

# Mise en place des outils en ligne de commande sur MacOS

La première étape consiste à installer le gestionnaire de paquet brew (c'est l'équivalent de apt sur Linux) :

Ouvrir un terminal et dans ce terminal, exécuter la commande (sur une seule ligne) :

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

## Installation de doxygen

Dans le terminal, exécuter la commande :

```
brew install doxygen
```

## Installation de SDL, SDL\_Image et SDL\_TTF

Dans le terminal, exécuter les commandes :

```
brew install sdl  
brew install sdl_image  
brew install sdl_ttf
```

## Installation de SFML

Dans le terminal, exécuter la commande :

```
Brew install sfml
```

## Installation de valgrind

Dans le terminal, exécuter les commandes :

```
brew tap LouisBrunner/valgrind  
brew install --HEAD LouisBrunner/valgrind/valgrind
```

## Installation de gdb

Dans le terminal, exécuter la commande :

```
brew install gdb
```

Une fois la commande terminée, testez l'exécution de gdb sur l'un de vos programmes :

```
gdb ./bin/exemple  
...  
(gdb) run
```

Lors de l'exécution de votre programme, gdb devrait vous afficher quelque chose comme cela :

```
Starting program: /Users/.../bin/exemple  
[New Thread 0x2503 of process 36397]
```

Si c'est le cas, tout est installé et vous pouvez travailler sans problème.

Si gdb vous affiche un message d'erreur :

```
Starting program: /Users/.../bin/exemple  
Unable to find Mach task port for process-id 18048: (os/kern) failure (0x5).  
(please check gdb is codesigned - see taskgated(8))
```

Dans ce cas il vous faut certifier manuellement l'exécutable de gdb.

## Certification de gdb

Ces étapes ne sont à faire que si l'exécution d'un programme dans gdb ne se fait pas.

### 1. Vérification du certificat

Dans le terminal, exécutez la commande

```
security find-certificate -c gdb-cert
```

Si vous obtenez le message :

```
security: SecKeychainSearchCopyNext: The specified item could not be found in the keychain.
```

Le certificat n'existe pas et il faut le créer (étape 3)

Si vous obtenez un message de la forme :

```
keychain: "/Library/Keychains/<<NOM>>.keychain"
```

Vérifiez que le certificat est bien dans le trousseau système : <<NOM>> == System.

Si oui passez à l'étape 4, sinon supprimez le certificat (étape 2) puis recréez-le (étape 3).

### 2. Suppression d'un certificat

- Ouvrir l'application trousseau (dans Application/Utilitaires/Trousseaux d'accès (icône de porte-clefs))
- Dans le volet haut-gauche (Trousseaux), choisir le bon emplacement (vu à l'étape 1 : <<NOM>>)
- Dans le volet droit, faire un clic droit sur le certificat à supprimer, puis choisir `Supprimer ...` dans le menu contextuel
- Saisissez votre mot de passe et cliquer sur `Modifier le trousseau`
- Fermer l'application `Trousseaux d'accès`

### 3. Création d'un certificat

- Ouvrir l'application trousseau (dans Application/Utilitaires/Trousseaux d'accès (icône de porte-clefs))
- Dans le menu `Trousseaux d'accès > Assistant de certificat`, choisir `Créer un certificat...`
- Dans la fenêtre qui s'ouvre inscrire `Nom: gdb-cert`, `Type d'identité: Racine auto-signée`, `Type de certificat: Signature de code`. Cocher la case `Me laisser ignorer les réglages par défaut` puis cliquer sur `Créer`
- Dans les 7 fenêtres suivantes (durée de validité, informations, informations bi-clé, Extension d'utilisation de clé, Extension d'utilisation de clé étendue, Extension des contraintes élémentaires, Extension de nom secondaire d'objet), ne changez rien et cliquez sur `Continuer`
- 4 Dans la fenêtre Indiquez l'emplacement du certificat`, Choisir le trousseau `Système` puis cliquer sur `Créer`
- Saisir votre mot de passe pour valider les modifications, puis cliquer sur `Terminer`
- Fermer l'application `Trousseaux d'accès` puis refaites l'étape 1 pour vérifier que le certificat a bien été créé au bon endroit (ie dans le bon trousseau) avant de passer à l'étape 4

### 4. Vérifier que le certificat n'est pas expiré

Dans le terminal, exécuter la commande :

```
security find-certificate -p -c gdb-cert | openssl x509 -checkend 0
```

Cette commande doit renvoyer

```
Certificate will not expire
```

Si le certificat a expiré -> étapes 2, 3 puis 4

## 5. Faire confiance au certificat pour la signature de code

- Ouvrir l'application trousseau (dans Application/Utilitaires/Trousseaux d'accès (icône de porte-clefs))
- Dans le volet haut-gauche (Trousseaux), choisir le trousseau `Système`
- Dans le volet droit, faire un clic droit sur le bon certificat (de nom `gdb-cert`), puis choisir `Lire les informations` dans le menu contextuel
- Dans la fenêtre qui s'ouvre, développer l'arborescence `Se fier`
- Dans la liste de saisie pour `signature de code`, choisir `Toujours approuver`
- Fermer la fenêtre du certificat
- Fermer l'application `Trousseaux d'accès`

## 6. Signature et validation de l'exécutable de gdb

- Créer un fichier **gdb-entitlement.xml** sur le bureau
- Éditer le fichier (attention avec TextEdit, pensez à convertir au format texte, menu Format) pour y mettre le contenu suivant

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>com.apple.security.cs.allow-jit</key>
  <true/>
  <key>com.apple.security.cs.allow-unsigned-executable-memory</key>
  <true/>
  <key>com.apple.security.cs.allow-dyld-environment-variables</key>
  <true/>
  <key>com.apple.security.cs.disable-library-validation</key>
  <true/>
  <key>com.apple.security.cs.disable-executable-page-protection</key>
  <true/>
  <key>com.apple.security.cs.debugger</key>
  <true/>
  <key>com.apple.security.get-task-allow</key>
  <true/>
</dict>
</plist>
</pre>
```

- Sauvegarder le fichier, puis fermer l'éditeur
- Pour signer gdb, dans le terminal, exécuter la commande :

```
codesign --entitlements gdb-entitlement.xml -fs gdb-cert $(which gdb)
```

- Vérifier le tout avec la commande

```
codesign -vv $(which gdb)
```

qui doit vous renvoyer

```
valid on disk
```

```
satisfies its Designated Requirement
```

## 7. Rafraichir les certificats

- Dans le terminal, exécuter la commande :

```
sudo killall taskgated
```

## 8. Vérification

- Tester gdb comme indiqué dans la partie installation