Software RAID on Red Hat Enterprise Linux® v6

Installation, Migration and Recovery

November 2010

Ashokan Vellimalai

Raghavendra Biligiri

Dell | Enterprise Operating Systems

November 2010

Contents

Introduction

Software RAID is RAID that is implemented at the software layer without the need for a dedicated hardware RAID controller on the system. Software RAID can be created on any storage block device independent of storage controllers. On Linux based operating system (OS), software RAID functionality is provided with the help of the md(4) (Multiple Device) driver and managed by the mdadm(8) utility. "md" and "mdadm" in RHEL 6 support RAID levels 0, 1, 4, 5, 6, and 10.

Some notable advantages in using Software RAID over hardware RAID are:

- Software RAID is controller Independent which makes it a cost-effective solution.
- The RAID solution can easily be migrated to any storage block device.
- The entire software stack runs on a host CPU, with modern multi-core CPUs, this ensures efficient CPU utilization.
- Software RAID provides a level of abstraction on underlying storage devices/technologies

This document attempts to provide step-by-step procedures that can be followed to:

- Set up software RAID in RHEL 6
- Migrate existing storage from Non-RAID to Software RAID
- Resize RAID volumes
- Recover from a broken RAID

This document uses RAID-1 as an example while working with Software RAID. The procedure can however be applied to other RAID types as applicable. Please consult the mdadm(8) man page for details on exact options for various RAID types.

We used a Dell PowerEdge™ R510 server with a Dell PERC H200 storage controller on the system for this procedure. All the storage volumes were exported directly to the OS without using any controller hardware RAID features.

Note: Ensure that all data backed up before performing any of these procedures

Setting up Software RAID in RHEL 6

Setup during Installation

The RHEL 6 installer (anaconda) has functionality that enables the OS to be installed on a software RAID partition. This section describes the steps to install RHEL 6 on a RAID-1 partition.

1. Start the RHEL 6 installer and follow the on-screen installation instructions and select the "Custom" layout for installation.
2. Create a minimum of two **partitions** to create **RAID-1 device** type and set the *File System Type* as software raid.
3. Create a **RAID-1 device** from RAID members created in Step2, select the filesystem and RAID-1 level

   Here is the *minimum* number of software RAID partitions required for each RAID level:

   - RAID 0,1,10 – 2 partitions
   - RAID 4,5 – 3 partitions
   - RAID 6 – 4 partitions

4. After creating all the necessary partitions (/boot, /, swap, etc.) on RAID-1 volume, proceed with the installation.

5. Once the installation is completed, the OS will boot successfully from the partitions on the RAID volume.

   Note: Ensure that the boot-loader is installed on the first disk and not on the RAID device. Installing boot-loader on the RAID device may result in failure to boot the OS after installation.

Setup after Installation

Software RAID volumes can be created on a running system post install as well. Ensure that the partition/s on which the OS is installed are not used for creating software RAID partitions, failure to do that may results in re-installing OS on the system.

Following section describes steps to create RAID-1 partition on the system.

1. Create the raid-1 md device using the `mdadm` command with /dev/sdb1 and /dev/sdc1. sdb1 and sdc1 are un-used partitions on this system

```
[root@Dell-PowerEdge-R510 ~]# mdadm --create /dev/md0 --level=1 --raid-
disks=2 /dev/sdb1 /dev/sdc1 --metadata=0.90
mdadm: array /dev/md0 started.
```

2. Create `ext4` filesystem layout on the md device and add array details `mdadm –-detail –scan` to `/etc/mdadm.conf` file. Mount the device `/dev/md0` on the system to use it.

```
[root@Dell-PowerEdge-R510 ~]# mkfs.ext4 /dev/md0
[root@Dell-PowerEdge-R510 ~]# mdadm --detail --scan >> /etc/mdadm.conf
[root@Dell-PowerEdge-R510 ~]# mount /dev/md0 /data/
```

3. Add a new entry in /etc/fstab file to auto mount the md raid partition, whenever system reboots.

```
[root@Dell-PowerEdge-R510 ~]# cat /etc/fstab
/dev/md0         /data                   ext3    defaults        1 1
```

Migration of Storage from Non-RAID to RAID Configurations

It is possible to migrate to software raid, the "/" partition without having to re-install the operating system if you installed RHEL 6 OS without software raid volumes. This section explains how migration of storage from non-raid to raid configuration can be achieved.

At a high level, here is how it can be achieved:

- Prepare the new storage volume
- Update fstab and grub configuration to boot from newly created storage volume
- Sync the data from the old partitions to the new storage volume
- Install the boot loader on new storage volume
- Add the old partition volume to the md raid-1 set

Prepare the new storage volume

1. Create the partition layout on the `/dev/sdb` volume similar to `/dev/sda`.

```
[root@Dell-PowerEdge-R510  ~]#  sfdisk  -d  /dev/sda  |  sfdisk  --force
/dev/sdb
Device Boot    Start       End   #sectors  Id  System
/dev/sdb1   *      2048  20482047   20480000  83  Linux
/dev/sdb2      20482048  24578047    4096000  82  Linux swap / Solaris

Warning: partition 1 does not end at a cylinder boundary
Successfully wrote the new partition table
```

2. Set the partition id of `/dev/sdb1` to Linux RAID

```
[root@Dell-PowerEdge-R510 ~]# sfdisk -c /dev/sdb 1 fd
Done
```

Create RAID-1 using `mdadm` utility:

1. Create the `raid-1 md` device using the mdadm command with `/dev/sdb1`. Mark the first volume as "missing", which will be sda volume, and it will be added later in the steps. Since sda has OS installed, we have to add this to raid array after copying the contents from sdb to sda drive.

```
[root@Dell-PowerEdge-R510 ~]# mdadm --create /dev/md0 --level=1 --raid-
disks=2 missing /dev/sdb1 --metadata=0.90
mdadm: array /dev/md0 started.
```

2. Create `ext4` filesystem layout on the `md device` and add the array details `mdadm –-detail –scan` to `/etc/mdadm.conf` file.

```
[root@Dell-PowerEdge-R510 ~]# mkfs.ext4 /dev/md0
[root@Dell-PowerEdge-R510 ~]# mdadm --detail --scan >> /etc/mdadm.conf
```

Update fstab and grub configuration to boot from newly created storage volume**:**

1. Modify the `/etc/fstab` and `/boot/grub/menu.lst` with `md device`.

```
[root@Dell-PowerEdge-R510 ~]# blkid | grep –i md0
/dev/md0: UUID="016db049-6802-4369-bf66-bd48aad15395" TYPE="ext4"

[root@Dell-PowerEdge-R510 ~]# cat /etc/fstab
#UUID=38b56dff-c6d2-434f-bb48-25efb97f3a58 /           ext4     defaults
1 1
UUID=016db049-6802-4369-bf66-bd48aad15395 /            ext4     defaults
1 1

[root@Dell-PowerEdge-R510 ~]# cat /boot/grub/menu.lst
```

2. Add sdb to the device map entry to install the grub on the sdb device.

```
[root@Dell-PowerEdge-R510 ~]# cat /boot/grub/device.map
# this device map was generated by anaconda
(hd0)     /dev/sda
(hd1)     /dev/sdb
```

Sync the data from the old partitions to the new storage volume**:**

1. Since we are trying to replicate the contents from currently running partition. It is recommended that you execute the following steps in run level1
2. Mount the array volume and copy the contents from sda1 to md0.

```
[root@Dell-PowerEdge-R510 ~]# mount /dev/md0 /mnt/
[root@Dell-PowerEdge-R510 ~]# rsync -aqxP / /mnt/
```

Install the boot loader on new storage volume**:**

1. Install the boot loader on the sdb device.

```
[root@Dell-PowerEdge-R510 ~]# grub-install /dev/sdb
Installation finished. No error reported.
This is the contents of the device map /boot/grub/device.map.
Check if this is correct or not. If any of the lines is incorrect,
fix it and re-run the script `grub-install'.

# this device map was generated by anaconda
(hd0)     /dev/sda
(hd1)     /dev/sdb
```

2. Reboot the system and verify the system has booted with the md device using the mount command.

Add the old partition volume to the md raid-1 set**:**

1. Change the partition of sda1 and add the sda1 device to md0 array and allow the resync to complete from sda1 to sdb1.

```
[root@Dell-PowerEdge-R510 ~]# sfdisk -c /dev/sda 1 fd
Done
[root@Dell-PowerEdge-R510 ~]# mdadm --add /dev/md0 /dev/sda1
mdadm: added /dev/sda1
[root@Dell-PowerEdge-R510 ~]# watch cat /proc/mdstat
```

2. Reboot the system to verify the md migration completed successfully.
3. Run the cat /proc /mdstat command to check the status of the running array.

Resizing an existing RAID Partition

The Linux software-RAID solution allows us to resize (increase or decrease) the RAID partition size. Following steps explain how to increase the size of existing software RAID partition (data and OS partitions).

Here is what is required:

- Prepare partitions of the new size desired
- Replace both RAID members with newly created partitions by breaking existing RAID
- Resize the RAID array
- If dealing with OS partitions, Prepare the new RAID volume to be bootable

Prepare partitions of the new size desired**:**

1. Initially create the `md0 raid-1` level with a size approximately 100 GB comprised of both the `sda1` and `sdb1` volumes. The `md-0` array RAID set will be increased to approximately 200 GB by using the `sdc1` and `sdd1` volumes.
2. Create the new RAID partition of increased size on `sdc` and `sdd`. In this example, we created a new RAID partition of approximately 200GB in size on the `sdc` volume.

```
[root@Dell-PowerEdge-R510 ~]# sfdisk -c /dev/sdc 1 fd
Done
[root@Dell-PowerEdge-R510 ~]# fdisk -l /dev/sdc

Disk /dev/sdc: 1000.2 GB, 1000204886016 bytes
255 heads, 63 sectors/track, 121601 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x4c0a9054

Device Boot      Start         End      Blocks   Id  System
/dev/sdc1      1       25000   200812468+  fd  Linux raid autodetect
```

Replace both RAID members with newly created partitions by breaking existing RAID**:**

1. Set the `sdb1` volume to faulty and remove the volume from the RAID set.

```
[root@Dell-PowerEdge-R510 ~]# mdadm /dev/md0 –-fail /dev/sdb1 –remove
/dev/sdb1
mdadm: set /dev/sdb1 faulty in /dev/md0
mdadm: hot removed /dev/sdb1 from /dev/md0
```

2. Add the new partition to the RAID set and allow the resync to complete on the new partition added to the RAID set. Run `cat /proc/mdstat` to show the status of resynchronization.

```
[root@Dell-PowerEdge-R510 ~]# mdadm --add /dev/md0 /dev/sdc1
mdadm: added /dev/sdc1

[root@Dell-PowerEdge-R510 ~]# cat /proc/mdstat

Personalities : [RAID-1]
md0 : active RAID-1 sdc1[2] sda1[0]
      102399928 blocks super 1.0 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk
```

```
unused devices: <none>
```

3. Repeat the above steps adding the /dev/sdd1 partition. Remove the /dev/sda1 partition from the RAID set. Allow the resynchronization to complete on the /dev/sdd1 partition.

```
[root@Dell-PowerEdge-R510 ~]# cat /proc/mdstat
Personalities : [RAID-1]
md0 : active RAID-1 sdd1[3] sdc1[2]
      102399928 blocks super 1.0 [2/1] [_U]
      [>.....................]    recovery  =    0.3%  (409472/102399928)
finish=20.7min speed=81894K/sec
      bitmap: 1/1 pages [4KB], 65536KB chunk

unused devices: <none>
```

Resize the RAID array:

1. Set the /dev/md0 partition size to use the new volume partition size and allow the resync to complete.

```
[root@Dell-PowerEdge-R510 ~]# mdadm --grow /dev/md0 --size=max
mdadm: Cannot set device size for /dev/md0: Device or resource busy
       Bitmap must be removed before size can be changed

[root@Dell-PowerEdge-R510 ~]# mdadm --grow /dev/md0 --bitmap none

[root@Dell-PowerEdge-R510 ~]# mdadm --grow /dev/md0 --size=max
mdadm: component size of /dev/md0 has been set to 200812396K
```

2. Resize the file system of the /dev/md0 partition to increase the file system size. Now the df –h command shows the increased md RAID set.

```
[root@Dell-PowerEdge-R510 ~]# resize2fs /dev/md0

resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/md0 is mounted on /; on-line resizing required
old desc_blocks = 7, new_desc_blocks = 12
Performing an on-line resize of /dev/md0 to 50203099 (4k) blocks.
The filesystem on /dev/md0 is now 50203099 blocks long.

[root@Dell-PowerEdge-R510 ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/md0              189G  1.4G  178G   1% /
tmpfs                 1.9G     0  1.9G   0% /dev/shm
```

If dealing with OS partitions, prepare the new RAID volume to be bootable**:**

Note: Following steps are required to boot the Linux system, if you are replacing the drive from RAID array which has boot loader and file system already installed.

Before resizing the RAID partition, ensure that the md-raid set has two active drives, sda1 and sdb1 and that grub is installed in /dev/sda. After the resizing operation, the md-raid set has been replaced with sdc1 and sdd1 volumes and boot loader has been installed on sdc volume.

1. Add the sdc and sdd entries in device.map file.

```
[root@Dell-PowerEdge-R510 ~]# cat /boot/grub/device.map
# this device map was generated by anaconda
(hd0)     /dev/sda
(hd1)     /dev/sdb
(hd2)     /dev/sdc
(hd3)     /dev/sdd
```

2. Install the `grub` boot loader on the `/dev/sdc` volume and then remove the older drives from the system.

```
[root@Dell-PowerEdge-R510 ~]# grub-install /dev/sdc
Installation finished. No error reported.
This is the contents of the device map /boot/grub/device.map.
Check if this is correct or not. If any of the lines is incorrect,
fix it and re-run the script `grub-install'.

# this device map was generated by anaconda
(hd0)     /dev/sda
(hd1)     /dev/sdb
(hd2)     /dev/sdc
(hd3)     /dev/sdd
```

3. Reboot the system and enter the storage controller BIOS configuration.
4. Change the boot drive to new drive where the boot loader is now installed.
5. Save the configuration and restart the system to boot from new RAID set partition.

Recovery from a Broken RAID

In case of RAID failures on a system running Linux Software RAID (md) solution (for example, media failure or disk driver failure), it is possible to recover the system by any of these methods

- replacing the faulty disk
- adding a new disk
- using the spare disk

In Linux software raid, recovery is achieved through "failover" mechanisms. Failover mechanism ensures data protection by providing additional drives (spares) and can be automatic or manual.

Automatic Failover

Linux md-raid solution has an intelligent monitor mechanism to detect hardware failure in RAID arrays. If any disk in the RAID array fails, the monitors sets the failed drive to faulty and starts using one of the available spare drives for regeneration. To check the status of the RAID array, look at /proc/mdstat file.

To replicate a raid failure to check how automatic failover scenario works, follow the steps.

1. Create a RAID-1 setup with three raid partition. Minimum two raid partitions are required to create RAID-1 device and third raid partition is used as a spare disk, will be used as a replacement if one of the active RAID partition fails.
2. Simply pull out one of the disks which are active in the raid array and check the status of the array using the `cat /proc/mdstat` command, will show the removed drive as faulty and use the spare drive as replacement for date re-generation.

Manual Failover

Faulty drives in the raid array can be replaced manually. Following steps discuss how to manually replace the faulty drive from the RAID-1 array:

1. Raid-1 setup is created with sda1 and sdb1 partitions. Set the sda1 drive to faulty.

```
[root@DELL-PowerEdge-R510 ~]# mdadm -f /dev/md0 /dev/sda1
mdadm: set /dev/sda1 faulty in /dev/md0
```

2. Remove the faulty drive sda1 from the array.

```
[root@DELL-PowerEdge-R510 ~]# mdadm -r /dev/md0 /dev/sda1
mdadm: hot removed /dev/sda1 from /dev/md0
```

3. Replace the faulty drive with adding a new one to the array.

```
[root@DELL-PowerEdge-R510 ~]# mdadm --add /dev/md0 /dev/sdc1
Device Added
```

4. Look at status of the RAID array by executing the `cat /proc/mdstat` command, showing `/dev/sdc1` added to the RAID array. Also check that the resynchronization is complete.

```
[root@DELL-PowerEdge-R510 ~]# cat /proc/mdstat
Personalities : [RAID-1]
md0 : active RAID-1 sdc1[2] sdb1[0]
      40958908 blocks super 1.1 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk
```

Adding a Spare Disk to the Array

Spare disks provide additional protection to a raid configuration. If a disk fails in a raid array, the spare disk automatically replaces the failed drive; also the raid can be rebuilt automatically in the background. Spare drives can be added to the RAID array during the time of creation of the array or later.

Adding spare disk during raid array creation

```
[root@Dell-PowerEdge-R510 ~]# mdadm --create /dev/md0 --level=1 --raid-
disks=2 /dev/sdb1 /dev/sdc1 --metadata=0.90 –-spare-devices=1 /dev/sdd1
mdadm: array /dev/md0 started.
```

Adding spare disk to an existing array.

```
[root@DELL-PowerEdge-R510 ~]# mdadm --add /dev/md0 /dev/sdc1
mdadm: added /dev/sdc1

[root@DELL-Poweredge-R510 ~]# cat /proc/mdstat
Personalities : [RAID-1]
md1 : active RAID-1 sdc1[2](S) sda1[0] sdb1[1]
```

References

- http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/index.html
- http://www.spinics.net/lists/raid/
- https://raid.wiki.kernel.org/index.php/Linux_Raid