

T<sub>1</sub>

# TERMUX POUR TOUS

*Tome 1 (Les Bases)*

*Auteur : Kal*

## A-PROPOS

Cet ebook est complet et c'est tout fait maison pour vous aider à avancer ou du moins à comprendre les bases de TERMUX pour faire du hacking que ce soit éthique ou autre alors vous allez bien vous régalez et dans le tome 2 ou on aboutiras avec les choses sérieuses mais en attendant retenez bien les bases car pour pouvoir programmer en n'importe langage( python, PHP, etc..) ou bien hacker votre voisin ou bien faire des CTF sur tryHackme ou root-ME, etc... il faut maîtriser les bases, et ce livre a été construit de telle sorte que si vous arrivez à tout maîtriser vous pourrez devenir de VRAI administrateur de serveur linux à l'avenir si ça vous tente ou bien devenir des testeur d'intrusion comme moi. Pour faire court sur l'à-propos je veux vous dire BON TRAVAIL.

## Tables des Matières

### 0. Présentation de l'auteur

#### 1. C'est quoi Termux ? Son utilisation ? Pourquoi ? où peut on le trouver ?

#### 2. Comment installer Termux sur Android ? ATTENTION IPHONE pas disponible

➤ Présentation de la première interface de Termux, le Welcome :D

#### 3. Comment l'enlever ?

➤ Le Profil d'utilisateur de Termux, comment en créer un ?

➤ Utiliser toilet, cmatrix, cowsay, figlet, etc...

➤ Bien personnaliser avec profile.h

#### 4. Comprendre le **bash.bashrc**

#### 5. Gestionnaire et installation des packages ? **apt install** ou **pkg install** quoi choisir ?

➤ pkg update && pkg upgrade, les indispensables pour les installations ?

➤ pkg unstable-repo && pkg root-repo

#### 6. L'éditeur de code NANO sur termux, pourquoi ?

#### 7. Stockage (termux-setup-storage)

#### 8. Additionnel Key & raccourci clavier

#### 9. Quelques commandes souvent utilisées ( cp, rm, mkdir, ls, ...)

#### 10. \$HOME, su & sudo, le \$ et #, history

#### 11. Les sessions

#### 12. Essayons de coder le fameux hello world en python avec Termux

Le Hacking avec Termux ( TOME 2 )

### REMERCIEMENTS

# **PRÉSENTATION DE L'AUTEUR**

Salut à tous chers lecteurs qui me connaissent ou pas, je tenais à vous dire WELCOME dans ma série de TERMUX POUR TOUS, j'espère que vous serez franche avec moi lorsque vous verrez que j'ai faussé à propos d'un truc. Sans trop bavarder je suis :

- Djefferson Saintilus connu sous le nom de Kal
- Nationalité Haïtienne;
- Administrateur du canal Termux & Linux depuis TELEGRAM MESSENGER ( <https://t.me/terlinux> )
- CEH v9 & LINUX PROFESSIONEL (LPIC-1) & Programmeur Python, PHP, C, ANSI,...

## **Mes références :**

- ✓ Tryhackme : <https://tryhackme.com/p/zorkalel>
- ✓ Root-me : <https://rootme.org//Djefferson-Saintilus?q=%2FZorkalel>
- ✓ Twitter : [https://twitter.com/djefferson\\_sain](https://twitter.com/djefferson_sain)
- ✓ Github : <https://github.com/Zorkalel>
- ✓ Linkedin : <https://linkedin.com/in/djefferson-saintilus>

...

# **TERMUX C'EST QUOI ?**

## ✓ Son Utilisation ? Pourquoi? Où trouver ?

Brièvement Termux est un environnement en ligne de commande android et une ligne de commande vous pouvez le savoir on peut le perfectionner à notre guise comme on veut, soit pour faire de la programmation ou bien du hacking ou autres mais nous, nous allons le perfectionner à faire du basic en attendant(TOME 2 hacking avec TERMUX).

Termux je le conseillerais beaucoup pour les nouveaux hackers(n00b ou newbie) et aussi pour les professionnelles car ça regorge une multitude d'options très utiles comme la possibilité de travailler sur linux depuis son android; et les administrateurs systèmes linux sachent bien de quoi je parle, concernant les n00bs qui voulaient se laisser aller par l'exploration de Termux sachez que ce seras très cool de débiter votre carrière de programmeur, administrateur ou pentesteur ou hacker( quelque soit le type) car sur termux tout y est même si certaines exploration demande le superSU ( root ou superUtilisateur) mais ne vous inquiétez pas car même si votre phone n'est pas superSU il est possible de tous utiliser sans crainte et aussi faut savoir que Termux à une communauté d'entraide très large, du coup lorsque vous avez des problèmes, que ce soit dans l'installation ou autres everyone peut vous aider car c'est très lucide la sortie d'erreur standard.

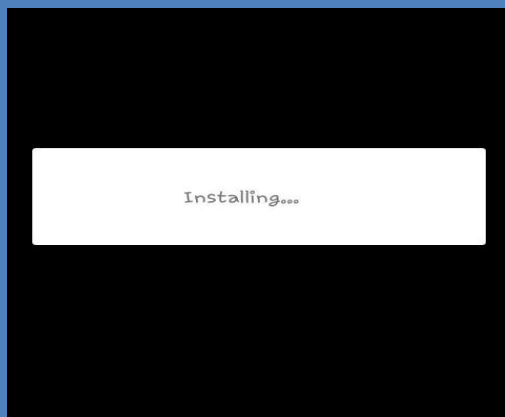
Conclusion, décidez d'utiliser Termux maintenant mon ami crois-moi ce seras pas en vain et quand même quelque soit ce que vous faites Termux vous donneras la réponse et croyez-moi. Moi ça m'y est utile dans mon entourage et j'espère que ce seras de même pour vous, sachez que vous trouverez Termux sur les store android comme Playstore, Uptodown, Apkpure, etc.. Termux est disponible uniquement avec android.

## **Comment installer Termux sur Android**

1- Allez sur le Playstore ou autre marché d'application cherchez le avec la syntaxe TERMUX



2- Ensuite patienter le temps de l'installation et vous allez tomber sur ces écrans :



3- après cela vous allez tomber devant un joli petit écran comme ci-dessous qui vous diras \*WELCOME TO TERMUX\*

```
Welcome to Termux!

Wiki:             https://wiki.termux.com
Community forum: https://termux.com/commu
ity
Gitter chat:      https://gitter.im/termux/
termux
IRC channel:      #termux on freenode

Working with packages:

* Search packages: pkg search <query>
* Install a package: pkg install <package>
>
* Upgrade packages: pkg upgrade

Subscribing to additional repositories:

* Root: pkg install root-repo

You are using legacy Termux environment.
Packages are unmaintained and will not
receive any updates.

$ █
```

Et voilà vous venez d'installer Termux de la manière la plus simple, mais je ne vais pas vous limiter la dessus je vais vous donner un lien méga Drive de téléchargement si vous n'arrivez pas à installer depuis Playstore.

Voici le lien ci-dessous :

<https://mega.nz/#!KAB3GKBQ!C4ZfUNkq3QAaIJOlxp4Wlf6HGKkd0iP3MdpE3OtP1pI>

**TERMUX NE PEUT ÊTRE INSTALLER SUR IPHONE, C'EST DISPONIBLE QUE SUR ANDROID JUSQU'À MAINTENANT.**

## **Présentation de la première interface de Termux, le Welcome**

```
Welcome to Termux!

Wiki:           https://wiki.termux.com
Community forum: https://termux.com/community
Gitter chat:    https://gitter.im/termux/termux
IRC channel:    #termux on freenode

Working with packages:

* Search packages:  pkg search <query>
* Install a package: pkg install <package>
>
* Upgrade packages: pkg upgrade

Subscribing to additional repositories:

* Root: pkg install root-repo

You are using legacy Termux environment.
Packages are unmaintained and will not
receive any updates.

$ █
```

Ceci est l'interface de Welcome offert par Termux et il vous explique plein de chose comme les basics jusqu'à plusss même, mais cette écran-là n'est pas si important pour nous maintenant puisque ceux qu'il raconte on le verra tout au long de ce livre alors mieux vaut l'enlever et laisser l'écran d'affichage vide juste pour ne pas se perdre comme débutant alors, comment le faire ?

D'abord faites-moi confiance en acceptant de faire ce que je vous dis de faire et vous verrez les résultats, croyez moi.


La première chose à faire c'est ouvrir votre Termux et d'y coller les codes ci-dessous dedans :

```
$ rm /data/data/com.termux/files/usr/etc/motd
```

```
$ exit
```

-voyez la pratique par mon écran-

```
$ rm /data/data/com.termux/files/usr/etc/motd
$ exit█
```



-Ainsi par la suite j'obtiens ça-

```
$ █
```

La remarque est que l'écran de Termux est vide maintenant, et lorsque c'est vide ça nous permet de mieux travailler. Ainsi on va commencer par la suite avec ce l'on appelle la personnalisation de TERMUX, qui est le fait d'arranger la ligne de commande à sa guise.

## Le Profil d'utilisateur de Termux, comment en créer un ?

Le Profil utilisateur c'est ce qui fait l'essence même de tout utilisateur de Termux car si il y'a des gens qui préfère utiliser son phone que son PC c'est parcequ'il le considère comme étant leur vie privée et comme étant leur bien non partageable et ce n'est pas sans raison. Bien souvent parceque leur phone possède un thème Hypercool (par exemple, NOVA THEME), une puissance de RAM extraordinaire ou bien une version genre 7.1, etc... Alors moi je vais proposer pour les bases, la personnalisation de TERMUX.

Commençons avec la liste des commandes (petit programme responsable de quelque chose Bien précis) à installer sur TERMUX pour débiter la personnalisation.

```
$ pkg update && pkg upgrade
```

```
$ pkg install cowsay toilet cmatrix figlet
```

```
$ cmatrix
```

```
$ Toilet
```

SUIVEZ LA PRATIQUE ->

```
$ pkg update && pkg upgrade
Get:1 https://termux.net stable InRelease
[1720 B]
Ign:2 https://dl.bintray.com/grimler/game-
packages-21 games InRelease
Ign:3 https://dl.bintray.com/grimler/scien-
ce-packages-21 science InRelease
Get:4 https://dl.bintray.com/grimler/game-
packages-21 games Release [5344 B]
Get:5 https://dl.bintray.com/grimler/scien-
ce-packages-21 science Release [5348 B]
Get:6 https://dl.bintray.com/grimler/game-
packages-21 games Release.gpg [475 B]
Get:7 https://dl.bintray.com/grimler/scien
```

- Je Viens d'installer les mises à jours

```
and 0 not upgraded.
$ pkg install figlet
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/grimler/game-
packages-21 games InRelease
Ign:3 https://dl.bintray.com/grimler/scien-
ce-packages-21 science InRelease
Get:4 https://dl.bintray.com/grimler/game-
packages-21 games Release [5344 B]
Hit:4 https://dl.bintray.com/grimler/game-
packages-21 games Release
Get:5 https://dl.bintray.com/grimler/scien-
ce-packages-21 science Release [5348 B]
Hit:5 https://dl.bintray.com/grimler/scien-
ce-packages-21 science Release
Reading package lists... Done
Building dependency tree... Done
All packages are up to date
```

- Je profite pour installer figlet seul

```
Preparing to unpack .../figlet_2.2.5-2_arm
.deb ...
Unpacking figlet (2.2.5-2) ...
Setting up figlet (2.2.5-2) ...
$ figlet kal
```

- Et voilà ce l'outil figlet permet de faire et si vous voulez plus d'options alors faites :

```
$ figlet --help
```

```
The following additional packages will be
installed:
perl
The following NEW packages will be install
ed:
cowsay perl
0 upgraded, 2 newly installed, 0 to remove
and 0 not upgraded.
Need to get 13.8 MB of archives.
After this operation, 60.0 MB of additiona
l disk space will be used.
Do you want to continue? [Y/n] n
Abort.
$ pkg install -y cowsay
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be
installed:
perl
The following NEW packages will be install
ed:
cowsay perl
0 upgraded, 2 newly installed, 0 to remove
```



- Ainsi je vais installer une autre outil pour la gestion qui est cowsay

```
Unpacking perl (5.30.0-2) ...
Selecting previously unselected package cowsay.
Preparing to unpack .../archives/cowsay_3.04-1_all.deb ...
Unpacking cowsay (3.04-1) ...
Setting up perl (5.30.0-2) ...
Setting up cowsay (3.04-1) ...
$ cowsay kal

<-- kal -->
  -----
    \      ^__^
     (oo)____\\
      ||----w |
      ||     ||
```

- Une fois exécuter voilà ce que peux faire cowsay mais ça reste pas là avec, il y a des tas d'options dont cette livre de quelques pages ne pourrait pas contenir

```
Need to get 959 kB of archives.
After this operation, 2249 kB of additional disk space will be used.
Do you want to continue? [Y/n] n
Abort.
$ pkg install toilet -y
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/griml/gamer-packages-21 games InRelease
Ign:3 https://dl.bintray.com/griml/science-packages-21 science InRelease
Get:4 https://dl.bintray.com/griml/gamer-packages-21 games Release [5344 B]
Hit:4 https://dl.bintray.com/griml/gamer-packages-21 games Release
Get:5 https://dl.bintray.com/griml/science-packages-21 science Release
```

- L'installation de notre dernière outil qui est toilet et qui va être utile pour la gestion de profil

```
Unpacking toilet (0.3-2) ...
Setting up libcaca (0.99.beta19-7) ...
Setting up toilet (0.3-2) ...
$ toilet kal

#          "          #
#  m      mmm      #
# m"      "      #
#"#      m""      #
#  "m    "mm"    "mm
```

- Une fois toilet exécuter voici ce qu'il nous a donné comme aperçu, que c'est LOL ;)

```
Subscribing to additional repositories:
* Root: pkg install root-repo

You are using legacy Termux environment.
Packages are unmaintained and will not
receive any updates.

$ toilet -f smblock 'hello guys'

hello guys

$ toilet -f smblock --filter metal 'kal is admin'

kal is admin
```

- Mais ça ne vas pas rester la avec toilet puisque voici en gros ce qu'il peut faire alors n'oubliez pas de consulter l'aide pour savoir.

\$ toilet -help

```
$ toilet -f smblock --filter border:metal 'TERMUX & LINUX'

TERMUX & LINUX
TERMUX & LINUX
TERMUX & LINUX
```

- Enfin on termine avec toilet avec ce petit affichage, la commande utiliser est :

\$ toilet -f smblock -filter border :metal ``

## Comment gérer son profil de TERMUX maintenant lorsque l'on a ces outils ?

1- AVANT toute gestion de profil nous allons créer un fichier qui va contenir nos infos pour la création stable du profil. Ce fichier va nous servir de pont pour communiquer avec le un fichier système nommé **bash.bashrc** sans problème. Ainsi nous allons installer NANO( éditeur de texte) pour le faire.

\$ pkg install nano

\$ nano .profile

Suivez la pratique ->

```
$ pkg install nano
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/grimmler/game-packages-21 game
s InRelease
Ign:3 https://dl.bintray.com/grimmler/science-packages-21 s
cience InRelease
Get:4 https://dl.bintray.com/grimmler/game-packages-21 game
s Release [5344 B]
Hit:4 https://dl.bintray.com/grimmler/game-packages-21 game
s Release
Get:5 https://dl.bintray.com/grimmler/science-packages-21 s
cience Release [5348 B]
```

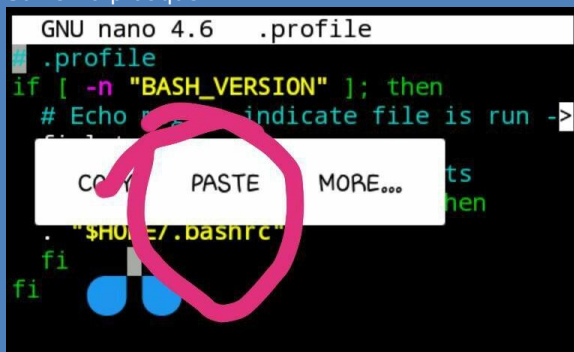
- Installation de NANO

```
$ nano .profile
```

2-Après avoir créé le fichier avec les commandes plus haut, vous allez ensuite copier/coller ce code ci-dessous pour le mettre dans Termux dans la fenêtre de NANO.

```
# .profile
if [ -n "BASH_VERSION" ]; then
#collez votre code en bas
figlet kal
# include .bashrc if it exists
if [ -f "$HOME/.bashrc" ]; then
. "$HOME/.bashrc"
fi
fi
```

Suivez la pratique ->



```
GNU nano 4.6 .profile
# .profile
if [ -n "BASH_VERSION" ]; then
# Echo msg to indicate file is run ->
figlet kal
# include .bashrc if it exists
if [ -f "$HOME/.bashrc" ]; then
. "$HOME/.bashrc"
fi
fi
```

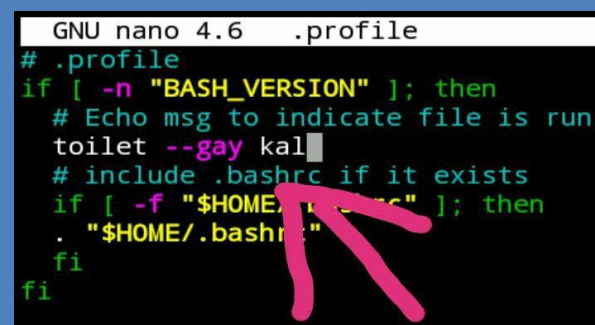
- pour coller dans termux, faites un appui long sur l'écran et vous verrez un menu et appuyez dessus.
- voici et le code sera collé, enregistrer avec ctrl + s puis y pour confirmer
- enfin faites ctrl + x pour quitter nano et après faites exit pour fermer termux et revenez après pour l'ouvrir.

3-Comme vous avez sûrement dû le remarquer dans le code il y'a une espace d'appel pour les programmes qui va créer le profil, si vous ne la voyez pas c'est du côté des commentaires(#), en dessous d'eux c'est là que nous allons appeler les programmes (packages) qu'on avait eu à installer en début. Comment va-t-on faire ?

1. Et bien vous aurez vu que dans le code que je vous ai montré il y a une espace de commentaire(#) qui dit collez votre code en bas, alors placez UN de ce qui suis là-dedans.

<-

- Toilet : toilet <option>  
<texteAfficher>



```
GNU nano 4.6 .profile
# .profile
if [ -n "BASH_VERSION" ]; then
# Echo msg to indicate file is run
toilet --gay kal
# include .bashrc if it exists
if [ -f "$HOME/.bashrc" ]; then
. "$HOME/.bashrc"
fi
fi
```

- J'ajoute des effets supplémentaires pour améliorer la qualité de mon profil

```
$ nano .profile
$ cd
$ nano .profile
$ exit
```

ESC → CTRL ALT - ↓ ↑

- Ce qu'il me reste c'est de fermer nano et termux avec CTRL + S et CTRL + X et ensuite venir exit dans termux

```
#
# m mm "#
# m" " #
#"# m" "#
#"# "m" "#
# "m "mm"#" "mm

$
```

Ainsi vous l'auriez surement compris que vous devez choisir un, parmi les outils et le mettre dans le code du fichier `profil.h` et après la sauvegarder et ensuite redémarrer votre Termux pour voir l'effet et idem que pour la personnalisation il existe de nombreuses packages qui peuvent vous aider à en faire plus mais moi je m'abtiens ici.

## **Tout comprendre sur le bash.bashrc**

Le `bash.bashrc` est un fichier système genre dire que bash ( Bourne Again Shell), sur linux le shell est l'interface en ligne de commande que vous avez pour communiquer avec le système de manière plus

efficace que autres. Avec le bash vous savez aussi ce qui se passe sur la ligne de commande genre comment est-ce indiquer l'affichage de l'utilisateur superSU ou non-superSU par des symboles, comment gère-t-on les symboles pour les chemins de fichiers, le listing de répertoire, la coloration de diverses types de fichiers et de privilèges sur les fichiers, la gestion des processus, bref pour ne citer que ceux-là sachez que le bash est TOUT alors toucher y seulement si vous en connaissez comment l'utiliser et c'est pour ça que dans la personnalisation on a utilisé un pont pour communiquer avec le bash.

Sachez que sous linux il existe différentes types type de shell exemple, le bash, zsh, tcsh, ksh, etc.. tous ces shells ont la particularité de faire des trucs différents dépendamment de l'environnement mais on conseille au débutant le BASH en terme de simplicité, facilite d'utilisation, efficacité, etc.. le fichier `bash.bashrc` faut jamais penser a la `rm -rf` (supprimer) parce que vous aurez de gros dégât, même si toutefois vous pouvez la recréer mais vous allez modifier un petit peu le fonctionnement de Termux, alors éviter d'y toucher un petit peu chers lecteurs.

```
$ screenfetch -A Trisquel
OS: g148ab99
android-af8c380843fd9ac8
OS: Android 5.1
Device: Z812 (xuantan)
ROM: Z812V1.0.0B19
Baseband: msm
Kernel: armv7l Linux 3.10.49-g3d22cfa-005
Uptime: 7h 46m
CPU: ARMv7 Processor rev 0 (v7l)
GPU: ARMv7 Processor rev 0 (v7l)
RAM: 606MiB / 896MiB
```

Ou

```
Setting up screenfetch (3.8.0) ...
$ screenfetch -A Windows
OS: 110232..
Device: 00000000
ROM: 00000000
Baseband: 00000000
Kernel: 00000000
Uptime: 00:00
CPU: 00000000
GPU: 00000000
RAM: 00000000
android-af8c380843fd9ac8
OS: Android 5.1
Device: Z812 (xuantan)
ROM: Z812V1.0.0B19
Baseband: msm
Kernel: armv7l Linux 3.10.49-g3d22cfa-00585-g148ab99
Uptime: 13h 38m
CPU: ARMv7 Processor rev 0 (v7l)
GPU: ARMv7 Processor rev 0 (v7l)
RAM: 597MiB / 896MiB
```

Ces capture sont la vue de mon Termux en estimant que je voulais savoir les infos sur mon téléphone en général mais c'est une belle vu du BASH aussi.

## **Comment installer quelque chose sur Termux ? apt-get install ou pkg install quoi choisir ?**

Installer quelque chose sur Termux est le besoin primaire de tout débutant et professionnel sur Termux car la meilleure façon de jouer d'une minimum d'efficacité dans le travail c'est de faire en sorte que les instruments sont disponibles alors il en va de soi que l'installation reste la meilleure

façon de travailler plus rapide, alors la question que l'on se pose alors c'est quel commande choisir pour installer quelque chose, doit-on voter pour le `apt-get install` ou bien le `pkg install` ? Pour y répondre faisons une parallèle entre les 2 pour voir quoi choisir car on ne peut juger l'un si l'on ne sait pas l'autre.

### **apt-get install VS pkg install**

`apt-get install` : Gestionnaire de paquets officiel de linux, si vous travaillez sur une distribution (debian, ubuntu, etc...) quelconque vous aurez toujours recours avec et bien c'est de même que sur Termux vous pouvez l'utiliser pour réaliser vos installations et c'est sans crainte.

`pkg install` : Gestionnaire de paquets officiel de Termux, le bash le reconnait très bien pour les installations et même les entraides que vous allez trouver a travers les communautés Termuxien(ne) vous conseillerons toujours `pkg`, et le débogage est facile à faire avec.

CONCLUSION, `apt-get install` & `pkg install` font très bien leur job et y'a pas a parier entre eux mais ceux que je peux dire c'est que je préfère `apt` que `pkg` parce que moi je travaille sur linux(debian) et c'est facile pour moi et j'hésite de me tromper

lorsque je switch de Termux a kali-linux.

## Exemple installation de nano

<pkg install nano>

```
$ pkg install nano
Testing the available mirrors:
[*] https://dl.bintray.com/termux/termux-packages-24: ok
[*] https://grimler.se/termux-packages-24: ok
[*] https://main.termux-mirror.ml: bad
[*] https://termux.mentality.rip/termux-packages-24: bad
Picking mirror: https://dl.bintray.com/termux/termux-packages-24
Ign:1 https://dl.bintray.com/termux/termux-packages-24 stable InRelease
Ign:2 https://dl.bintray.com/grimler/game-packages-24 games InRelease
Ign:3 https://dl.bintray.com/grimler/science-packages-24 science InRelease
Ign:4 https://dl.bintray.com/xeffyr/unstable-packages unstable InRelease
Get:5 https://dl.bintray.com/termux/termux-packages-24 stable Release [8255 B]
Get:6 https://dl.bintray.com/grimler/game-packages-24 games Release [5344 B]
Get:7 https://dl.bintray.com/grimler/science-packages-24 science Release [6191 B]
Get:8 https://dl.bintray.com/xeffyr/unstable-packages unstable Release [6065 B]
Get:9 https://dl.bintray.com/termux/termux-packages-24 stable Release.gpg [821 B]
Get:10 https://dl.bintray.com/grimler/game-packages-24 games Release.gpg [475 B]
Get:11 https://dl.bintray.com/grimler/science-packages-24 science Release.gpg [475 B]
Get:12 https://dl.bintray.com/xeffyr/unstable-packages unstable Release.gpg [821 B]
Get:13 https://dl.bintray.com/termux/termux-packages-24 stable/main arm Packages [123 kB]
Fetched 151 kB in 32s (4770 B/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
Reading package lists... Done
Building dependency tree
Reading state information... Done
nano is already the newest version (5.3).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

<image apt-get install >

```
* Install a package: pkg install <package>
* Upgrade packages:  pkg upgrade

Subscribing to additional repositories:

* Root:      pkg install root-repo
* Unstable:  pkg install unstable-repo
* X11:       pkg install x11-repo

Report issues at https://termux.com/issues

$ apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
nano is already the newest version (5.3).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
$ pkg install nano
Testing the available mirrors:
[*] https://dl.bintray.com/termux/termux-packages-24: ok
[*] https://grimler.se/termux-packages-24:  █
```

## Exemple de désinstallation nano

<pkg remove nano>

```
[*] https://dl.bintray.com/termux/termux-packages-24: ok
[*] https://grimler.se/termux-packages-24: ok
[*] https://main.termux-mirror.ml: bad
[*] https://termux.mentality.rip/termux-packages-24: ok
Picking mirror: https://termux.mentality.rip/termux-packages-24
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
$ pkg remove nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  nano
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 791 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 7500 files and directories currently installed.)
Removing nano (5.3) ...
$ █
```

<apt remove nano>

```
After this operation, 791 kB of additional disk space will be used.
Selecting previously unselected package nano.
(Reading database ... 7441 files and directories currently installed.)
Preparing to unpack .../apt/archives/nano_5.3_arm.deb ...
Unpacking nano (5.3) ...
Setting up nano (5.3) ...
update-alternatives: using /data/data/com.termux/files/usr/bin/nano to provide /data/data/com.termux/files/usr/bin/editor (editor) in auto mode
$ apt remove nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  nano
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 791 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 7500 files and directories currently installed.)
Removing nano (5.3) ...
$ █
```



## pkg update && pkg upgrade

*pkg-update* : mise à jour, c'est indispensable sur Termux si vous voulez bien travailler car avec cette option Termux se met à jour pour voir s'il y a du nouveau concernant une installation qui a été déjà faite.

```
$ pkg update
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/grimler/game-packa
ges-21 game InRelease
Ign:3 https://dl.bintray.com/grimler/science-pa
ckages-21 science InRelease
Get:4 https://dl.bintray.com/grimler/game-packa
ges-21 games Release [5344 B]
Hit:4 https://dl.bintray.com/grimler/game-packa
ges-21 games Release
Get:5 https://dl.bintray.com/grimler/science-pa
ckages-21 science Release [5348 B]
Hit:5 https://dl.bintray.com/grimler/science-pa
ckages-21 science Release
0% [Release.gpg gpgv 5344 B]
```

*pkg-upgrade* : mise à jour mais cette fois ci il va télécharger les mises à jour qui existent concernant une installation qui a été déjà faite.

```
$ pkg upgrade
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/grimler/game-packa
ges-21 game InRelease
Ign:3 https://dl.bintray.com/grimler/science-pa
ckages-21 science InRelease
Get:4 https://dl.bintray.com/grimler/game-packa
ges-21 games Release [5344 B]
Hit:4 https://dl.bintray.com/grimler/game-packa
ges-21 games Release
Get:5 https://dl.bintray.com/grimler/science-pa
ckages-21 science Release [5348 B]
Hit:5 https://dl.bintray.com/grimler/science-pa
ckages-21 science Release
0% [Release.gpg gpgv 5344 B]
```

En résumé, *pkg update* vérifie s'il existe des mises à jour, et *pkg upgrade* lui les télécharge et les installe.

## • *pkg install unstable-repo*

Cette option concerne les outils qui sont en dehors de la source directe de Termux avec cette commande vous pouvez être sûr que vos installations d'outils pour faire du hacking sont rassurant, exemple pour installer *sqlmap*, *metasploit*, *tshark*, etc... il est très utile pour faire du hacking avec Termux. La liste est longue mais toutefois vous pouvez vérifier sur le site officiel de Termux.

```
Unpacking nano (5.3) ...
Setting up nano (5.3) ...
update-alternatives: using /data/data/com.termux/files/usr/
bin/nano to provide /data/data/com.termux/files/usr/bin/edi
tor (editor) in auto mode
$ apt remove nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  nano
0 upgraded, 0 newly installed, 1 to remove and 0 not upgrad
ed.
After this operation, 791 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 7500 files and directories currently
installed.)
Removing nano (5.3) ...
$ pkg install unstable-repo
Checking availability of current mirror: ok
Reading package lists... Done
Building dependency tree
Reading state information... Done
unstable-repo is already the newest version (4.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgrad
ed.
$
```

## • *pkg install root-repo(..>)*

Cette option concerne de fournir les outils qui sont en dehors de Termux les privilèges dont ils ont besoin pour s'exécuter exemple avec cette commande vous pouvez être sûr que vos installations d'outils pour faire du hacking ont toutes les privilèges qu'ils ont besoin pour s'exécuter et vous

donnez les résultats comme ça devrait comme pour installer sqlmap et les autres.

```
Removing nano (5.3) ...
$ pkg install unstable-repo
Checking availability of current mirror: ok
Reading package lists... Done
Building dependency tree
Reading state information... Done
unstable-repo is already the newest version (4.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
$ pkg install root-repo
Checking availability of current mirror: ok
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  root-repo
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1120 B of archives.
After this operation, 53.2 kB of additional disk space will be used.
Get:1 https://termux.mentality.rip/termux-packages-24 stable/main arm root-repo all 1.6 [1120 B]
Fetched 1120 B in 1s (1150 B/s)
Selecting previously unselected package root-repo.
(Reading database ... 7441 files and directories currently installed.)
Preparing to unpack .../archives/root-repo_1.6_all.deb ...
Unpacking root-repo (1.6) ...
Setting up root-repo (1.6) ...
Downloading updated package list ...
Ign:1 https://dl.bintray.com/grimler/game-packages-24 games InRelease
Ign:2 https://termux.mentality.rip/termux-packages-24 stable InRelease
Ign:3 https://dl.bintray.com/grimler/termux-root-packages-24 root InRelease
Hit:4 https://termux.mentality.rip/termux-packages-24 stable Release
Ign:5 https://dl.bintray.com/grimler/science-packages-24 science InRelease
Ign:6 https://dl.bintray.com/xeffyr/unstable-packages unstable InRelease
Get:7 https://dl.bintray.com/grimler/game-packages-24 games Release [5344 B]
Get:8 https://dl.bintray.com/grimler/termux-root-packages-24 root Release [6185 B]
Get:9 https://dl.bintray.com/grimler/science-packages-24 science Release [6191 B]
Get:10 https://dl.bintray.com/xeffyr/unstable-packages unstable Release [6065 B]
Get:11 https://dl.bintray.com/grimler/termux-root-packages-24 root Release.gpg [475 B]
```

## **L'éditeur de code NANO sur Termux, pourquoi ?**

Un éditeur de code facile et simple d'utilisation. Bien souvent vous allez tomber dans un besoin de modifier un fichier, genre ce qui est écrit dedans et la y a pas moyen

de plus rapide que nano, l'installation est rapide et très adaptable. Comment l'installer et comment l'ouvrir ?

## INSTALLATION

```
$ pkg install nano
```

```
$ nano <nomfichier>
```

*Suivez la pratique ->*

```
$ pkg install nano
Hit:1 https://termux.net stable InRelease
Ign:2 https://dl.bintray.com/grimler/game-packages-21 games InRelease
Ign:3 https://dl.bintray.com/grimler/science-packages-21 science InRelease
Get:4 https://dl.bintray.com/grimler/game-packages-21 games Release [5344 B]
Hit:4 https://dl.bintray.com/grimler/game-packages-21 games Release
Get:5 https://dl.bintray.com/grimler/science-packages-21 science Release [5348 B]
```

- *j'installe nano*

```
$ ls
Hash-Buster saycheese texte
$ nano texte
```

- *je crée mon fichier de test*

```
GNU nano 4.6      texte
salut a tous
```

- *enfin je tape une petite texte dans mon éditeur de code NANO*

Sur NANO il y a des raccourci clavier à respecter exemple :

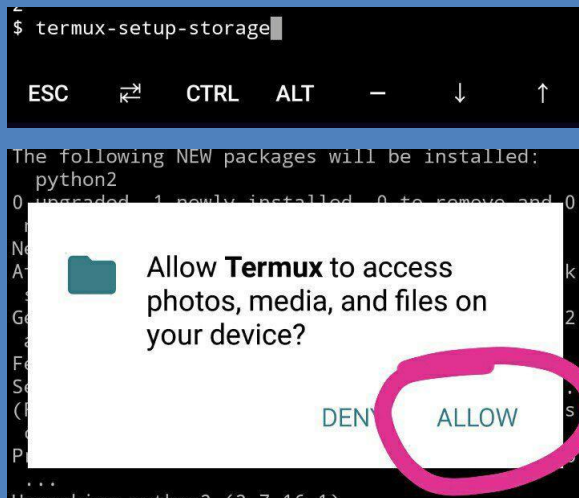
- *Ctrl + s = sauvegarder*
- *Ctrl + z = quitter*

*Etc... le reste à vous de les trouver.*

## **Stockage (termux-setup-storage)**

Sur Termux la plus grande des choses est d'accéder à son stockage interne ou externe, parlant de carte mémoire, et Bien sur Termux c'est très rapide pour accéder avec et aussi pour donner les privilèges avec termux accéder au stockage. La commande c'est bien `termux-setup-storage` et lorsque vous allez l'entrez ça va vous demander quelque chose genre acceptez-vous de donner les privilèges à Termux pour qu'il puisse accéder à votre stockage interne et c'est là que vous allez ACCEPTER.

\$ `termux-setup-storage`



- Ainsi pour accéder au stockage interne du téléphone vous devez donner l'autorisation et par la suite vous verrez un répertoire nommer storage ( sur mon téléphone mais il y'a certains ça se nomme sdcard).

## **Additionnel Key & raccourci clavier**

Quand Termux est START, vous allez remarquer que il y a des touches additionnelles qui s'ajoute à votre clavier et bien pourquoi ? parce que les développeurs de Termux ont constaté que il existe beaucoup de clavier qui sont pas HACKERS Keyboard, c'est-à-dire qui n'a pas presque tous les touches d'un clavier PC sous android et bien il ont fait en sorte que des touches additionnelles puissent être ajouter sans inquiétude via des commandes passer au clavier. Alors il en existe beaucoup mais vous pouvez adapter a votre besoin.

Voici le lien pour perfectionner votre clavier a votre guise : <https://wiki.termux.com/wiki/Touch Keyboard>

Exemple:

- CTRL+A → déplace le curseur pour le mettre vers l'avant
  - CTRL+C → annuler une action en cours
  - CTRL+D → fermé la session en cours
  - CTRL+E → déplace le curseur pour le mettre en fin de ligne
- Etc....

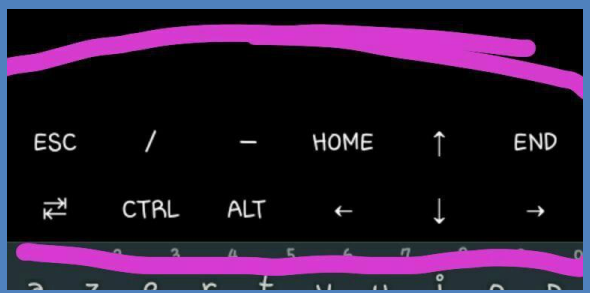


Copier le code ci-dessous et coller le dans votre termux et tout va bien fonctionner.

```
mkdir $HOME/.termux/ ;echo "extra-keys = [[ 'ESC', '/', '-', 'HOME', 'UP', 'END'], ['TAB', 'CTRL', 'ALT', 'LEFT', 'DOWN', 'RIGHT']] " >> $HOME/.termux/termux.properties && termux-reload-settings && sleep 1 && logout
```



```
$ mkdir $HOME/.termux/ ;echo "extra-keys = [[ 'ESC', '/', '-', 'HOME', 'UP', 'END'], ['TAB', 'CTRL', 'ALT', 'LEFT', 'DOWN', 'RIGHT']] " >> $HOME/.termux/termux.properties && termux-reload-settings && sleep 1 && logout
```



## Quelques commandes souvent utilisées

chmod - donne des autorisations spécifiques à un fichier spécifique.

```
$ ls
kal newNameFolder newfile
$ chmod 277 newfile
$ ls -la
total 24
drwx----- 4 u0_a120 u0_a120 4096 Jun 14 09:52 .
drwx----- 4 u0_a120 u0_a120 4096 Jun 7 13:33 ..
-rw----- 1 u0_a120 u0_a120 1363 Jun 11 16:44 .bash_history
-rw----- 1 u0_a120 u0_a120 279 Jun 9 20:55 .profile
drwx----- 3 u0_a120 u0_a120 4096 Jun 14 09:49 kal
drwx----- 3 u0_a120 u0_a120 4096 Jun 14 09:49 newNameFolder
--w-rwxrwx 1 u0_a120 u0_a120 0 Jun 14 09:52 newfile
$ chmod +x newfile
$ ls -la
total 24
drwx----- 4 u0_a120 u0_a120 4096 Jun 14 09:52 .
drwx----- 4 u0_a120 u0_a120 4096 Jun 7 13:33 ..
-rw----- 1 u0_a120 u0_a120 1363 Jun 11 16:44 .bash_history
-rw----- 1 u0_a120 u0_a120 279 Jun 9 20:55 .profile
drwx----- 3 u0_a120 u0_a120 4096 Jun 14 09:49 kal
drwx----- 3 u0_a120 u0_a120 4096 Jun 14 09:49 newNameFolder
--w-rwxrwx 1 u0_a120 u0_a120 0 Jun 14 09:52 newfile
$
```

cat - regarde, modifie ou combine un fichier.

```
$ ls
file1 file2 file3 file4 file5 php projet
$ echo "bonjour tout le monde" > file1
$ cat file1
bonjour tout le monde
$
```

mkdir - créer un ou des répertoires.

```
$ ls
$ mkdir bonjour
$ mkdir folder
$ mkdir kal
$ ls
bonjour folder kal
$ mv bonjour/ kal/
$ ls
folder kal
```

cd - passer d'un répertoire à un autre.

```
$ cd /
$ pwd
/
$ cd -
/data/data/com.termux/files/home
```

cd .. : permet de revenir en arrière

cd . : permet de rester dans le répertoire courant

cd / : permet d'aller à la racine

cd - : permet de revenir à la source avant que le terminal soit close.

cp - Copie un fichier d'un endroit à un autre.

```
$ ls
fichier1 folder
$ cp -r fichier1 folder/
$ cd folder/
$ ls
fichier1
$ pwd
/data/data/com.termux/files/home/folder
$
```

exit - quitte le shell dans lequel vous êtes.

ifconfig - configuration de l'interface réseau. Bon pour obtenir votre propre adresse IP.

```
$ ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txq
ueuelen 0 (UNSPEC)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

rmnet0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 143
0
    inet 10.0.0.79 netmask 255.255.255.0 destination 10.9.79.
9
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txq
ueuelen 0 (UNSPEC)
    RX packets 555 bytes 63150009 (60.2 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 56902 bytes 10332268 (9.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

$
```

ls - répertorie tous les fichiers dans un répertoire.

```
/data/data/com.termux/files/home
$ ls
Hash-Buster bash.bashrc saycheese texte
$ ls -lat
total 48
drwx----- 3 u0_a120 u0_a120 4096 May  3 15:36
saycheese
-rw----- 1 u0_a120 u0_a120 2082 May  3 11:25
```

Ls -lat : permet de voir les plus de détails sur les fichiers lister

mv - Déplace un fichier d'un endroit à un autre ou renommer un fichier.

```
bonjour folder kal
$ mv bonjour/ kal/
$ ls
folder kal
```

- Il le renomme ainsi pour déplacer c'est la même processus mais on le pointeras Termux sur l'endroit à déposer.

ps - liste des processus. Utile pour voir ce qui fonctionne.

rm - supprime un ou plusieurs fichier spécifié

```
$ ls
fichier1 folder
$ rm -rf fichier1
$ ls
folder
$ rm -rf folder
$ ls
$ ls
$
```

-r : récursive

-f : forcer c'est a dire ne pas demander autorisation

pwd – affiche le répertoire courant

```
$ pwd
/data/data/com.termux/files/home
$ cd /
$ pwd
/
```

chemin aboslu: par de la racine  
/root/home

chemin relatif: par depuis un répertoire  
/home/kal/

## **\$HOME, su ou # et \$**

\$HOME : variable environnement qui a pour valeur le répertoire par défaut des utilisateurs. Tout ce passe le HOME avec Termux et c'est pourquoi si on start Termux et on tape dans la ligne de commande pwd il nous indiquera qu'on est dans le HOME. Je vais trop vous en dire plus car il va de soi que vous comprenez le FHS(Hiérarchisation du Système de Fichiers) de Termux mais sachez que si vous faites ça :

/home = \$HOME = ~ ;

```
$ echo $HOME
/data/data/com.termux/files/home
$ pwd
/data/data/com.termux/files/home
$ echo ~
/data/data/com.termux/files/home
$
```

Vous aurez le même résultat car ça signifie la même chose.

su ou # : superUtilisateur, signifie que l'utilisateur en cours possède toutes les privilèges sur la ligne de commande en cours et avec la commande « su » ça valide le session utilisateur de jouir de beaucoup de droits comme supprimer fichier système, faire des trucs genre illégales,etc...

\$ : signifie que la session en cours ne possède pas toutes les droits sur la ligne de commande c'est-à-dire que les accès sont restreints (appelé non-su).

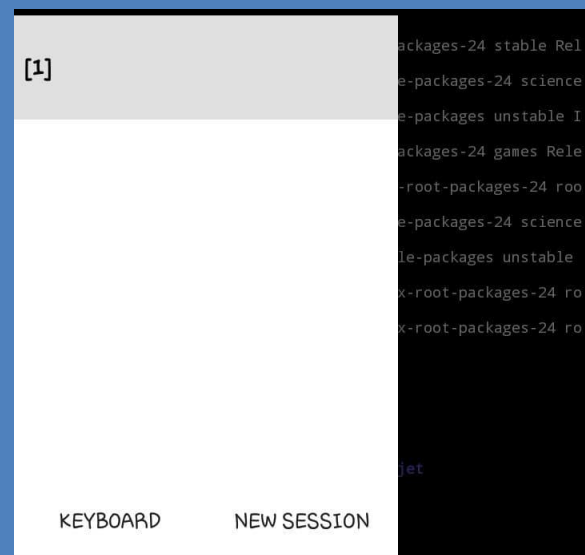
## **Les Sessions**

*Session : Espace utilisateur. Sur Termux ça signifie l'espace de travail de chaque utilisateur qui a une durée de fonctionnement et en gros ça nous rapporte a voir a ce que l'utilisateur fais sur son espace, exemple lorsque l'on parlait du BASH, il y a une rapport très étroite entr'eux puisque si vous détruisez le BASH votre session risque d'être inaccessible,etc... le session est très important et sur Termux vous*

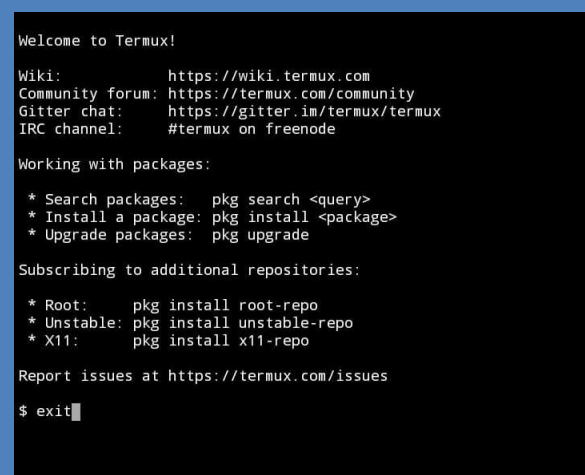
*pouvez en démarrer une multitude pour faire votre boulot. Comment démarrer ou fermer une seconde session sur termux ?*

*<démarrer session>*

- Cliquer sur NEW SESSION



*<fermer session>*



- Ainsi la session se terminera tout seule grâce à la commande exit

## **Remerciements && Références**

Je tenais à remercier d'abord Dieu, pour la disponibilité qu'il nous à accorder de pouvoir écrire ce livre pour vous aider mais on ne vas pas rester là puisque nous prévoyions de faire davantage de tome(édition) mais qui plutôt seras pas générale mais plutôt particulière a un outil sur Termux, ainsi pour faire suite je remercie Ssociety ( communauté haïtienne pour la sécurité informatique) et DoctypeCoders ( communauté de programmeur haïtien ) pour m'avoir conseiller et accompagner à réaliser ce tome 1 ainsi dans le tome 2,on se concentreras plus sur le hacking éthique avec Termux ou l'on va voir Metasploit(pour hack global), aircrack-ng(pour hack le wifi du voisin) fbbrute( pour bruteforcer le compte facebook de quelqu'un), zphisher( pour phisher le compte paypal, insta et autres), etc..... Sur ceux je vous dis Bye à la prochaine dans le tome 2 mais toutefois restez connecté sur notre blog <https://hackbookin.blogspot.com> ou <https://t.me/terlinux>

### SPONSORS:

#### DOCTYPE CODERS

- <http://doctypecoders.com>

#### SECURE SOCIETY

- <http://ssociety.doctypecoders.com>