

# SPEEDY SOLUTIONS

27  
4 / 2015

## Cordova - Gerando e Publicando apk do Android

Semana passada enfrentei um problema ao gerar o apk para Android utilizando o Cordova. A solução que achei irei explicar abaixo.

### 1 - Crie uma keystore para o seu apk

Para publicar seu apk na Play Store, é necessário que ele esteja assinado. Para isso, primeiro é preciso gerar uma keystore. Abra o CMD e digite o código abaixo:

```
keytool -genkey -v -keystore nome-da-keystore.keystore -alias  
nome-do-alias -keyalg RSA -keysize 2048 -validity 10000
```

O nome da keystore pode ser qualquer um, mas para melhor identificação gosto de colocar algo que envolva o nome do app. Por exemplo meuprimeiroapp.keystore. O nome do alias é importante e essa informação deve ser guardada. Pode conter underline para melhor leitura. Por exemplo meu-primeiro-app

Após digitar as informações acima do CMD, ele irá pedir uma senha. Coloque uma senha confiável e quando essa informação aparecer a hora de assinar o apk ela se

Também serão req  
Guarde todas as in



**speedysolutions**  
Speedy Solutions

Seguir

```

Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> keytool -genkey -v -keystore meuprimeiroapp.keystore -alias meu-primeiro-app -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Fulano
What is the name of your organizational unit?
[Unknown]: Fulano
What is the name of your organization?
[Unknown]: Fulano
What is the name of your City or Locality?
[Unknown]: São Paulo
What is the name of your State or Province?
[Unknown]: SP
What is the two-letter country code for this unit?
[Unknown]: BR
Is CN=Fulano, OU=Fulano, O=Fulano, L=São Paulo, ST=SP, C=BR correct?
[no]: yes

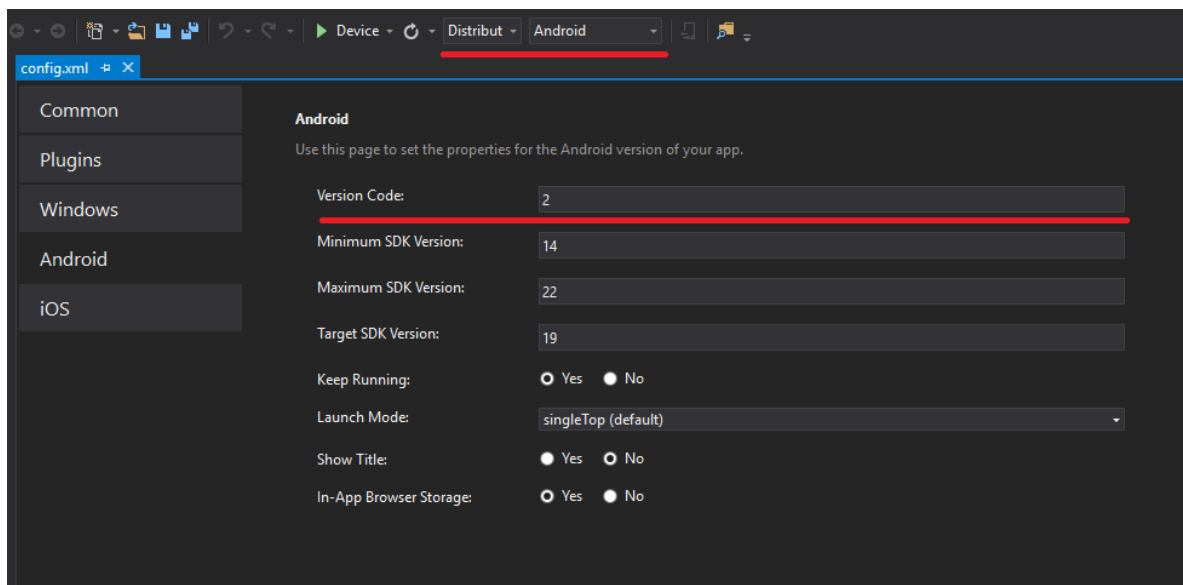
Generating 2048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
for: CN=Fulano, OU=Fulano, O=Fulano, L=São Paulo, ST=SP, C=BR
Enter key password for <meu-primeiro-app>
<RETURN if same as keystore password>:
Re-enter new password:
[Storing meuprimeiroapp.keystore]
PS C:\WINDOWS\system32>

```

## 2 - Gerando o apk

Essa foi a parte em que mais tive problemas. Aparentemente com o Cordova, ao dar build no projeto já deveria gerar o apk assinado, porém não foi isso o que aconteceu.

A solução foi, setar o “Solution Configurations” como “Distribution” e o “Solution Platform” como “Android”. Lembrando que é preciso configurar o config.xml na aba “Android”. É necessário que a cada versão gerada para ser publicada na PlayStore, o version code sera incrementado. Exemplo:



Após isso, aperte F6 (ou dê build no projeto). Pode ser que dê algum erro, mas o importante é que o apk sem assinatura tenha sido gerado. Esse arquivo deve ficar dentro da pasta “bld\Distribution\platforms\android\ant-build” ter o nome “CordovaApp-release-unsigned.apk”

## 3 - Assinando o apk

Em uma pasta separada e de fácil navegação, copie e cole o

“CordovaApp-release-unsigned.apk” e a keystore que foi gerada anteriormente “meu-primeiro-app.keystore”.

Abra novamente o CMD e navegue até a pasta onde estão o APK e a keystore e digite a linha de comando abaixo:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1  
-keystore my-release-key.keystore  
my_application.apk alias_name
```

Substitua o “my-release-key.keystore” pelo nome do arquivo keystore gerado anteriormente e que está na pasta. Substitua “my\_application.apk” pelo arquivo que está na pasta que vou gerado no build. Substitua o “alias\_name” pelo nome do alias que você colocou anteriormente na keystore. Nessa etapa é requisitada a senha que você gerou junto a keystore.

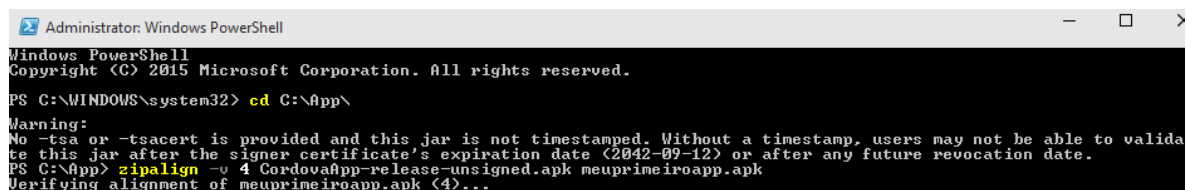


```
Administrator: Windows PowerShell  
Windows PowerShell  
Copyright (C) 2015 Microsoft Corporation. All rights reserved.  
PS C:\WINDOWS\system32> cd C:\App\  
PS C:\App> jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore meu-primeiro-app.keystore CordovaApp-release-unsigned.apk meuprimeiroapp
```

## 4 - Fazer zipalign




Após assinar o apk, é preciso fazer o zipalign para publicá-lo na PlayStore. Para isso abra o cmd e navegue novamente a pasta criada anteriormente e use o comando abaixo:

```
zipalign -v 4 your_project_name-unaligned.apk  
your_project_name.apk
```



```
Administrator: Windows PowerShell  
Windows PowerShell  
Copyright (C) 2015 Microsoft Corporation. All rights reserved.  
PS C:\WINDOWS\system32> cd C:\App\  
Warning:  
No -tsa or -tsacert is provided and this jar is not timestamped. Without a timestamp, users may not be able to validate this jar after the signer certificate's expiration date (2042-09-12) or after any future revocation date.  
PS C:\App> zipalign -v 4 CordovaApp-release-unsigned.apk meuprimeiroapp.apk  
Verifying alignment of meuprimeiroapp.apk (4)...
```

Substitua “your\_project\_name-unaligned.apk” pelo arquivo apk que está na pasta que vou gerado no build e já está assinado. Substitua “your\_project\_name.apk” pelo nome final do seu apk. Após rodar a linha de comando, esse arquivo final será gerado:

PC > Local Disk (C:) > App				Search
Name	Date modified	Type	Size	
 CordovaApp-release-unsigned.apk	27/04/2015 12:06	APK File	7.291 KB	
 meuprimeiroapp.apk	27/04/2015 12:11	APK File	7.291 KB	
 meu-primeiro-app.keystore	27/04/2015 12:02	KEYSTORE File	3 KB	


Agora basta subir na PlayStore e esperar a aprovação.


Tags: [cordova](#), [android](#), [apk](#)

[Link permanente](#) [2 notas](#)

---

 [joseballester](#) gostou desta postagem

 [vacacateta](#) gostou desta postagem

 [speedysolutions](#) publicou esta postagem

[« Postagem anterior](#)

[Arquivo](#)   [Ask](#)   [RSS](#)

Powered by [Tumblr](#) and designed by [Kcmr](#).