Advanced Installation

Previous Next

Software RAID

Redundant Array of Independent Disks "RAID" is a method of using multiple disks to provide different balances of increasing data reliability and/or increasing input/output performance, depending on the RAID level being used. RAID is implemented in either software (where the operating system knows about both drives and actively maintains both of them) or hardware (where a special controller makes the OS think there's only one drive and maintains the drives 'invisibly').

The RAID software included with current versions of Linux (and Ubuntu) is based on the *'mdadm'* driver and works very well, better even than many so-called 'hardware' RAID controllers. This section will guide you through installing Ubuntu Server Edition using two RAID1 partitions on two physical hard drives, one for */* and another for *swap*.

Partitioning

Follow the installation steps until you get to the *Partition disks* step, then:

1. Select *Manual* as the partition method.
2. Select the first hard drive, and agree to *"Create a new empty partition table on this device?"*.

   Repeat this step for each drive you wish to be part of the RAID array.
3. Select the *"FREE SPACE"* on the first drive then select *"Create a new partition"*.
4. Next, select the *Size* of the partition. This partition will be the *swap* partition, and a general rule for swap size is twice that of RAM. Enter the partition size, then choose *Primary*, then *Beginning*.

   > A swap partition size of twice the available RAM capacity may not always be desirable, especially on systems with large amounts of RAM. Calculating the swap partition size for servers is highly dependent on how the system is going to be used.
5. Select the *"Use as:"* line at the top. By default this is *"Ext4 journaling file system"*, change that to *"physical volume for RAID"* then *"Done setting up partition"*.
6. For the */* partition once again select *"Free Space"* on the first drive then *"Create a new partition"*.
7. Use the rest of the free space on the drive and choose *Continue*, then *Primary*.
8. As with the swap partition, select the *"Use as:"* line at the top, changing it to *"physical volume for RAID"*. Also select the *"Bootable flag:"* line to change the value to *"on"*. Then choose *"Done setting up partition"*.
9. Repeat steps three through eight for the other disk and partitions.

RAID Configuration

With the partitions setup the arrays are ready to be configured:

1. Back in the main "Partition Disks" page, select *"Configure Software RAID"* at the top.
2. Select *"yes"* to write the changes to disk.
3. Choose *"Create MD device"*.
4. For this example, select *"RAID1"*, but if you are using a different setup choose the appropriate type (RAID0 RAID1 RAID5).

   > In order to use *RAID5* you need at least *three* drives. Using RAID0 or RAID1 only *two* drives are required.
5. Enter the number of active devices *"2"*, or the amount of hard drives you have, for the array. Then select *"Continue"*.
6. Next, enter the number of spare devices *"0"* by default, then choose *"Continue"*.
7. Choose which partitions to use. Generally they will be sda1, sdb1, sdc1, etc. The numbers will usually match and the different letters correspond to different hard drives.

   For the *swap* partition choose *sda1* and *sdb1*. Select *"Continue"* to go to the next step.
8. Repeat steps *three* through *seven* for the */* partition choosing *sda2* and *sdb2*.
9. Once done select *"Finish"*.

Formatting

There should now be a list of hard drives and RAID devices. The next step is to format and set the mount point for the RAID devices. Treat the RAID device as a local hard drive, format and mount accordingly.

1. Select *"#1"* under the *"RAID1 device #0"* partition.
2. Choose *"Use as:"*. Then select *"swap area"*, then *"Done setting up partition"*.
3. Next, select *"#1"* under the *"RAID1 device #1"* partition.
4. Choose *"Use as:"*. Then select *"Ext4 journaling file system"*.
5. Then select the *"Mount point"* and choose *"/ - the root file system"*. Change any of the other options as appropriate, then select *"Done setting up partition"*.
6. Finally, select *"Finish partitioning and write changes to disk"*.

If you choose to place the root partition on a RAID array, the installer will then ask if you would like to boot in a *degraded* state. See Degraded RAID for further details.

The installation process will then continue normally.

Degraded RAID

At some point in the life of the computer a disk failure event may occur. When this happens, using Software RAID, the operating system will place the array into what is known as a *degraded* state.

If the array has become degraded, due to the chance of data corruption, by default Ubuntu Server Edition will boot to *initramfs* after thirty seconds. Once the initramfs has booted there is a fifteen second prompt giving you the option to go ahead and boot the system, or attempt manual recover. Booting to the initramfs prompt may or may not be the desired behavior, especially if the machine is in a remote location. Booting to a degraded array can be configured several ways:

1. The *dpkg-reconfigure* utility can be used to configure the default behavior, and during the process you will be queried about additional settings related to the array. Such as monitoring, email alerts, etc. To reconfigure *mdadm* enter the following:

   ```
   sudo dpkg-reconfigure mdadm
   ```

2. The `dpkg-reconfigure mdadm` process will change the `/etc/initramfs-tools/conf.d/mdadm` configuration file. The file has the advantage of being able to pre-configure the system's behavior, and can also be manually edited:

   ```
   BOOT_DEGRADED=true
   ```

   The configuration file can be overridden by using a Kernel argument.

3. Using a Kernel argument will allow the system to boot to a degraded array as well:

   1. When the server is booting press **Shift** to open the *Grub* menu.
   2. Press **e** to edit your kernel command options.
   3. Press the **down** arrow to highlight the kernel line.
   4. Add *"bootdegraded=true"* (without the quotes) to the end of the line.
   5. Press **Ctrl**+**x** to boot the system.

Once the system has booted you can either repair the array see RAID Maintenance for details, or copy important data to another machine due to major hardware failure.

RAID Maintenance

The *mdadm* utility can be used to view the status of an array, add disks to an array, remove disks, etc:

1. To view the status of an array, from a terminal prompt enter:

   ```
   sudo mdadm -D /dev/md0
   ```

   The *-D* tells *mdadm* to display *detailed* information about the `/dev/md0` device. Replace `/dev/md0` with the appropriate RAID device.

2. To view the status of a disk in an array:

   ```
   sudo mdadm -E /dev/sda1
   ```

   The output if very similar to the `mdadm -D` command, adjust `/dev/sda1` for each disk.

3. If a disk fails and needs to be removed from an array enter:

   ```
   sudo mdadm --remove /dev/md0 /dev/sda1
   ```

   Change `/dev/md0` and `/dev/sda1` to the appropriate RAID device and disk.

4. Similarly, to add a new disk:

   ```
   sudo mdadm --add /dev/md0 /dev/sda1
   ```

Sometimes a disk can change to a *faulty* state even though there is nothing physically wrong with the drive. It is usually worthwhile to remove the drive from the array then re-add it. This will cause the drive to re-sync with the array. If the drive will not sync with the array, it is a good indication of hardware failure.

The `/proc/mdstat` file also contains useful information about the system's RAID devices:

```
cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sda1[0] sdb1[1]
      10016384 blocks [2/2] [UU]

unused devices: <none>
```

The following command is great for watching the status of a syncing drive:

```
watch -n1 cat /proc/mdstat
```

Press *Ctrl+c* to stop the *watch* command.

If you do need to replace a faulty drive, after the drive has been replaced and synced, *grub* will need to be installed. To install *grub* on the new drive, enter the following:

```
sudo grub-install /dev/md0
```

Replace `/dev/md0` with the appropriate array device name.

Resources

The topic of RAID arrays is a complex one due to the plethora of ways RAID can be configured. Please see the following links for more information:

1. Ubuntu Wiki Articles on RAID.
2. Software RAID HOWTO
3. Managing RAID on Linux

Logical Volume Manager (LVM)

Logical Volume Manager (LVM)

Logical Volume Manger, or *LVM*, allows administrators to create *logical* volumes out of one or multiple physical hard disks. LVM volumes can be created on both software RAID partitions and standard partitions residing on a single disk. Volumes can also be extended, giving greater flexibility to systems as requirements change.

Overview

A side effect of LVM's power and flexibility is a greater degree of complication. Before diving into the LVM installation process, it is best to get familiar with some terms.

1. *Physical Volume (PV):* physical hard disk, disk partition or software RAID partition formatted as LVM PV.
2. *Volume Group (VG):* is made from one or more physical volumes. A VG can can be extended by adding more PVs. A VG is like a virtual disk drive, from which one or more logical volumes are carved.
3. *Logical Volume (LV):* is similar to a partition in a non-LVM system. A LV is formatted with the desired file system (EXT3, XFS, JFS, etc), it is then available for mounting and data storage.

Installation

As an example this section covers installing Ubuntu Server Edition with `/srv` mounted on a LVM volume. During the initial install only one Physical Volume (PV) will be part of the Volume Group (VG). Another PV will be added after install to demonstrate how a VG can be extended.

There are several installation options for LVM, *"Guided - use the entire disk and setup LVM"* which will also allow you to assign a portion of the available space to LVM, *"Guided - use entire and setup encrypted LVM"*, or *Manually* setup the partitions and configure LVM. At this time the only way to configure a system with both LVM and standard partitions, during installation, is to use the Manual approach.

1. Follow the installation steps until you get to the *Partition disks* step, then:
2. At the *"Partition Disks* screen choose *"Manual"*.
3. Select the hard disk and on the next screen choose "yes" to *"Create a new empty partition table on this device"*.
4. Next, create standard */boot*, *swap*, and */* partitions with whichever filesystem you prefer.
5. For the LVM */srv*, create a new *Logical* partition. Then change *"Use as"* to *"physical volume for LVM"* then *"Done setting up the partition"*.
6. Now select *"Configure the Logical Volume Manager"* at the top, and choose *"Yes"* to write the changes to disk.
7. For the *"LVM configuration action"* on the next screen, choose *"Create volume group"*. Enter a name for the VG such as *vg01*, or something more descriptive. After entering a name, select the partition configured for LVM, and choose *"Continue"*.
8. Back at the *"LVM configuration action"* screen, select *"Create logical volume"*. Select the newly created volume group, and enter a name for the new LV, for example *srv* since that is the intended mount point. Then choose a size, which may be the full partition because it can always be extended later. Choose *"Finish"* and you should be back at the main *"Partition Disks"* screen.
9. Now add a filesystem to the new LVM. Select the partition under *"LVM VG vg01, LV srv"*, or whatever name you have chosen, the choose *Use as*. Setup a file system as normal selecting */srv* as the mount point. Once done, select *"Done setting up the partition"*.
10. Finally, select *"Finish partitioning and write changes to disk"*. Then confirm the changes and continue with the rest of the installation.

There are some useful utilities to view information about LVM:

1. *pvdisplay:* shows information about Physical Volumes.
2. *vgdisplay:* shows information about Volume Groups.
3. *lvdisplay:* shows information about Logical Volumes.

Extending Volume Groups

Continuing with *srv* as an LVM volume example, this section covers adding a second hard disk, creating a Physical Volume (PV), adding it to the volume group (VG), extending the logical volume `srv` and finally extending the filesystem. This example assumes a second hard disk has been added to the system. In this example, this hard disk will be named `/dev/sdb` and we will use the entire disk as a physical volume (you could choose to create partitions and use them as different physical volumes)

> Make sure you don't already have an existing `/dev/sdb` before issuing the commands below. You could lose some data if you issue those commands on a non-empty disk.

1. First, create the physical volume, in a terminal execute:

```
sudo pvcreate /dev/sdb
```

2. Now extend the Volume Group (VG):

```
sudo vgextend vg01 /dev/sdb
```

3. Use *vgdisplay* to find out the free physical extents - Free PE / size (the size you can allocate). We will assume a free size of 511 PE (equivalent to 2GB with a PE size of 4MB) and we will use the whole free space available. Use your own PE and/or free space.

   The Logical Volume (LV) can now be extended by different methods, we will only see how to use the PE to extend the LV:

```
sudo lvextend /dev/vg01/srv -l +511
```

   The *-l* option allows the LV to be extended using PE. The *-L* option allows the LV to be extended using Meg, Gig, Tera, etc bytes.

4. Even though you are supposed to be able to *expand* an ext3 or ext4 filesystem without unmounting it first, it may be a good practice to unmount it anyway and check the filesystem, so that you don't mess up the day you want to reduce a logical volume (in that case unmounting first is compulsory).

   The following commands are for an *EXT3* or *EXT4* filesystem. If you are using another filesystem there may be other utilities available.

```
sudo umount /srv
sudo e2fsck -f /dev/vg01/srv
```

   The *-f* option of *e2fsck* forces checking even if the system seems clean.

5. Finally, resize the filesystem:

```
sudo resize2fs /dev/vg01/srv
```

6. Now mount the partition and check its size.

```
mount /dev/vg01/srv /srv && df -h /srv
```

Resources

1. See the Ubuntu Wiki LVM Articles.
2. See the LVM HOWTO for more information.
3. Another good article is Managing Disk Space with LVM on O'Reilly's linuxdevcenter.com site.
4. For more information on *fdisk* see the fdisk man page.

iSCSI

The iSCSI protocol can be used to install Ubuntu on systems with or without hard disks attached, and iBFT can be used to automate iSCSI setup on installation and boot.

Installation on a diskless system

The first steps of a diskless iSCSI installation are identical to the Installation using debian-installer section up to "Hard drive layout".

1. The installer will display a warning with the following message:

```
No disk drive was detected. If you know the name of the driver needed by your disk drive, you can
select it from the list.
```

2. Select the item in the list titled *login to iSCSI targets.*
3. You will be prompted to Enter an IP address to scan for iSCSI targets with a description of the format for the address. Enter the IP address for the location of your iSCSI target and navigate to *<continue>* then hit **ENTER**
4. If authentication is required in order to access the iSCSI device, provide the *username* in the next field. Otherwise leave it blank.
5. If your system is able to connect to the iSCSI provider, you should see a list of available iSCSI targets where the operating system can be installed. The list should be similar to the following :

```
Select the iSCSI targets you wish to use.

iSCSI targets on 192.168.1.29:3260:

[ ] iqn.2016-03.TrustyS-iscsitarget:storage.sys0

<Go Back>                          <Continue>
```

6. Select the iSCSI target that you want to use with the space bar. Use the arrow keys to navigate to the target that you want to select.
7. Navigate to *<Continue>* and hit **ENTER**.

If the connection to the iSCSI target is successful, you will be prompted with the *[!!] Partition disks* installation menu. The rest of the procedure is identical to any normal installation on attached disks. Once the installation is completed, you will be asked to reboot.

Installation on a system with disk attached

Again, the iSCSI installation on a normal server with one or many disks attached is identical to the Installation using debian-installer section until we reach the disk partitioning menu. Instead of using any of the Guided selection, we need to perform the following steps :

1. Navigate to the Manual menu entry
2. Select the Configure iSCSI Volumes menu entry
3. Choose the Log into iSCSI targets
4. You will be prompted to Enter an IP address to scan for iSCSI targets. with a description of the format for the address. Enter the IP address and navigate to *<continue>* then hit **ENTER**
5. If authentication is required in order to access the iSCSI device, provide the *username* in the next field or leave it blank.
6. If your system is able to connect to the iSCSI provider, you should see a list of available iSCSI targets where the operating system can be installed. The list should be similar to the following :

```
Select the iSCSI targets you wish to use.

iSCSI targets on 192.168.1.29:3260:

[ ] iqn.2016-03.TrustyS-iscsitarget:storage.sys0

<Go Back>                          <Continue>
```

7. Select the iSCSI target that you want to use with the space bar. Use the arrow keys to navigate to the target that you want to select
8. Navigate to <Continue> and hit **ENTER**.
9. If successful, you will come back to the menu asking you to Log into iSCSI targets. Navigate to Finish and hit **ENTER**

The newly connected iSCSI disk will appear in the overview section as a device prefixed with SCSI. This is the disk that you should select as your installation disk. Once identified, you can choose any of the partitioning methods.

> Depending on your system configuration, there may be other SCSI disks attached to the system. Be very careful to identify the proper device before proceeding with the installation. Otherwise, irreversible data loss may result from performing an installation on the wrong disk.

Installation with iBFT

In order to setup iSCSI based on the iBFT (iSCSI Boot Firmware Table) on the installation and boot, append these options at the installer prompt (or to the preseed file):

```
disk-detect/ibft/enable=true partman-iscsi/iscsi_auto=true
```

This should probe for iBFT information and configure network interface(s) and iSCSI target(s) accordingly during the installation, and configure system boot (initramfs) to do that too in order to find the root device.

> The support for iBFT is available in the debian-installer on netboot images as of 2019-06-20 and (expected) on ISO images for the 18.04.3 point release and later.

Rebooting to an iSCSI target

The procedure is specific to your hardware platform. As an example, here is how to reboot to your iSCSI target using iPXE

```
iPXE> dhcp

Configuring (net0 52:54:00:a4:f2:a9)....... ok

iPXE> sanboot iscsi:192.168.1.29::::iqn.2016-03.TrustyS-iscsitarget:storage.sys0
```

If the procedure is successful, you should see the Grub menu appear on the screen.

Previous Next