

Termux



Termux : maîtrisez l'appareil Android avec ce guide détaillé Dernière mise à jour: 20 Octobre, 2025
Auteur: Lorena Figuero

Guides Android » Applications Android » Termux : Maîtrisez le terminal Android grâce à ce guide complet

- Installez Termux depuis F-Droid, mettez à jour les packages et activez le stockage pour travailler en toute sécurité avec les fichiers sur votre téléphone.
- Apprenez les commandes clés, utilisez les éditeurs (Nano/Vim) et étendez les capacités avec pkg/apt, SSH, Python et les utilitaires réseau.
- Automatisez avec Bash, les alias et cron, augmentez la productivité avec tmux/htop et explorez des distributions entières avec proot-distro.

Maîtriser Termux

Si l'idée d'avoir un environnement GNU/Linux dans votre poche vous séduit, Termux transforme votre téléphone portable en un terminal complet avec de véritables paquets et des commandes familières. En quelques minutes, vous pouvez passer de la simple curiosité à la maîtrise d'un système vous permettant de naviguer dans les dossiers, d'automatiser des tâches, de modifier des fichiers, de programmer et de gérer des services sans avoir à rooter votre téléphone.

Ce guide vous accompagne pas à pas : installation recommandée, configuration de l'environnement, commandes essentielles et avancées, éditeurs de texte, gestion des paquets, SSH, serveur web et paquets supplémentaires pour l'image, la vidéo ou les scripts. Vous y trouverez des astuces, des conseils et des solutions aux erreurs courantes pour une expérience rapide, sécurisée et productive, que vous soyez novice ou utilisateur régulier de Linux.

Qu'est-ce que Termux et pourquoi vaut-il la peine de l'utiliser ?

Termux est une application libre et gratuite qui émule un terminal Linux sur Android et intègre son propre gestionnaire de paquets. En d'autres termes, vous aurez accès à des utilitaires comme git, vim, python, openssh, curl et nodejs sans privilèges root et avec des performances remarquables sur les appareils modernes.

Son principal atout par rapport aux autres émulateurs réside dans son écosystème de paquets basé sur apt (également gérable avec la commande `pkg``), qui permet d'installer en quelques secondes tout le nécessaire, des éditeurs et outils réseau aux bibliothèques de développement. Grâce à cette approche, l'application est idéale pour apprendre à utiliser le terminal, configurer de petits serveurs, écrire des scripts, tester des automatisations ou administrer des machines distantes.

Un autre avantage réside dans son isolement du système Android : Termux fonctionne dans son propre espace utilisateur, minimisant les risques et permettant l'expérimentation. De plus, sa communauté très active maintient une documentation, des dépôts et des utilitaires qui étendent ses fonctionnalités et répondent aux questions fréquemment posées.

Si vous avez toujours rêvé d'« utiliser Linux sur un téléphone portable », vous trouverez ici un flux de travail cohérent pour tout mettre en place étape par étape : de la navigation dans les répertoires et l'édition de fichiers au démarrage de services, en passant par l'utilisation de SSH, le tunneling ou l'exécution de distributions complètes sous Android.

Termux Développeur: Fredrik Fornwall Prix: Gratuit Installation correcte sur Android et premières étapes essentielles

Installation de Termux sur Android

La meilleure façon d'installer Termux aujourd'hui est de passer par F-Droid, où sont publiées les versions les plus récentes. Rendez-vous sur le site web de F-Droid, téléchargez le fichier APK et installez-le. Ensuite, dans F-Droid, recherchez Termux et installez la version stable la plus récente afin d'éviter les versions plus anciennes ou limitées parfois disponibles sur le Play Store.

Dès l'ouverture de l'application, mettez à jour la base de données des packages et les composants

préinstallés. Exécutez ces commandes et confirmez à l'invite : `apt update && apt upgrade -y` Le faire dès le début évite les erreurs ultérieures et vous garantit services publics de pointe dans votre environnement de travail.

Accordez l'accès au stockage en exécutant `termux-setup-storage`. Une boîte de dialogue d'autorisation Android apparaîtra : acceptez-la pour obtenir des liens directs vers les téléchargements, les images, la musique, etc., et travaillez confortablement avec fichiers depuis le téléphone lui-même depuis le terminal via des itinéraires en `~/storage/`.

Un petit conseil pratique : activez la touche Tabulation virtuelle si elle n'apparaît pas sur votre clavier. Sur de nombreux appareils, maintenez le bouton Volume + enfoncé et, dans Termux, appuyez sur la touche Q pour activer une touche supplémentaire faisant office de touche Tabulation, ce qui facilite grandement la saisie semi-automatique des chemins et des paquets.

À partir d'ici, vous pouvez exploiter pleinement le terminal. N'oubliez pas ce raccourci pratique : `Ctrl + C` pour interrompre une commande en cours et `Ctrl + Z` pour la suspendre ; cela vous facilitera grandement la tâche lors du test d'utilitaires. Structure des dossiers et chemins utiles dans Android

Termux place votre répertoire personnel dans `/data/data/com.termux/files/home` (également accessible en `$HOME`). C'est ici que vous souhaitez enregistrer vos scripts, projets et paramètres. Pour accéder aux dossiers partagés Android, après avoir accordé les autorisations avec `termux-setup-storage`, tu as `~/storage/`. À l'intérieur, vous trouverez des raccourcis vers téléchargements, images, musique, films et partage, vous pouvez ainsi déplacer, modifier et copier des fichiers sans aucun problème.

Cette séparation permet de maintenir l'ordre et la sécurité. Par exemple, il est recommandé de travailler en `$HOME` avec vos scripts et synchronisez-les avec GitHub ou GitLab, tandis que les données que vous souhaitez échanger avec d'autres applications ou partager avec le PC peuvent aller vers `~/storage/downloads` pour les avoir toujours à portée de main.

Si vous vous perdez, la commande `pwd` montre où vous êtes, et avec `ls -la` Vous verrez du contenu détaillé, y compris du contenu caché. En combinaison avec des outils tels que `tree` (installable avec `pkg install tree`) vous pouvez visualiser l'ensemble de la hiérarchie en un coup d'œil, ce qui est utile lors de la gestion projets avec plusieurs dossiers. Commandes essentielles et gestion avancée des fichiers

Pour se déplacer avec aisance, il est important d'intérioriser les bases : `ls` lister les fichiers, `cd` changer de répertoire, `pwd` montre l'itinéraire actuel, `clear` nettoyer l'écran et `man` ouvrir l'aide intégrée, par exemple `man ls`. Essayez des variantes comme `ls -l`, `ls -a` ou `ls -color=auto` pour obtenir des détails et des couleurs qui facilitent la lecture.

Créer et gérer du contenu est tout aussi simple : `touch notas.txt` génère un fichier vide ; `mkdir proyectos` créer un dossier; `echo "Hola" > hola.txt` écrire du texte en écrasant et `echo "más" >> hola.txt` ajouter à la fin. Afficher le contenu avec `cat` ou consultez les première et dernière lignes avec `head` y `tail` aide à inspecter journaux et configurations rapidement.

Pour copier ou déplacer, utilisez `cp` y `mv`: `cp fichero.txt copia.txt`, `cp -r carpeta1 carpeta2`, `mv notas.txt notas_antiguas.txt`. Recommandation : ajoutez des indicateurs de sécurité tels que `cp -n` (ne pas écraser), `mv -i` (demandez avant) ou informatif tel que `-v` pour savoir ce qui a été copié ou déplacé, réduisant ainsi les risques de écrasements indésirables.

Pour rechercher, `find . -name "*.jpg"` Suivi par motif. Et si vous avez besoin de localiser du texte dans des fichiers, `grep` est votre allié : essayez-le `grep -R "cadena"` . pour une recherche récursive. Avec

`awk` y `wc` Vous pouvez filtrer les colonnes ou compter les lignes, les mots et les caractères, ce qui facilite la tâche. analyse rapide des données dans le terminal lui-même.

Vous allez également devoir gérer la compression. `unzip` `archivo.zip` extraire le ZIP, `tar -xf fichero.tar` gère le goudron et avec des utilitaires comme `gzip`, `bzip2` o `7z` (installable) étend la compatibilité des formats. Pour supprimer, rappelez-vous : `rm` `archivo` supprimer un fichier; `rm -r carpeta/` supprimer des répertoires ; et `rm -rf carpeta` le fait sans demander, utilisez-le à bon escient car il n'y a pas de retour en arrière.

Dans le domaine des réseaux, `ifconfig` o `ip addr` Afficher les interfaces et les adresses IP. Ceci est crucial lors de la configuration de connexions SSH, de la diffusion de contenu sur votre réseau local ou du test de services avec `curl`, `wget` y `ncat`, des outils qui simplifient la débogage des API et des ports sans quitter votre mobile. Gestionnaire de paquets et utilitaires essentiels

Paquets essentiels dans Termux

La magie de Termux réside dans son gestionnaire de paquets. Vous pouvez l'utiliser de manière interchangeable. `pkg` o `apt` pour installer, mettre à jour et rechercher des utilitaires. Veuillez d'abord vérifier avec `pkg search` nombre et quand vous avez tout clair, installez avec `pkg install` `paquete` agrandir capacités instantanées.

Recommandations initiales: `pkg install nano` (éditeur simple), `pkg install vim` (éditeur puissant), `pkg install openssh` (SSH), `pkg install python` (Scripting Python), `pkg install git` (contrôle de version) et `pkg install coreutils` (utilitaires essentiels s'ils ne l'étaient pas déjà). Tout cela vous permet d'écrire des scripts, de cloner des dépôts, se connecter aux serveurs et automatiser.

Au-delà du minimum, il existe des packages qui améliorent votre flux de travail : `man` pour les manuels, `imagemagick` traiter des images, `ffmpeg` comme un couteau suisse de l'audio/vidéo, `mc` (Midnight Commander) si vous préférez un gestionnaire de fichiers de type panneau, ou `bash-completion` pour la saisie semi-automatique plus intelligent dans la coquille.

Pour la surveillance et les processus, `htop` offre une vue claire du système, tandis que `top` C'est l'option la plus légère. Avec `pkill` vous pouvez tuer des processus par leur nom, et `tmux` vous permet de gérer plusieurs sessions de terminal en parallèle, idéal lors de l'exécution serveurs ou tâches de longue durée Depuis le téléphone.

Si vous devez monter des systèmes de fichiers distants, `sshfs` Il vous permet d'accéder aux répertoires d'une autre machine via SSH, comme s'ils étaient locaux. Et pour des recherches ultra-rapides, `updatedb` y `locate` Ils maintiennent une base de données pour retrouver des fichiers en un clin d'œil, quelque chose de très pratique dans projets gourmands en ressources. Éditeurs de terminaux : `Nano` et `Vim` sans crainte

Vous devrez modifier fréquemment des fichiers : configurations, scripts et notes. `Nano` est simple et efficace : `nano archivo.txt` ouvert, vous vous déplacez avec les flèches et sortez avec `Ctrl + X` (Enregistrer sur demande). C'est idéal pour débiter et apporter des modifications rapides sans difficulté d'apprentissage.

`Vim` est un peu plus exigeant, mais il redonne puissance et rapidité. Ouvrir avec `vim archivo.txt`; presse `i` pour entrer en mode insertion, `Esc` pour quitter le mode normal et utiliser `:wq` pour enregistrer et quitter ou `:q` Si vous ne souhaitez pas enregistrer, vous pouvez personnaliser les raccourcis, la coloration syntaxique et les macros au fur et à mesure de votre progression.

programmation et montage intensifs.

Pour les scripts et les fichiers de configuration, Vim offre une grande flexibilité, bien que Nano reste excellent pour l'édition à la volée. Choisissez l'éditeur qui vous convient le mieux ; l'important est de l'intégrer à votre routine et de gagner en rapidité lors de l'écriture et de l'édition sur votre appareil mobile. Personnalisation et automatisation : alias, invites et scripts dans Bash

L'archive ~/.bashrc est votre terrain de jeu. Vous pouvez définir des alias comme alias updg="apt update && apt upgrade" pour raccourcir les tâches répétitives. Vous pouvez également ajuster l'invite en modifiant la variable PS1, par exemple PS1=":\w\$ ", afin que le terminal vous montre l'itinéraire actuel et que vous ayez un environnement plus informatif et plus propre.

Créer des scripts ne prend que quelques minutes. Ouvrir un fichier, par exemple. nano mi_script.sh, ajoutez le shebang approprié et vos instructions, enregistrez, puis attribuez les autorisations avec chmod +x mi_script.sh. À partir de ce moment, vous pouvez l'exécuter avec ./mi_script.sh et automatiser les copies, les rapports, les synchronisations ou tâches de maintenance.

Le démarrage d'un script Bash typique pourrait ressembler à :

```
#!/data/data/com.termux/files/usr/bin/bash. À partir de là, ajoutez des conditions, des boucles et des fonctions. Par exemple, des variables comme nombre="Juan"; un si : if [ "$edad" -ge 18 ]; then echo "Eres mayor de edad"; fi; une boucle for : for f in *.txt; do echo "$f"; cat "$f"; done; ou des fonctions comme saludar() { echo "Hola $1"; } Avec cela, vous pouvez créer utilitaires personnalisés pour votre vie quotidienne.
```

Si vous êtes intéressé par la programmation au-delà de Bash, avoir python installé ouvre la porte à des scripts complexes, des requêtes HTTP avec curl ou requestset des automatisations avec des API. De plus, avec git Vous versionnez votre travail et le synchronisez avec des dépôts distants, ce qui ferme un cercle très solide dans productivité mobile. SSH et accès à distance : contrôlez votre téléphone depuis votre PC

Avec OpenSSH, vous pouvez vous connecter à votre Termux depuis un ordinateur. Commencez par installer pkg install openssh. Ensuite, si vous souhaitez une sécurité maximale, générez une paire de clés sur votre PC et copiez la clé publique sur votre mobile (par exemple en termux.pub en ~/storage/downloads/). Ajoutez-le ensuite à ~/.ssh/authorized_keys avec: cat ~/storage/downloads/termux.pub » ~/.ssh/authorized_keys autoriser connexion sans mot de passe.

Démarrez le serveur avec sshd (vous pouvez l'arrêter avec pkill sshd) et découvrez l'IP mobile avec ifconfig. Si vous préférez un nom d'utilisateur-mot de passe, définissez-en un avec passwd Depuis votre PC, connectez-vous comme ceci : ssh usuario@direccion-ip -p 8022 (8022 est le port par défaut dans Termux). De là, vous pouvez taper confortablement avec un clavier physique, transférer des fichiers avec scp ou sftp et courir flux de travail à distance sans toucher le mobile.

Pour déplacer des fichiers, exemples avec scp: grimper scp archivo.txt usuario@ip:/destino/; télécharger scp usuario@ip:/origen/archivo.txt ./ . Si vous préférez des synchronisations robustes et différentielles, rsync C'est fantastique, et avec sshfs Vous pouvez monter le système de fichiers mobile sur votre PC ou vice versa pour fonctionner comme s'il l'était un dossier local. Serveur Web et utilitaires réseau à la volée

Maîtriser Termux sur Android

Pour tester ou partager des fichiers sur votre réseau local, il vous suffit pkg install python et alors

python -m http.server 8080 dans le dossier souhaité. Accès <http://IP-del-movil:8080> Depuis un autre appareil, vous verrez ces fichiers servis. C'est idéal pour des validations rapides et pour augmenter les chances de succès. prototypes Web en quelques secondes sans rien installer d'autre.

Dans le domaine du réseau/sécurité, Termux prend en charge des outils précieux pour la formation et l'audit dans des environnements autorisés, tels que nmap (balayage du réseau), openssl (cryptage et TLS/SSL), netcat/ncat (tests de port et de service) ou tmux (plusieurs sessions). Gardez à l'esprit que des utilitaires tels que hydra ou sqlmap Ils ne doivent être utilisés qu'avec autorisation explicite dans des laboratoires ou des environnements contrôlés, en respectant cadres juridiques et éthiques.

Si vous êtes intéressé par la confidentialité, vous pouvez installer tor avec pkg install tor et exécutez-le avec tor pour acheminer le trafic et, si nécessaire, configurer des services .onion. Une autre option pour exposer temporairement un service local consiste à utiliser des tunnels de type ngrok, utiles pour afficher une démonstration externe sans ouvrir de ports sur le routeur, toujours avec une configuration soucieuse de la sécurité.

Bots, API et automatisation connectée

Créer des bots pour Telegram ou de petites intégrations avec des API est une tâche très naturelle dans Termux. curl pour les requêtes HTTP et jq pour traiter JSON (pkg install jq curl) vous pouvez envoyer des notifications, répondre à des commandes ou intégrer le mobile dans des flux de travail surveillance et contrôle qui vivent dans le cloud.

La clé réside dans le jeton du bot et dans l'écriture de scripts qui gèrent les points de terminaison, valident les réponses et appliquent la logique. Combinez Bash et Python si cela vous convient : utilisez Python pour la logique complexe et Bash pour l'interface système, ce qui permet d'obtenir des solutions légères fonctionnant 24 h/24 et 7 j/7, avec la possibilité d'utiliser tmux ou cronie pour une exécution planifiée.

Conseils quotidiens, raccourcis et dépannage

Si vous rencontrez des erreurs d'autorisation avec les fichiers Android, vérifiez ce que vous avez corrigé. termux-setup-storage et que vous avez accordé les autorisations dans les Paramètres système. Travaillez toujours dans ~/storage/ pour les fichiers partagés évite les surprises et maintient les packages à jour avec pkg update && pkg upgrade vous libère de incompatibilités et échecs étrangers.

Concernant le clavier, si la touche Tabulation n'apparaît pas, activez l'option mentionnée précédemment (Volume + et Q). Pour copier/coller, utilisez le menu contextuel Android ou les options de Termux en appuyant longuement sur l'écran. Un conseil important : ne vous contentez pas de copier-coller ; comprendre ce que vous faites réduit les erreurs et vous donne les outils nécessaires pour diagnostiquer vous-même les problèmes.

Raccourcis qui font la différence : saisie semi-automatique avec Tab, historique avec flèches haut/bas et recherche de commandes avec Ctrl + R. Lignes de commentaire avec le caractère # dans vos scripts et automatisez avec cronie Si vous avez besoin de tâches périodiques, surveillez les processus avec htop et garder plusieurs tâches en parallèle avec tmux pour que rien ne vienne interrompre longues séances.

Si vous préférez une interface graphique, vous pouvez utiliser un serveur VNC ou X11 moyennant une configuration supplémentaire. Ce n'est pas obligatoire, mais cela peut s'avérer utile pour les outils nécessitant une interface graphique. Cependant, l'objectif principal de Termux est d'exploiter le terminal, vous offrant ainsi légèreté, contrôle et rapidité sur un appareil mobile. Linux complet dans Android avec proot-distro

Si vous souhaitez aller plus loin, proot-distro vous permet de déployer des distributions complètes au sein de Termux, telles qu'Ubuntu, Debian, Kali ou Alpine. Vous avez ainsi accès aux gestionnaires de paquets de chaque distribution (apt, apk, dnf, etc.) et pouvez installer des logiciels qui ne sont pas disponibles dans les dépôts de Termux. C'est un excellent moyen d'expérimenter, de vous entraîner et d'étendre vos capacités de développement ou d'audit sans avoir besoin des droits d'administrateur.

Il y a bien sûr une période d'apprentissage, mais en contrepartie, vous obtenez un véritable « laboratoire portable » pour étudier, préparer des démonstrations ou reproduire des environnements. Grâce à une utilisation judicieuse des scripts, de SSH et de l'automatisation, votre téléphone portable devient un outil de travail performant qui vous accompagne partout avec une flexibilité surprenante.

Pour boucler la boucle, pensez à synchroniser votre \$HOME Grâce aux dépôts Git, sauvegardez vos fichiers dotfiles et assurez la cohérence des commandes et des alias sur tous vos appareils. Ainsi, quel que soit l'endroit où vous l'utilisez, votre shell se comporte de la même manière et offre des performances optimales, ce qui impacte directement votre productivité et confiance quand je travaille.

Termux offre une base Linux de poche avec une installation facile via F-Droid, des commandes familières, des packages puissants, un éditeur de texte fluide, SSH, un serveur web rapide, des outils de sécurité éducatifs, l'automatisation avec Bash et Python, et la possibilité d'exécuter des distributions complètes ; avec un peu de pratique, votre Android devient un environnement polyvalent pour apprendre, créer et résoudre des tâches du monde réel où que vous soyez.

Configurer les autorisations et le stockage

Après avoir installé Termux, il est essentiel de lui donner accès au stockage :

```
termux-setup-storage
```

Cette commande crée des raccourcis vers des dossiers dans le système de fichiers Android, tels que Téléchargements, DCIM, Musique et d'autres, au sein du conseil d'administration ~/storage/. Mettre à jour les packages

Pour garantir que nous disposons de la dernière version des packages disponibles dans Termux, il est recommandé d'exécuter :

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

Cela mettra à jour la liste des packages et installera les dernières versions disponibles.

Commandes de base de Termux

Si vous n'avez jamais utilisé de terminal sous Linux, voici quelques commandes essentielles :

- ls – Répertorie le contenu d'un dossier.
- cd – Changer de répertoire.
- pwd – Affiche le chemin du répertoire actuel.
- cp – Copier des fichiers ou des dossiers.
- mv – Déplacer ou renommer des fichiers.
- rm – Supprimer les fichiers. À utiliser avec précaution.
- mkdir – Créer un nouveau dossier.
- clear – Nettoyez l'écran du terminal.

Gestion des paquets dans Termux

Termux utilise APT (le même système de gestion de paquets que celui utilisé par Debian et Ubuntu) pour installer des outils et des programmes supplémentaires.

Installer des packages

Pour installer un paquet , utilisez :

```
pkg install nombre_del_paquete
```

exemple:

```
pkg install vim
```

Rechercher des packages

Si vous souhaitez trouver un paquet spécifique , utilisez :

```
pkg search nombre_del_paquete
```

Supprimer des paquets

Si vous n'avez plus besoin d'un paquet, vous pouvez le supprimer avec :

```
pkg uninstall nombre_del_paquete
```

Connectez Termux à d'autres appareils avec SSH

SSH (Secure Shell) vous permet d'accéder à des serveurs à distance et en toute sécurité. Pour installer SSH sur Termux :

```
pkg install openssh
```


Démarrer le serveur SSH

Termux peut servir de serveur SSH , permettant les connexions à distance :

```
sshd
```

Pour vous connecter à Termux depuis un autre appareil, vous aurez besoin de l'adresse IP de votre téléphone :

```
ifconfig
```

Ensuite, connectez-vous depuis un autre appareil en utilisant :

```
ssh usuario@IP -p 8022
```

Automatisation et scripts dans Termux

Termux permet d' exécuter des scripts pour automatiser les tâches répétitives. Voici un exemple de script simple pour afficher « Bonjour, monde » :

```
echo "echo Hola, mundo" > hola.sh
```

```
chmod +x hola.sh
```

```
./hola.sh
```

Personnalisation de Termux

Nous pouvons personnaliser le terminal éditer le fichier .bashrc:

```
vim ~/.bashrc
```

Exemple de personnalisation :

```
PS1="\[\033[1;32m\]\u@\h:\[\033[1;34m\]\w\[\033[0m\]$ "
```

Cela modifiera l'apparence de la ligne de commande. L'exploration des possibilités de Termux ouvre des opportunités infinies pour utiliser des commandes avancées, développer des scripts et se connecter à des serveurs depuis notre Android. Avec ce guide, vous aurez tout ce dont vous avez besoin pour commencer à utiliser Termux efficacement et en tirer le meilleur parti.

From:

<https://chanterie37.fr/fablab37110/> - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:

<https://chanterie37.fr/fablab37110/doku.php?id=start:linux:termux&rev=1788766700>



Last update: **2026/09/07 09:38**