

Ad Scan your Web-Server for Malware with ISPProtect now. Get Free Trial.

Replacing A Failed Hard Drive In A Software RAID1 Array. This guide shows how to remove a failed hard drive from a Linux RAID1 array (software RAID), and how to add a new hard disk to the RAID1 array without losing data.

NOTE: There is a new version of this tutorial available that uses `gdisk` instead of `sfdisk` to support [GPT partitions](#).

On this page

- [1 Preliminary Note](#)
- [2 How Do I Tell If A Hard Disk Has Failed?](#)
- [3 Removing The Failed Disk](#)
- [4 Adding The New Hard Disk](#)

1 Preliminary Note

In this example I have two hard drives, `/dev/sda` and `/dev/sdb`, with the partitions `/dev/sda1` and `/dev/sda2` as well as `/dev/sdb1` and `/dev/sdb2`.

`/dev/sda1` and `/dev/sdb1` make up the RAID1 array `/dev/md0`.

`/dev/sda2` and `/dev/sdb2` make up the RAID1 array `/dev/md1`.

`/dev/sda1 + /dev/sdb1 = /dev/md0`

`/dev/sda2 + /dev/sdb2 = /dev/md1`

`/dev/sdb` has failed, and we want to replace it.

2 How Do I Tell If A Hard Disk Has Failed?

If a disk has failed, you will probably find a lot of error messages in the log files, e.g. `/var/log/messages` or `/var/log/syslog`.

You can also run

```
cat /proc/mdstat
```

and instead of the string `[UU]` you will see `[U_]` if you have a degraded RAID1 array.

3 Removing The Failed Disk

To remove `/dev/sdb`, we will mark `/dev/sdb1` and `/dev/sdb2` as failed and remove them from their respective RAID arrays (`/dev/md0` and `/dev/md1`).

First we mark `/dev/sdb1` as failed:

should look like this:

```
server1:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid5] [raid4] [raid6] [raid10]
md0 : active raid1 sda1[0] sdb1[2] (F)
      24418688 blocks [2/1] [U_]

md1 : active raid1 sda2[0] sdb2[1]
      24418688 blocks [2/2] [UU]

unused devices: <none>
```

Then we remove `/dev/sdb1` from `/dev/md0`:

```
mdadm --manage /dev/md0 --remove /dev/sdb1
```

The output should be like this:

```
server1:~# mdadm --manage /dev/md0 --remove /dev/sdb1
mdadm: hot removed /dev/sdb1
```

And

```
cat /proc/mdstat
```

should show this:

```
server1:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid5] [raid4] [raid6] [raid10]
md0 : active raid1 sda1[0]
      24418688 blocks [2/1] [U_]

md1 : active raid1 sda2[0] sdb2[1]
      24418688 blocks [2/2] [UU]

unused devices: <none>
```

Now we do the same steps again for `/dev/sdb2` (which is part of `/dev/md1`):

```
mdadm --manage /dev/md1 --fail /dev/sdb2
```

```
cat /proc/mdstat
```

```
server1:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid5] [raid4] [raid6] [raid10]
md0 : active raid1 sda1[0]
      24418688 blocks [2/1] [U_]

md1 : active raid1 sda2[0] sdb2[2] (F)
      24418688 blocks [2/1] [U_]

unused devices: <none>
```

```
mdadm --manage /dev/md1 --remove /dev/sdb2
```

```
server1:~# mdadm --manage /dev/md1 --remove /dev/sdb2
mdadm: hot removed /dev/sdb2
```

```
cat /proc/mdstat
```

```
24418688 blocks [2/1] [U_]
unused devices: <none>
```

Then power down the system:

```
shutdown -h now
```

and replace the old `/dev/sdb` hard drive with a new one (**it must have at least the same size as the old one - if it's only a few MB smaller than the old one then rebuilding the arrays will fail**).

4 Adding The New Hard Disk

After you have changed the hard disk `/dev/sdb`, boot the system.

The first thing we must do now is to create the exact same partitioning as on `/dev/sda`. We can do this with one simple command:

```
sfdisk -d /dev/sda | sfdisk /dev/sdb
```

You can run

```
fdisk -l
```

to check if both hard drives have the same partitioning now.

Next we add `/dev/sdb1` to `/dev/md0` and `/dev/sdb2` to `/dev/md1`:

```
mdadm --manage /dev/md0 --add /dev/sdb1
```

```
server1:~# mdadm --manage /dev/md0 --add /dev/sdb1
mdadm: re-added /dev/sdb1
```

```
mdadm --manage /dev/md1 --add /dev/sdb2
```

```
server1:~# mdadm --manage /dev/md1 --add /dev/sdb2
mdadm: re-added /dev/sdb2
```

Now both arrays (`/dev/md0` and `/dev/md1`) will be synchronized. Run

```
cat /proc/mdstat
```

to see when it's finished.

During the synchronization the output will look like this:

```
server1:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid5] [raid4] [raid6] [raid10]
md0 : active raid1 sda1[0] sdb1[1]
      24418688 blocks [2/1] [U_]
      [=>.....] recovery = 9.9% (2423168/24418688) finish=2.8min speed=127535K/sec

md1 : active raid1 sda2[0] sdb2[1]
      24418688 blocks [2/1] [U_]
      [=>.....] recovery = 6.4% (1572096/24418688) finish=1.9min speed=196512K/sec
```

```
Personalities : [linear] [multipath] [raid0] [raid1] [raid5] [raid4] [raid6] [raid10]
md0 : active raid1 sda1[0] sdb1[1]
      24418688 blocks [2/2] [UU]

md1 : active raid1 sda2[0] sdb2[1]
      24418688 blocks [2/2] [UU]

unused devices: <none>
```

That's it, you have successfully replaced `/dev/sdb`!

 [view as pdf](#) |  [print](#)

Share this page:

[Tweet](#)

[Follow @howtoforgecom](#)

28.9K followers

[Recommend 12](#)





Suggested articles


79 Comment(s)

Add comment

Name *

Email *



B *I* 

p

Comments

From:

[Reply](#)

Thanks for the great article. This seems to be the best case scenario for a drive failure in a mirrored RAID array (i.e. drive 2 failing in a 2 drive mirror).

Perhaps a useful addition to the article would be to detail how to recover when the first drive (e.g. /dev/sda in this article) fails. Physically removing /dev/sda would allow the system to run from /dev/sdb (so long as the boot loader was installed on /dev/sdb!), but if you put a new HD in /dev/sda, I don't think you would be able to reboot...

You would probably need to remove /dev/sda, then move /dev/sdb to /dev/sda, and then install a new /dev/sdb.

From: Ben F

[Reply](#)

Just to add - I've just had a 2TB sda disk fail which was part of a RAID 1 mirror to sdb.

The disks were connected to a AMD SB710 controller and the server was running Centos 5.7

I did have problems getting the system to boot from sdb (fixed by re-installing grub to sdb) but I'd thought I'd report I was able to successfully disconnect the failed sda and hot-plug the new drive in, with it showing up as a 'blank' disk with fdisk -l.

Copying the partition table from sdb to sda (fdisk as above plus using the --force as noted due to Centos) I could then add back in the partitions to the different arrays as detailed in the and watch the disks rebuild. The four 2TB disk RAID5 array took around 6 hours to rebuild.

Have to also got to say, this is an excellent how-to.

From: Anonymous

[Reply](#)

Hi, I followed exactly same steps as of your. But I got some surprise

Before adding the disk I just did fdisk this was the output

```
root@host ~]# fdisk -l
```

Disk /dev/sda: 1000.2 GB, 1000204886016 bytes

255 heads, 63 sectors/track, 121601 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	65	522081	fd	Linux raid autodetect
/dev/sda2		66	121601	976237920	fd	Linux raid autodetect

Disk /dev/sdb: 1000.2 GB, 1000204886016 bytes

255 heads, 63 sectors/track, 121601 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1	*	1	65	522081	fd	Linux raid autodetect
/dev/sdb2		66	121601	976237920	fd	Linux raid autodetect

Disk /dev/md1: 999.6 GB, 999667531776 bytes

2 heads, 4 sectors/track, 244059456 cylinders

Units = cylinders of 8 * 512 = 4096 bytes

Disk /dev/md1 doesn't contain a valid partition table

Disk /dev/md0: 534 MB, 534511616 bytes

2 heads, 4 sectors/track, 130496 cylinders

Units = cylinders of 8 * 512 = 4096 bytes

Disk /dev/md0 doesn't contain a valid partition table

=====

But when I tried to add the sda1 into my md0 raid it went perfect but when I tried to add sda2 into md1 it failed telling that no such device found. And when I did fdisk -l again I saw

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1	*	1	65	522081	fd	Linux raid autodetect
/dev/sdb2		66	121601	976237920	fd	Linux raid autodetect

Disk /dev/md1: 999.6 GB, 999667531776 bytes
 2 heads, 4 sectors/track, 244059456 cylinders
 Units = cylinders of 8 * 512 = 4096 bytes

Disk /dev/md1 doesn't contain a valid partition table

Disk /dev/md0: 534 MB, 534511616 bytes
 2 heads, 4 sectors/track, 130496 cylinders
 Units = cylinders of 8 * 512 = 4096 bytes

Disk /dev/md0 doesn't contain a valid partition table

Disk /dev/sdc: 1000.2 GB, 1000204886016 bytes
 255 heads, 63 sectors/track, 121601 cylinders
 Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1	*	1	65	522081	fd	Linux raid autodetect
/dev/sdc2		66	121601	976237920	fd	Linux raid autodetect

You have new mail in /var/spool/mail/root
 =====

Suprisingly linux detected the new drive suddenly as sdc1. And now if I want to delete the sda1 from md0 so that I could add sdc1 its not allowing me saying sda1 no such device. Please help...
 Dmesh at below pastebin
<http://fpaste.org/qwdh/>

From: hash

[Reply](#)

you should copy partition table in a file then reboot it

From:

[Reply](#)

Hello there,

i m missing the part for the bootloader (lilo/grub).
 maybe you can add it?

a part for replacing the first disk (as said by the previous poster) would be good, also a part if the bootloader was not added to the bootloader (rescue-disc, chrooting, ...)

regards,
 som-a

From:

[Reply](#)

Hi,
 This link worked well for me: <http://lists.us.dell.com/pipermail/linux-poweredge/2003-July/008898.html>
 Regards
 Rikard

From:

[Reply](#)

This is something that should be added to the howto. On debian it is simply a matter of running "grub-install /dev/sdb".
 I'm sure this was just an oversight on part of the author as otherwise Falko Timmes RAID howtos have been very correct and god send.
 Keep up the good work!

This type of tutorial is invaluable. The man page for 'mdadm' is over 1200 lines long and it can be easy for the uninitiated to get lost. My only question when working through the tutorial was is it necessary to --fail all of the remaining partitions on a disk in order to remove them from the array (in preparation to replace the disk)? The answer is 'yes', easily found in the man page once I knew the option existed. One of the follow-up comments included a link to a post from the Linux-PowerEdge mailing list entitled 'Software Raid and Grub HOW-TO' (yes, 'software' is misspelled in the post's title). Although this paper is dated 2003 and the author refers to 'raidtools' instead of 'mdadm', there are two very useful sections. The most useful is on using grub to install the master boot record to the second drive in the array. The other useful section is on saving the partition table, and using this to build a new drive. (In my own notes this I add saving the drive's serial number so I have a unambiguous confirmation of what device maps to what physical drive.) Merging these tips to Falco's instructions gave me a system bootable from either drive, and easily rebuilt when I replaced a 'failed' drive with a brand-new unpartitioned hard drive. Thanks to Falko and the other helpful posters.

From: Stephen Jones

[Reply](#)

Class tutorial - just repaired a failed drive remotely (with a colleagues assistance at the location) flawlessly - hope its as easy if sda falls over
.....

From: Kris

[Reply](#)

Thanks for the step-by-step guide to replacing a failed disk, this went much smoother than I was expecting - Now I just have to sit and wait 2.5 hours for the array to rebuild itself...

Thanks again!

From: Paul Bruner

[Reply](#)

I think the author needs to put in how to find the physical drive though. Every time my server reboots it seems to put the drives in different dev nodes. (ex, sdb1 is now sda1, and so on)
Not everyone can dig through the commands for that:P

From: Benjamin

[Reply](#)

Regarding how to find the failed drive....

I believe that the (F) will be beside the failed drive when you cat /proc/mdstat.

(But I'm not 100% certain)

However, you don't need to know the letter of the drive to remove it.

for example: mdadm --manage /dev/md0 --remove failed

Will remove the failed drive. Comparing /proc/mdstat from before and after will confirm the drive that failed. If you're still not sure which drive to physically remove, run a badblocks scan on the drive that was removed. It will go to 100% activity -- watch for the pretty lights... :)

From: Gary Dale

[Reply](#)

The question refers, I believe, to the physical drive to be replaced. Unfortunately with SATA it's not always easy to determine which drive is the faulty one. Unlike the IDE drives, the drive assignments don't come from the cable.

Even the hot-swap drive cages don't usually give you individual lights for the different drives. Pulling the wrong one with a degraded array will probably cause your computer to lock up.

If you can shut down your computer, you can disconnect the SATA cables one by one to see which one still allows the MD array to start.

If you can't shut down your computer, you may have to dig out the motherboard manual to see which SATA ports are which then hope they line up with drive letters (i.e. SATA 0 <--> /dev/sda, SATA 1 <--> /dev/sdb, etc.). This may not work.

If you can leave the non-failed arrays running, and if you have a hot swap setup, you may be able to get away with pulling drives until you find the right one. For RAID 1 you have a 50% chance of getting the right one.

If you have a hot-spare, you can rebuild the array before doing this. This works even with RAID 1. You can have two redundant disks in a RAID 1 array, for example, so losing one means you can still pull drives without killing the array.

If you have a hot-swap cage or can shut down the machine, I recommend adding the new drive and rebuilding the array before trying to remove the defective drive. This can be done with any array type. It just requires having enough SATA connections.

From: the_rusted_one

[Reply](#)

Actually, it is much easier than all that. Here's what you do (PREFERABLY BEFORE the failure event, but still this is useful):
in bash (or your favourite shell - just translate my bash-isms below):

hopefully you can see enough of the drive to get enough of the label to find a matching 'number'.

Now, when you are all done, be sure to label the locations where you put your drives so that this doesn't happen to you next time!

2 side notes:

1 - "ME TOO" - I am moderately experienced, but this was the simplest reminder of what I needed to do to recover my degraded RAID. Thanks!

2 - In my case, when I shut down, removed the failing drive (and did the above mapping and labelling!), and then rebooted - the removes had been done for me 'automagically'.

3 - (yes, i was a math major, why do you ask???) Also in my case:

a - the drives are all bought at different times, and/or are from different manufacturers.

b - the boot drive is NOT RAIDed. If I care much about rebuilding the / partition, then I will make an image (dd image) of the system for ease of restoration. However, this makes the 'what happens when your boot partition goes bad' a different issue ;-).

Anyway, thanks to Falko Timme for a great and to-the-point howto.

From: bobbyjimmy

[Reply](#)

Thanks - This worked perfectly against my raid5 as well.

From: mpy

[Reply](#)

Thank you very much for this tutorial... especially the sfdisk trick is really clever!

I only have one comment: Perhaps it'll be smarter to wait with the re-addition of /dev/sdb2 until sdb1 is sync'd completely. Then the load of the HDD (writing to two partitions simultaneously) will be reduced.

From: ttr

[Reply](#)

Nope, if there are multiple arrays on one drive to be sync, they will be queued and syncing will be done one-by one, so there is no need to wait with adding other partitions.

From: Anonymous

[Reply](#)

Interesting... thanks for clarifying this. It was just a thought, as in the example above it looks like the sync'ing is done simultaneously (md0 at 9.9% and md1 at 6.4%).

From: dino

[Reply](#)

Very helpful, thanks.

Any advice on a /dev/sda master mirror disk failure? I'm having some difficulty tracking anything down about this on the Internet. All information seems to refer to a slave disk failure /dev/sdb.

Cheers and thanks.

From: pupu

[Reply](#)

I can add the procedure I've just used to replace failed /dev/sda on my Fedora system. I'm assuming you have your bootloader in MBR; if not, adjust arguments at point 7 and 8 1. After you have finished the procedure described in the article, boot from rescue cd/dvd/usb stick/whatever 2. Let the rescue procedure process to the point you are offered shell 3. Check for the location of your '/boot' directory on physical disks. Mine was on /dev/sda3 or /dev/sdb3; it means (hd0,2) or (hd1,2) in grub syntax (check grub docs if you are not sure) 4. run 'chroot /mnt/sysimage' 5. run 'grub' 6. At grub prompt, type 'root (hd0,2)' when the argument is the path you've found at the point 3 7. type 'install (hd0)' 8. type 'install (hd1)' 9. leave grub shell, leave chroot, leave rescue shell and reboot

From: Anonymous

[Reply](#)

The computer hard drives have become a short-board, then the hard drive performance is really not be able to enhance through other means? The answer is no, in fact, short-board hard disk RAID technology can be compensated for before the RAID technology has been used in high-performance servers, etc. However, as the popularity of an integrated RAID controller board, this technology can be used in our daily life .here is my blog about [What is the difference between RAID 0 and RAID 5E](#)
[How to achieve drives raid](#) [How to set up RAID drives and enhance hard disk performance](#)

If your controller supports hot-swapping, then the reboot is NOT required.

You'll run `rescan-scsi-bus.sh` after replacing the drive, then proceed with creating / setting the partition type on the new drive. (assuming you're using partitions, and not just adding the device directly to the array)

From: Anonymous

[Reply](#)

Hello,

im also interested with hot-swap,

1. do you mean download the <http://rescan-scsi-bus.sh> to server and run it ?
2. how can i make sure if my board support hot-swap ? if yes,should i enable any option under my bios ?

thank you

From: scsi hot swap

[Reply](#)

Wow.

This was just the article I needed after one of my disks failed and I had to get the array back up and running. Linux is an amazing OS, but when you start to run mission critical services on there and don't employ or train people to support it properly, it is pages like this that are a big BIG help.

Thanks again.

From: ObiDrunk

[Reply](#)

first, ty, this its a very complete tutorial, cost a lot find info like this on the web.

i have a question, i have a Raid 1 by software, same cfg that you, the md0 its the swap partition and the md1 its the / when i first start, after the instalation i run on a shell
`watch -n1 cat /proc/mdstat`

and the md1 appears to be on sync status, this its normal? can i reboot while the sync its on?, ty

From: Roger K.

[Reply](#)

I had a failed drive in a 4 disk RAID-5 array under Linux. Your instructions made it quick and painless to replace the drive and not lose any data. The array is rebuilding at this moment. THANK YOU SIR!

-- Roger

From: Anonymous

[Reply](#)

Great guide!

I have just had to do exactly this, worked like a charm. Very satisfying to be able to replace a failed hard drive with less than half an hours down time. This guide made good sense and I was able to proceed confident I understood what I was doing. Disaster averted!

From: Mark Copper

[Reply](#)

Worked for me, too. A couple of gotchas in my case (using lilo and sata drives, failed device sda): lilo must be patched and drive must be ejected in order for machine to be re-boot-able with degraded array.

Thanks for the guide.

From: solo

[Reply](#)

Excellent guide! Worked like a charm, thanx!

```
root@rescue ~ # cat /proc/mdstat
```

```
Personalities : [raid1]
```

```
md1 : active raid1 sdb2[2] sda2[0]
```

```
291981248 blocks [2/1] [U_]
```

```
[===>.....] recovery = 20.4% (59766080/291981248) finish=67.7min speed=57086K/sec
```

:-)

From: Fierman

[Reply](#)

On new hard drivers with 4k sector size instead of 512b sfdisk cannot copy partition table because it internally uses cylinders instead of sectors. It says:

```
sfdisk: ERROR: sector 0 does not have an msdos signature
/dev/sdb: unrecognized partition table type
Old situation:
No partitions found
Warning: given size (3898640560) exceeds max allowable size (3898635457)
```

sfdisk: bad input

Is there a way to copy partition table using another tool? Don't want to create it by hand ;)

From:

[Reply](#)

Hi, If you look at this tutorial, which is newer, you can use the "--force" switch: <http://www.howtoforge.com/how-to-set-up-software-raid1-on-a-running-system-incl-grub2-configuration-ubuntu-10.04-p4>
sfdisk -d /dev/sda | sfdisk --force /dev/sdb
It also suggests this at the command line. Hope that helps

From: Kris

[Reply](#)

This is a great guide but unfortunately, I could not apply it to my failed Raid-1 situation. Please forgive me for asking for help in here but I could not find section in the forum that talks about the Raid-1 failed disks in such details.

My system was originally setup with two identical Seagate 1TB drives and partitioned as follow:

```
/dev/md0    /boot      Raid-1
/dev/md2/    /           Raid-1
/dev/md3/    /var/data   Raid-1
```

Here is the output from the mdstat command that I ran from bootable BT4 CD as I was not able to boot the actual system that was configured as a Raid-1:

```
# cat /proc/mdstat
Personalities : [linear] [raid0] [raid1] [raid10] [raid6] [raid5] [raid4] [multipath]
md3 : active raid1 sdb2[1]
      870040128 blocks [2/1] [_U]

md2 : active raid1 sdb3[1]
      102398208 blocks [2/1] [_U]
md0 : active raid1 sda1[0]
      104320 blocks [2/1] [U_]
unused devices: <none>
```

Does this mean that both drives have failed? At this point, I do not care if I rebuild or fix the Raid-1 but at least I would like to recover my data that is stored on md3. How do I proceed? Any help will be greatly appreciated. Thank you.

Kris

From: 3rensho

[Reply](#)

THANKS!!! Just did it and your steps worked perfectly.

From: Jeremy Rayman

[Reply](#)

These instructions worked well. Some people may be concerned by a message at this step:

```
sfdisk -d /dev/sda | sfdisk /dev/sdb
sfdisk may stop on some systems and refuse to clone the partition table, saying:
"Warning: extended partition does not start at a cylinder boundary.
DOS and Linux will interpret the contents differently.
[...snip...]
Warning: partition 1 does not end at a cylinder boundary
```

sfdisk: I don't like these partitions - nothing changed.
(If you really want this, use the --force option.)"

This message about not ending at a cylinder boundary is something Linux users don't need to worry about. See the explanation here:

After some googling, we concluded that having to align the partition boundaries with the cylinders was a DOS legacy issue, and was not something that would cause a problem for Linux.

So to copy the partitions from the working disk to the new disk we used the following:"

```
sfdisk -d /dev/sda | sfdisk --Linux /dev/sdb
```

Using the --Linux switch made it go ahead and clone the partition table. This likely gives the same end result as using --force, but people may prefer to use --Linux instead.

From: Peter

[Reply](#)

Excellent description which also works pretty good on a 4 disk software RAID 10. 5 stars! Greetz Peter

From: Anonymous

[Reply](#)

I am having the same issue now. My dev/sdb is going bad so I need to replace it. I have the previously used hard drive with the same size.

(1). Do I need to format it before put it on the linux server? If so, what are the steps I should take to format it on my windows machine before take it to the data center? (2). Some reason this Western Digital HD shows 74GB when I attempted to format it on my windows machine but the actual size is 80GB though. Any advice? Thanks

From: mike

[Reply](#)

hello,it is great,but will the MBR also been copy to the new hd ? it mean the new single hd can boot by itself.thank you.

From: Yago_bg

[Reply](#)

Grate article. Exactly what I needed the array is rebuilding right now. Fingers crossed
Thanks

From: Dr. Matthew R Roller

[Reply](#)

Thank you so much! I have used your guide twice now with great results, once for a failed hard drive, and once because I took one drive out and booted it on another identical computer then when I put it back in it didn't know what to do with it.

From: Rodger

[Reply](#)

Thanks for the information, though once the drive has failed, all data will be lost, so I guess this is more of a consolation, for me.

From: Ruslan

[Reply](#)

Thanks for good instruction! It works!

```
md3 : active raid1 sda4[2] sdd4[1]
      250042236 blocks super 1.1 [2/1] [_U]
      resync=DELAYED
      bitmap: 2/2 pages [8KB], 65536KB chunk
md2 : active raid1 sda1[2] sdd1[1]
      31456188 blocks super 1.1 [2/1] [_U]
      [==>.....] recovery = 14.6% (4614400/31456188) finish=37.4min speed=11936K/sec
md1 : active raid1 sda2[2] sdd2[1]
      10484668 blocks super 1.1 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk
md0 : active raid1 sda3[2] sdd3[1]
      1048564 blocks super 1.0 [2/2] [UU]
```

From: Brian J. Murrell

[Reply](#)

It seems to me that first adding the new disk, waiting for the resync to complete and then going through the fail/remove steps is safer since you now have an array with multiple devices in it should you mess up your removal steps somehow.

Of course, this depends on being able to install the new disk before having to remove the failed one.

From: Jason H

[Reply](#)

This is easily one of the best tutorials written. I really hope you are getting paid well at your job! If you are doing stuff like this to be helpful to the masses, I can't imagine what you are like at work. Thanks again- J

if a sector is marked as bad it can be relocated). If it happens a lot, then you would be wise to look in to it (and always wise to have backups). Actually, keeping track of your disks is always a good idea. As for the bad sectors point: smartmontools will show that and has other tests, too. I guess what I'm saying is it depends (for logs). The other part below may add to confusion to some about disk dying, disk dead, versus an array issue itself. So, the article says this: "You can also run `cat /proc/mdstat` and instead of the string [UU] you will see [U_] if you have a degraded RAID1 array." Yes, that's right that it means degraded. But that does NOT mean (by any stretch of the imagination) that disk is failing or failed. Consider a disk being disconnected temporarily (and booted before reconnected), a configuration problem. There's other possibilities. Actually, mdadm itself has the possibility to specify 'missing' for a disk, upon creation (and it would show a _ in its place in `/proc/mdstat`). So while it's true that it could be an issue, a degraded array does not necessarily equate to a dead disk as such. It might but it might not. Why do I mention that? Simply because having to get a new disk is not fun and I've also seen arrays degraded and the disks are fine. And believe me, when I say it's not fun, I will go further and say I've had more than one disk die at the same time. Similar has happened to a good friend of mine. I know you're not writing about backups but I'll say it anyway: Disk arrays is not a backup solution. Will repeat that: it is NOT a backup solution. If you remove a file by error or purposefully from the array and there's no backup, then what? Similar, what if all disks die at the same time (like what I mentioned above)?

From: James

[Reply](#)

To confirm which physical drive failed, try

```
sudo hdparm -I /dev/sdb
```

(Which may give you the serial number of the drive and remove the confusion as to which device is which drive.)

From: M@

[Reply](#)

Thanks! Exactly what I needed to add to my toolbox. Well written, easy to follow (--force/--Linux was obvious enough at prompt, only noticed it was in comments after).

Tested procedure in VMware Workstation8 CentOS6.x-64 guest, 2x10GB vmdk (md0 /boot, md1 swap, md2 /tmp, md3 /). Removed 1 vmdk, reboot, verified only sda, shutdown, added 1 new 10GB vmdk, duplicated partitions, verified partitions, rebuilt array, perfect.

Next: to add converting existing single disk install to RAID1 array.

From: Anonymous

[Reply](#)

Excellent tutorial for recovering a failed drive of a cross partition Raid-1 array.

To get a refreshing status of the rebuild process you can optionally use

```
watch cat /proc/mdstat
```

Which will periodically refresh the `cat /proc/mdstat` so you don't need to.

```
server1:~# mdadm --manage /dev/md3 --fail /dev/sdd1
mdadm: set /dev/sdd1 faulty in /dev/md3
```

```
server1:~# mdadm --manage /dev/md3 --remove /dev/sdd1
mdadm: hot removed /dev/sdd1
```

```
server1:~# mdadm --manage /dev/md3 --add /dev/sdd1
mdadm: re-added /dev/sdd1
```

```
server1:~# watch cat /proc/mdstat
Every 2.0s: cat /proc/mdstat
Personalities : [raid1]
md3 : active raid1 sdd1[2] sdb1[1]
      976759936 blocks [2/1] [_U]
      [=>.....] recovery = 8.6% (84480768/976759936) finish=337.0min speed=44121K/sec
```

```
md0 : active raid1 sdc1[1] sda1[0]
      256896 blocks [2/2] [UU]
```

```
md1 : active raid1 sdc2[1] sda2[0]
      2048192 blocks [2/2] [UU]
```

```
md2 : active raid1 sdc3[1] sda3[0]
      122728448 blocks [2/2] [UU]
```

From: Guido A

[Reply](#)

Excelente Howto, thank you very much. It was very useful to me.

Just one thing I would add, when you explain how to copy the partition table, I would make a BIG note stating that one should have care on what drive is being used as the source and which one as the destination. An error on this could cause big problems, I guess.

Thanks again!

From: Anonymous

[Reply](#)

From: Martin

[Reply](#)

What a relief !!! this was exactly the piece of information I was missing. I could not get the exact same make and model for my replacement HD. Nevertheless the disks are of exact the same size and geometry. I partitioned the new one with gdisk but could not add it to the array. This is how it looked like in fdisk: # fdisk -l /dev/sda GNU Fdisk 1.2.5 Copyright (C) 1998 - 2006 Free Software Foundation, Inc. This program is free software, covered by the GNU General Public License. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. Disk /dev/sda: 2000 GB, 2000396321280 bytes 255 heads, 63 sectors/track, 243201 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Device Boot Start End Blocks Id System /dev/sda1 1 81092 651371458 83 Linux Warning: Partition 1 does not end on cylinder boundary. /dev/sda2 81092 243202 1302148575 83 Linux Warning: Partition 2 does not end on cylinder boundary. and# fdisk -l /dev/sdb GNU Fdisk 1.2.5 Copyright (C) 1998 - 2006 Free Software Foundation, Inc. This program is free software, covered by the GNU General Public License. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. Disk /dev/sdb: 2000 GB, 2000396321280 bytes 255 heads, 63 sectors/track, 243201 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Device Boot Start End Blocks Id System /dev/sdb1 1 81092 651371458 83 Linux Warning: Partition 1 does not end on cylinder boundary. /dev/sdb2 81092 243202 1302148575 83 Linux Warning: Partition 2 does not end on cylinder boundary. I could find no differences. BUT # mdadm --manage /dev/md/Home --add /dev/sda1 mdadm: /dev/sda1 not large enough to join array CONFUSION! I found out, that during the partitioning I got other block sizes for the new drive: # blockdev --report R0 RA SSZ BSZ StartSec Size Device rw 256 512 4096 0 2000398934016 /dev/sdb rw 256 512 512 34 666999982592 /dev/sdb1 rw 256 512 4096 1302734375 1333398917120 /dev/sdb2 ... rw 256 512 4096 0 2000398934016 /dev/sda rw 256 512 1024 34 666996163584 /dev/sda1 rw 256 512 512 1302726916 1333402736128 /dev/sda2 So this seemed to be the cause of the trouble. The above hint saved me! now I have a running and currently syncing raid again. Thanks!

From: Anonymous

[Reply](#)

Hello,
Thank you for the good tutorial, I replace a disk which have bad sectors.
Then I have a question: Where can I get the program sgdisk?
I use debian (wheezy) and there gives nothing with the name sgdisk.

From: me

[Reply](#)

apt-cache search gdiskdisk - GPT fdisk text-mode partitioning tool

From: doyle

[Reply](#)

This is a great tutorial. Thank you.

From:

[Reply](#)

Thanks! work great on Ubuntu server 12.04 software RAID 1.

From: Anonymous

[Reply](#)

A great tutorial!
It might be a good idea to include the usage of mdadm with the --zero-superblock option, just like you do at your other great tutorial "*How To Set Up Software RAID1 On A Running System*":
To make sure that there are no remains from previous RAID installations on /dev/sdb, we run the following commands:
mdadm --zero-superblock /dev/sdb1
mdadm --zero-superblock /dev/sdb2
mdadm --zero-superblock /dev/sdb3

From: Anonymous

[Reply](#)

With newer Linux versions and the uncertainties of disk enumeration order, I recommend using `/dev/disk/by-id/disk-part-id` rather than `/dev/sdxy`.

From: Anonymous

[Reply](#)

Many Thanks
Works exactly to the point!

From: Taurus II

[Reply](#)

This was a great help but, I got an error whilst trying to copy the partitions table to the replacement drive.
`/dev/sdb: Permission denied sfdisk: cannot open /dev/sdb read-write Warning: extended partition does not start at a cylinder boundary. DOS and Linux will interpret the contents differently.`
The solution was
`sudo sfdisk -d /dev/sda | sudo sfdisk --force /dev/sdb`
Taurus II, Ubuntu 14.04.2 LTS

From: Dan

[Reply](#)

Thank you very much!

From: Thankful

[Reply](#)

Thanks for the article - I've used it around 10 times without issue - I should probably have memorised the steps by now!

From: Thomas

[Reply](#)

Gr8, just repaired a faulty drive in my server, ubuntu 14.x software RAID0
1. Bought a exact identical drive
2. use `cat /proc/mdstat` to see that md1 was on sda2 and md2 was on sda3
3. did all the above, but failed on `sfdisk -d /dev/sda | sfdisk /dev/sdb`
instead
`$apt-get install gdisk`
`$sgdisk --replicate=/dev/sdb /dev/sda` (NOTE! `--replicate=/dev/target /dev/source`)
4. Continued on the step to add sdb2 to md1 and sdb3 to md2 (use `cat /proc/mdstat` to see mapping)
Resync of disk started beautifully.
Know i'm going to clean my house while waiting for sync

From: guest

[Reply](#)

Excellent tutorial!

From: Stefan

[Reply](#)

Worked as charm. Thank you

From: kevin Fitzgerald

[Reply](#)

I have a failed hard drive sdb
However the readout from `cat /proc/mdstat` reads as follows:
`md127 : active raid1 sda[1] sdb[0] 976759808 blocks super external:/md0/0 [2/2] [UU] md0 : inactive sdb[1](S) sda[0](S) 5288 blocks super external:imsm`
Would I still use this Tutorial as is?
Appreciate your help

From: Jon Jarker

[Reply](#)

Thanks for the clear guide. You should add a caution note to first copy the master boot record and install grub on the new disk. You are using sda as the example of a failed drive.

Thank you for the article. It's very very well written and helped me alot. Keep up the good work.

From: Bob Gustafson

[Reply](#)

Excellent HowTo - thanks much. A slight addition to the original plus the excellent comments:

When you replace a raid component disk with a brand new disk, keep in mind that it is blank and has no boot information. If your Bios (which probably doesn't know about raid disk pairs) is pointing to the disk slot with the blank disk, you are not going to boot!

You have a 50/50 chance that your Bios is pointing to the blank disk. If so, then go into the Bios (after Ctrl-Alt-Del) and switch the disks boot order. This may be tricky because the labels for both disks are probably going to be identical in the Bios display. If you have to try a few times (the Bios on-screen help is helpful), no problem - you are not going to break anything. Have fun

From: Kurogane

[Reply](#)

What happen if i did not Removing The Failed Disk before

```
mdadm --manage /dev/mdx --fail /dev/sdbx
mdadm --manage /dev/mdx --remove /dev/sdbx
just power down the system and insert the new disk?
```

From: Anonymous

[Reply](#)

Thanks so much for putting this guide together. One of my RAID drives is dead to the point that it is not even recognisable by BIOS. I followed the instruction to rebuild the RAID skipping the one step of removing the faulty drive/partition from the RAID as the step was not needed in my case. It is working like a charm and the RAID is literally being rebuilt as we speak. Kudos to you!!

From: Jacques

[Reply](#)

awesome! worked exactly as described :-)

From: Lars

[Reply](#)

One of the best step-by-step tutorials to replace a failed hard drive, thanks!

One question: would you mind adding a note about the possibility to check/change the speed of the resync via the commands:

```
cat /proc/sys/dev/raid/speed_limit_min
cat /proc/sys/dev/raid/speed_limit_max
```

respectively:

```
echo 100000 > /proc/sys/dev/raid/speed_limit_min
echo 250000 > /proc/sys/dev/raid/speed_limit_max
```

The last two commands speed up the synchronization speed, but you have to check if this might conflict with the expected access speed of your array at all, as most of the available hard drive speed will be used to resync the raid array (and not for applications requesting data).

On a home server, the above settings make sense - but better check your users before you do it on a productive enterprise system.

From: Kevin

[Reply](#)

Hi, Thank you for this excellent tutorial. I tried to use it to help me resolving my problem but I still don't understand. I just want to add a new disk I have just bought to the array but I can't.

I have md126 and md127.

The volume I have created is called "raid". I can't understand why I can't add my sdb disk to the raid array:

```
mdadm --manage /dev/md/raid --add /dev/sdb1
```

```
mdadm: Cannot add disks to a 'member' array, perform this operation on the parent container
```

I'm under Ubuntu server Xenial with a 5-1TB disks RAID5 Array.

```
#####
fdisk -l
```

...

```
Périphérique Start      Fin    Secteurs  Size Type
/dev/sdb1    2048 1953523711 1953521664 931,5G RAID Linux
```

Disque /dev/sdc : 931,5 GiB, 1000204886016 octets, 1953525168 secteurs

Unités : sectors of 1 * 512 = 512 octets

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Unités : sectors of 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disque /dev/sdf : 931,5 GiB, 1000204886016 octets, 1953525168 secteurs
Unités : sectors of 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disque /dev/md126 : 2,7 TiB, 3000571527168 octets, 5860491264 secteurs
Unités : sectors of 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 131072 bytes / 393216 bytes

#####

mdadm --detail /dev/md/raid

/dev/md/raid:

Container : /dev/md/ims0, member 0

Raid Level : raid5

Array Size : 2930245632 (2794.50 GiB 3000.57 GB)

Used Dev Size : 976748672 (931.50 GiB 1000.19 GB)

Raid Devices : 4

Total Devices : 4

State : clean

Active Devices : 4

Working Devices : 4

Failed Devices : 0

Spare Devices : 0

Layout : left-asymmetric

Chunk Size : 128K

UUID : 2cf106c8:2d9d14c7:ebd9eb51:fcd2586d

Number	Major	Minor	RaidDevice	State
--------	-------	-------	------------	-------

3	8	32	0	active sync /dev/sdc
---	---	----	---	----------------------

2	8	48	1	active sync /dev/sdd
---	---	----	---	----------------------

1	8	64	2	active sync /dev/sde
---	---	----	---	----------------------

0	8	80	3	active sync /dev/sdf
---	---	----	---	----------------------

#####

mdadm --detail /dev/md126

/dev/md126:

Container : /dev/md/ims0, member 0

Raid Level : raid5

Array Size : 2930245632 (2794.50 GiB 3000.57 GB)

Used Dev Size : 976748672 (931.50 GiB 1000.19 GB)

Raid Devices : 4

Total Devices : 4

State : clean

Active Devices : 4

Working Devices : 4

Failed Devices : 0

Spare Devices : 0

Layout : left-asymmetric

Chunk Size : 128K

UUID : 2cf106c8:2d9d14c7:ebd9eb51:fcd2586d

Number	Major	Minor	RaidDevice	State
--------	-------	-------	------------	-------

Version : imsm
Raid Level : container
Total Devices : 5

Working Devices : 5

UUID : 2250d73c:cb29afea:75455eca:049ecb21
Member Arrays : /dev/md/raid

Number	Major	Minor	RaidDevice
0	8	32	- /dev/sdc
1	8	80	- /dev/sdf
2	8	64	- /dev/sde
3	8	48	- /dev/sdd
4	8	16	- /dev/sdb

#####

mdadm --detail-platform

Platform : Intel(R) Matrix Storage Manager

Version : 12.7.0.1936

RAID Levels : raid0 raid1 raid10 raid5

Chunk Sizes : 4k 8k 16k 32k 64k 128k

2TB volumes : supported

2TB disks : supported

Max Disks : 6

Max Volumes : 2 per array, 4 per controller

I/O Controller : /sys/devices/pci0000:00/0000:00:1f.2

(SATA)#####

cat /proc/mdstat

Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0] [raid1] [raid10]

md126 : active raid5 sdc[3] sdd[2] sde[1] sdf[0]

2930245632 blocks super external:/md127/0 level 5, 128k chunk, algorithm 0 [4/4] [UUUU]

md127 : inactive sdb[4](S) sdd[3](S) sde[2](S) sdf[1](S) sdc[0](S)

15765 blocks super external:imsm

unused devices: <none>

#####Thanks for your help.

From: Jeff

[Reply](#)

Great tutorial!

Thanks so much for writing it ????

From: Leandro Branco

[Reply](#)

Great tutorial. But in my case, when I add a new disk my array looks like this:root@ubuntu:/home/servidor# mdadm -D /dev/md0/dev/md0:
Version : 1.2 Creation Time : Tue Jun 10 19:00:15 2014 Raid Level : raid1 Array Size : 727406400 (693.71 GiB 744.86 GB) Used Dev Size :
727406400 (693.71 GiB 744.86 GB) Raid Devices : 2 Total Devices : 2 Persistence : Superblock is persistent Update Time : Thu Feb 23
18:46:20 2017 State : clean, degraded Active Devices : 1Working Devices : 2 Failed Devices : 0 Spare Devices : 1 Name : DELL-CS24-SC:0
UUID : 7307b163:e96d260a:52b16438:0262e681 Events : 156079 Number Major Minor RaidDevice State 0 0 0 0 removed 1 8 33 1 active
sync /dev/sdc1 2 8 17 - spare /dev/sdb1How can i put the disk as active and make my raid back up and running?can you help me? thank
you...

From: Jack Zimmermann

[Reply](#)

Thanks for the article. Saved my a**!

[Sign up now!](#)

Tutorial Info

Author: falko
Tags: [linux](#), [other](#), [storage](#)

Share This Page

Tweet

Recommend 12



