Using Large Hard Drives in Linux

Presented by Kevin McGregor
Manitoba UNIX User Group
March 12, 2013

The Problem

- Because the Master Boot Record (MBR) data structures use 32-bit pointers for LBA (Logical Block Addressing) and sectors are assumed to be 512 bytes long, maximum disk size is ~2.2 TB (2 TiB)

So what? I Won't Use Partitions

- Mark whole drive as an LVM physical volume
  e.g. pvcreate /dev/sdb
  and make logical volumes out of that
- Works fine!

That *Usually* Works Fine…

- But other GPT-unaware OSs may still see it (e.g. on a SAN)
- But the disk looks empty (with standard tools) even when it isn't
- But it's hard to tell what the disk contains
- But it's hard to tell what the disk is for
- But mistakes happen

So Label Your Disk

- With MBR?
  - 2^{32} sector limit
  - Single Point Of Failure (SPOF) – one copy
  - Maximum four primary partitions
  - Extended/Logical partitions are lame and fragile (Single-linked list!)
  - Cylinders? Heads? Sectors per track?
    Irrelevant cruft now

GUID Partition Table (GPT)

- Up to 2^{64} sectors (8 Giga-Terabytes [ZiB])
- Two copies; start and end of disk
- Variable number of partitions (default 128)
- LBA 0 is a "Protective MBR"; a dummy partition table with one partition of type 0xEE covering whole disk (up to a maximum of 2 TiB)

GPT

- LBA 1 is GPT header
- Defines
  - Maximum number of partitions
  - Number and size of table entries
  - Disk UUID
  - Location of GPT, backup GPT
  - Checksums
- GPT entries include
  - 64-bit start LBA and end LBA (not length)
  - 128-bit UUID for partition type
  - Name (up to 36 UTF-16LE "code units")

GUID Partition Types

- Linux/Windows data
  EBD0A0A2-B9E5-4433-87C0-68B6B72699C7
- Linux swap
  0657FD6D-A4AB-43C4-84E5-0933C84B4F4F
- Linux LVM
  E6D6D379-F507-44C2-A23C-238F2A3DF928
- Linux RAID
  A19D880F-05FC-4D3B-A006-743F0F84911E
- Good thing we don't have to memorize them!

Okay, how?

- Don't use fdisk; it's for MBR-only disks
- fdisk will warn you if it detects a GPT-labeled disk

- Use parted:
- mklabel gpt # create the disklabel
- p # list the GPT partitions
- q # exit parted, writing changes

Basic parted

- Create a basic data partition
  - mkpart <name> <start> <end>
  - e.g. mkpart home 1G 2G

- Create a swap partition
  - mkpart <name> linux-swap <start> <end>
  - e.g. mkpart swap linux-swap 2G 3G

Basic parted (continued)

- Create a LVM partition
  - Make a normal data partition
  - Mark as LVM
    - parted <drive-device> set <partition#> lvm on
    - e.g. parted /dev/sda set 2 lvm on
    - NOT parted /dev/sda2 lvm on
  - Marks /dev/sda2 with "Linux LVM" GUID

Basic parted (continued)

- Create a software RAID partition
  - Make a normal data partition
  - Mark as RAID
    - parted /dev/sda set 3 raid on
  - Marks /dev/sda3 with "Linux RAID" GUID

Booting from GPT

- All current Linux distros can use GPT-labeled secondary disks
- To boot from GPT, your system must support the uEFI boot process
- The "Protective MBR" no longer contains bootloader
- First partition on boot disk is EFI System Partition (ESP) – a FAT filesystem, usually mounted on /boot/efi
- See also efibootmgr

Hybrid MBR/GPT

- You can do this, technically… but it's a bad idea
- Not generally supported
- Prone to error

A Quick Sample

```
$ parted /dev/sdb
GNU Parted 2.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: ATA WDC WD30EFRX-68A (scsi)
Disk /dev/sdb: 3001GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt

Number  Start   End     Size    File system  Name        Flags
 1      1049kB  3001GB  3001GB               Linux RAID  raid

(parted) align-check opt 1
1 aligned
(parted) unit MiB
(parted) p
Model: ATA WDC WD30EFRX-68A (scsi)
Disk /dev/sdb: 2861588MiB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt

Number  Start    End         Size        File system  Name        Flags
 1      1.00MiB  2861588MiB  2861587MiB               Linux RAID  raid
```

References

- This presentation was largely copied from https://www.redhat.com/summit/2011/presentations/summit/taste_of_training/wednesday/Bonneville_Getting_Beyond_2_Terabytes_Using_GPT_with_Storage_Devices.pdf
- See also
  https://en.wikipedia.org/wiki/Master_boot_record
  https://en.wikipedia.org/wiki/GUID_Partition_Table
- And many other sources

Questions