Cloning a laptop over NVME TCP

Posted on Mar 10, 2024 By copyninja under devops

Recently, I got a new laptop and had to set it up so I could start using it. But I wasn't really in the mood to go through the same old steps which I had explained in this post earlier. I was complaining about this to my colleague, and there came the suggestion of why not copy the entire disk to the new laptop. Though it sounded like an interesting idea to me, I had my doubts, so here is what I told him in return.

1. I don't have the tools to open my old laptop and connect the new disk over USB to my new laptop.
2. I use full disk encryption, and my old laptop has a 512GB disk, whereas the new laptop has a 1TB NVME, and I'm not so familiar with resizing LUKS.

He promptly suggested both could be done. For step 1, just expose the disk using NVME over TCP and connect it over the network and do a full disk copy, and the rest is pretty simple to achieve. In short, he suggested the following:

1. Export the disk using nvmet-tcp from the old laptop.
2. Do a disk copy to the new laptop.
3. Resize the partition to use the full 1TB.
4. Resize LUKS.
5. Finally, resize the BTRFS root disk.

Exporting Disk over NVME TCP

The easiest way suggested by my colleague to do this is using systemd-storagetm.service. This service can be invoked by simply booting into *storage-target-mode.target* by specifying *rd.systemd.unit=storage-target-mode.target*. But he suggested not to use this as I need to tweak the dracut initrd image to involve network services as well as configuring WiFi from this mode is a painful thing to do.

So alternatively, I simply booted both my laptops with GRML rescue CD. And the following step was done to export the NVME disk on my current laptop using the nvmet-tcp module of Linux:

```
modprobe nvmet-tcp
cd /sys/kernel/config/nvmet
mkdir ports/0
cd ports/0
echo "ipv4" > addr_adrfam
echo 0.0.0.0 > addr_traaddr
echo 4420 > addr_trsvcid
echo tcp > addr_trtype

cd /sys/kernel/config/nvmet/subsystems
mkdir testnqn
echo 1 >testnqn/allow_any_host
mkdir testnqn/namespaces/1
```

```
cd testnqn
# replace the device name with the disk you want to export
echo "/dev/nvme0n1" > namespaces/1/device_path
echo 1 > namespaces/1/enable

ln -s "../../subsystems/testnqn" /sys/kernel/config/nvmet/ports/0/subsystems/testnqn
```

These steps ensure that the device is now exported using NVME over TCP. The next step is to detect this on the new laptop and connect the device:

```
nvme discover -t tcp -a <ip> -s 4420
nvme connectl-all -t tcp -a <> -s 4420
```

Finally, `nvme list` shows the device which is connected to the new laptop, and we can proceed with the next step, which is to do the disk copy.

Copying the Disk

I simply used the `dd` command to copy the root disk to my new laptop. Since the new laptop didn't have an Ethernet port, I had to rely only on WiFi, and it took about 7 and a half hours to copy the entire 512GB to the new laptop. The speed at which I was copying was about 18-20MB/s. The other option would have been to create an initial partition and file system and do an rsync of the root disk or use BTRFS itself for file system transfer.

```
dd if=/dev/nvme2n1 of=/dev/nvme0n1 status=progress bs=40M
```

Resizing Partition and LUKS Container

The final part was very easy. When I launched `parted`, it detected that the partition table does not match the disk size and asked if it can fix it, and I said yes. Next, I had to install `cloud-guest-utils` to get `growpart` to fix the second partition, and the following command extended the partition to the full 1TB:

```
growpart /dev/nvem0n1 p2
```

Next, I used `cryptsetup-resize` to increase the LUKS container size.

```
cryptsetup luksOpen /dev/nvme0n1p2 ENC
cryptsetup resize ENC
```

Finally, I rebooted into the disk, and everything worked fine. After logging into the system, I resized the BTRFS file system. BTRFS requires the system to be mounted for resize, so I could not attempt it in live boot.

```
btfs fielsystem resize max /
```

Conclussion

The only benefit of this entire process is that I have a new laptop, but I still feel like I'm using my existing laptop. Typically, setting up a new laptop takes about a week or two to completely get adjusted, but in this case, that entire time is saved.

An added benefit is that I learned how to export disks using NVME over TCP, thanks to my colleague. This new knowledge adds to the value of the experience.

Tagged as: clone, debian, nvmet,
