Increasing Disk Space in Azure

Following is the process to increase disk space for a Lumeta system in Azure. The first two steps are representative in that they apply only to the Azure platform. If you plan to run your Lumeta system another supported platform and need assistance, please contact us.

a. Login to the Azure portal select the VM. On the **Disks** pane, under **Data disks**, select **Create and attach a new disk**.
b. Enter a name for your managed disk. Review the default settings, and update the **Storage type**, **Size (GiB)**, **Encryption** and **Host caching (read/write)** as necessary; 100GB for Cloud Scout, 500GB or more for Command Center.
c. When you are done, select **Save** at the top of the page to create the managed disk and update the VM configuration.Then perform the following steps utilize the new disk space:
d. Login to Asset Manager via ssh as the user you created during the deployment. Run:
    i. sudo su
    ii. provide password for the user
    iii. (you will become root)

View Existing Disk Space Allocations

Run fdisk -l to obtain information about Disk Partitions. For Asset Manager the /dev/xvda is the default disk with 3 Logical Volumes xvda1 to xvda3 already in use.

```
Device       Start      End   Sectors  Size Type
/dev/xvda1    2048     4095      2048    1M BIOS boot
/dev/xvda2    4096   2052095   2048000 1000M Linux filesystem
/dev/xvda3 2052096 209713151 207661056   99G Linux LVM
```

For Asset Manager the Volume Group is called “vg_sys”

There are five disk partitions:

- · /home: Small space to store files for individual users (2% of vg_sys, mounted nosuid to comply with V-72041)
- · /var/log/audit: Location for the Linux Audit Framework to store data. These logs show who did what and when. While this is technically a log, STIGs specify that the audit log must be in it's own space (2% of vg_sys).
- · /var: A place for persistent data to be stored for applications such as the Lumeta database. Also, logs are stored in this location (70% of vg_sys).
- · /tmp: A space for temporary files to be stored. Lumeta does not currently clean this (2% of vg_sys).
- · : The root of the file system. This is the top level directory for all other files (10% of vg_sys).

Add Disk Space to existing VM Disk

Create a New Partition that takes up the remaining space and is of filesystem type 8e (Logical Volume).

- fdisk /dev/xvda
- Now enter n to create a new partition, and choose p to create a new primary partition.
- Partition Number: 4 - using the default
- Select default first available cylinder to the default last cylinder
- enter t to change a partition’s system id
- enter 4 for partition number
- Set type to 30 for LVM
- p to view the new partition layout.
- w to write the partition layout to disk

Create the Physical Volume as a basis for your LVM

Type fdisk -l Your new partition layout now includes /dev/sdc1.

```
Device       Start      End   Sectors  Size Type
/dev/xvda1    2048     4095      2048    1M BIOS boot
/dev/xvda2    4096   2052095   2048000 1000M Linux filesystem
/dev/xvda3 2052096 209713151 207661056   99G Linux LVM
/dev/xvda4 209713152 629145566 419432415  200G Linux filesystem
```

Create the physical volume.

- pvcreate /dev/xvda4

Add the xvda4 new physical volume to Volume Group vg_sys

- vgextend vg_sys /dev/xvda4

Type vgdisplay. You’ll see the physical volume is now in the volume groups size. In this example 200GB have been added to the 100GB Volume Group.

```
Alloc PE / Size       76548 / <99.02 GiB

Free  PE / Size       200GB
```

Allocate Disk to Partitions

Now that the New Physical Volume in the Volume Group has been created, it can be distributed to the Partitions. Below is a template on disk space distribution.

Please note each customer has their unique needs on how to allocate the disk.

1. First allocate 70 percent of the new disk space to lv_var
    a. Type lvextend -l +70%FREE /dev/mapper/vg_sys-lv_var
    b. Trigger online resizing of the live and mounted filesystem so the new disk space can be utilized immediately:
        i. resize2fs /dev/mapper/vg_sys-lv_var

2. Second we will allocate 15 percent (half of the 30% left) of the new disk space to lv_root
    a. Type lvextend -l +50%FREE /dev/mapper/vg_sys-lv_root
    b. Trigger online resizing of the live and mounted filesystem so the new disk space can be utilized immediately:
        i. resize2fs /dev/mapper/vg_sys-lv_root

3. Third we will allocate 5 percent (third of the 15% left) of the new disk space to lv_home
    a. Type lvextend -l +33%FREE /dev/mapper/vg_sys-lv_home
    b. Trigger online resizing of the live and mounted filesystem so the new disk space can be utilized immediately:
        i. resize2fs /dev/mapper/vg_sys-lv_home

4. Fourth we will allocate 5 percent (half of the 10% left) of the new disk space to lv_var_log_audit
    a. Type lvextend -l +50%FREE /dev/mapper/vg_sys-lv_var_log_audit
    b. Trigger online resizing of the live and mounted filesystem so the new disk space can be utilized immediately:
        i. resize2fs /dev/mapper/vg_sys-lv_var_log_audit

5. Finally we will allocate 5 percent (All of the 5% left) of the new disk space to lv_tmp
    a. Type lvextend -l +100%FREE /dev/mapper/vg_sys-lv_tmp
    b. Trigger online resizing of the live and mounted filesystem so the new disk space can be utilized immediately:
        i. resize2fs /dev/mapper/vg_sys-lv_tmp

6. Run df -h, lvs, and vgs to view new Disk Space Allocation