

An EndeavourOS based Simple Home Server.

July 23, 2020

Backing up enosServer. Preparing an External Storage Device

rsync should have been installed during OS installation. If the external DATA SSD was partitioned, formatted, and mounted by the script, then the /serverbkup mount point should have been created.

From a Linux client Computer, ssh into enosServer and login as root

```
# pacman -Q rsync      (should list rsync as installed, if not install with pacman)
rsync 3.2.2-2
# ls -l /
drwxrwxr-- root users  server          (edited for clarity)
drwxrwxr-- root users  serverbkup
```

Check /server and /serverbkup for ownership (root users) and permissions drwxrwxr--

If /serverbkup does not exist

```
# mkdir /serverbkup
```

If the permissions and ownership are incorrect:

```
# chown root:users /serverbkup
```

```
# chmod 774 /serverbkup
```

Partition and format the BKUP USB3 SSD

Ideally, the external USB 3 backup SSD drive should be the same size as the DATA SSD drive. If you try to rsync a 500 GB SSD with 130 GB of data on it TO a 250 GB backup SSD, things will not go well. Even though the data to be backed up is less than the 250 GB BKUP SSD. Trust me on this. The backup SSD must be the same size or larger than the DATA SSD.

Connect the external USB 3.0 backup SSD to the enosServer SBC.

```
# lsblk -o NAME,FSTYPE,FSSIZE,LABEL,MOUNTPOINT
```

NAME	FSTYPE	FSSIZE	LABEL	MOUNTPOINT
sda				
└─sda1	ext4	233.7G	DATA	/server
sdb				
└─sdb1	ext4	XXXG	XXXX	
mmcblk1				
└─mmcblk1p1	ext4	58.3G	Odroid-XU4	/

The Micro SD card or eMMC card has a partition mmcblk1p1 labeled Odroid-XU4 and has a mount point of / (root). That is the partition that contains the OS.

The device named sda has a partition of sda1 labeled DATA and has a mount point of /server. That is the DATA SSD.

The device named sdb has a partition of sdb1 with a random label and no mount point. That will become our BKUP SSD. Take note of the SSD's name. In this case sdb. Yours may be slightly different.

Format the external USB 3 BKUP SSD

```
# fdisk /dev/sdb          (change device name if necessary)
  Command: o              (thats lower case letter o --create an new empty DOS partition table)
  Command: n              (add a new partition)
    Partition type: p      (primary partition)
    Partition number: 1    (first partition)
    First sector:          (Enter to accept default)
    Last sector:           (Enter to accept default)
      Partition #1 contains a xxxx signature. (this warning may not appear, if so yes)
      Do you want to remove the signature? yes
Command: w                (write table to disk and exit)
```

```
# mkfs.ext4 -L BKUP /dev/sdb1    (the -L option sets the volume label for the partition)
```

The BKUP USB 3 SSD is now partitioned as a single partition and formatted to ext4. The SSD is ready to be mounted.

```
# mount /dev/sdb1 /serverbkup
```

```
# lsblk -o NAME,FSTYPE,FSSIZE,LABEL,MOUNTPOINT
```

NAME	FSTYPE	FSSIZE	LABEL	MOUNTPOINT
sda				
└─sda1	ext4	233.7G	DATA	/server
sdb				
└─sdb1	ext4	228.2G	BKUP	/serverbkup
mmcblk1				
└─mmcblk1p1	ext4	58.3G	Odroid-XU4	/

The BKUP SSD is now ready to use for backups.
To clean things up for the next page.

```
# umount /serverbkup
```

Next, on it's own page, so it makes a good cheat sheet, Performing a Backup.

PERFORMING A BACKUP OF THE DATA DRIVE

To do a backup connect the external BKUP SSD drive to an available USB port.
In a Linux client computer, ssh into enosServer as root.

```
# ll /server
```

should show a bunch of folders on your DATA SSD as it was mounted to /server at boot up

```
# ll /serverbkup
```

```
Total 0
```

Should show "Total 0" since the BKUP SSD is not mounted yet

As connecting USB Drives to different connectors may alter the device name, always determine which device name is being used for the USB BKUP Drive.

```
# lsblk -o NAME,FSTYPE,LABEL,MOUNTPOINT
```

NAME	FSTYPE	LABEL	MOUNTPOINT
sda			
└─sda1	ext4	DATA	/server
sdb			
└─sdb1	ext4	BKUP	
mmcblk1			
└─mmcblk1p1	ext4	Odroid-XU4	/

Now you know why specific Volume labels were suggested to be used when formatting the partitions on the storage devices. This easily shows which SSD has which device name.

Now we need to mount the BKUP SSD to /serverbkup. In this case /dev/sdb1

```
# mount /dev/sdb1 /serverbkup (adjust device name as required)
```

```
# ll /serverbkup
```

The first time the device is used after a format, you will probably only see the lost+found file.
If this device has been used previously for backups, you should see a lot of directories.
This proves the mount command was successful.

```
# su pshare (IMPORTANT: perform the backup as user pshare Do not backup as root)
```

```
$ rsync -av - -delete /server/ /serverbkup (a slash at the end of /server/ is very important)
```

```
$ exit (Exit back to root)
```

```
# umount /serverbkup
```

```
# ll /serverbkup
```

```
Total 0
```

Shows that the BKUP drive was un-mounted and safe to remove

unplug the BKUP SSD from the USB port.