Installation d'une VM Android

Cette page n'est plus actualisée. À partir de BlueMind 4.8, veuillez consulter la **nouvelle documentation BlueMind**

Présentation

Cette page vous guide dans l'installation d'un environnement Android virtuel sous Linux.

Si vous n'avez pas besoin d'Android Studio (rich IDE), vous pouvez n'installer que le strict nécessaire pour monter une machine virtuelle Android.

Toutes les informations sur les outils en ligne de commande sont disponibles en ligne : https://developer.android.com/studio/command-line

Sur cette page :

- Présentation
- Prérequis
  - Linux générique
  - Instructions spécifiques
- Android Virtual Device
  - Gérer les SDK et outils
  - Gérer les AVDs
  - Astuces
- Screenshots
- Problèmes connus
  - Erreur sdkmanager "Could not create settings"
  - Erreur "Warning: Failed to read or create install properties file."
  - Erreur emulator "PANIC: Broken AVD system path. Check your ANDROID_SDK_ROOT value [/opt/android-sdk]!"
  - Erreur emulator
- Références

Prérequis

Avant d'installer la machine virtuelle en elle-même, il est nécessaire de mettre en place un certain nombre de variables d'environnement et installer le *sdkmanager*.

Vous trouverez ci-dessous les actions à effectuer en fonction de votre distribution (actions génériques, commandes correspondantes pour certaines distributions).

Linux générique

1. Installer Java 8, plus précisément openjdk-8 (cf. https://doc.ubuntu-fr.org/java)
2. Télécharger les outils en ligne de commande : https://developer.android.com/studio#command-tools
3. Décompresser son contenu dans `/opt/android-sdk`
   a. `sudo mkdir -p /opt/android-sdk`
4. Mettre les variables d'environnement en place (cf. https://doc.ubuntu-fr.org/variables_d_environnement#variables_d_environnement_persistantes) :
   a. **`ANDROID_SDK_ROOT`**`=/opt/android-sdk`
   b. ajouter à **`PATH`** :
      i. `/opt/android-sdk/emulator`
      ii. `/opt/android-sdk/platform-tools`
      iii. `/opt/android-sdk/tools/bin`
5. Donner les droits au dossier pour le groupe `android-sdk`
   a. `groupadd android-sdk`
   b. `gpasswd -a $USER android-sdk`
   c. `setfacl -R -m g:android-sdk:rwx /opt/android-sdk`
   d. `setfacl -d -m g:android-sdk:rwX /opt/android-sdk`
   e. `chown :android-sdk /opt/android-sdk -R`
   f. `chmod u+w /opt/android-sdk`
6. vous devez vous relogguez ou exécuter `newgrp android-sdk`
7. Installer `tools` : `sdkmanager --sdk_root=/opt/android-sdk "tools"`

Désormais vous avez tout le nécessaire pour utiliser les outils en ligne de commande afin de gérer votre environnement.

Instructions spécifiques

Retrouvez ici les lignes de commandes correspondant aux étapes décrites dans le chapitre ci-dessus.

Ubuntu

```
$ cd ~
$ wget https://dl.google.com/android/repository/commandlinetools-linux-6609375_latest.zip
$ sudo mkdir -p /opt/android-sdk
$ sudo unzip ~/commandlinetools-linux-6609375_latest.zip -d /opt/android-sdk
$ groupadd android-sdk
$ gpasswd -a $USER android-sdk
$ setfacl -R -m g:android-sdk:rwx /opt/android-sdk
$ setfacl -d -m g:android-sdk:rwX /opt/android-sdk
$ chown :android-sdk /opt/android-sdk -R
$ chmod u+w /opt/android-sdk
$ newgrp android-sdk
$ sdkmanager --sdk_root=/opt/android-sdk "tools"
```

Arch Linux

cf. https://wiki.archlinux.org/index.php/Android#SDK_packages et https://wiki.archlinux.org/index.php/Android#Making_/opt/android-sdk_group-writeable

```
$ yay -S android-sdk
$ groupadd android-sdk
$ gpasswd -a $USER android-sdk
$ setfacl -R -m g:android-sdk:rwx /opt/android-sdk
$ setfacl -d -m g:android-sdk:rwX /opt/android-sdk
$ chown :android-sdk /opt/android-sdk -R
$ chmod u+w /opt/android-sdk
$ newgrp android-sdk
$ sdkmanager --sdk_root=/opt/android-sdk "tools"
```

Android Virtual Device

Gérer les SDK et outils

Quelques commandes utiles de sdkmanager :

```
$ sdkmanager --list
$ sdkmanager --update
```

Avant de continuer assurez-vous d'avoir tous les prérequis en place :

- vérifier le PATH pour l'accès aux commandes

```
$ echo $PATH
…:/opt/android-sdk/emulator:/opt/android-sdk/platform-tools:/opt/android-sdk/tools/bin:/opt/android-sdk
/emulator
```

- vérifier ANDROID_SDK_ROOT

```
$ echo $ANDROID_SDK_ROOT
/opt/android-sdk
```

- vérifier les droits dans le dossier d'installation

```
$ ll -g $ANDROID_SDK_ROOT
Permissions Size User      Group Date Modified Name
drwxrwxr-x@    - user android-sdk 14 Sep 13:13  emulator
drwxrwxr-x@    - user android-sdk 14 Sep 13:19  licenses
drwxrwxr-x@    - user android-sdk 14 Sep 13:11  patcher
drwxrwxr-x@    - user android-sdk 14 Sep 13:13  platform-tools
drwxrwxr-x@    - user android-sdk 14 Sep 13:19  system-images
drwxrwxr-x@    - user android-sdk 14 Sep 13:15  tools
```

- vérifier ce qui est installé

```
$ sdkmanager --list
Installed packages:=====================] 100% Computing updates...
  Path           | Version | Description                | Location
  -------        | ------- | -------                    | -------
  emulator       | 30.0.26 | Android Emulator           | emulator/
  patcher;v4     | 1       | SDK Patch Applier v4       | patcher/v4/
  platform-tools | 30.0.4  | Android SDK Platform-Tools | platform-tools/
```

Gérer les AVDs

Avant de créer un AVD (*Android Virtual Device*), il vous faut une image système installée sur votre environnement.

Ensuite vous pouvez créer un AVD selon un profil de hardware.

Si ce n'est pas le cas, installez les éléments nécessaires :

```
$ sdkmanager "platform-tools" "emulator" "platforms;android-30" "system-images;android-30;google_apis;x86_64"
```

Enfin, vous pouvez créer un AVD suivant l'image souhaitée :

```
$ avdmanager create avd -n avd-android30-pixel -k "system-images;android-30;google_apis;x86_64" --device
"pixel"
```

Ensuite vous pouvez démarrer votre AVD à partir de la ligne de commande :

```
$ emulator -avd avd-android30-pixel
```

Astuces

Clavier physique

Pour utiliser le clavier physique de votre ordinateur dans l'émulateur, modifier la valeur de la variable `hw.keyboard` dans le fichier `~/.android/avd/avd-android30-pixel.avd/config.ini`

config.ini

```
hw.keyboard = yes
```

Manipuler l'émulateur

On peut envoyer des commandes à l'émulateur pour simuler des activités particulières : recevoir un sms, modifier la position du gps, la puissance du gsm, etc.

La liste des actions possibles est disponible ici : https://developer.android.com/studio/run/emulator-console

```
telnet localhost 5554

Trying ::1...
Connected to localhost.
Escape character is '^]'.
Android Console: Authentication required
Android Console: type 'auth <auth_token>' to authenticate
Android Console: you can find your <auth_token> in
'/home/user/.emulator_console_auth_token'
OK
auth FEHdxl/bQ4B24S2l
Android Console: type 'help' for a list of commands
OK
sms send 4085555555 hi there
OK
```

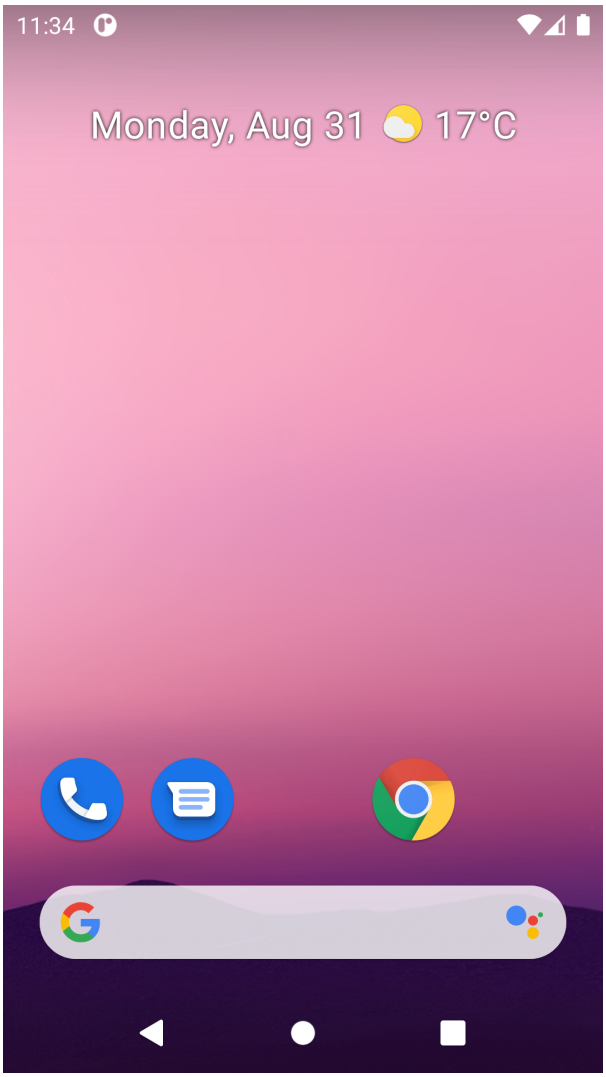
Ouvrir un root shell

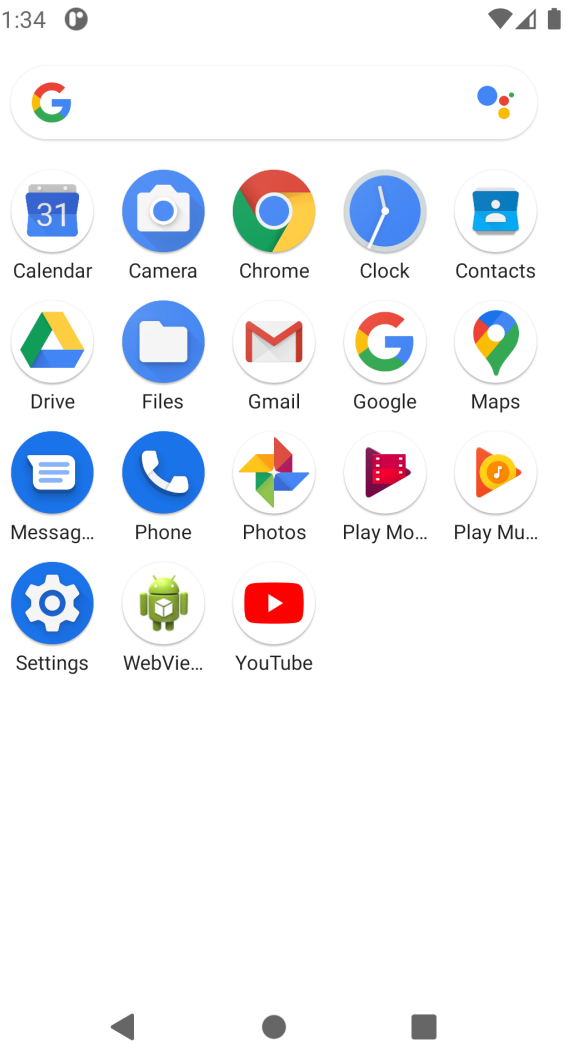
Créez une console *root shell* sur le terminal actuel. Elle diffère de la commande *shell adb* des manières suivantes :

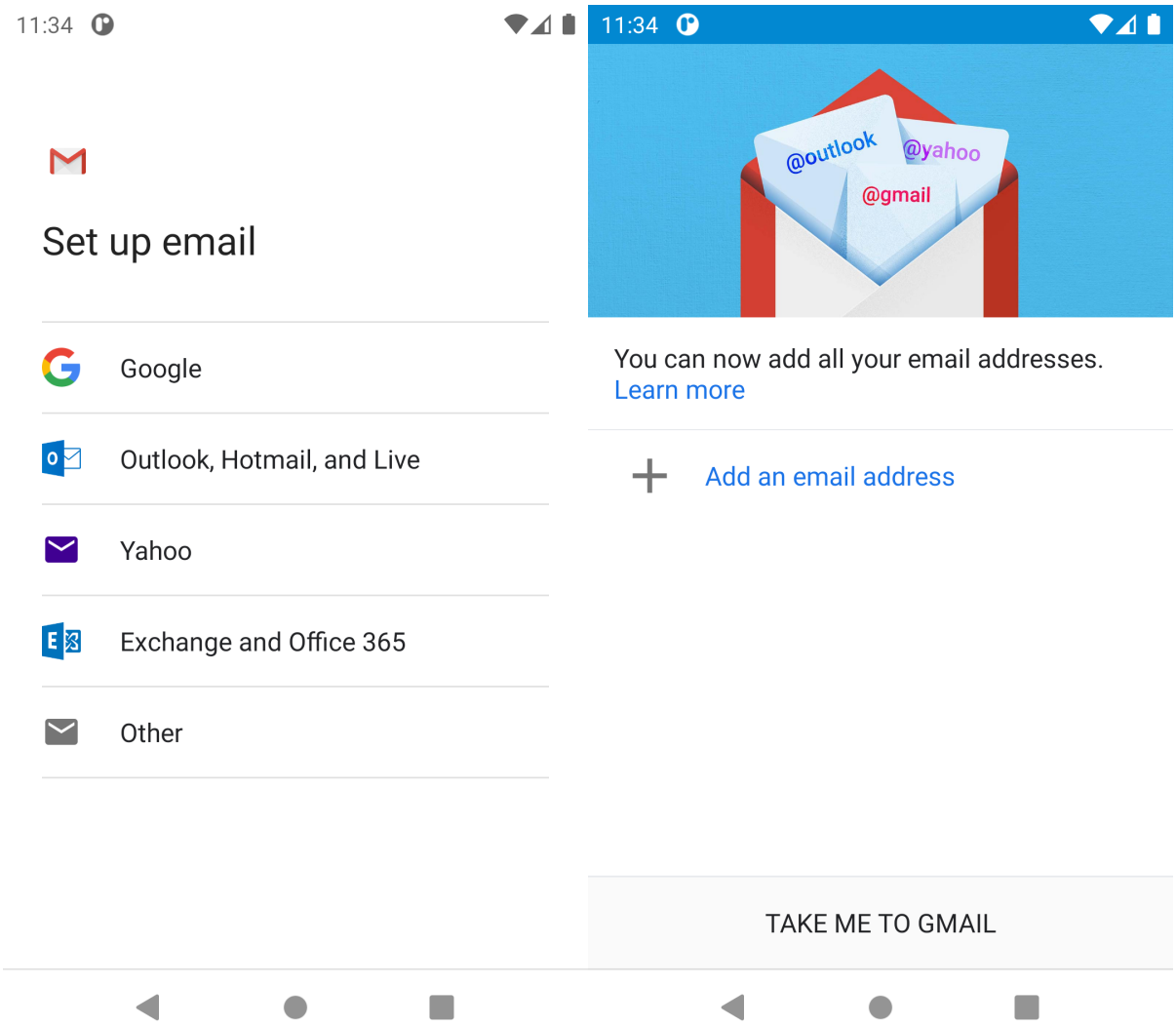
- Elle crée un shell racine qui vous permet de modifier de nombreuses parties du système.
- Elle fonctionne même si le démon adb dans le système émulé est défectueux.
- En appuyant sur Ctrl+C (C), l'émulateur s'arrête à la place du shell.

cf. https://developer.android.com/studio/run/emulator-commandline#advanced

Screenshots


11:34
Monday, Aug 31 17°C


11:34
Calendar
Camera
Chrome
Clock
Contacts
Drive
Files
Gmail
Google
Maps
Messag...
Phone
Photos
Play Mo...
Play Mu...
Settings
WebVie...
YouTube



Problèmes connus

Erreur sdkmanager "Could not create settings"

Quelle que soit la commande de `sdkmanager`, on obtient cette réponse :

```
$ sdkmanager --list
Warning: Could not create settings
java.lang.IllegalArgumentException
        at com.android.sdklib.tool.sdkmanager.SdkManagerCliSettings.<init>(SdkManagerCliSettings.java:428)
        at com.android.sdklib.tool.sdkmanager.SdkManagerCliSettings.createSettings(SdkManagerCliSettings.
java:152)
        at com.android.sdklib.tool.sdkmanager.SdkManagerCliSettings.createSettings(SdkManagerCliSettings.
java:134)
        at com.android.sdklib.tool.sdkmanager.SdkManagerCli.main(SdkManagerCli.java:57)
        at com.android.sdklib.tool.sdkmanager.SdkManagerCli.main(SdkManagerCli.java:48)
```

Solution

Il faut aider l'application java à trouver le root sdk :

```
$ sdkmanager --sdk_root=/opt/android-sdk --list
```

Erreur "Warning: Failed to read or create install properties file."

Impossible d'écrire dans le répertoire `/opt/android-sdk`.

Solution

```
$ sudo chmod g+w /opt/android-sdk
```

Erreur emulator "PANIC: Broken AVD system path. Check your ANDROID_SDK_ROOT value [/opt/android-sdk]!"

cf. https://stackoverflow.com/questions/39645178/panic-broken-avd-system-path-check-your-android-sdk-root-value

Erreur emulator

Il y a deux binaires dans l'installation `/opt/android-sdk/tools/emulator` et `/opt/android-sdk/emulator/emulator`. C'est un peu obscur.

Moins de problèmes ont été rencontré avec la version installée dans `/opt/android-sdk/emulator` (cf. `sdkmanager --list`). Pour autant selon la variable `$PATH`, c'est l'autre binaire qui sera utilisé (cf. `which emulator`)

Références

https://medium.com/michael-wallace/how-to-install-android-sdk-and-setup-avd-emulator-without-android-studio-aeb55c014264