fe_l_2?ie=UTF8&node=16427261&no=3435361&me=A36L942TSJ2AJA

-- This may be of interest re. High Availability

Dear AWS Customer:

Many of you have asked us to let you know ahead of time about features and services that are currently under development so that you can better plan for how that functionality might integrate with your applications. To that end, we are excited to share some early details with you about a new offering we have under development here at AWS -- a content delivery service.

This new service will provide you a high performance method of distributing content to end users, giving your customers low latency and high data transfer rates when they access your objects. The initial release will help developers and businesses who need to deliver popular, publicly readable content over HTTP connections. Our goal is to create a content delivery service that:

Lets developers and businesses get started easily - there are no minimum fees and no commitments. You will only pay for what you actually use. Is simple and easy to use - a single, simple API call is all that is needed to get started delivering your content. Works seamlessly with Amazon S3 - this gives you durable storage for the original, definitive versions of your files while making the content delivery service easier to use. Has a global presence - we use a global network of edge locations on three continents to deliver your content from the most appropriate location.

You'll start by storing the original version of your objects in Amazon S3, making sure they are publicly readable. Then, you'll make a simple API call to register your bucket with the new content delivery service. This API call will return a new domain name for you to include in your web pages or application. When clients request an object using this domain name, they will be automatically routed to the nearest edge location for high performance delivery of your content. It's that simple.

We're currently working with a small group of private beta customers, and expect to have this service widely available before the end of the year. If you'd like to be notified when we launch, please let us know by clicking here.

Sincerely,

The Amazon Web Services Team

Share Improve this answer Follow

edited Sep 18, 2008 at 14:29

answered Sep 17, 2008 at 6:23

pro

963 ● 2 ● 9 ● 24

1 S3 does not have particularly fantastic availability. It is great in many ways, but does not fit the "high availability" requirement the OP is asking for. – Stu Thompson Sep 17, 2008 at 7:02

0

May I suggest you visit the F5 site and check out http://www.f5.com/solutions/virtualization/file/

Share Improve this answer Follow

answered Apr 18, 2009 at 21:40

McGovernTheory

6,654 ● 5 ● 42 ● 75

You can look at Mirror File System. It does the file replication on file system level. The same file on both

0

primary and backup systems are live file.

http://www.linux-ha.org/RelatedTechnologies/Filesystems

Share Improve this answer

Follow

answered Jan 30, 2010 at 8:01

fish.ada94

1