

# Lab 03.1.: Configuring Guest Disks

## ID720 Virtualisation

### Introduction

Up to this point we have used LVM to create volumes providing principal hard drives for our guests. However, we might want to provide additional disk storage for guests, in particular to create storage options not linked to a particular guest instance. In this lab we will add a disk to a Xen guest domain.

### 1 The plan

1. Create a new logical volume on Dom0;
2. Add a specification for the volume to a guest's diskspec;
3. Start the guest;
4. From the guest, create a partition table on the new volume;
5. Set up a file system on the volume;
6. Mount the volume on the guest.

### 2 Create a new logical volume

We've already used lvm a few times to create logical volumes in our volume group. This case is no different.

```
lvcreate -L 1GB -n storagevol /dev/<volume group name>
```

### 3 Add the volume to a guest's diskspec

In our guests' configuration files, there is a diskspec line that looks like this:

```
disk = [ '/dev/my-vg/clone1,raw,xvda,rw' ]
```

We just need to add a second string to this list that provides the details for our new volume. That string will look like this:

```
'/dev/my-vg/storagevol,raw,xvdb,rw'
```

Don't forget to substitute your systems volume group name. Note that we label this new device **xvdb** on the guest, since **xvda** is already used for the first disk device.

Once you finish modifying the diskspec, start the reconfigured guest domain. All the following steps will be carried out on the guest's console.

## 4 Create a partition table on the new volume

From the point of view of the guest domain, we have added a new unformatted drive to the system. We need to do some additional work before we can mount it and use it. First, we'll use **parted** to create the partition table:

(Below we first invoke **parted**, specifying the new volume as it's seen by the guest. Then we enter **parted** commands where we see the (**parted**) prompt.)

```
parted /dev/xvdb
(parted) mklabel msdos
(parted) mkpart
Partition type? primary/extended? primary
File system type? [ext2]? ext4
Start? 1
End? 100%
<output snipped>
(parted)quit
```

Note that you may get a warning about the sector alignment. For our purposes today it's safe to ignore it.

The result of this is that we will now have a disk device with a partition table on it. We need to do one more thing to get it ready to use.

## 5 Create a file system on the new volume

Before we can mount the volume, it needs a file system. Create a new **ext4** system on the volume with the command

```
mkfs.ext4 /dev/xvdb1
```

Note the "1" above. We create the file system on the first (and only) partition.

Once this is done we can mount and use the new volume.

## 6 Mount the volume

Create a mount point on the guest domain with the command

```
mkdir /mnt/storage
```

Then mount it with the command

```
mount /dev/xvdb1 /mnt/storage
```

Finally, use **df** to verify that the new volume is mounted and available.

## 7 Epilogue

Now that the new volume is fully formatted, we can use it on other guest domains by simply adding the volume to the guest's **diskspec** configuration. Then we can immediately mount and use the disk on the guest. Any files that we create using one guest will be seen by another that uses the volume.