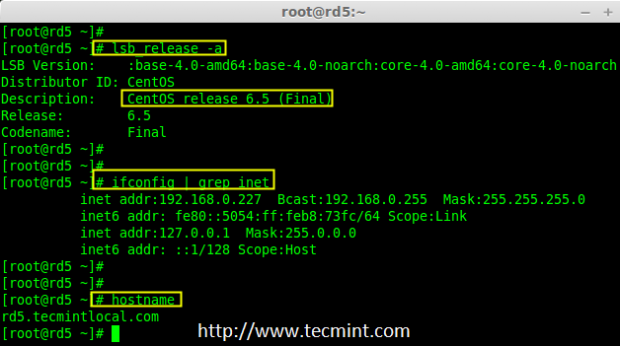
### **Step 1: Installing mdadm and Verify Drives**

**1.** As we said earlier, that we’re using CentOS 6.5 Final release for this raid setup, but same steps can be followed for RAID setup in any Linux based distributions.

# lsb\_release -a

# ifconfig | grep inet



CentOS 6.5 Summary

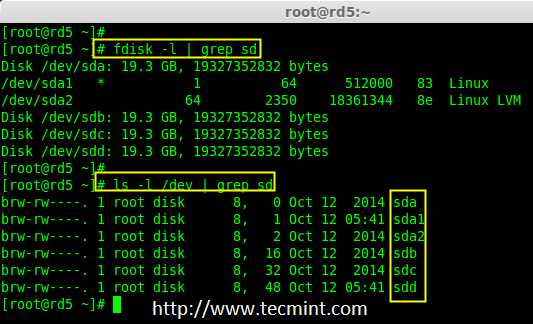
**2.** If you’re following our raid series, we assume that you’ve already installed ‘**mdadm**‘ package, if not, use the following command according to your Linux distribution to install the package.

# yum install mdadm [on RedHat systems]

# apt-get install mdadm [on Debain systems]

**3.** After the ‘**mdadm**‘ package installation, let’s list the three 20GB disks which we have added in our system using ‘**fdisk**‘ command.

# fdisk -l | grep sd

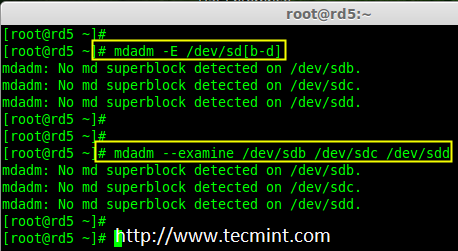


Install mdadm Tool

**4.** Now it’s time to examine the attached three drives for any existing RAID blocks on these drives using following command.

# mdadm -E /dev/sd[b-d]

# mdadm --examine /dev/sdb /dev/sdc /dev/sdd



Examine Drives For Raid

**Note**: From the above image illustrated that there is no any super-block detected yet. So, there is no RAID defined in all three drives. Let us start to create one now.

### **Step 2: Partitioning the Disks for RAID**

**5.** First and foremost, we have to partition the disks (**/dev/sdb**, **/dev/sd**c and **/dev/sdd**) before adding to a RAID, So let us define the partition using ‘fdisk’ command, before forwarding to the next steps.

# fdisk /dev/sdb

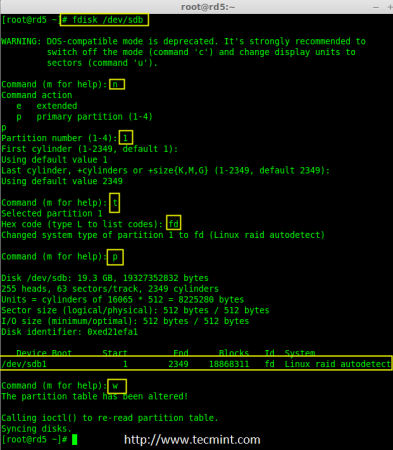
# fdisk /dev/sdc

# fdisk /dev/sdd

##### **Create /dev/sdb Partition**

Please follow the below instructions to create partition on **/dev/sdb** drive.

1. Press ‘**n**‘ for creating new partition.
2. Then choose ‘**P**‘ for Primary partition. Here we are choosing Primary because there is no partitions defined yet.
3. Then choose ‘**1**‘ to be the first partition. By default it will be **1**.
4. Here for cylinder size we don’t have to choose the specified size because we need the whole partition for RAID so just Press Enter two times to choose the default full size.
5. Next press ‘**p**‘ to print the created partition.
6. Change the Type, If we need to know the every available types Press ‘**L**‘.
7. Here, we are selecting ‘**fd**‘ as my type is RAID.
8. Next press ‘**p**‘ to print the defined partition.
9. Then again use ‘**p**‘ to print the changes what we have made.
10. Use ‘**w**‘ to write the changes.



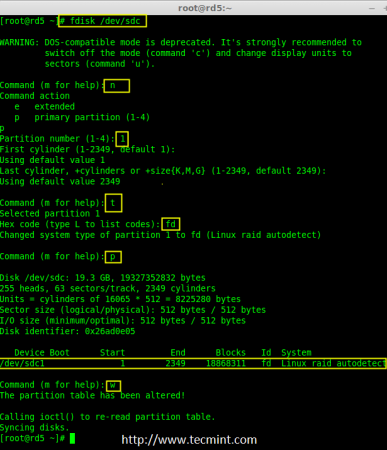
Create sdb Partition

**Note**: We have to follow the steps mentioned above to create partitions for **sdc** & **sdd** drives too.

##### **Create /dev/sdc Partition**

Now partition the **sdc** and **sdd** drives by following the steps given in the screenshot or you can follow above steps.

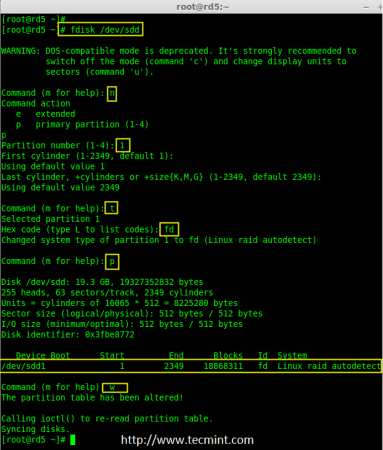
# fdisk /dev/sdc



Create sdc Partition

##### **Create /dev/sdd Partition**

# fdisk /dev/sdd



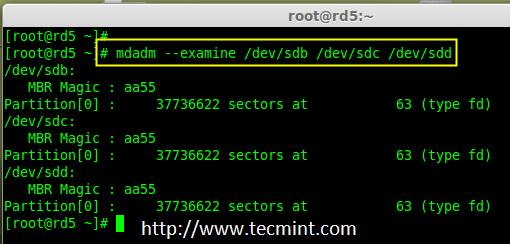
Create sdd Partition

**6.** After creating partitions, check for changes in all three drives sdb, sdc, & sdd.

# mdadm --examine /dev/sdb /dev/sdc /dev/sdd

or

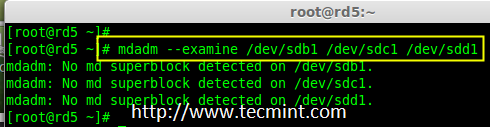
# mdadm -E /dev/sd[b-c]



Check Partition Changes

**Note**: In the above pic. depict the type is fd i.e. for RAID.

**7.** Now Check for the RAID blocks in newly created partitions. If no super-blocks detected, than we can move forward to create a new RAID 5 setup on these drives.



Check Raid on Partition

### **Step 3: Creating md device md0**

**8.** Now create a Raid device ‘**md0**‘ (i.e. **/dev/md0**) and include raid level on all newly created partitions (sdb1, sdc1 and sdd1) using below command.

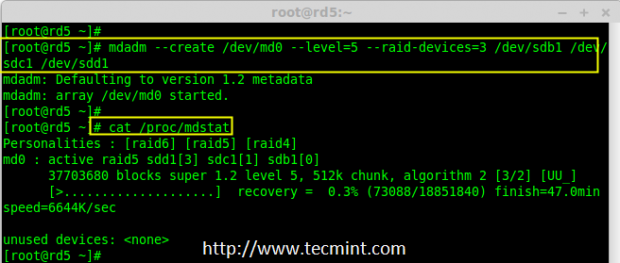
# mdadm --create /dev/md0 --level=5 --raid-devices=3 /dev/sdb1 /dev/sdc1 /dev/sdd1

or

# mdadm -C /dev/md0 -l=5 -n=3 /dev/sd[b-d]1

**9.** After creating raid device, check and verify the RAID, devices included and RAID Level from the **mdstat** output.

# cat /proc/mdstat



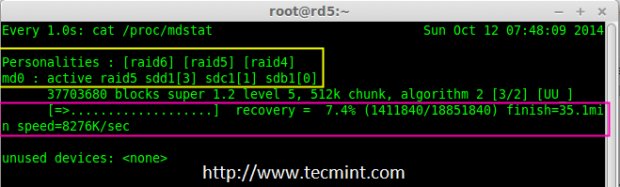
Verify Raid Device

If you want to monitor the current building process, you can use ‘**watch**‘ command, just pass through the ‘**cat /proc/mdstat**‘ with watch command which will refresh screen every **1** second.

# watch -n1 cat /proc/mdstat



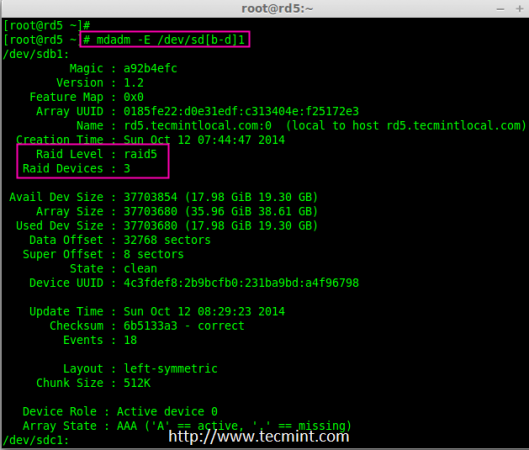
Monitor Raid 5 Process



Raid 5 Process Summary

**10.** After creation of raid, Verify the raid devices using the following command.

# mdadm -E /dev/sd[b-d]1

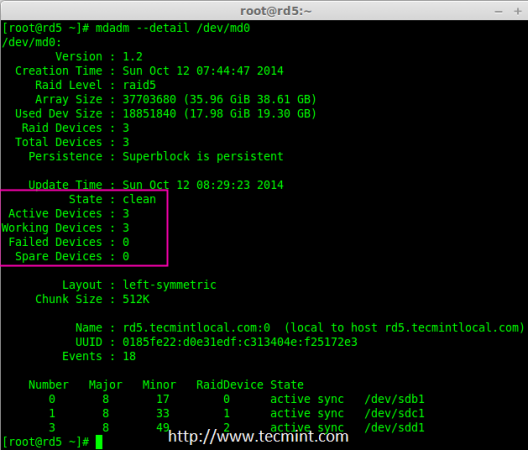


Verify Raid Level

**Note**: The Output of the above command will be little long as it prints the information of all three drives.

**11.** Next, verify the RAID array to assume that the devices which we’ve included in the RAID level are running and started to re-sync.

# mdadm --detail /dev/md0

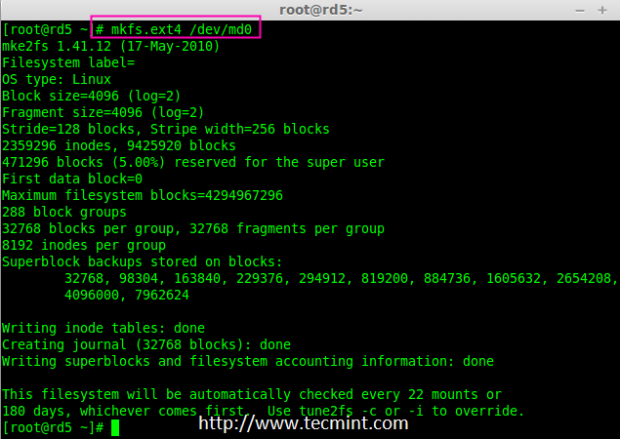


Verify Raid Array

### **Step 4: Creating file system for md0**

**12.** Create a file system for ‘**md0**‘ device using **ext4** before mounting.

# mkfs.ext4 /dev/md0



Create md0 Filesystem

**13.** Now create a directory under ‘**/mnt**‘ then mount the created filesystem under **/mnt/raid5** and check the files under mount point, you will see **lost+found** directory.

# mkdir /mnt/raid5

# mount /dev/md0 /mnt/raid5/

# ls -l /mnt/raid5/

**14.** Create few files under mount point **/mnt/raid5** and append some text in any one of the file to verify the content.

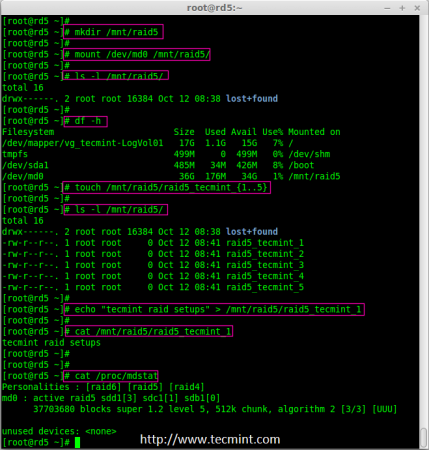
# touch /mnt/raid5/raid5\_tecmint\_{1..5}

# ls -l /mnt/raid5/

# echo "tecmint raid setups" > /mnt/raid5/raid5\_tecmint\_1

# cat /mnt/raid5/raid5\_tecmint\_1

# cat /proc/mdstat

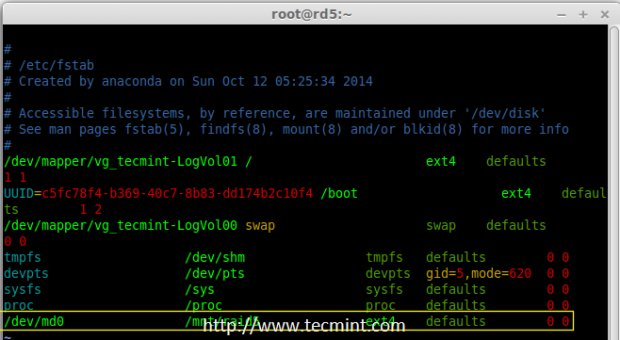


Mount Raid Device

**15.** We need to add entry in **fstab**, else will not display our mount point after system reboot. To add an entry, we should edit the fstab file and append the following line as shown below. The mount point will differ according to your environment.

# vim /etc/fstab

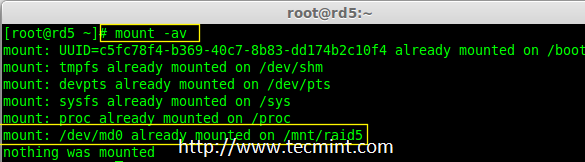
/dev/md0 /mnt/raid5 ext4 defaults 0 0



Raid 5 Automount

**16.** Next, run ‘**mount -av**‘ command to check whether any errors in fstab entry.

# mount -av



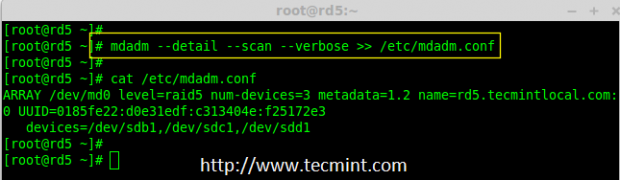
Check Fstab Errors

### **Step 5: Save Raid 5 Configuration**

**17.** As mentioned earlier in requirement section, by default RAID don’t have a config file. We have to save it manually. If this step is not followed RAID device will not be in md0, it will be in some other random number.

So, we must have to save the configuration before system reboot. If the configuration is saved it will be loaded to the kernel during the system reboot and RAID will also gets loaded.

# mdadm --detail --scan --verbose >> /etc/mdadm.conf



Save Raid 5 Configuration

**Note**: Saving the configuration will keep the RAID level stable in md0 device.

### **Step 6: Adding Spare Drives**

**18.** What the use of adding a spare drive? its very useful if we have a spare drive, if any one of the disk fails in our array, this spare drive will get active and rebuild the process and sync the data from other disk, so we can see a redundancy here.