



RAID LINUX

MISE EN PLACE ET
CONFIGURATION

TANET Margaux

CPI 2024-2025

Table des matières

I-	Création des différents RAID.....	4
a)	RAID 0 : mise en place.....	4
b)	RAID 1 : mise en place.....	6
c)	RAID 5 : mise en place.....	7
II-	Phase de tests	9
a)	Test sur le RAID 0 : déconnection d'un disque	9
b)	Test sur le RAID 1 : déconnection d'un disque	11
c)	Test sur le RAID 5 : déconnection d'un disque	12
III-	Comparaison des RAID : Debian/ Windows Server	15
IV-	Conclusion	15

Pré requis :

- Machine virtuelle Debian
- Installation des disques et vérification
- 2 disques durs de 1Go chacun pour le RAID 0
- 2 disques durs de 1Go chacun pour le RAID 1
- 3 disques durs de 1Go chacun pour le RAID 5

Introduction :

Nous mettrons en place des RAID 0, RAID 1 et RAID 5 sur un serveur Debian. Nous réaliserons des tests afin de savoir dans quel cas nous perdons les données si c'est le cas.

	Nombre de disque minimum	Tolérance aux pannes	Espace utilisable
RAID 0	2	Aucune	2 disques de 1 Go = 2Go
RAID 1	2	1 disque peut tomber	50 %
RAID 5	3	1 disque peut tomber	Nombre de disques – 1 disque / capacité Go

❖ Qu'est-ce qu'un RAID logiciel ?

Le **RAID logiciel** est une méthode qui permet de gérer un RAID directement à partir du système d'exploitation, sans avoir besoin d'un matériel spécifique comme un contrôleur RAID matériel. Il utilise les ressources du processeur et de la mémoire de l'ordinateur pour organiser les disques en RAID.

❖ Comment fonctionne un RAID logiciel ?

Il regroupe plusieurs disques et les faire fonctionner comme un seul ensemble.

❖ Qu'est-ce que mdadm logiciel ?

Mdadm regroupe plusieurs disques pour former un RAID. Il remplace un contrôleur RAID matériel en gérant directement les disques via le système d'exploitation. Il surveille l'état du RAID et permet de reconstruire les données si un disque tombe en panne.

I- Création des différents RAID

L'installation du paquet « mdadm » permet de créer, de gérer et de surveiller l'ensemble de multidisques. On l'installe avec cette commande : **apt install mdadm -y**

```
root@debian:~# apt install mdadm -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
```

Pour voir l'ensemble des disques disponibles ainsi que leur taille, il faut taper la commande **fdisk -l**:

```
Disque /dev/sdb : 1 GiB, 1073741824 octets, 2097152 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
taille d'E/S (minimale / optimale) : 512 octets / 512 octets

Disque /dev/sdc : 1 GiB, 1073741824 octets, 2097152 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
taille d'E/S (minimale / optimale) : 512 octets / 512 octets

Disque /dev/sdd : 1 GiB, 1073741824 octets, 2097152 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
taille d'E/S (minimale / optimale) : 512 octets / 512 octets

Disque /dev/sde : 1 GiB, 1073741824 octets, 2097152 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
taille d'E/S (minimale / optimale) : 512 octets / 512 octets
```

On voit ici une partie de mes disques prêt pour les différents RAID et faisant tous 1Go.

a) RAID 0 : mise en place

Je vais créer ce que l'on appelle une « grappe de disques ». C'est l'ensemble des disques durs qui sont regroupés pour former un RAID. Je choisis les disques **sdb** et **sdc** pour le RAID 0. J'indique tout cela grâce à cette commande :

Mdadm – create – verbose /dev/md0 - - level=0 - -raid-devices=2 /dev/sdb /dev/sdc

Description de la commande :

Mdadm comme décrit, plus haut, permet de gérer l'ensemble des RAID.

- - create permet de créer/ initialiser un nouveau volume de RAID.

-- *verbose* permet de connaître des informations pendant le processus de création, sur ce qu'il est en train de faire.

-- *level=0* signifie le niveau du RAID. Où 0 veut dire « striping » ce qui revient à RAID 0.

-- *raid-devices=2* précise le nombre de disques qui vont faire partie de la grappe de RAID.

/dev/sdb /dev/sdc : « **sdb** » et « **sdc** » sont les deux disques que je souhaite utiliser pour mon RAID 0. */dev* correspond au chemin vers les disques.

Md0 correspond à mon RAID 0.

```
root@debian:~# mdadm --create --verbose /dev/md0 --level=0 --raid-devices=2 /dev/sdb /dev/sdc
mdadm: chunk size defaults to 512K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

Ma grappe de disques est bien créée pour le RAID 0.

- La commande « **lsblk** » permet d'afficher des informations sur les disques. Je réalise cette commande :

```
root@debian:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
fd0          2:0    1    4K  0 disk
sda          8:0     0   40G  0 disk
├─sda1       8:1     0   39G  0 part  /
├─sda2       8:2     0    1K  0 part
└─sda5       8:5     0   975M  0 part  [SWAP]
sdb          8:16    0    1G  0 disk
└─md0        9:0     0    2G  0 raid0
sdc          8:32    0    1G  0 disk
└─md0        9:0     0    2G  0 raid0
sdd          8:48    0    1G  0 disk
sde          8:64    0    1G  0 disk
sdf          8:80    0    1G  0 disk
sdg          8:96    0    1G  0 disk
sdh          8:112   0    1G  0 disk
sr0         11:0    1 1024M  0 rom
```

Je vois que mes deux disques ont été ajoutés dans le RAID 0 (md0). Je me retrouve avec 2 Go tout est correct.

- Prochaine étape, la configuration pour formater le fichier avec le système de fichiers **ext4**. Cela se fait avec cette commande : **mkfs.ext4 /dev/md0**

Avec la commande **mkfs.ext4**, un système de fichiers ext4 va être créé sur le volume */dev/md0*.

```

root@debian:~# mkfs.ext4 /dev/md0
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 523264 4k blocks and 130816 inodes
Filesystem UUID: 9a03e04d-7fa9-4d75-a63f-129283eba52e
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

```

b) RAID 1 : mise en place

Pour la mise en place du RAID 1, cela se fait de la même manière qu'un RAID 0.

- Création de ma grappe de disque avec mes disques **sdd** et **sde** de 1 Go :

Voici la commande :

Mdadm - -create - -verbose /dev/md1 - -level=1 - -raid-devices=2 /dev/sdd /dev/sde

Pour le niveau de RAID (« level »), indiquer « =1 » comme nous passons en RAID 1.

- Visualisation des disques et raids grâce à la commande « **lsblk** »

```

root@debian:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
fd0          2:0    1   4K  0 disk
sda          8:0     0  40G  0 disk
├─sda1       8:1     0   39G  0 part /
├─sda2       8:2     0    1K  0 part
└─sda5       8:5     0  975M  0 part [SWAP]
sdb          8:16    0    1G  0 disk
└─md0        9:0     0    2G  0 raid0
sdc          8:32    0    1G  0 disk
└─md0        9:0     0    2G  0 raid0
sdd          8:48    0    1G  0 disk
└─md1        9:1     0    2G  0 raid1
sde          8:64    0    1G  0 disk
└─md1        9:1     0    2G  0 raid1
sdf          8:80    0    1G  0 disk
sdg          8:96    0    1G  0 disk
sdh          8:112   0    1G  0 disk
sr0         11:0    1 1024M  0 rom

```

On voit ici que les disques sélectionnés font bien partie du RAID 1 avec 2 Go au total, ce qui est correct comme chaque disque fait 1Go.

- Modification du format des disques pour le RAID 1 : **mkfs.ext4 /dev/md1**

```

root@debian:~# mkfs.ext4 /dev/md1
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 523264 4k blocks and 130816 inodes
Filesystem UUID: 471ad67e-881b-4a8c-a54b-9b0439390dde
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

```

c) RAID 5 : mise en place

- Création de la grappe de disques pour le RAID 5 avec les disques **sdf**, **sdg** et **sdh**.

La commande : **Mdadm --create --verbose /dev/md5 --level=5 --raid-devices=3 /dev/sdf /dev/sdg /dev/sdh**

Ici, le niveau de disque a changé : level=5 pour RAID 5 et le nombre de disques aussi au niveau de --raid-devices=3 comme il y a 3 disques.

```

root@debian:~# mdadm --create --verbose /dev/md5 --level=5 --raid-devices=3 /dev/sdf /dev/sdg /dev/sdh
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 1046528K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md5 started.

```

- Visualisation des disques et RAID grâce à la commande « **lsblk** »

sdf	8:80	0	1G	0	disk
└md5	9:5	0	2G	0	raid5
sdg	8:96	0	1G	0	disk
└md5	9:5	0	2G	0	raid5
sdh	8:112	0	1G	0	disk
└md5	9:5	0	2G	0	raid5

Pour le RAID 5, mes 3 disques font chacun 1 Go avec un total de 2 Go pour les 3.

En effet, la volumétrie totale est égale au **nombre de disques – 1 disque X la capacité d'un disque** : 3 disques – 1 disque X 1 Go ce qui revient donc à **2 Go** au total.

- Modification du format des disques pour le RAID 5 : **mkfs.ext4 /dev/md5**

```
root@debian:~# mkfs.ext4 /dev/md5
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 523264 4k blocks and 130816 inodes
Filesystem UUID: 6b36666d-d093-4450-a4ee-593baa4492dd
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

Une fois cette attribution des disques faite avec chaque RAID, nous pouvons vérifier cela dans la **gestion des disques** sous Debian :



Je peux voir ici les disques que j'ai créés avant et qui sont maintenant attribués à chaque RAID ainsi que les différents RAID que j'ai pu mettre en place.

II- Phase de tests

Avant chaque test, un fichier test.txt a été créer sur chaque disque.

```
margaux@debian: /media/margaux/a92ff59c-bbf9-4652-8263-97a009504316$ ls
lost+found
margaux@debian: /media/margaux/a92ff59c-bbf9-4652-8263-97a009504316$ su -
Mot de passe :
root@debian:~# ls
test.txt
root@debian:~# s
```

	Debian GNU/Linux	33,3 Go / 40,9 Go de disponibles	/
	Volume de 2,1 Go	1,9 Go / 2,1 Go de disponibles	/dev/md0
	Volume de 1,1 Go	965,0 Mo / 1,0 Go de disponibles	/dev/md1
	Volume de 1,1 Go		/dev/md11
	Volume de 2,1 Go	1,9 Go / 2,1 Go de disponibles	/dev/md5

a) Test sur le RAID 0 : déconnection d'un disque

Avant chaque déconnexion, réaliser le statut sur le disque qui va être déconnecté :

La commande pour réaliser cette action : **mdadm --detail /dev/md0**

```
root@debian:~# mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
    Creation Time : Thu Jan 30 13:38:14 2025
    Raid Level : raid0
    Array Size : 2093056 (2044.00 MiB 2143.29 MB)
    Raid Devices : 2
    Total Devices : 2
    Persistence : Superblock is persistent

    Update Time : Thu Jan 30 13:38:14 2025
    State : clean
    Active Devices : 2
    Working Devices : 2
    Failed Devices : 0
    Spare Devices : 0

    Layout : -unknown-
    Chunk Size : 512K

Consistency Policy : none
```

Number	Major	Minor	RaidDevice	State
0	8	16	0	active sync /dev/sdb
1	8	32	1	active sync /dev/sdc

Les points intéressants pour regarder le statut du disque :

- State= clean
 - Mes deux disques sont activés.
- Commençons par la déconnexion du disque

La commande a réalisé pour la déconnexion du disque : **mdadm --fail /dev/md0 /dev/sdb**

J'indique ici que je vais déconnecter le disque **sdb** faisant partie du RAID 0 (*md0*).

Je ressaisis la commande pour avoir en détail le statut de mon disque : **mdadm --detail /dev/md0**

```
State : broken
Active Devices : 2
Working Devices : 2
Failed Devices : 0
Spare Devices : 0

Layout : -unknown-
Chunk Size : 512K

Consistency Policy : none

Name : debian:0 (local to host debian)
UUID : 71aec72:01b4c948:edd02b1e:b64113c8
Events : 0
```

Il est indiqué que le disque est cassé, il ne fonctionne donc plus.

Quand je me rends dans la gestion de disque pour accéder à mon volume RAID 0, il est impossible d'y accéder. Je n'ai donc plus accès à mon lecteur et mes données sont perdues.



b) Test sur le RAID 1 : déconnection d'un disque

Avant chaque déconnexion, réaliser le statut sur le disque qui va être déconnecté :

La commande pour réaliser cette action : **mdadm - -detail /dev/md1**

Le disque cible que je veux déconnecter est **sdd** sur le RAID 1.

La commande a réalisé pour déconnecter le disque **sdd** est la suivante **mdadm - -fail /dev/md1 /dev/sdd**.

Je réalise de nouveau la commande **mdadm - -detail /dev/md1** pour savoir le statut de mon disque et j'obtiens ce résultat :

```
Used Dev Size : 1046528 (1022.00 MiB 1071.64 MB)
Raid Devices : 2
Total Devices : 2
Persistence : Superblock is persistent

Update Time : Thu Jan 30 14:54:47 2025
State : clean, degraded
Active Devices : 1
Working Devices : 1
Failed Devices : 1
Spare Devices : 0

Consistency Policy : resync

Name : debian:1 (local to host debian)
UUID : df78c233:b61438d5:5bee3d2c:4badbb12
Events : 23

Number Major Minor RaidDevice State
 0         8      32         0  active sync  /dev/sdc
-          0         0         1  removed
 1         8      48         -  faulty   /dev/sdd
@debian:~#
```

On voit que le statut du disque en question est dégradé et que ce même disque n'est plus actif (« faulty »). De plus, il est indiqué qu'un de mes disques a rencontré une erreur /panne (« failed devices=1 »).

Pour vérifier si j'accède toujours à mon RAID, je me rends dans mon gestionnaire de disque et je tente d'accéder au RAID 1. Cela fonctionne, je regarde ensuite sur le disque si mon fichier test.txt est toujours là : le fichier est toujours présent.



```
margaux@debian: /media/margaux/a92ff59c-bbf9-4652-8263-97a009504...$ su -
Mot de passe :
root@debian:~# ls
test.txt
root@debian:~#
```

c) Test sur le RAID 5 : déconnection d'un disque

Avant chaque déconnexion, réaliser le statut sur le disque qui va être déconnecté :

La commande pour réaliser cette action : **mdadm - -detail /dev/md5**

```
margaux@debian: ~  
Raid Level : raid5  
Array Size : 2093056 (2044.00 MiB 2143.29 MB)  
Used Dev Size : 1046528 (1022.00 MiB 1071.64 MB)  
Raid Devices : 3  
Total Devices : 3  
Persistence : Superblock is persistent  
  
Update Time : Thu Jan 30 14:51:42 2025  
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:5 (local to host debian)  
UUID : 4bd66faa:9a5b46da:b97ec3b3:a31ed0e8  
Events : 20  
  
Number Major Minor RaidDevice State  
0 8 64 0 active sync /dev/sde  
1 8 80 1 active sync /dev/sdf  
3 8 96 2 active sync /dev/sdg
```

Tous les disques sont actifs et sur chacun d'eux il y a aucune erreur détectée.

Le disque cible que je veux déconnecter est **sdf** sur le RAID 5.

La commande à réaliser pour déconnecter le disque sdf est la suivante **mdadm - -fail /dev/md5 /dev/sdf**.

Je réalise de nouveau la commande **mdadm - -detail /dev/md5** pour savoir l'état de mon disque après la déconnexion et j'obtiens ce résultat :

```
root@debian:~# mdadm - -fail /dev/md127 /dev/sdf  
mdadm: set /dev/sdf faulty in /dev/md127
```

```

margaux@debian: ~
Used Dev Size : 1046528 (1022.00 MiB 1071.64 MB)
Raid Devices : 3
Total Devices : 3
Persistence : Superblock is persistent

Update Time : Thu Jan 30 15:57:37 2025
State : clean, degraded
Active Devices : 2
Working Devices : 2
Failed Devices : 1
Spare Devices : 0

Layout : left-symmetric
Chunk Size : 512K

Consistency Policy : resync

Name : debian:5 (local to host debian)
UUID : 4bd66faa:9a5b46da:b97ec3b3:a31ed0e8
Events : 22

Number Major Minor RaidDevice State
0 8 64 0 active sync /dev/sde
- 0 0 1 removed
3 8 96 2 active sync /dev/sdg
1 8 80 - faulty /dev/sdf

```

Dans le détail, à la ligne « state », un des disques est dégradé. En bas du détail, on peut voir quel disque a été impacté. C'est le disque **sdf** comme choisi, qui n'est plus actif (« faulty »). De plus, il est indiqué qu'un de mes disques a rencontré une erreur /panne (« failed devices=1 »).

Au niveau des données, avec la commande « lsblk » aucune perte est encore visible :

sde	8:64	0	1G	0	disk
└md127	9:127	0	2G	0	raid5
sdf	8:80	0	1G	0	disk
└md127	9:127	0	2G	0	raid5
sdg	8:96	0	1G	0	disk
└md127	9:127	0	2G	0	raid5

Si on se rend maintenant dans la gestion des disques, sur le RAID 5 pour regarder si le fichier est toujours là :

```
margaux@debian: /media/margaux/6b36666d-d093-4450-a4ee-593baa44...
margaux@debian: /media/margaux/6b36666d-d093-4450-a4ee-593baa4492dd$ SU
bash: SU : commande introuvable
margaux@debian: /media/margaux/6b36666d-d093-4450-a4ee-593baa4492dd$ su
Mot de passe :
root@debian:~# ls
test.txt
root@debian:~#
```

Le fichier est bien présent, il y a seulement un disque dégradé, mais les données ne sont pas perdues.

Bilan de l'état des disques ayant été déconnectés :

La commande **cat /proc/mdstat** permet d'afficher l'état des ensemble RAID :

```
root@debian:~# cat /proc/mdstat
Personalities : [raid0] [raid1] [raid6] [raid5] [raid4] [linear] [multipath] [raid10]
md126 : active raid1 sdi[1] sdh[0]
      1046528 blocks super 1.2 [2/2] [UU]

md127 : active raid5 sdg[3] sde[0] sdf[1](F)
      2093056 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/2] [U_U]

md1 : active raid1 sdc[0] sdd[1](F)
      1046528 blocks super 1.2 [2/1] [U_]
```

Le md127 est mon RAID 5.

Au niveau du RAID 5 (md127), **Sdf [1] (F)** indique que le disque **sdf** est en panne, **[u_u]** veut dire que le raid fonctionne en mode dégradé, c'est-à-dire qu'il y a un disque en moins mais encore fonctionnel comme il y a la parité sur les disques. Pour aller plus loin, il faudrait remplacer le disque **sdf** afin de reconstruire le raid cela reste encore possible pour le raid 5. Aucune perte de données.

Pour le RAID 1, **Sdd[1] (f)** signifie que le disque **sdd** est en panne, **[U_]** veut dire qu'un seul disque fonctionne. Le RAID est toujours accessible, mais il n'y a plus aucune redondance. Aucune perte de données.

Pour le RAID 0, un des deux disques est en panne. Comme un des deux disques ne fonctionne plus cela entraîne la perte des données comme il n'y a pas de redondance sur un raid 0.

III- Comparaison des RAID : Debian/ Windows Server

Pour le RAID 0, sous Debian, il n'y a plus d'accès possible au disque et il y a une perte totale des données. Sous Windows, le lecteur ne s'affiche plus et nous perdons aussi les données.

Pour le RAID 1, sous Debian, le disque est dégradé, il n'y a aucune perte de données et on peut reconstruire un disque manuellement avec **mdadm**. Sous Windows, le disque est aussi dégradé mais les données sont toujours accessibles grâce à la redondance. Il est possible de mettre en place un processus de reconstruction du disque.

Même chose pour le RAID 5, sous Debian, le disque est dégradé la reconstruction du disque reste possible, aucune perte de données. Sous Windows, le disque est également dégradé, la reconstruction du disque reste possible et aucune perte de donnée grâce à la parité.

IV- Conclusion

Dans ce document, nous avons mis en place différents RAID sous Debian, où nous provoquons à chaque fois la perte d'un disque volontairement. Nous avons vu que pour le RAID 0, il était impossible d'y accéder et donc l'accès aux données est impossible en déconnectant un des deux disques. Pour le RAID 1 et le RAID 5, il est toujours possible d'accéder aux données malgré la perte d'un disque. Il suffira par la suite de reconstruire un nouveau disque pour le bon fonctionnement de ces RAID.