

# Debian

Debian est un système d'exploitation libre et gratuit, principalement basé sur Linux. Créé en 1993, il est connu pour sa stabilité, sa sécurité et son large choix de logiciels. Debian supporte de nombreuses architectures matérielles et sert souvent de base à d'autres distributions comme Ubuntu.

- [Commandes communes](#)
- [Mise à jour](#)
- [Étendre une partition sur Debian \(MBR\) sans perte de données](#)

# Commandes communes

## Utiliser la commande `ss`

**tag** : <voir port>

La commande `ss` est un outil moderne et rapide pour afficher les connexions réseau. Vous pouvez l'utiliser pour lister les ports ouverts.

Pour afficher tous les ports ouverts et les services qui les utilisent :

```
sudo ss -tuln
```

- `-t` : Affiche les connexions TCP.
- `-u` : Affiche les connexions UDP.
- `-l` : Affiche les ports à l'écoute (Listening).
- `-n` : Affiche les adresses et les ports sous forme numérique (sans résolution de noms).

Exemple de sortie :

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port
tcp	LISTEN	0	128	*:22	*:*
tcp	LISTEN	0	128	127.0.0.1:631	*:*
udp	UNCONN	0	0	0.0.0.0:123	*:*

Cela vous donne une vue d'ensemble des ports ouverts et des adresses locales auxquelles ils sont liés.

Exemple pour un port spécifique (par exemple, le port 8080) :

```
sudo ss -tuln | grep ':8080'
```

Cette commande affiche les connexions TCP et UDP en écoute et filtre celles qui utilisent le port 8080.

Si vous souhaitez également afficher le PID, ajoutez l'option `-p` :

```
sudo ss -tulnp | grep ':8080'
```

## Utiliser la commande `ip -c a`

**tag :** <voir ip, voir adresse ip>

`ip -c a` (ou `ip -c addr`) affiche toutes les interfaces réseau et leurs informations, avec une sortie colorée pour améliorer la lisibilité.

```
ip -c a
```

### Exemple de sortie de `ip -c a` :

Voici un exemple de ce à quoi pourrait ressembler la sortie d'une commande `ip -c a` :

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen
1000
    inet 192.168.1.10/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe2a:0bfa/64 scope link
        valid_lft forever preferred_lft forever
```

- `lo` est l'interface de boucle locale (loopback), avec l'adresse `127.0.0.1`.
- `eth0` est une interface Ethernet avec une adresse IPv4 (`192.168.1.10`) et une adresse IPv6 (`fe80::...`).

Avec l'option `-c`, ces informations sont colorées pour les rendre plus faciles à analyser visuellement.

### En résumé :

- `ip` : Utilitaire pour gérer les interfaces réseau, les routes, les adresses IP, etc.
- `-c` : Active la coloration de la sortie.
- `a` : Affiche les adresses réseau et les interfaces.

La commande `ip -c a` est donc un moyen simple et rapide de visualiser l'état des interfaces réseau de votre système avec une sortie colorée pour une lecture plus facile.

# Mise à jour

## ? Commande sans risque (mise à jour des index uniquement)

```
sudo apt update
```

Met à jour la liste des paquets disponibles sans installer quoi que ce soit. Aucun changement sur le système.

## ?? Commande plus complète (mise à jour des paquets installés)

```
sudo apt update && sudo apt upgrade -y
```

Met à jour la liste des paquets **et installe les mises à jour disponibles** pour les paquets déjà installés. Le `-y` permet de valider automatiquement les changements.

## ? Commande ultra complète (mise à jour + nettoyage)

```
sudo apt update && sudo apt full-upgrade -y && sudo apt autoremove --purge -y && sudo apt autoclean
```

En plus de la mise à jour : `full-upgrade` gère les changements de dépendances (peut installer ou supprimer des paquets). `autoremove --purge` supprime les paquets inutiles et leurs fichiers de config. `autoclean` nettoie les fichiers `.deb` obsolètes.

# Étendre une partition sur Debian (MBR) sans perte de données

Ce guide explique comment étendre une partition / (root) sous Debian lorsque le disque virtuel a été agrandi dans l'hyperviseur, mais que la partition principale est bloquée par une partition étendue contenant le swap (`sda2` + `sda5`).

Fonctionne pour :

- VMware, Proxmox, Hyper-V, VirtualBox
- Debian 10 → Debian 13+
- Disques MBR (label `dos`)

---

## ? Avant de commencer

Vérifier que le disque virtuel a déjà été agrandi dans l'hyperviseur :

```
lsblk
```

Vous devez voir par ex. :

```
sda 50G └─sda1 23G
```

---

## 1?? Désactiver le swap existant

```
swapoff -a
```

---

## 2?? Modifier la table de partitions avec fdisk

Lancer `fdisk` :

```
fdisk /dev/sda
```

Dans `fdisk`, effectuer exactement les actions suivantes :

## 1. Supprimer l'ancienne swap (sda5)

```
d  
5
```

## 2. Supprimer la partition étendue (sda2)

```
d  
2
```

## 3. Supprimer la partition root (sda1)

⚠ Sans danger tant que le secteur de début est conservé.

```
d  
1
```

## 4. Recréer /dev/sda1 sur tout le disque

```
n  
p  
1  
2048 ← le même secteur de début qu'avant  
<Entrée> ← fin par défaut = fin du disque
```

## 5. Remettre le drapeau boot

```
a  
1
```

## 6. Sauvegarder et quitter

```
w
```

Un redémarrage peut être nécessaire :

```
reboot
```

---

## 3?? Étendre le système de fichiers ext4

Une fois la machine revenue :

```
resize2fs /dev/sda1
```

Le système de fichiers s'adapte alors à la nouvelle taille (50 Go dans cet exemple).

Vérifier :

```
df -h
```

## 4?? Créer un swap via un fichier (recommandé)

Un swapfile est plus simple et plus flexible qu'une partition swap.

Création du fichier swap :

```
fallocate -l 1G /swapfile && chmod 600 /swapfile && mkswap /swapfile && swapon /swapfile
```

## Rendre le swap permanent

Ajouter dans `/etc/fstab` :

```
echo '/swapfile none swap sw 0 0' >> /etc/fstab
```

ou

```
/swapfile none swap sw 0 0
```

## 5?? Vérifications finales

```
lsblk && df -h && free -h
```

Vous devriez voir :

- `/dev/sda1` = ~50 Go monté sur `/`
- `/swapfile` actif

## ? Résultat final

Votre système Debian utilise maintenant toute la taille du disque agrandi, sans perte de données, avec un swap propre et moderne via fichier.

---

## ? Bonus : résolution du problème “commandes sbin introuvables”

Si des commandes comme `fdisk`, `parted`, `ip`, `reboot`... n'étaient plus accessibles :

Ajouter ceci dans `/root/.bashrc` :

```
export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

Puis :

```
source /root/.bashrc
```