

Installer des logiciels

- [Installer Stocat sur LMDE7 \(64 bits\)](#)
- [Installer manuellement RustDesk sous LMDE 7](#)
- [Installer NoComprendo 2.X sur Ubuntu](#)

Installer Stocat sur LMDE7 (64 bits)

Installer Stocat sur LMDE 7

1. Télécharger la version Linux x64 (.zip)

Stocat n'a pas encore de paquet officiel `.deb` / Flathub pour Debian/LMDE — la page officielle renvoie vers les releases GitHub.

1. Va sur la page officielle de téléchargement : **<https://stoat.chat/download>**
2. Dans la section **Linux**, clique sur **GitHub** (lien vers les releases : `https://github.com/stoatchat/for-desktop/releases`)
3. Sur la page de la dernière release (tout en haut, ex. `v1.4.0`), déroule la section **Assets**
4. Repère l'archive correspondant à Linux 64 bits. Le nom suit généralement ce format :

```
Stocat-linux-x64-<version>.zip
```

(vérifie bien `linux` + `x64` dans le nom, à ne pas confondre avec `arm64` qui est pour les architectures ARM, ou avec les fichiers `.deb` / `.rpm` / `.AppImage` / `.flatpak` s'ils sont proposés)

5. Télécharge ce fichier `.zip` (par défaut dans `~/Téléchargements`)

“ ☐ Alternative : le dépôt communautaire [stoat-desktop-for-linux](https://github.com/stoatchat/for-desktop) propose aussi des builds Linux, mais privilégie toujours en premier les assets officiels de `stoatchat/for-desktop` .

2. Extraire l'archive

```
cd ~/Téléchargements
unzip Stoaat-linux-x64-*.zip -d ~/Applications/stoaat
```

(crée le dossier `~/Applications/stoaat` s'il n'existe pas encore : `mkdir -p ~/Applications`)

L'exécutable principal se trouve normalement à la racine du dossier extrait, sous un nom comme `stoaat` ou `Stoaat` (Electron).

3. Rendre l'exécutable lançable

```
chmod +x ~/Applications/stoaat/stoaat
```

Adapte le nom du binaire s'il diffère (vérifie avec `ls ~/Applications/stoaat`).

4. Créer le raccourci (fichier .desktop)

```
nano ~/.local/share/applications/stoaat.desktop
```

Colle ce contenu (adapte les chemins) :

```
[Desktop Entry]
Type=Application
Name=Stoaat
Comment=Lancer Stoaat
Exec=/home/TON_USER/Applications/stoaat/stoaat
Icon=/home/TON_USER/Applications/stoaat/resources/app/icon.png
Terminal=false
Categories=Network;InstantMessaging;
```

- Remplace `TON_USER` par ton nom d'utilisateur (`whoami`)
- `Icon=` est optionnel : cherche un fichier `.png` dans le dossier extrait (souvent dans `resources/`), ou supprime la ligne pour une icône générique

Sauvegarde (Ctrl+O, Entrée, Ctrl+X), puis rends le fichier exécutable :

```
chmod +x ~/.local/share/applications/stoat.desktop
```

5. Copier le raccourci sur le bureau

```
cp ~/.local/share/applications/stoat.desktop ~/Desktop/  
chmod +x ~/Desktop/stoat.desktop
```

Au premier double-clic, Cinnamon/Nemo demandera de confirmer la confiance : clic droit sur l'icône → « **Autoriser le lancement** ».

6. Lancement automatique au démarrage de session

Méthode graphique : Menu → « Applications au démarrage » (Startup Applications) → **Ajouter** → sélectionne `stoat.desktop`.

Méthode terminal :

```
mkdir -p ~/.config/autostart  
cp ~/.local/share/applications/stoat.desktop ~/.config/autostart/
```

Stoat se lancera automatiquement à chaque ouverture de session.

Récapitulatif rapide

Étape	Commande clé
Extraction	<code>unzip Stoat-linux-x64-*.zip -d ~/Applications/stoat</code>
Rendre exécutable	<code>chmod +x ~/Applications/stoat/stoat</code>
Créer raccourci	<code>~/.local/share/applications/stoat.desktop</code>
Copier sur le bureau	<code>cp ... ~/Desktop/</code>
Démarrage auto	<code>cp ... ~/.config/autostart/</code>

Mettre à jour Stoaat plus tard

Comme il n'y a pas de gestionnaire de paquets, la mise à jour est manuelle : retélécharge la nouvelle archive depuis les releases GitHub, extrais-la par-dessus l'ancien dossier (ou dans un nouveau dossier en mettant à jour le chemin `Exec=` du fichier `.desktop` si besoin).

Installer manuellement 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.

Source : <https://github.com/rustdesk/rustdesk/releases>

Procédure d'installation

1. Mise à jour du système

Exécuter dans le terminal :

```
sudo apt update
```

Puis :

```
sudo apt upgrade
```

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.

Installer NoComprendo 2.X sur Ubuntu

“ Basé sur le wiki officiel du projet : invent.kde.org/beroot/nocomprendo — [Install on Debian](#) et sur une session de dépannage réelle sur Ubuntu (septembre 2026).

NoComprendo est un outil de commande vocale, de dictée et de synthèse vocale (Text-to-Speech) pour Linux, développé par Bruno Anselme. Il utilise :

- **Vosk** pour la reconnaissance vocale,
- **dotool** pour simuler les entrées clavier/souris (compatible X11/Wayland),
- **svox-pico** / eSpeak pour la synthèse vocale.

1. Prérequis

1.1 Installer dotool (simulation clavier/souris)

```
git clone https://git.sr.ht/~geb/dotool
cd dotool
./build.sh
sudo ./build.sh install
```

Autoriser dotool à créer des périphériques d'entrée virtuels via une règle udev :

```
sudo udevadm control --reload && sudo udevadm trigger && sudo usermod -a -G input USER_NAME
```

“ Remplace `USER_NAME` par ton nom d'utilisateur (ex. `staline`).

Redémarre ton ordinateur pour que les nouvelles règles udev prennent effet. Sans ce redémarrage, seuls le lancement de programmes et les méta-commandes fonctionneront — pas le contrôle du clavier/de la souris.

1.2 Installer l'environnement Qt6

```
sudo apt install qt6-base-dev qt6-multimedia-dev qt6-tools-dev qt6-l10n-tools
```

1.3 Installer Vosk (reconnaissance vocale)

Sur Ubuntu, la bibliothèque Vosk s'installe via pip et fournit `libvosk.so`, mais **pas** le header C `vosk_api.h`, nécessaire à la compilation :

```
pip3 install vosk
```

Télécharger ensuite le header manquant depuis le dépôt officiel et le copier dans le même dossier que `libvosk.so` :

```
wget https://raw.githubusercontent.com/alphacep/vosk-api/master/src/vosk_api.h -P /tmp/  
sudo cp /tmp/vosk_api.h /usr/local/lib/python3.12/dist-packages/vosk/
```

“ Adapte le chemin (`python3.12`) à ta version de Python — vérifie-le avec la commande de la section 2.2.

2. Compilation

2.1 Récupérer et préparer les sources

```
tar zxvf nocomprendo-2.x.tar.gz  
cd nocomprendo-2.x
```

2.2 Localiser libvosk.so

```
sudo find / -name "libvosk.so" 2>/dev/null
```

Résultat typique sur Ubuntu (pip, install système) :

```
/usr/local/lib/python3.12/dist-packages/vosk/libvosk.so
```

2.3 Adapter `nocomprendo.pro`

Le fichier `.pro` d'origine référence un chemin Fedora/RedHat (`/usr/lib64/libvosk.so.0`) qui n'existe pas sur Ubuntu, et ne définit **aucun** `INCLUDEPATH` vers le header `vosk_api.h` — c'est la cause de l'erreur `fatal error: vosk_api.h: Aucun fichier ou dossier de ce nom` (que vous aurez en suivant la documentation initiale du développeur)

Remplacer :

```
LIBS += /usr/lib64/libvosk.so.0
# LIBS += -lvosk
# For debians check libvosk.so location. May be next line
# LIBS += /usr/lib/python3.10/site-packages/vosk/libvosk.so
```

par (en adaptant le chemin trouvé à l'étape 2.2) :

```
LIBS += /usr/local/lib/python3.12/dist-packages/vosk/libvosk.so
INCLUDEPATH += /usr/local/lib/python3.12/dist-packages/vosk
```

2.4 Compiler

Utiliser `qmake6` (et non `qmake`, au cas où `qmake5` serait aussi présent sur le système) :

```
qmake6
make
```

2.5 Problème de traduction (.qm manquant)

Erreur possible :

```
RCC: Error in 'qmake_qmake_qm_files.qrc': Cannot find file '../.qm/nocomprendo_fr_FR.qm'
```

Le fichier de traduction compilé (.qm) n'est pas généré automatiquement par lrelease générique sur certaines configs Ubuntu. Solution :

```
# Si "lrelease" seul échoue avec "could not find a Qt installation"
/usr/lib/qt6/bin/lrelease nocomprendo_fr_FR.ts

# Déplacer le fichier généré dans le dossier attendu par le build
mkdir -p .qm
mv nocomprendo_fr_FR.qm .qm/

# Relancer la compilation (sans make clean)
make
```

2.6 Installer l'application

```
sudo make install
```

2.7 Créer le dossier des modèles de langue Vosk

```
sudo mkdir -p /usr/share/vosk-models/
sudo chmod 777 /usr/share/vosk-models/
```

2.8 Lancer NoComprendo

```
./nocomprendo
```

3. Utilisation de NoComprendo

3.1 Premier démarrage — modèle de langue

Au premier lancement, NoComprendo se connecte à **alphacephei.com** et propose de télécharger un modèle de langue Vosk. Pour le français, deux options principales :

- **fr** : modèle complet, plus précis mais plus lourd (~1.4 Go)
- **small-fr** : modèle léger, plus rapide mais moins précis.

L'application propose aussi de choisir la langue de l'interface (**fr_FR** ou **en_US**) au premier démarrage.

3.2 Dictée vocale

- Clique sur l'icône du **micro** dans l'interface et parle distinctement.
- Vosk transcrit la parole en texte au fur et à mesure.
- Commandes de ponctuation orales disponibles : « *point* », « *virgule* », « *point à la ligne* », etc.
- Pour démarrer rapidement : dire « *Je commence à dicter* ».
- Si seule la dictée t'intéresse (pas les commandes vocales), tu peux désactiver les groupes de commandes dans les réglages.

3.3 Commandes vocales

- NoComprendo est livré avec des **groupes de commandes de démonstration**, orientés pour l'environnement KDE/Plasma — à vérifier et adapter selon ton propre environnement (GNOME sur Ubuntu par défaut, par exemple).
- Chaque commande associe une phrase prononcée à une action : raccourci clavier, lancement de programme, etc.
- Pour modifier une commande : **double-clic dessus**, puis ré-enregistrer la nouvelle formulation.

3.4 Conseils pour une bonne reconnaissance

- Vosk fonctionne mieux avec des **phrases syntaxiquement correctes** plutôt que des mots isolés.
 - Préférer « *Fermer la fenêtre* » à « *fermer fenêtre* ».

- Le résultat exact peut varier selon le modèle (`fr` vs `small-fr`) : par exemple « *Fermez la fenêtre* » ou « *Fermé la fenêtre* » pour la même intention.
- Articuler distinctement, éviter les commandes monosyllabiques et les mots proches phonétiquement (homophonies).
- Ne pas taper au clavier pendant que l'écoute vocale est active (peut perturber la reconnaissance).

3.5 Synthèse vocale (Text-to-Speech)

- Fonctionne en copiant du texte dans le presse-papier, puis en demandant à NoComprendo de le vocaliser.
- Six langues disponibles pour la synthèse vocale au moment de la rédaction de cette doc.

3.6 Fonctionnement hors-ligne

Reconnaissance et synthèse vocales sont effectuées **entièrement hors-ligne** — aucune donnée vocale n'est envoyée à des serveurs tiers.

4. Ressources

- Wiki officiel (installation Debian/Ubuntu) : https://invent.kde.org/beroot/nocomprendo/-/wikis/debian_install
- Dépôt Vosk (header et sources) : <https://github.com/alphacep/vosk-api>
- dotool : <https://git.sr.ht/~geb/dotool>