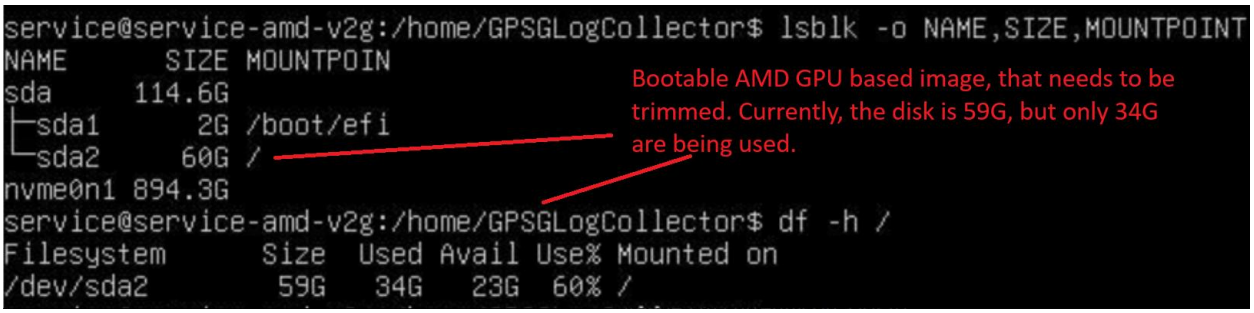
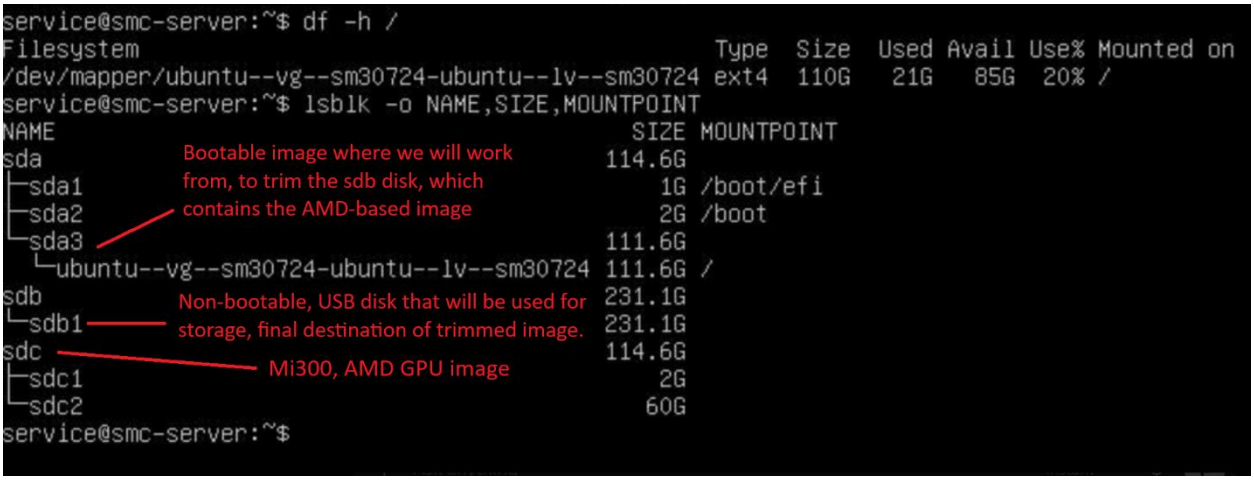
Trimming AMD-based Image – That will be used for hundreds of Field Application Engineers

As a Supermicro Employee, I was tasked to trim and OS image that will be downloaded hundreds of times by Field Application Engineers to test AMD GPU systems that are MI300X. Here's what I did.

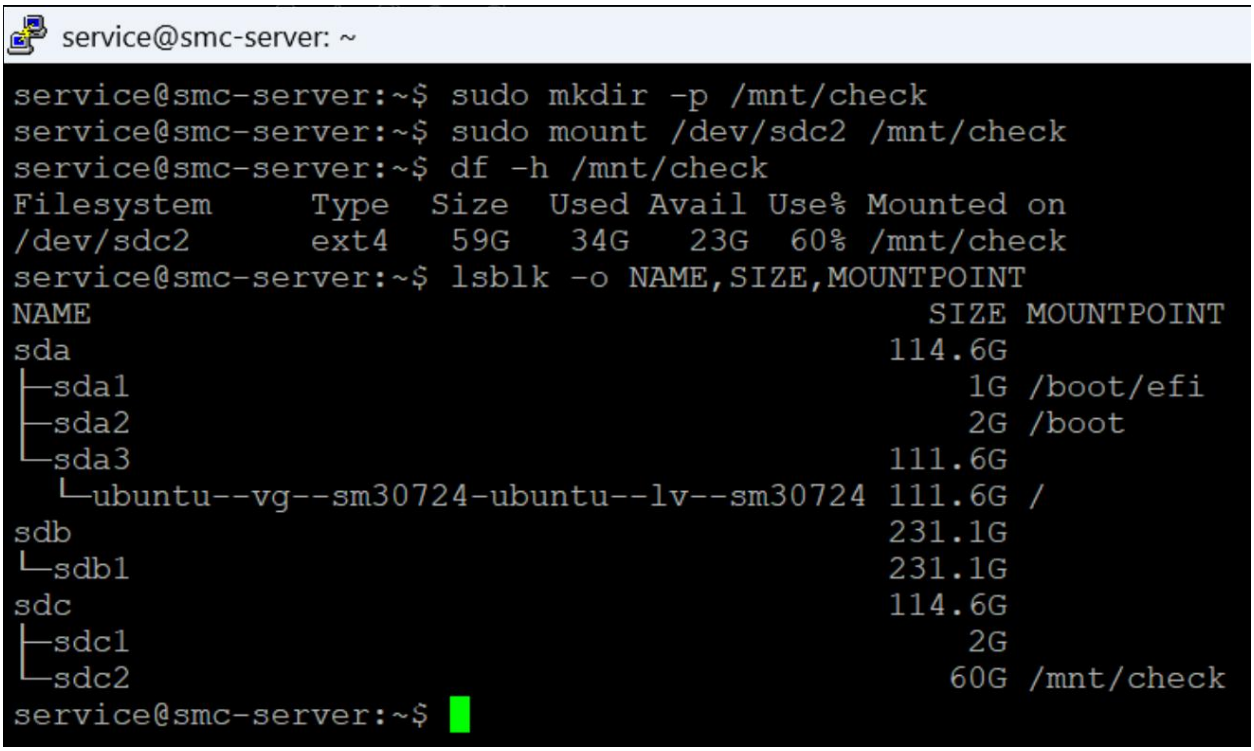
Currently, we are booting into an Mi300 and have this image on this partition that we need to trim:



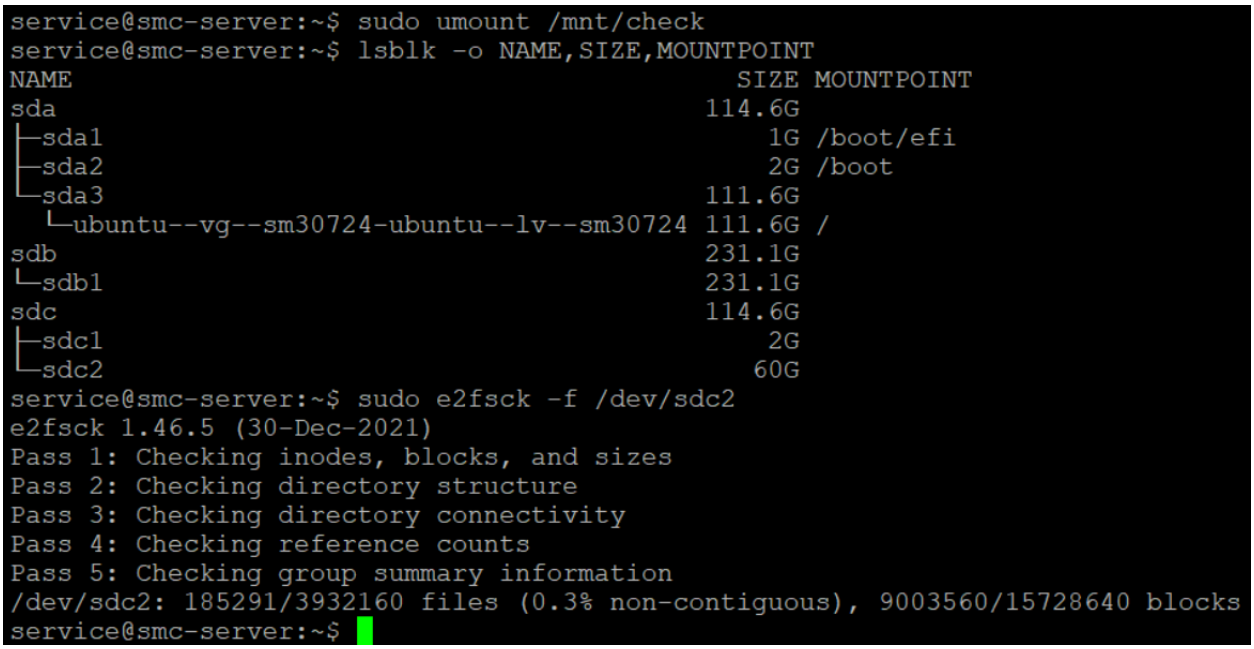
To trim that image, I first powered off the system, and on the USB port, inserted a USB hub with three ports. The first port, contains this same AMD-based image. But I also plug in two more usb keys: one,will be bootable (from where we will do the work) and another will be a non-bootable usb key that will be used to store the final trimmed AMD-based image.



We then mount sdc (where the AMD image is), check the available space to confirm it matches with our previous df -h output, and we confirm it is mounted:



We then umount /mnt/check, and do a check of the filesystem (verifying the internal integrity and consistency of the filesystem). We do the umount first since we shouldn't e2fsck a mounted filesystem:



The filesystem has no errors, needs no fixes, and shows 9003560/15728640 blocks. The filesystem easily fits inside 40G, our intended size for the full disk img. Now, we do a resize2fs command to resize the filesystem to 40G.

```
service@smc-server:~$ sudo resize2fs /dev/sdc2 40G
resize2fs 1.46.5 (30-Dec-2021)
Resizing the filesystem on /dev/sdc2 to 10485760 (4k) blocks.
The filesystem on /dev/sdc2 is now 10485760 (4k) blocks long.

service@smc-server:~$
```

sdc2 → now 40G filesystem.

10485760 (4k) blocks, matches 40G.

With the filesystem now reduced, we now proceed to reduce the partition itself, sdc.

```
service@smc-server:~$ sudo fdisk /dev/sdc

Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

GPT PMBR size mismatch (130025535 != 240328703) will be corrected by write.

Command (m for help): p

Disk /dev/sdc: 114.6 GiB, 123048296448 bytes, 240328704 sectors
Disk model:  SanDisk 3.2Gen1
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 91CBF1A3-B8A7-4D8D-8E57-A686B85970A5

Device       Start       End   Sectors Size Type
/dev/sdc1     2048   4196351   4194304   2G EFI System
/dev/sdc2  4196352 130025471 125829120  60G Linux filesystem

Command (m for help):
```

sdc2 start sector = 4196352

Now we delete and recreate the partition starting from the start sector, and adding 40 gigas:

```
service@smc-server:~$ sudo fdisk /dev/sdc

Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

GPT PMBR size mismatch (130025535 != 240328703) will be corrected by write.

Command (m for help): p

Disk /dev/sdc: 114.6 GiB, 123048296448 bytes, 240328704 sectors
Disk model:  SanDisk 3.2Gen1
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 91CBF1A3-B8A7-4D8D-8E57-A686B85970A5

Device       Start       End   Sectors Size Type
/dev/sdc1     2048   4196351   4194304   2G EFI System
/dev/sdc2  4196352 130025471 125829120  60G Linux filesystem

Command (m for help): d
Partition number (1,2, default 2): 2

Partition 2 has been deleted.

Command (m for help): n
Partition number (2-128, default 2): 2
First sector (4196352-240328670, default 4196352): 4196352
Last sector, +/-sectors or +/-size{K,M,G,T,P} (4196352-240328670, default 240328670): +40G

Created a new partition 2 of type 'Linux filesystem' and of size 40 GiB.
Partition #2 contains a ext4 signature.

Do you want to remove the signature? [Y]es/[N]o: n

Command (m for help): w

The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

service@smc-server:~$
```

We keep the signature, and write to save changes.

We then reload partition table (partprobe) and verify that the filesystem still matches (should see a "nothing to do msg"), then we do zero-fill to dramatically improve compression/cleanliness

```
service@smc-server:~$ sudo partprobe
service@smc-server:~$ sudo resize2fs /dev/sdc2
resize2fs 1.46.5 (30-Dec-2021)
The filesystem is already 10485760 (4k) blocks long.  Nothing to do!

service@smc-server:~$ sudo mkdir -p /mnt/fill
service@smc-server:~$ sudo mount /dev/sdc2 /mnt/fill
service@smc-server:~$ sudo dd if=/dev/zero of=/mnt/fill/zero.fill bs=1M status=progress
6388973568 bytes (6.4 GB, 6.0 GiB) copied, 381 s, 16.8 MB/s
dd: error writing '/mnt/fill/zero.fill': No space left on device
6099+0 records in
6098+0 records out
6394900480 bytes (6.4 GB, 6.0 GiB) copied, 380.675 s, 16.8 MB/s
service@smc-server:~$ sudo rm /mnt/fill/zero.fill
service@smc-server:~$ sync
service@smc-server:~$ sudo umount /mnt/fill
service@smc-server:~$
```

We had ~6GB free remaining inside the 40G filesystem after shrinking.
The dd if=/dev/zero filled that free space with zeros until "No space left on device."
We clean up by unmounting, removing the zero-fill directory, and syncing.
Now, we get the final sector:

```
service@smc-server:~$ sudo fdisk -l /dev/sdc
Disk /dev/sdc: 114.6 GiB, 123048296448 bytes, 240328704 sectors
Disk model:  SanDisk 3.2Gen1
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 91CBF1A3-B8A7-4D8D-8E57-A686B85970A5

Device       Start      End  Sectors Size Type
/dev/sdc1     2048  4196351  4194304   2G EFI System
/dev/sdc2  4196352 88082431 83886080  40G Linux filesystem
service@smc-server:~$
```

Final sector count = 88082431 + 1 = 88082432
Then we mount sdb1 (because that is where we will save our AMD image), and we run the DD command to save sdc to sdb1.

```
service@smc-server:~$ sudo mkdir -p /mnt/bigusb
service@smc-server:~$ sudo mount /dev/sdb1 /mnt/bigusb
service@smc-server:~$ sudo dd if=/dev/sdc of=/mnt/bigusb/gpsg_amd.img bs=512 count=88082432 status=progress
43834688000 bytes (44 GB, 41 GiB) copied, 1850 s, 23.7 MB/s
```

```
service@smc-server:~$ sudo dd if=/dev/sdc of=/mnt/bigusb/gpsg_amd.img bs=512 count=88082432 status=progress
45089068032 bytes (45 GB, 42 GiB) copied, 1913 s, 23.6 MB/s
88082432+0 records in
88082432+0 records out
45098205184 bytes (45 GB, 42 GiB) copied, 1918.46 s, 23.5 MB/s
service@smc-server:~$
```

88082432+0 records in
88082432+0 records out

That means the FULL requested sector range was copied successfully.

45098205184 bytes, which means ~42 GiB (~45 GB decimal).

The final image name is gpsg_amd.img. Final cleanup:

```
service@smc-server:~$ sync
service@smc-server:~$ sudo umount /mnt/bigusb
service@smc-server:~$ lsblk -o NAME,SIZE,MOUNTPOINT
NAME                                      SIZE MOUNTPOINT
sda                                     114.6G
├─sda1                                     1G /boot/efi
├─sda2                                     2G /boot
└─sda3                                  111.6G
  └─ubuntu--vg--sm30724-ubuntu--lv--sm30724 111.6G /
sdb                                     231.1G
└─sdb1                                  231.1G
sdc                                     114.6G
├─sdc1                                     2G
└─sdc2                                    40G
service@smc-server:~$
```

Finished:

sdb1 has NO mountpoint
destination USB cleanly unmounted
sdc2 = 40G

We can now simply disconnect usb key holding the image.

Optional:

```
service@smc-server:~$ sudo mount /dev/sdb1 /mnt/bigusb
service@smc-server:~$ ls -lh /mnt/bigusb
total 105G
drwxr-xr-x 2 root root 128K May  4 11:38 'System Volume Information'
-rwxr-xr-x 1 root root  34K May  4 11:38  autorun.ico
-rwxr-xr-x 1 root root  192 May  4 11:38  autorun.inf
-rwxr-xr-x 1 root root  32G May  7 11:32  b200_v2.img
-rwxr-xr-x 1 root root  32G May  5 14:41  b200v1.img
-rwxr-xr-x 1 root root  43G May 19 23:37  gpsg_amd.img
service@smc-server:~$ lsblk -o NAME,SIZE,MOUNTPOINT
NAME                                      SIZE MOUNTPOINT
sda                                     114.6G
├─sda1                                     1G /boot/efi
├─sda2                                     2G /boot
└─sda3                                  111.6G
  └─ubuntu--vg--sm30724-ubuntu--lv--sm30724 111.6G /
sdb                                     231.1G
└─sdb1                                  231.1G /mnt/bigusb
sdc                                     114.6G
├─sdc1                                     2G
└─sdc2                                    40G /mnt/source
service@smc-server:~$
```

We then use WinSCP to move the image from the system to my laptop.

Optional 2: Don't forget to unmount what needs to be unmounted:

```
service@smc-server:~$ lsblk -o NAME,SIZE,MOUNTPOINT
NAME                                      SIZE MOUNTPOINT
sda                                     114.6G
├─sda1                                     1G /boot/efi
├─sda2                                     2G /boot
└─sda3                                  111.6G
  └─ubuntu--vg--sm30724-ubuntu--lv--sm30724 111.6G /
sdb                                     231.1G
└─sdb1                                  231.1G /mnt/bigusb
sdc                                     114.6G
├─sdc1                                     2G
└─sdc2                                    40G /mnt/source
service@smc-server:~$
```

sudo umount /mnt/source
sudo umount /mnt/bigusb
verify: lsblk -o NAME,SIZE,MOUNTPOINT