How to create a signed APK file using Cordova command line interface?

Step 1:

```
D:\projects\Phonegap\Example> cordova plugin rm org.apache.cordova.console --save
```

add the `--save` so that it removes the plugin from the `config.xml` file.

Step 2:

To generate a release build for Android, we first need to make a small change to the `AndroidManifest.xml` file found in platforms/android. Edit the file and change the line:

```
<application android:debuggable="true" android:hardwareAccelerated="true"
android:icon="@drawable/icon" android:label="@string/app_name">
```

and change `android:debuggable` to `false`:

```
<application android:debuggable="false" android:hardwareAccelerated="true"
android:icon="@drawable/icon" android:label="@string/app_name">
```

As of cordova 6.2.0 remove the android:debuggable tag completely. Here is the explanation from cordova:

Explanation for issues of type "HardcodedDebugMode": It's best to leave out the android:debuggable attribute from the manifest. If you do, then the tools will automatically insert android:debuggable=true when building an APK to debug on an emulator or device. And when you perform a release build, such as Exporting APK, it will automatically set it to false.

If on the other hand you specify a specific value in the manifest file, then the tools will always use it. This can lead to accidentally publishing your app with debug information.

Step 3:

Now we can tell cordova to generate our release build:

```
D:\projects\Phonegap\Example> cordova build --release android
```

Then, we can find our unsigned APK file in `platforms/android/ant-build`. In our example, the file was `platforms/android/ant-build/Example-release-unsigned.apk`

Step 4:

Note : We have our keystore `keystoreNAME-mobileapps.keystore` in this Git Repo, if you want to create another, please proceed with the following steps.

Key Generation:

Syntax:

```
keytool -genkey -v -keystore <keystoreName>.keystore -alias <Keystore AliasName> -
keyalg <Key algorithm> -keysize <Key size> -validity <Key Validity in Days>
```

Egs:

```
keytool -genkey -v -keystore NAME-mobileapps.keystore -alias NAMEmobileapps -keyalg
RSA -keysize 2048 -validity 10000
```

```
keystore password? : xxxxxxx
What is your first and last name? :  xxxxxx
What is the name of your organizational unit? :  xxxxxxxx
What is the name of your organization? :  xxxxxxxxx
What is the name of your City or Locality? :  xxxxxxx
What is the name of your State or Province? :  xxxxx
What is the two-letter country code for this unit? :  xxx
```

Then the Key store has been generated with name as NAME-mobileapps.keystore

Step 5:

Place the generated keystore in

old version cordova

```
D:\projects\Phonegap\Example\platforms\android\ant-build
```

New version cordova

```
D:\projects\Phonegap\Example\platforms\android\build\outputs\apk
```

To sign the unsigned APK, run the jarsigner tool which is also included in the JDK:

Syntax:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore <keystorename
<Unsigned APK file> <Keystore Alias name>
```

Egs:

```
D:\projects\Phonegap\Example\platforms\android\ant-build> jarsigner -verbose -sigalg
SHA1withRSA -digestalg SHA1 -keystore NAME-mobileapps.keystore Example-release-
unsigned.apk xxxxxmobileapps
```

OR

```
D:\projects\Phonegap\Example\platforms\android\build\outputs\apk> jarsigner -verbose -
sigalg SHA1withRSA -digestalg SHA1 -keystore NAME-mobileapps.keystore Example-release-
unsigned.apk xxxxxmobileapps

Enter KeyPhrase as 'xxxxxxxx'
```

This signs the apk in place.

Step 6:

Finally, we need to run the zip align tool to optimize the APK:

```
D:\projects\Phonegap\Example\platforms\android\ant-build> zipalign -v 4 Example-
release-unsigned.apk Example.apk
```

OR

```
D:\projects\Phonegap\Example\platforms\android\ant-build> C:\Phonegap\adt-bundle-
windows-x86_64-20140624\sdk\build-tools\android-4.4W\zipalign -v 4 Example-release-
unsigned.apk Example.apk
```

OR

```
D:\projects\Phonegap\Example\platforms\android\build\outputs\apk> C:\Phonegap\adt-
bundle-windows-x86_64-20140624\sdk\build-tools\android-4.4W\zipalign -v 4 Example-
release-unsigned.apk Example.apk
```

Now we have our final release binary called example.apk and we can release this on the Google Play Store.

An update to @malcubierre for Cordova 4 (and later)-

Create a file called `release-signing.properties` and put in `APPFOLDER\platforms\android` folder

Contents of the file: edit after = for all except 2nd line

```
storeFile=C:/yourlocation/app.keystore
storeType=jks
keyAlias=aliasname
keyPassword=aliaspass
storePassword=password
```

Then this command should build a release version:

```
cordova build android --release
```