

リリース用署名

<http://magpad.jugem.jp/?eid=78>

前提

- Android アプリケーションを Android 端末へインストールするためには、apk ファイルへの署名が必要。
- 署名に必要な非公開鍵の生成と管理を行うのが keytool。
- keytool で生成された非公開鍵を使用して、apk ファイルへ署名を行うのが jarsigner。
- keytool で非公開鍵を生成し、jarsigner で署名する。

keytool

- 非公開鍵を格納するためのキーストアファイル (.keystore) の作成と非公開鍵 (=キー) の生成を行う。
- キーストアとキーそれぞれに対してパスワードを設定することができる。
- キーストアには複数のキーを格納することができる。(alias で区別する。)
- -genkey オプションは -genkeypair の旧名称だが、引き続き使用可能。
- Android では 1 つのキーストアの中に 1 つのキーを設定すればよい。
- Android ではデバッグ用とアプリケーション公開用の 2 つのキーストアファイルが必要。

キーストア / キーの生成を行う方法

キーストア名 : hoge.keystore

キーストアのパスワード : pass_keystore

キーのエイリアス : alias_key1

キーのパスワード : pass_key1

CN : CN=HOGE,O=FUGA,C=JP

生成アルゴリズム : RSA

証明書の有効日数 : 10000 日

として、DOS からキーストア / キーの生成を行う。

```
C:¥>keytool -genkey -v ^
-keystore hoge.keystore ^
-storepass pass_keystore ^
-alias alias_key1 ^
-keypass pass_key1 ^
-dname "CN=HOGE,O=FUGA,C=JP" ^
-keyalg RSA ^
-validity 10000
```

alias と keypass を変えて -genkey を実行すれば、同一キーストアに対して複数のキーを登録することができる

生成されたキーストアの内容を表示する方法

```
C:¥>keytool -list -v ^
-keystore hoge.keystore ^
-storepass pass_keystore
```

別名や発行者などが合っていれば OK。

jarsigner

- ・ keytool で生成したキーストアファイルと非公開鍵を使って、apk ファイルに署名を行う。
- ・ 非公開鍵の指定は alias で行う。
- ・ 「-verify」 オプションで署名の検証を行える。
- ・ コマンド引数の順番は「jarsigner オプション apk ファイル名 エイリアス名」の順で行う。

apk ファイルに署名を行う方法

上記で生成した「hoge.keystore」を使って「piyo.apk」に署名を行う。

```
C:¥>jarsigner -verbose ^
-keystore hoge.keystore ^
-storepass pass_keystore ^
-keypass pass_key1 ^
piyo.apk ^
alias_key1
```

署名の検証を行う方法

```
C:¥>jarsigner -verify -verbose -certs piyo.apk
```

デバッグ用署名

通常は、自動的にデバッグ用の署名が作られるので必要なし。

<http://magpad.jugem.jp/?eid=77>

デバッグ用のキーストア / キーは以下の値で生成する。

キーストア名「debug.keystore」

キーストアのパスワード「android」

キーのエイリアス「androiddebugkey」

キーのパスワード「android」

CN「CN=Android Debug,O=Android,C=US」

DOS から

```
C:\>keytool -genkey -v ^
-keystore debug.keystore ^
-alias androiddebugkey ^
-keyalg RSA ^
-validity 10000 ^
-dname "CN=Android Debug,O=Android,C=US"
```

これで「debug.keystore」が生成されるので、-list オプションを使って確認。

```
C:\>keytool -list -v ^
-keystore debug.keystore ^
-storepass android
```