

Dépannage Linux

- [Connecter un clavier Keychron K3 Pro en bluetooth sous linux](#)
- [Blacklister des secteur de mémoire corrompue](#)
- [Lecture du SMART en USB UAS](#)
- [Vérifier l'intégrité de la mémoire flash d'une carte SD / clé USB](#)
- [Vérifier l'état de la batterie sous Linux](#)
- [Installation manuelle de RustDesk sous LMDE 7](#)

Connecter un clavier Keychron K3 Pro en bluetooth sous linux

Il semble que le firmware de ce clavier nécessite une suite d'actions dans un ordre spécifique pour fonctionner en bluetooth sous linux.

Voici l'ordre exact des étapes à réaliser dans Bluetooth-manager (l'outil le plus courant, mais les étapes sont les mêmes avec un autre gestionnaire bluetooth):

- Supprimer le clavier des périphériques bluetooth connus s'il y est
- Sur le clavier, maintenez Fn+le chiffre à assigner à cet ordinateur (le clavier peut se connecter en bluetooth à 3 appareils différents). Le clavier passe en mode jumelage.
- Dans le gestionnaire, actualiser la liste des périphériques bluetooth.
- Ne vous connectez pas directement au clavier, faites un clic droit dessus dans la liste, et faire "Jumeler/Appairer".
- Le jumelage prend quelques secondes, suite à quoi le clavier redémarre, patientez jusqu'à ce qu'il réapparaisse dans la liste des périphériques bluetooth.
- Maintenant, connectez-vous au clavier en double-cliquant dessus dans la liste.
- Une fois le clavier connecté, clic-droit dessus dans la liste, et indiquez "Faire confiance/Approuver".

Blacklister des secteur de mémoire corrompue

Intro

Si vous avez des instabilités, que vous faites un memtest86+ et que vous avez des erreurs, la solution est de remplacer la ram, mais vu le prix de la RAM aujourd'hui et les rams soudées à la carte mere de plus en plus démocratisées, ce n'est pas toujours possible. Donc, pourquoi ne pas empêcher le système de s'en servir au demarrage ?

badram

Badram est un utilitaire dans GRUB permettant de faire juste ca. Il suffit pour cela de lui passer en argument l'adresse mémoire, suivi du masque, le tout en hexadécimal dans le fichier de config par défaut de grub (/etc/default/grub) dans la clé "BAD_RAM=" et faire un petit grub_update

Récupérer les secteur et masques

Facile il suffit de lancer memtest86+, et dans les options de signaler qu'on veut le retour en format badram (F1, F4, F4) et de recopier les secteurs,masques

Résolution des problèmes

Il peut arriver que grub refuse de demarrer après la mise en place du badram. C'est un bug connu sur les kernels 64bits. Il suffit de delancer le système en passant par SuperGrubDisk, remplacer le premier F des masques par des 7, et relancer update-grub

Lecture du SMART en USB UAS

SAT avec UAS sous Linux — Guide de dépannage pour smartmontools

Problème

Dans certaines situations, le pilote Linux `uas` désactive les transferts SAT (ATA passthrough), ce qui empêche `smartmontools` (et d'autres outils comme `hdparm`) de communiquer avec un disque SATA connecté en USB.

Le symptôme typique est une erreur de ce type :

```
Read Device Identity failed: scsi error unsupported field in scsi command
A mandatory SMART command failed: exiting.
```

Ou encore :

```
Unknown USB bridge [0xXXXX:0xXXXX (0xXXXX)]
Please specify device type with the -d option.
```

Pourquoi ?

Certains chipsets bridges USB-SATA ne gèrent pas correctement les commandes SAT en mode UAS. Le noyau Linux active alors automatiquement le flag `NO_ATA_1X` pour ces chipsets, ce qui désactive le passthrough SAT et bloque smartmontools.

Les appareils concernés incluent notamment les boîtiers Seagate, les chipsets Initio INIC-3069 et VIA VL711.

Étape 1 — Identifier les IDs USB du périphérique

```
lsusb
```

Repérez la ligne correspondant à votre disque, par exemple :

```
Bus 002 Device 003: ID XXXX:YYYY Nom du fabricant
```

- `XXXX` = vendor ID
- `YYYY` = product ID

Notez ces deux valeurs, elles seront utilisées dans toutes les commandes suivantes.

Étape 2 — Vérifier le pilote utilisé

```
lsusb -t
```

Si votre périphérique utilise le pilote `uas`, c'est probablement la cause du problème.

Étape 3 — Forcer le type de device avec smartctl

Avant tout, essayez simplement de forcer le type SAT :

```
sudo smartctl -d sat -a /dev/sdX
```

Si ça échoue, tentez avec une longueur de commande différente :

```
sudo smartctl -d sat,12 -a /dev/sdX  
sudo smartctl -d sat,16 -a /dev/sdX
```

Remplacez `/dev/sdX` par le chemin réel de votre disque (`/dev/sda`, `/dev/sdb`, etc.).

Étape 4 — Basculer temporairement vers usb-storage

Si les commandes ci-dessus échouent, désactivez temporairement UAS pour votre périphérique :

```
sudo modprobe -r uas
sudo modprobe usb-storage quirks=XXXX:YYYY:u
```

Débranchez puis rebranchez le disque, puis relancez smartctl :

```
sudo smartctl -d sat -a /dev/sdX
```

Étape 5 — Solutions permanentes

Option A — Désactiver UAS (recommandé)

Crée un fichier de configuration modprobe :

```
sudo nano /etc/modprobe.d/usb-storage-quirks.conf
```

Ajoutez :

```
options usb-storage quirks=XXXX:YYYY:u
```

Le flag `:u` active `IGNORE_UAS` — le noyau utilisera `usb-storage` à la place de `uas`.

Option B — Forcer le SAT passthrough en gardant UAS

⚠ À vos risques et périls — peut causer des instabilités sur certains chipsets.

Même fichier, mais avec le flag `:t` qui force le passthrough SAT malgré UAS :

```
options usb-storage quirks=XXXX:YYYY:t
```

Pour plusieurs périphériques

Séparez les entrées par des virgules :

```
options usb-storage quirks=XXXX:YYYY:u,AAAA:BBBB:u
```

Mise à jour de l'initramfs

Après avoir créé ou modifié le fichier de configuration :

```
sudo update-initramfs -u
```

Option via GRUB

Modifiez `/etc/default/grub` et ajoutez le paramètre à `GRUB_CMDLINE_LINUX_DEFAULT` :

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash usb-storage.quirks=XXXX:YYYY:u"
```

Puis mettez à jour GRUB :

```
sudo update-grub
```

Référence des flags

Flag	Valeur hex	Lettre	Effet
<code>IGNORE_UAS</code>	<code>0x00800000</code>	<code>:u</code>	Désactive UAS, bascule sur usb-storage
<code>NO_ATA_IX</code>	<code>0x02000000</code>	<code>:t</code>	Force le SAT passthrough (risqué)

Vérification

Après redémarrage ou reconnexion du disque :

```
sudo smartctl -a /dev/sdX
```

Un résultat valide commencera par une section `=== START OF INFORMATION SECTION ===` avec les informations de votre disque.

Sources

- [smartmontools Wiki — SAT with UAS under Linux](#)
- [Kernel Parameters Documentation](#)

Vérifier l'intégrité de la mémoire flash d'une carte SD / clé USB

Systeme utilisé

- **Distribution** : LMDE 7 (Linux Mint Debian Edition)

1. Installation de f3

f3 (Fight Flash Fraud) est un utilitaire pour tester la capacité réelle et détecter les clés USB ou cartes mémoire frauduleuses.

Documentation officielle : <https://fight-flash-fraud.readthedocs.io/en/latest/introduction.html>

Pour installer f3 sur LMDE 7, ouvrez un terminal et tapez :

```
sudo apt update
sudo apt install f3
```

2. Identifier la clé USB

Avant de tester la clé, il faut connaître son identifiant de périphérique. Branchez la clé USB et exécutez :

```
lsblk
```

Vous verrez une liste des disques et partitions. Exemple de sortie :

```
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda         8:0    0  500G  0 disk
├─sda1     8:1    0  500G  0 part /
sdb         8:16    1   32G  0 disk
└─sdb1     8:17    1   32G  0 part /media/usb
```

⚠ **Important** : Notez bien le périphérique (`/dev/sdX`). Si vous choisissez le mauvais disque, vous risquez de supprimer des données importantes.

3. Tester la clé USB avec f3probe

La commande `f3probe` permet de vérifier la capacité réelle et la vitesse d'écriture de la clé.

⚠ **Cette opération efface complètement la clé.**

```
sudo f3probe --destructive --time-ops /dev/sdX
```

- Remplacez `/dev/sdX` par l'identifiant de votre clé (ex. `/dev/sdb`).
- `--destructive` : efface toutes les données sur la clé.
- `--time-ops` : affiche le temps de chaque opération pour mesurer la vitesse.

Exemple de sortie :

```
F3 probe 7.0
...
Probe finished: 31.2 GiB, speed 20.1 MB/s
No bad sectors found
```

⚠ Si des secteurs sont « bad », cela indique que la clé est défectueuse ou frauduleuse.

4. Résumé des commandes

```
sudo apt install f3           # Installer f3
lsblk                        # Identifier la clé USB
sudo f3probe --destructive --time-ops /dev/sdX # Tester la clé
```

Vérifier l'état de la batterie sous Linux

Vérifier l'état de la batterie sous Linux avec UPower

1. Ouvrir le terminal

Ouvrir l'application Terminal sur le système Linux.

2. Lister les périphériques d'alimentation

Utiliser la commande suivante pour afficher tous les périphériques liés à l'alimentation :

```
upower --enumerate
```

Cette commande retourne une liste de chemins correspondant aux différents périphériques détectés.

Exemple de sortie :

```
/org/freedesktop/UPower/devices/line_power_AC  
/org/freedesktop/UPower/devices/battery_BAT0  
/org/freedesktop/UPower/devices/mouse_dev_CD_27_F1_E6_00_B3  
/org/freedesktop/UPower/devices/mouse_hidpp_battery_10  
/org/freedesktop/UPower/devices/DisplayDevice
```

3. Identifier le chemin de la batterie

Repérer dans la liste le chemin correspondant à la batterie. Dans la plupart des cas, il ressemble à :

```
/org/freedesktop/UPower/devices/battery_BAT0
```

4. Afficher les informations de la batterie

Utiliser la commande suivante en remplaçant le chemin par celui de la batterie :

```
upower -i /org/freedesktop/UPower/devices/battery_BAT0
```

5. Comprendre les informations affichées

Cette commande affiche des informations détaillées sur la source d'alimentation, notamment :

- **Niveau de charge actuel**
- **État** (en charge, décharge, etc.)
- **Capacité de la batterie**
- **État de santé de la batterie** (battery health)
- **Temps restant estimé**

Installation manuelle de RustDesk sous LMDE 7

Installation manuelle de RustDesk sous LMDE 7

Contexte matériel

- **Distribution** : LMDE 7 64 bits

Contexte

Suite à l'installation de Linux LMDE 7, il peut arriver que RustDesk ne soit pas présent dans la logithèque. Il faut alors l'installer manuellement.

Procédure d'installation

1. Mise à jour du système

```
sudo apt update
```

2. Vérifier la présence dans la logithèque

Ouvrir la logithèque et rechercher **RustDesk**.

“ Si RustDesk est présent, l'installer depuis la logithèque et arrêter la procédure ici. Sinon, continuer.

3. Installation manuelle

Ouvrir un terminal et exécuter les commandes suivantes :

```
wget https://github.com/rustdesk/rustdesk/releases/download/1.4.0/rustdesk-1.4.0-x86_64.deb
sudo dpkg -i rustdesk-1.4.0-x86_64.deb
sudo apt -f install
```

4. Redémarrage

Redémarrer l'ordinateur pour finaliser l'installation.

Liens associés

- [Résolution du problème - Reddit](#)
-

Notes

Activer le mode **headless Linux** dans RustDesk pour les machines sans clavier ni écran.