**一、添加插件**

1. 新建Cordova项目

cordova create CordovaHotCode com.ezample.hotcode

1. 添加android平台

在新建项目的根目录下，进入命令行，cordova platform add android

1. 添加自动更新插件

cordova plugin add cordova-hot-code-push-plugin

1. 添加开发拓展

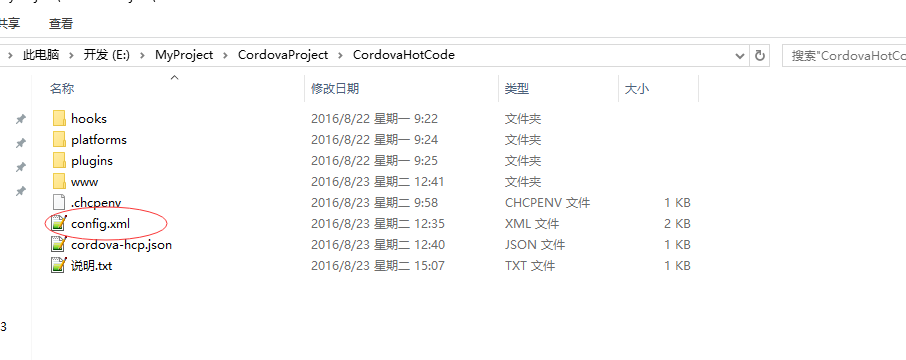
cordova plugin add cordova-hot-code-push-local-dev-addon

1. 添加cordova hot code push客户端，用于生成www目录下文件的hash码，更新的时候对比使用。

npm install -g cordova-hot-code-push-cli

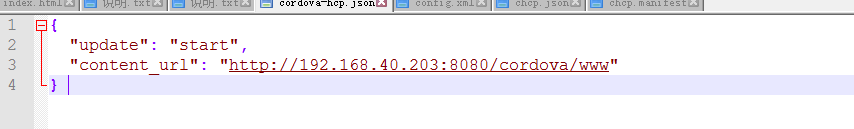
**二、配置**

1、修改根目录下的config.xml





1. 上一步执行完之后，需要在项目根目录下的命令行中执行cordova build，编译项目。
2. 项目根目录下执行cordova-hcp build，这个命令在项目根目录下的www目录中生成chcp.json和chcp.manifest两个文件。Chcp.json包含了项目的版本信息，是否立即更新，服务器的更新地址。Chcp.manifest包含了www目录下的文件和对应的hash码值。为了chcp.json文件生成的方便，我们需要在项目的根目录下新建一个文件cordova-hcp.json，内容为：



**{**

**"update": "start",**

**"content\_url": "http://192.168.40.203:8080/cordova/www"**

**}**

1. 上面执行完之后，可以将项目导入到android studio中或者直接在根目录执行cordova build,然后在CordovaHotCode\platforms\android\build\outputs\apk这个目录下生成了apk文件，直接安装就行。

**三、服务器端配置**

1、服务器端就比较简单了，我们修改项目根目录下的www目录之后，在根目录下执行cordova-hcp build.

2、直接把根目录下的www目录拷贝到服务器中，注意地址不要写错。然后启动我们的app,第一次启动的时候检查更新并下载更新，第二次启动的时候就会使用更新的版本。

参考网址：

<http://blog.csdn.net/lovelyelfpop/article/details/50848524#javascript-module>

http://blog.csdn.net/qq\_34160678/article/details/51580975