

## Le Nas ( Network Attached Storage )

### Les différents systèmes de stockage, leurs avantages et inconvénients :

- **DAS (Direct Attached Storage)** : Ce système de stockage permet de connecter les disques directement à un seul serveur, offrant ainsi une solution rapide et locale. Idéal pour les applications nécessitant des performances élevées et une faible latence.
  - Avantages :
    - Performances élevées
    - Faible latence
  - Inconvénients :
    - Pas de partage sur un réseau
- **NAS (Network Attached Storage)** : Ce dispositif de stockage est connecté à un réseau local, permettant le stockage partagé et l'accès aux fichiers via divers protocoles.
  - Avantages :
    - Stockage partagé
    - Accès aux fichiers via le réseau
  - Inconvénients :
    - Performances parfois limitées en comparaison avec DAS
- **SAN (Storage Area Network)** : Ce réseau dédié au stockage connecte des serveurs et des dispositifs de stockage via des protocoles de transfert de blocs.
  - Avantages :
    - Hautes performances pour les transferts de blocs
    - Grande évolutivité
  - Inconvénients :
    - Coût élevé
    - Complexité de configuration
- **Baie de stockage** : Il s'agit d'un boîtier physique contenant des disques durs ou des SSD, souvent utilisé dans les centres de données ou les environnements d'entreprise.
  - Avantages :
    - Stockage centralisé et organisé
    - Grande capacité de stockage
  - Inconvénients :
    - Coût élevé initialement
    - Limitations physiques de capacité de stockage

### Pourquoi faire un NAS :

- Besoin d'une solution de stockage sécurisée (synchroniser toutes ses données multimédias notamment via la technologie RAID qui duplique les données sur les différentes baies)
- Besoin d'une grande capacité de stockage
- Volonté de créer un serveur autonome, capable de fonctionner de façon isolée
- Souhait de profiter d'un équipement hors tension, pouvant se mettre en veille de lui-même

### Composition d'un NAS :

- Une carte mère (si vous recyclez un vieux PC pour votre projet de création de NAS, prenez garde à ce qu'elle ne soit pas trop ancienne pour pouvoir gérer toutes vos applis) ;
- De la mémoire RAM
- Un processeur
- Un boîtier dont le nombre de baies est en adéquation avec le nombre de disques durs à installer
- Plusieurs câbles SATA et plusieurs câbles d'alimentation
- De kits de montage pour disque dur.

## Créer son NAS

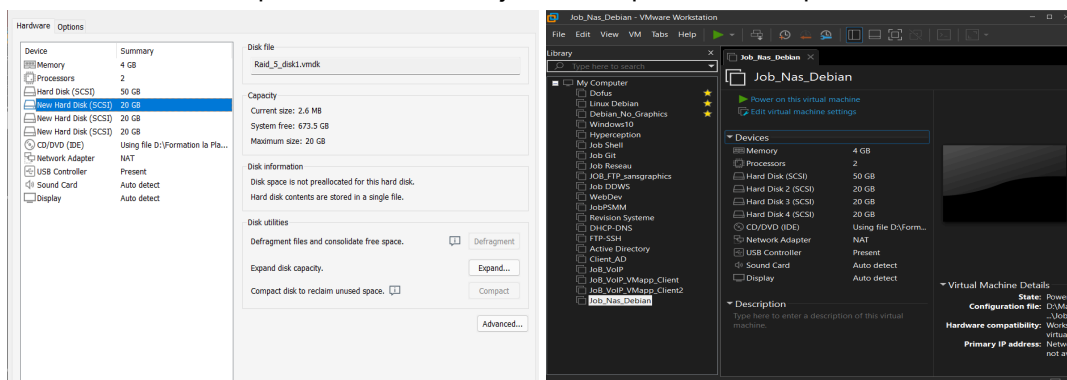
<https://www.macway.com/guide/20060/creer-serveur-nas>

<https://lapinlibre.net/geek/nas-sous-debian/>

## Le stockage en RAID

### Mise en Place du Raid et du Nas sur ma VM

1. Manipulation de base communes à toutes mes VM sans interface
  - Copie un VM sans interface graphique de debian12
  - Connection en Root avec admin comme Mdp
  - Suppression de l'utilisateur "x" via : `userdel x`
  - Suppression de son home via : `rm -r x`
  - Modification du fichier sources d'apt via : `nano /etc/apt.sources.list`
  - Mise à jour via : `apt update && apt upgrade`
  - Installation du groupe sudo via : `apt install sudo`
  - Affichage de l'ipv4 au démarrage via : `echo "IPv4 \4" >> /etc/issue`
  - Affichage de l'ipv6 au démarrage via : `echo "IPv6 \6" >> /etc/issue`
  - On autorise la connection au root avec putty via :  
`echo "PermitRootLogin Yes" > /etc/ssh/sshd_config.d/ssh.conf`
  - On redémarre la VM
2. Mise en Place du Raid qui permettra de Sécuriser les données de notre NAS
  - Dans les options de la VM on ajoute 3 disques minimum pour créer un RAID 5



Via la commande `ls -l /dev/sd*` on vérifie que nos disques sont bien pris en compte par notre système

```
root@debian:~# ls -l /dev/sd*
brw-rw---- 1 root disk 8,  0 20 mai  11:40 /dev/sda
brw-rw---- 1 root disk 8,  1 20 mai  11:40 /dev/sda1
brw-rw---- 1 root disk 8,  2 20 mai  11:40 /dev/sda2
brw-rw---- 1 root disk 8,  5 20 mai  11:40 /dev/sda5
brw-rw---- 1 root disk 8, 16 20 mai  11:40 /dev/sdb
brw-rw---- 1 root disk 8, 32 20 mai  11:40 /dev/sdc
brw-rw---- 1 root disk 8, 48 20 mai  11:40 /dev/sdd
root@debian:~#
```

Ici nous voyons notre disque sda et ses 3 partitions sda1/sda2 et sda5 qui contiennent notre système. Ainsi que sdb/sdc/sdd qui correspondent au 3 disques composant notre futur RAID 5

#### - Création du RAID5

Pour ce faire nous avons besoin des paquets `gdisk` qui est un gestionnaire de partitionnement de disques et `mdadm` pour la création du raid via : `apt install gdisk` et `apt install mdadm`.

On crée le périphérique `/dev/md0` (les volumes RAID ont des noms de type `/dev/md*` ), on lui dit que c'est du RAID 5, qu'il y a 3 disques en RAID `/dev/sdb` , `/dev/sdc` et `/dev/sdd` via la commande :  
`sudo mdadm --create /dev/md0 --level=raid1 --raid-devices=2 /dev/sdb /dev/sdc /dev/sdd`

```
root@debian:~# mdadm --create /dev/md0 --level=raid5 --raid-devices=3 /dev/sdb /dev/sdc /dev/sdd
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
root@debian:~#
```

On vérifie la création de notre raid via `cat /proc/mdstat`

```
root@debian:~# cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 sdd[3] sdc[1] sdb[0]
      41908224 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
root@debian:~#
```

## - Paramétrage du RAID5

Afin de fixer les paramètres de votre RAID et éviter que votre périphérique ne change de nom au prochain démarrage, il est conseillé de rajouter l'UUID de notre RAID au fichier mdadm.conf. Pour ce faire, on récupère l'UUID du raid via : mdadm --query --detail /dev/md0

```
root@debian:~# mdadm --query --detail /dev/md0
/dev/md0:
    Version : 1.2
    Creation Time : Mon May 20 12:02:47 2024
    Raid Level : raid5
    Array Size : 41908224 (39.97 GiB 42.91 GB)
    Used Dev Size : 20954112 (19.98 GiB 21.46 GB)
    Raid Devices : 3
    Total Devices : 3
    Persistence : Superblock is persistent

    Update Time : Mon May 20 12:04:32 2024
    State : clean
    Active Devices : 3
    Working Devices : 3
    Failed Devices : 0
    Spare Devices : 0


    Layout : left-symmetric
    Chunk Size : 512K

Consistency Policy : resync


    Name : debian:0 (local to host debian)
    UUID : f767efbe:9da0225f:6dec2a0a:e9e055dc
    Events : 18


   Number Major Minor RaidDevice State
    ----
    0         8      16           0 active sync  /dev/sdb
    1         8      32           1 active sync  /dev/sdc
    3         8      48           2 active sync  /dev/sdd
root@debian:~#
```

Puis nous modifions le fichier `/etc/mdadm/mdadm.conf` après le commentaire `# definitions of existing MD arrays`

```
# mdadm.conf
#
# !NB! Run update-initramfs -u after updating this file.
# !NB! This will ensure that initramfs has an uptodate copy.
#
# Please refer to mdadm.conf(5) for information about this file.
#
# by default (built-in), scan all partitions (/proc/partitions) and all
# containers for MD superblocks. alternatively, specify devices to scan, using
# wildcards if desired.
#DEVICE partitions containers

# automatically tag new arrays as belonging to the local system
HOMEHOST <system>

# instruct the monitoring daemon where to send mail alerts
MAILADDR root

# definitions of existing MD arrays
/dev/md0 level=raid5 num-devices=3 UUID : f767efbe:9da0225f:6dec2a0a:e9e055dc
# This configuration was auto-generated on Mon, 20 May 2024 11:56:52 +0200 by mkconf
```

Le système de démarrage de votre système a sa propre copie du fichier `/etc/mdadm/mdadm.conf`. Pour que les modifications sur ce fichier soient prises en compte au prochain démarrage, il faut également lancer la commande : `update-initramfs -u`

- Partitionnement, formatage et montage du disque

Un raid malgré qu'il soit composé de plusieurs disques correspond pour le système à un seul périphérique. Commençons par obtenir des informations sur nos disques avec l'option `gdisk -l` suivie du fichier périphérique à examiner.

```
root@debian:~# gdisk -l /dev/md0
GPT fdisk (gdisk) version 1.0.9

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.
Disk /dev/md0: 83816448 sectors, 40.0 GiB
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): CCB58929-5970-4990-9660-6F1C658CFFC2
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 83816414
Partitions will be aligned on 2048-sector boundaries
Total free space is 83816381 sectors (40.0 GiB)

Number  Start (sector)    End (sector)  Size      Code  Name
root@debian:~#
```

Sans option, gdisk est un programme interactif.

En tapant ? puis "Entrée", nous voyons la liste des commandes disponibles.

```
root@debian:~# gdisk /dev/md0
GPT fdisk (gdisk) version 1.0.9

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.

Command (? for help): ?
b          back up GPT data to a file
c          change a partition's name
d          delete a partition
i          show detailed information on a partition
l          list known partition types
n          add a new partition
o          create a new empty GUID partition table (GPT)
p          print the partition table
q          quit without saving changes
r          recovery and transformation options (experts only)
s          sort partitions
t          change a partition's type code
v          verify disk
w          write table to disk and exit
x          extra functionality (experts only)
?          print this menu
```

Entrez **n** pour ajouter une nouvelle partition, comme indiqué dans la liste des commandes.

```
Command (? for help): n
Partition number (1-128, default 1):
First sector (34-83816414, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-83816414, default = 83814399) or {+-}size{KMGTP}:
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'
```

Vous pouvez alors accepter toutes les valeurs par défaut. Ce qui nous amène à créer une partition numéro 1 qui commence au premier secteur disponible et termine au dernier secteur disponible (tout le disque donc). Un code indique alors que le système de fichiers de la partition sera un "système de fichiers Linux", ce qui nous va très bien.

```

Command (? for help): p
Disk /dev/md0: 83816448 sectors, 40.0 GiB
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 22A75870-DE21-49F4-8C4B-6BFF50100C1B
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 83816414
Partitions will be aligned on 2048-sector boundaries
Total free space is 4029 sectors (2.0 MiB)

Number  Start (sector)    End (sector)  Size      Code  Name
   1           2048         83814399   40.0 GiB   8300   Linux filesystem

```

N'hésitez pas à vérifier que votre nouvelle partition correspond bien à ce que nous voulons par la commande `p` puis enregistrez les changements sur le disque avec `w`.

```

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): y
OK; writing new GUID partition table (GPT) to /dev/md0.
The operation has completed successfully.
root@debian:~#

```

`gdisk -l /dev/md0` pour vérifier la création de notre partition

```

root@debian:~# gdisk -l /dev/md0
GPT fdisk (gdisk) version 1.0.9

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/md0: 83816448 sectors, 40.0 GiB
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 22A75870-DE21-49F4-8C4B-6BFF50100C1B
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 83816414
Partitions will be aligned on 2048-sector boundaries
Total free space is 4029 sectors (2.0 MiB)

Number  Start (sector)    End (sector)  Size      Code  Name
   1           2048         83814399   40.0 GiB   8300   Linux filesystem
root@debian:~#

```

blkid pour voir les disques / partitions de mon systèmes

```
root@debian:~# blkid
/dev/sdd: UUID="f767efbe-9da0-225f-6dec-2a0ae9e055dc" UUID_SUB="2c2e0e12-9e55-ce46-fa72-334a3269a7a8" LABEL="debian:0" TYPE="linux_raid_member"
/dev/sdb: UUID="f767efbe-9da0-225f-6dec-2a0ae9e055dc" UUID_SUB="29465790-3286-412c-2814-578711927906" LABEL="debian:0" TYPE="linux_raid_member"
/dev/md0p1: PARTLABEL="Linux filesystem" PARTUUID="c4a3011c-b42e-4cbb-bef2-26082bef7aad"
/dev/sr0: BLOCK_SIZE="2048" UUID="2023-07-22-12-15-07-00" LABEL="Debian 12.1.0 amd64 1" TYPE="iso9660" PTUUID="274631fb" PTTYPE="dos"
/dev/sdc: UUID="f767efbe-9da0-225f-6dec-2a0ae9e055dc" UUID_SUB="e492da84-3182-30dc-0987-6ef862398de2" LABEL="debian:0" TYPE="linux_raid_member"
/dev/sda5: UUID="a7b77f9f-7185-45d0-b089-1c8fa762467a" TYPE="swap" PARTUUID="afd9affc-05"
/dev/sda1: UUID="f8688c04-0c6b-4302-9b92-49e9a984392d" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="afd9affc-01"
```

À ce stade, vous avez juste écrit une table de partition au début de votre disque de données qui indique que ce disque contient une partition. Il vous faut maintenant la formater.  
Dans notre cas, il s'agit de md0p1.

Pour ce faire, on tape `mkfs -t ext4 /dev/md0p1`. La commande `mkfs` permet de créer un système de fichier. Avec `-t ext4`, on précise qu'on veut un système ext4 et on donne en argument le fichier device `/dev/md0p1` qu'on veut formater.

```
root@debian:~# mkfs -t ext4 /dev/md0p1
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 10476544 4k blocks and 2621440 inodes
Filesystem UUID: 5cc21bff-a5c5-4ed6-9da6-214273adaa34
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624

Allocating group tables: done
Writing inode tables: done
Creating journal (65536 blocks): done
Writing superblocks and filesystem accounting information: done

root@debian:~#
```

Votre nouveau disque dur est prêt. Avant de le monter nous pouvons optimiser une partition ext4 de manière à ne pas perdre d'espace sur notre raid via `tune2fs`.

Sur une partition dédiée à du stockage de données, vous n'avez pas besoin de blocs réservés. Vous allez donc supprimer cette réservation (option `-m 0`).

```
root@debian:~# tune2fs -m 0 /dev/md0p1
tune2fs 1.47.0 (5-Feb-2023)
Setting reserved blocks percentage to 0% (0 blocks)
root@debian:~#
```

**Ici il est important de retenir que `tune2fs` permet de gérer / optimiser à notre guise une partition ext4.**



Notre disque est prêt il faut maintenant le monter. Dans mon cas, je souhaite que les fichiers des utilisateurs soient enregistrés sur ce disque en particulier je vais donc le monter sur /home.

Pour ce faire, avec la commande `mount`, je vais monter la partition `/dev/md0p1` dans le répertoire `/home` en précisant que le système de fichier est de type `ext4`.

Ce qui me donne : `mount -t ext4 /dev/md0p1 /home`

Puis nous vérifions les disques montés sur notre vm via : `df -h`

```
root@debian:~# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev                1,9G      0  1,9G   0% /dev
tmpfs               389M    784K  388M   1% /run
/dev/sda1           48G    2,2G   44G   5% /
tmpfs               1,9G      0  1,9G   0% /dev/shm
tmpfs               5,0M      0  5,0M   0% /run/lock
tmpfs               389M      0  389M   0% /run/user/0
/dev/md0p1          40G    24K   40G   1% /home
root@debian:~#
```

Afin que notre nouveau disque dur puisse être monté automatiquement au démarrage du système, il faut rajouter la ligne suivante à la fin du fichier `/etc/fstab` :

`UUID="5cc21bff-a5c5-4ed6-9da6-214273adaa34" /home ext4 defaults 0 0`

```
GNU nano 7.2 /etc/fstab *
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=f8688c04-0c6b-4302-9b92-49e9a984392d / ext4 errors=remount-ro 0
# swap was on /dev/sda5 during installation
UUID=a7b77f9f-7185-45d0-b089-1c8fa762467a none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0

# Montage de la partition de mon RAID5 au démarrage
UUID="5cc21bff-a5c5-4ed6-9da6-214273adaa34" /home ext4 defaults 0 0
```

**RAPPEL L'UUID se récupère via la commande blkid**

Lors du redémarrage de ma VM je n'ai rencontré aucun problème, ce qui signifie que mon disque correspondant au raid5 est bien monté. Voilà ce que j'obtiens quand je vais un `df -h`.

```
Debian GNU/Linux 12 debian tty1

IPv4 192.168.198.135
IPv6 fe80::20c:29ff:fec0:fe27
debian login: root
Password:
Linux debian 6.1.0-21-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.90-1 (2024-05-03) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 20 11:40:57 CEST 2024 from 192.168.198.1 on pts/0
root@debian:~# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
udev                1,9G      0   1,9G   0% /dev
tmpfs               389M    780K   388M   1% /run
/dev/sda1           48G     2,2G   44G    5% /
tmpfs               1,9G      0   1,9G   0% /dev/shm
tmpfs               5,0M      0   5,0M   0% /run/lock
/dev/md0p1          40G     24K   40G    1% /home
tmpfs               389M      0   389M   0% /run/user/0
root@debian:~# _
```

3. Création des utilisateurs et installations/configuration des services permettant le partage  
<https://www.it-connect.fr/linux-configuration-dun-espace-de-stockage-securise-avec-sftp/>

Pour commencer il est nécessaire de créer un utilisateur qui aura les droits sudo pour administrer l'ensemble, la création des autres utilisateurs se fera par l'intermédiaire d'un script une fois tous les services configurés.

- `adduser --allow-bad-names LaPlateforme` avec comme mdp `LaPlateforme13`
- `adduser LaPlateforme sudo` pour l'ajouter en tant que sudoers
- et on redémarre pour appliquer les droits

```
root@debian:~# adduser --allow-bad-names LaPlateforme
Autoriser l'usage de noms d'utilisateur contestables.
Ajout de l'utilisateur « LaPlateforme » ...
Ajout du nouveau groupe « LaPlateforme » (1000) ...
Ajout du nouvel utilisateur « LaPlateforme » (1000) avec le groupe « LaPlateforme » (1000) ...
Création du répertoire personnel « /home/LaPlateforme » ...
Copie des fichiers depuis « /etc/skel » ...
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
Modifier les informations associées à un utilisateur pour LaPlateforme
Entrer la nouvelle valeur, ou appuyer sur ENTER pour la valeur par défaut
  NOM []: LaPlateforme
  Numéro de chambre []:
  Téléphone professionnel []:
  Téléphone personnel []:
  Autre []:
Cette information est-elle correcte ? [O/n]o
Ajout du nouvel utilisateur « LaPlateforme » aux groupes supplémentaires « users » ...
Ajout de l'utilisateur « LaPlateforme » au groupe « users » ...
root@debian:~# adduser LaPlateforme sudo
Ajout de l'utilisateur « LaPlateforme » au groupe « sudo » ...
Fait.
root@debian:~# shutdown -h now
root@debian:~# █
```