

Unifi OS

Tout ce qui concerne Unifi OS

- [Storage Upgrade](#)

Storage Upgrade

PC Preparation

You'll need a desktop PC with two SATA ports and available power, running Linux directly or via a live disk. I prefer Debian, but any distro should work. Using USB adapters or docks is *possible*, but the transfer will take longer. Run the computer on a battery backup to avoid starting over due to a power outage. It took me 13 hours to clone a 10TB disk directly in a computer.

Dream Machine Preparation

Export a config backup of all your applications!

For my purposes the console was only running Unifi Network, Unifi Protect, Unifi Talk, and Unifi InnerSpace.

Protect and Talk both store their recordings and voicemails on the 3.5" HDD. Network stores client NetFlow logs on it. I didn't find InnerSpace using it at all. We'll need to stop these applications and prevent them from starting back up while doing the upgrade. SSH into your console and run:

```
systemctl stop unifi unifi-talk unifi-protect ms
systemctl disable unifi unifi-talk unifi-protect ms
```

Verify that all of the applications have stopped with:

```
systemctl status unifi unifi-talk unifi-protect ms
```

Now that the application's services have stopped, we need to verify that nothing is accessing the HDD.

```
lsof /volume1/
```

That command will output a table of what processes are reading/writing to the HDD. If you still see anything listed there, the first column is the command from the offending service, stop/disable it if possible.

Once all services that use the drive have been stopped, we can unmount it with

```
umount /volume1
```

If the drive is not busy it will succeed. If you get an error, verify there is still nothing accessing the drive and troubleshoot from there.

With the drive unmounted we push in the removal latch, wait a few seconds for the drive to stop spinning, and then fully remove it.

I recommend going ahead and popping the new drive into the sled for quicker replacement.

While the drive is out it's important that your console stay powered up with the services stopped and disabled. **If you reboot your console it will try to fix itself and you won't be able to upgrade while keeping your data.**

Drive Cloning

Connect both drives to your linux computer and identify their device id:

```
lshw -class disk -class storage -short
```

The output should show you the model names and their dev id, something like `/dev/sda` or `/dev/sdb`.

Once you are 100% sure which drive is which we can start the clone.

In my situation the original 10TB drive was `/dev/sda` and the new 18TB drive was `/dev/sdb`.

There are many method to clone a disk but I like to use `pv`. Install it if you don't have it already.

The first device in the command below is the source, the second is the destination.

```
pv < /dev/sda > /dev/sdb
```

The above command uses `pv` to show you the cloning progress based on the size of the source disk. It will copy the entire disk, partition table and all. This process will take hours if using a large drive that's full. It took about 13 hours to clone a 10TB disk. With 13 hours of down time I'm sure you can understand why I wanted to find a way to do this while keeping the network up!

Expand the Partition

Once the drive has finished cloning we will need to expand the partition size.

Run `gdisk` with the dev ID of your new drive:

```
gdisk /dev/sdb
```

Execute a partition write to move the secondary GPT header from the middle of the drive to the end. Just enter `w` and press enter then confirm the action.

Once the partition header is fixed, we'll expand the size with `parted`:

```
parted /dev/sdb
```

Enter `print` to list the current partition layout. You should only see one listed.

Enter `resizepart 1 100%` Enter `q` to exit.

Once the partition has been maximized, we must check the filesystem and then increase it to the full size:

```
e2fsck -f /dev/sdb1  
resize2fs /dev/sdb1
```

That's it, you're done cloning!

Dream Machine Completion

Now that you're done cloning, reinsert the new larger HDD into the bay.

UnifiOS will automatically remount it to `/volume1`, you can validate it by running the `mount` command.

Once it's remounted navigate to the web UI's OS Settings → Storage to confirm it's visible, showing the correct size, and confirm it's compatible.

If all checks out, reenable and start the services we stopped before:

```
systemctl enable --now unifi unifi-talk unifi-protect ms
```

Your console should now be starting up the services and recording should have resumed for Protect. All your old data should be available.

[Source](#)