

# Google Play

## Table of Contents

Release Version .....2

Generate Signed Bundle / APK .....3

Data Safety ..... 6

Advertising ID .....7

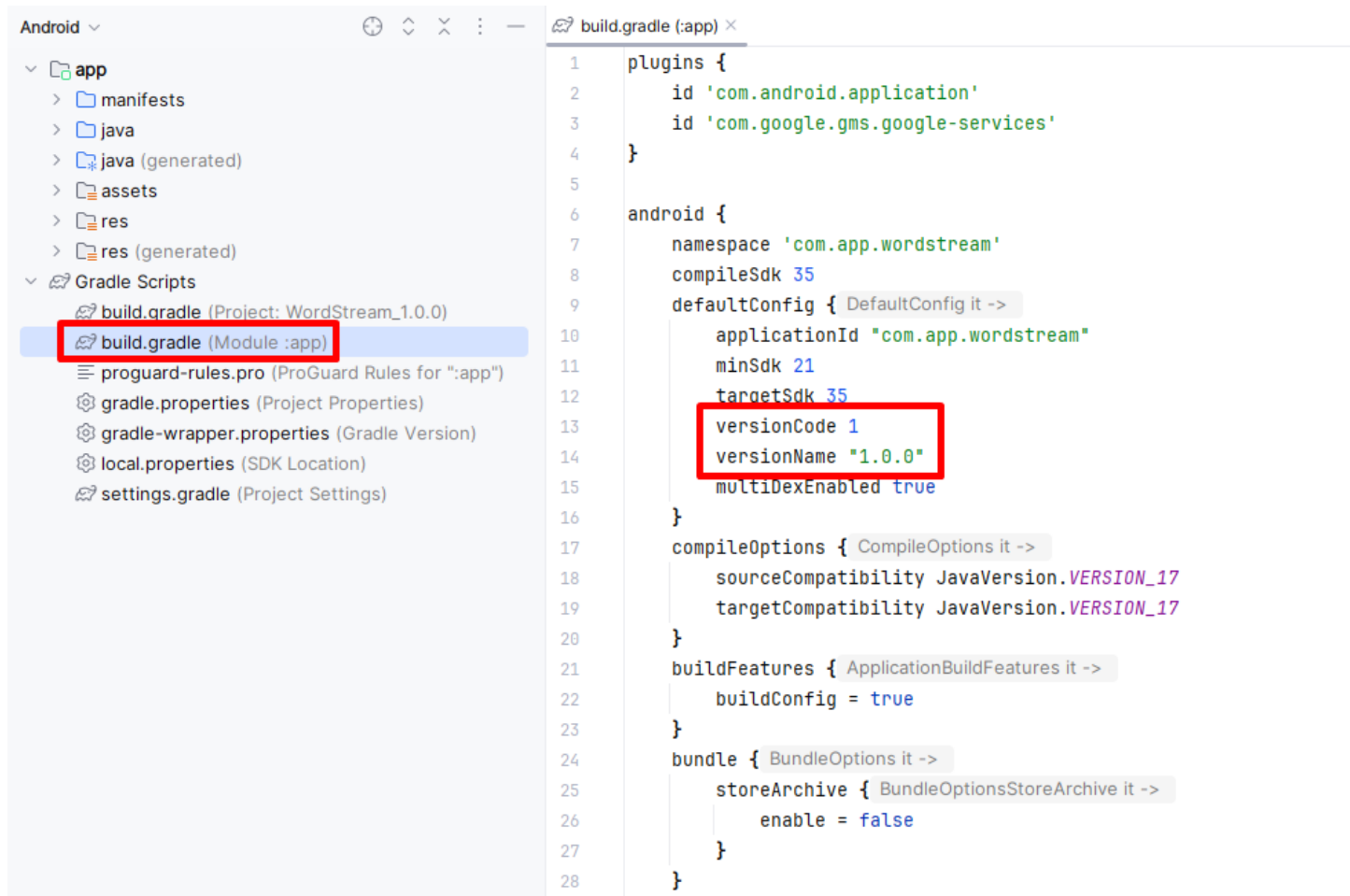
Foreground Service Permissions ..... 8

Cleartext traffic ..... 9

Warnings.....10

# Release Version

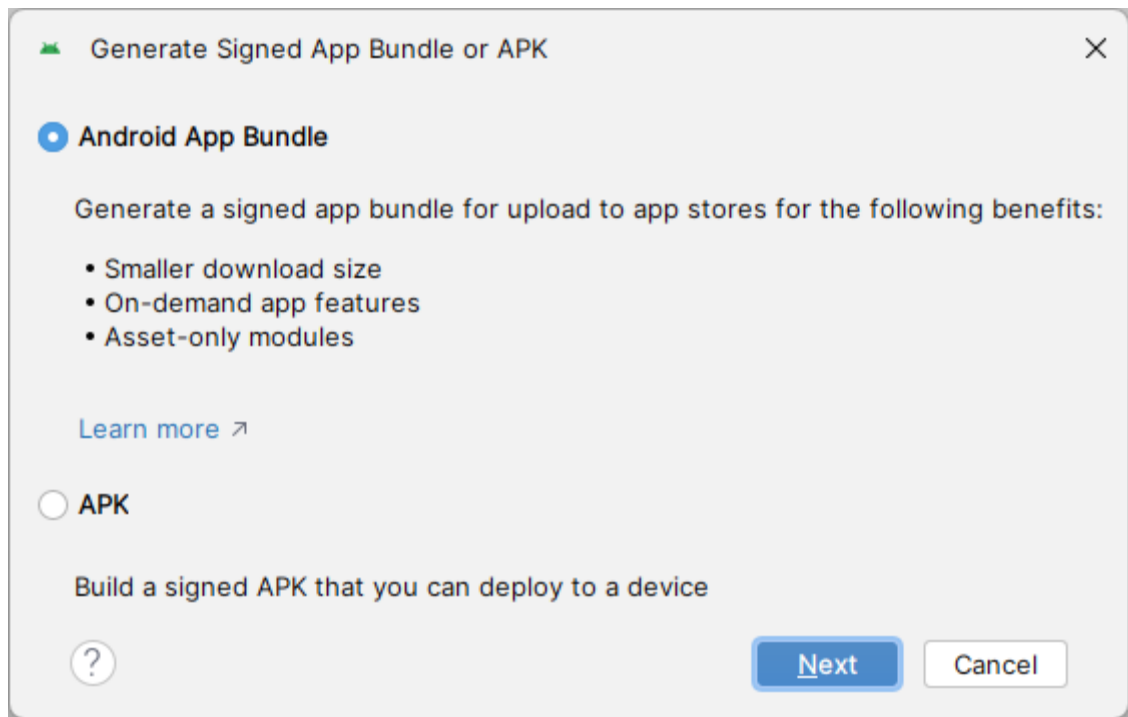
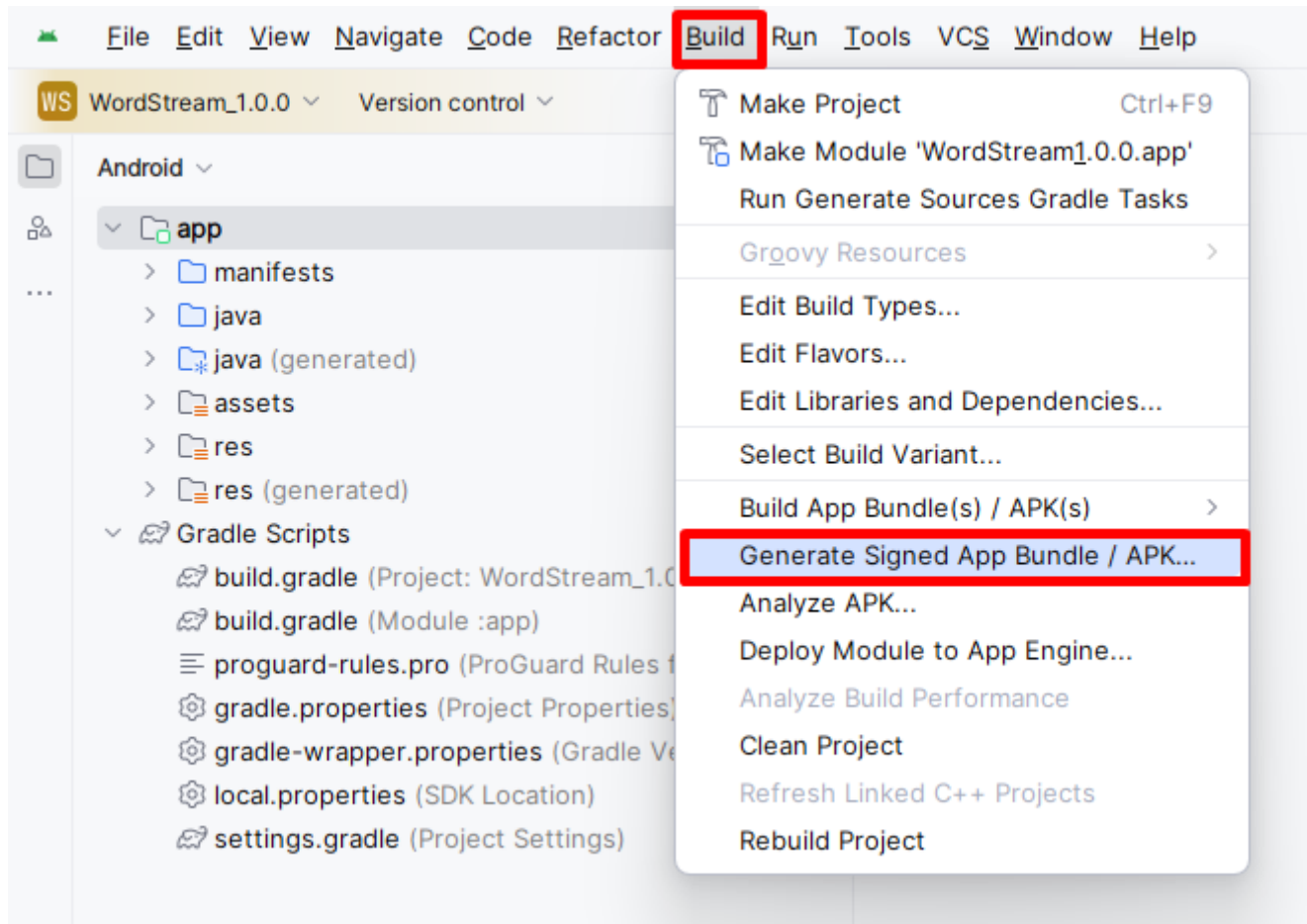
Before release your application, make sure you have change the app **versionCode** and **versionName** in the **build.gradle (Module: app)**. If you just release as new application, just set **versionCode 1** and **versionName 1.0.0** for first release. If you want to update your application, update your app version code and version to higher from previous version.



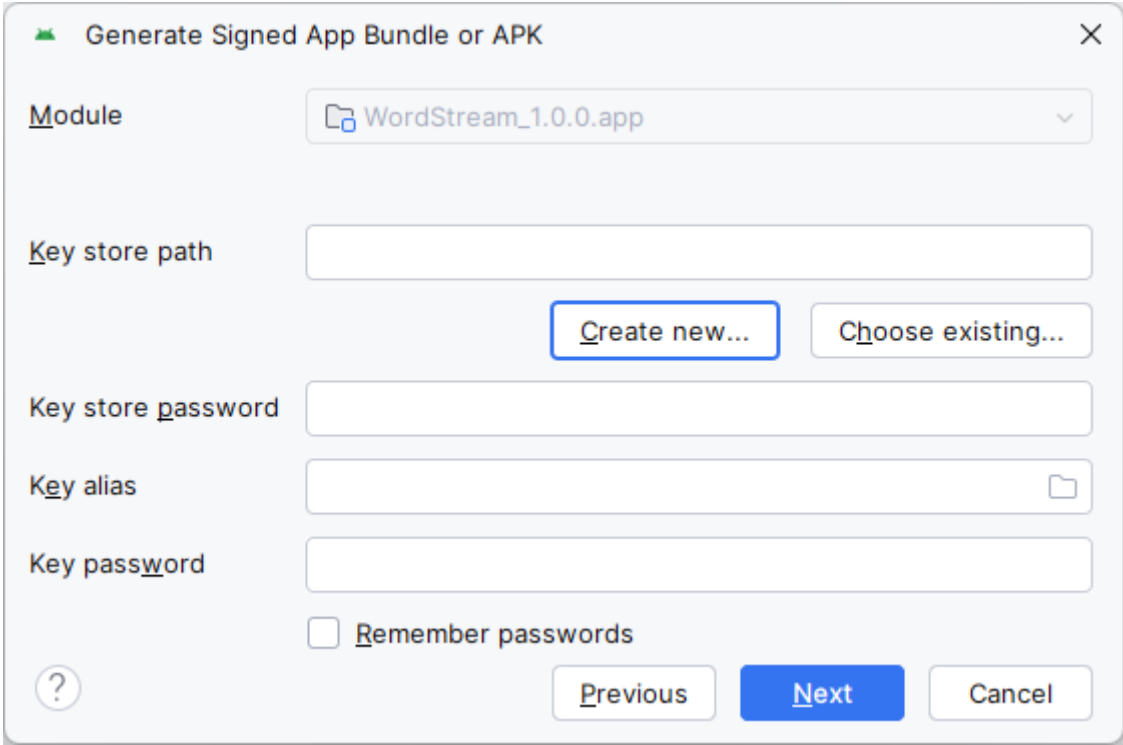
# Generate Signed Bundle / APK

If you have finished reskin the application and want to publish your application on Play Store, to sign your app in release mode in Android Studio, follow these steps :

On the menu bar, click **Build** → **Generate Signed Bundle / APK** → **APK**



On the Generate Signed Bundle or APK Wizard window, click **Create new...** to create your new keystore or If you already have a keystore, select **Choose existing...**



Generate Signed App Bundle or APK

Module: WordStream\_1.0.0.app

Key store path:

Create new... Choose existing...

Key store password:

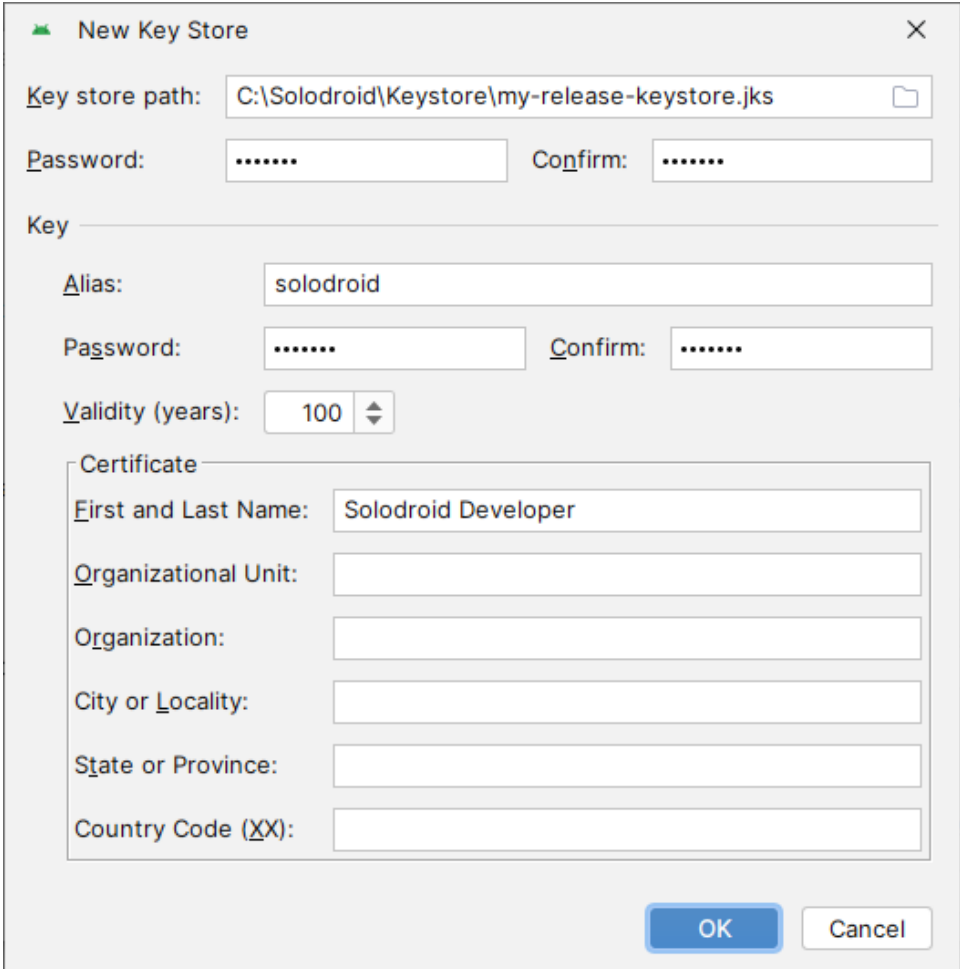
Key alias:

Key password:

☐ Remember passwords

Previous Next Cancel

In the New Key Store window, provide the required information, on the Certificate form you must at least fill in First and Last Name, although other fields are optional, it is better if you fill in all fields, your key must be valid for at least 25 years, so you can Sign application updates with the same key for the life of your application.



New Key Store

Key store path: C:\Solodroid\Keystore\my-release-keystore.jks

Password: ..... Confirm: .....

Key

Alias: solodroid

Password: ..... Confirm: .....

Validity (years): 100

Certificate

First and Last Name: Solodroid Developer

Organizational Unit:

Organization:

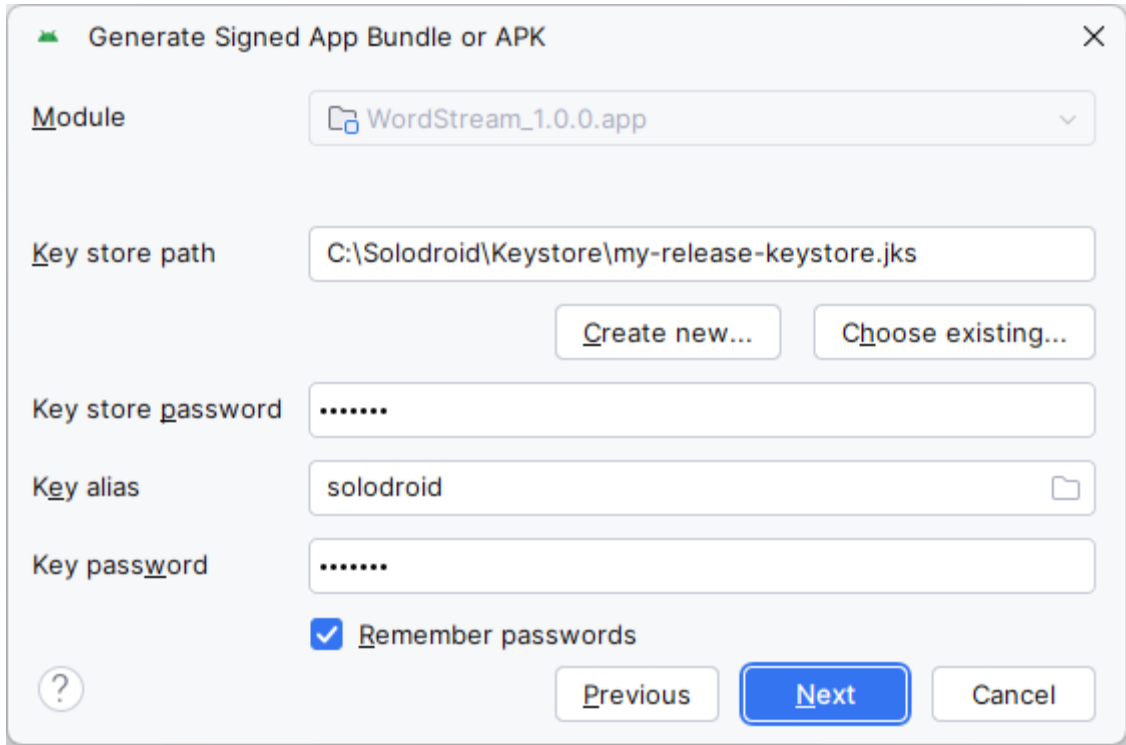
City or Locality:

State or Province:

Country Code (XX):

OK Cancel

On the Generate Signed Bundle or APK Wizard window, select a keystore, a private key, and enter the passwords for both. Then click Next



Generate Signed App Bundle or APK

Module: WordStream\_1.0.0.app

Key store path: C:\Solodroid\Keystore\my-release-keystore.jks

Key store password: .....

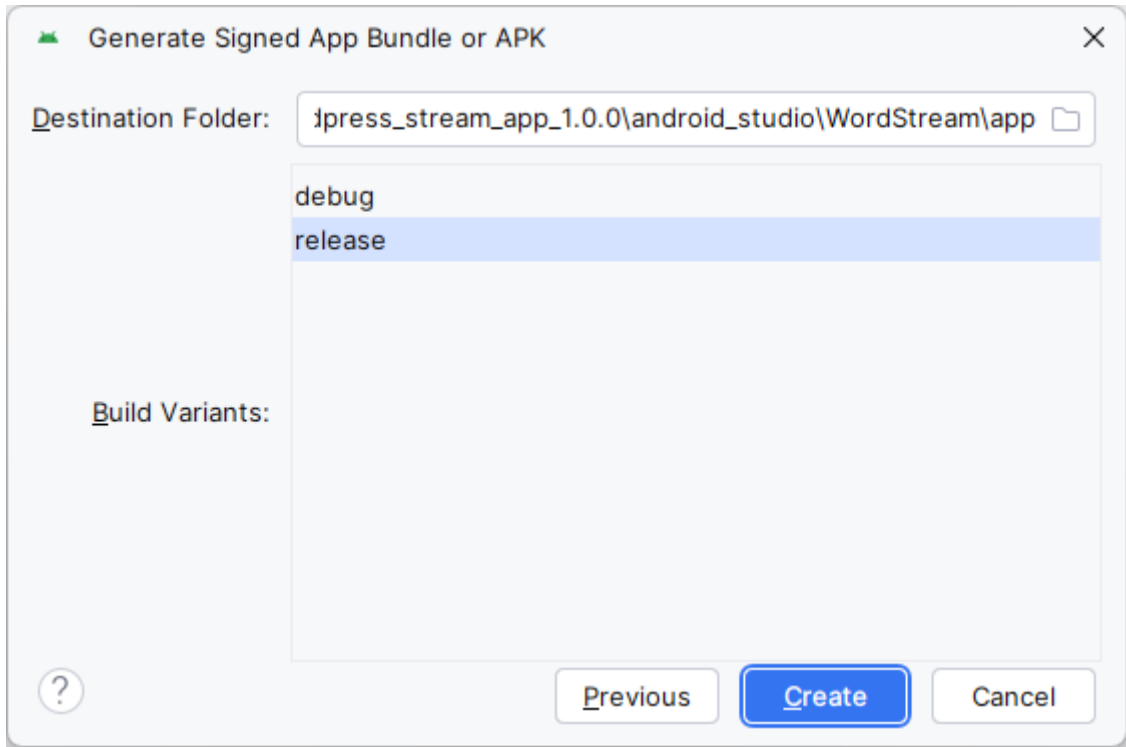
Key alias: solodroid

Key password: .....

☒ Remember passwords

Previous Next Cancel

On the next window, select a destination for the signed Bundle.



Generate Signed App Bundle or APK

Destination Folder: d:\press\_stream\_app\_1.0.0\android\_studio\WordStream\app

Build Variants:

- debug
- release

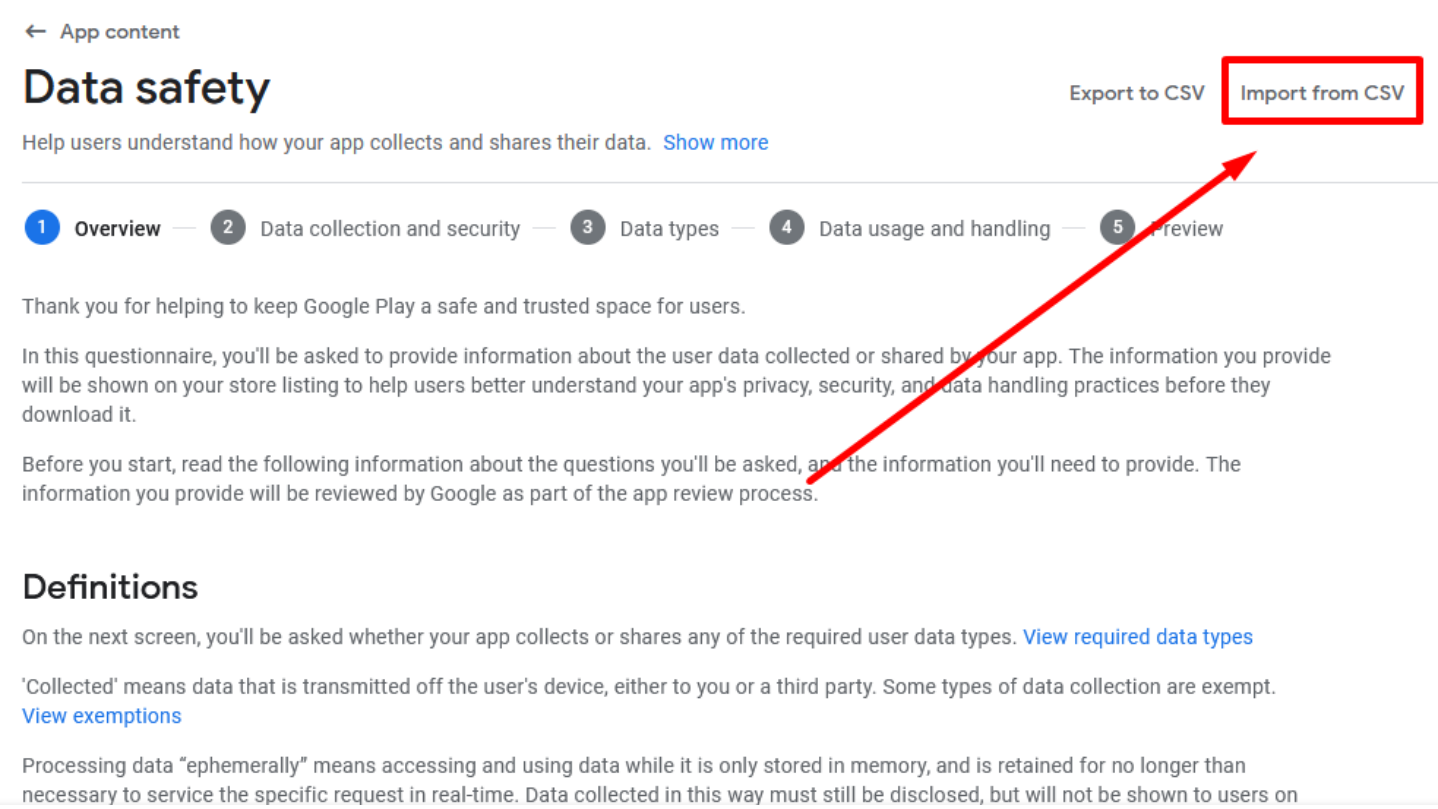
Previous Create Cancel

# Data Safety

Recently, many applications received warnings about data security/data security. Data security will be enforced in April 2022, so all detected apps/games related to data collection need to be updated. Referring to the data security article, We try to describe the population according to code by looking at the following points:

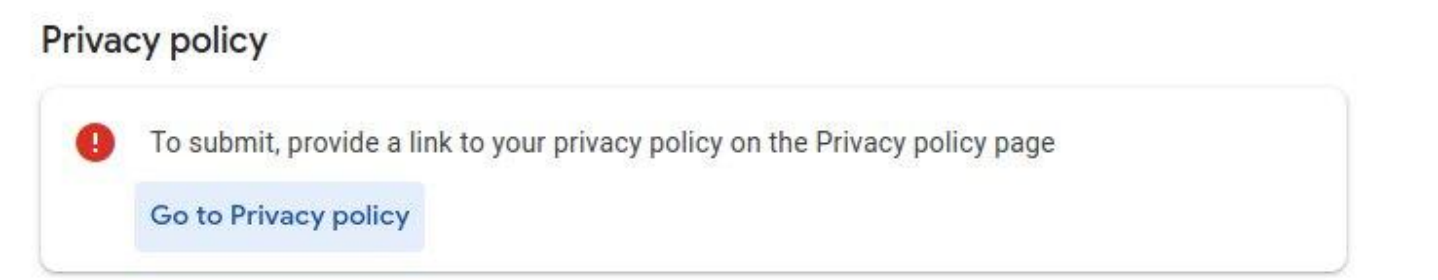
Automatic filling out forms (Import CSV file)

Import a CSV file for faster autofill based on the tutorial : [DOWNLOAD CSV](#)



Fill your privacy policy link

Finally, prepare your privacy policy link. You can use the app's [privacy policy generator](#) to create your privacy policy.



Until the stage of filling out the privacy policy has been completed, but for some source code there may be additional filling in accordance with the permissions used.

# Advertising ID

In addition to filling the Data Safety form, the new Play Store policy is to answer questions or fill in data related to Advertising ID, this app is using ads for monetization purposes, so when you want to publish your app on Google Play, it will detect the use of Advertising ID, this is a common thing because some of the libraries like google play services ads sdk and other related libraries used do integrate it for the needs of displaying ads, also Advertising ID declaration needed for sending push notification and analytics features.

## Advertising ID

### Does your app use advertising ID?

This includes any SDKs that your app imports that use advertising ID

 Your manifest file includes the `com.google.android.gms.permission.AD_ID` permission. This means your app declares the use of advertising ID. Answer 'yes' or remove this permission from your manifest.

☐ No

☒ Yes

When you answer this question, make sure to verify if any third-party SDK code in your app uses advertising ID. If so, you must declare that your app uses it. Some SDKs, such as the Google Mobile Ads SDK (Play Services-ads) may already declare the `com.google.android.gms.permission.AD_ID` permission in the SDK's library manifest. If your app uses these SDKs as dependencies, the `com.google.android.gms.permission.AD_ID` permission from the SDK's library manifest will be merged with your app's main manifest by default, even if you don't explicitly declare the permission in your app's main manifest. [Learn more](#)

Why does your app need to use advertising ID? This includes any SDKs your app imports that use advertising IDs.

Select all that apply

☐ App functionality

Used for features in your app. For example, to enable functionality, or authenticate users.

☒ Analytics

Used to collect data about how users use your app, or how your app performs. For example, to see how many users are using a particular feature, to monitor app health, to diagnose and fix bugs or crashes, or to make future performance improvements.

☐ Developer communications

Used to send news or notifications about you or your app. For example, sending a push notification to inform users about an important security update, or informing users about new features in your app.

☒ Advertising or marketing

Used to display or target ads or marketing communications, or measure ad performance. For example, displaying ads in your app, sending push notifications to promote other products or services, or sharing data with advertising partners.

☐ Fraud prevention, security, and compliance

Used for fraud prevention, security, or compliance with laws. For example, monitoring failed login attempts to identify possible fraudulent activity.

☐ Personalization

Used to customize your app, such as showing recommended content or suggestions. For example, suggesting playlists based on users' listening habits, or delivering local news based on a user's location.

# Foreground Service Permissions

**FOREGROUND\_SERVICE\_MEDIA\_PLAYBACK** is a permission that must be declared in **AndroidManifest.xml** for the needs of streaming media playback on the notification bar, without this permission, the app will immediately crash when trying to play streaming audio, when the app uses this permission, a warning or error will appear in the Google Play Developer Console that requires you to declare **Foreground service permissions**.

Google Play Console

Crashes and ANRs

App size

Monetize

Products

Price experiments

Promo codes

Financial reports

Monetization setup

Policy and programs

Policy status

App content

Teacher Approved

Search Play Console

App content

Let us know about the content of your app. This is to make sure your app complies with Google Play policies. [Learn more](#)

Some sections have errors, or aren't complete

Need attention (1)

Actioned

1 declaration needs attention

Policy declarations that require your attention are shown here. Fix any issues and complete the declarations before the relevant deadlines.

Foreground service permissions

When apps that target Android 14 use a foreground service, developers must declare the appropriate foreground service permission for that specific foreground service type. [Learn more](#)

Why this impacts your app

One or more of your app bundles or APKs includes the Foreground service permissions in its manifest

Start declaration

Google Play Console

Crashes and ANRs

App size

Monetize

Products

Price experiments

Promo codes

Financial reports

Monetization setup

Policy and programs

Policy status

App content

Teacher Approved

Search Play Console

App content

Foreground service permissions

Your app uses the FOREGROUND\_SERVICE\_MEDIA\_PLAYBACK permission. You can only use this permission if your app performs tasks noticeable to the user when they're not directly interacting with your app.

View app bundles and APKs

Learn more

Media playback

What tasks require your app to use the FOREGROUND\_SERVICE\_MEDIA\_PLAYBACK permission?

Media playback

Provide a video demonstrating how your app uses the FOREGROUND\_SERVICE\_MEDIA\_PLAYBACK permission for the tasks you've selected

Video link

Show picture in picture

Other

You must provide a video demonstrating how your app uses the permission, Therefore, you must create video when you test the app, play one of the radio streaming, wait for the stream to play then swipe down in your status bar to show the play bar notification. Publish your video on YouTube then copy your video link and insert it into the Video link form field.



# Cleartext traffic

## Security and trust

### Cleartext traffic allowed for all domains

 Warning • Privacy

Your app's Network Security Configuration allows cleartext traffic for all domains. This could allow eavesdroppers to intercept data sent by your app. If that data is sensitive or user-identifiable it could impact the privacy of your users.

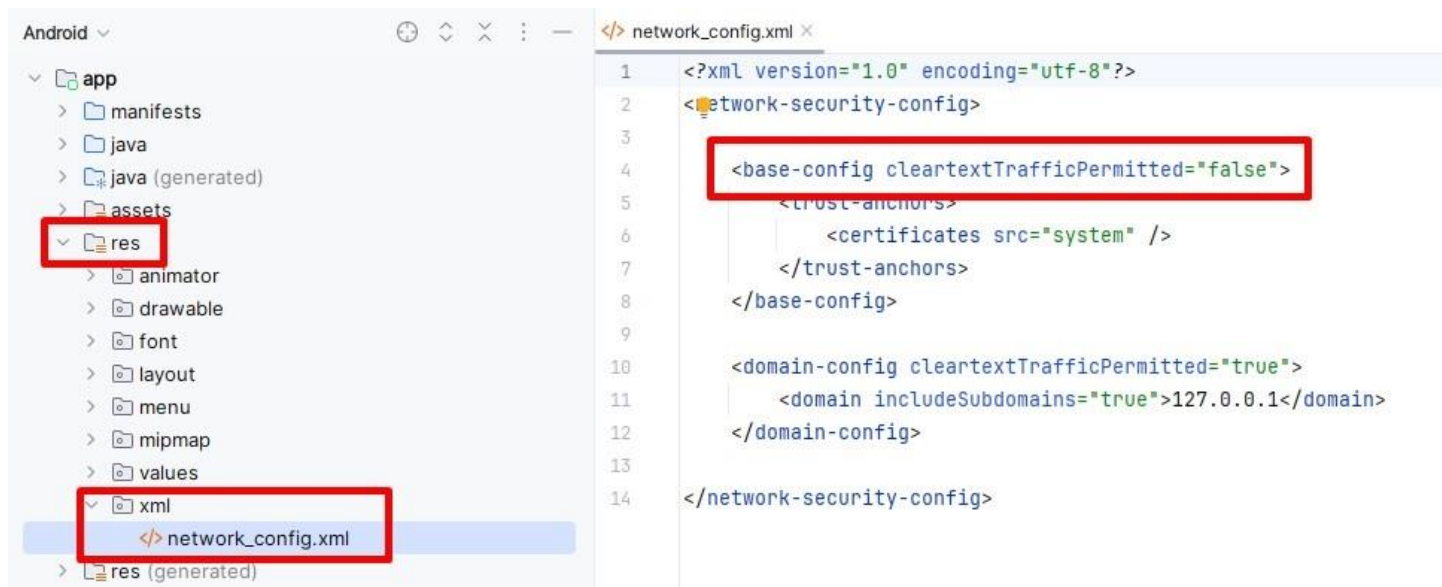
Consider only permitting encrypted traffic by setting the [cleartextTrafficPermitted](#) flag to false, or adding an encrypted policy for specific domains.

[Learn more](#)

The Cleartext traffic configuration is required for the app to run on **http and https protocols**, the cleartext traffic warning is not an error, but a warning which can be ignored.


But, if you want to solve or remove the warning, it's also can be removed, but with consequences and with note that the app will only run on **https protocol**, if you use http protocol, the content will not be able to be loaded from the server to the app.

To remove the warning, in the Android Studio project, open **res/xml/network\_config.xml**, then, in the **base-config cleartextTrafficPermitted** configuration, change the value from **true** to **false**, by using false value, the warnings message will be disappear when you publish the app on Google Play.



# Warnings

When you create an app release, you will get 2 warning messages regarding deobfuscation and native code, but don't worry, they are just warning messages which can be safely ignored.

 2 Warnings  
[Show less](#) ^

2 MESSAGES FOR VERSION CODE 1

 Warning

There is no deobfuscation file associated with this App Bundle. If you use obfuscated code (R8/proguard), uploading a deobfuscation file will make crashes and ANRs easier to analyze and debug. Using R8/proguard can help reduce app size. [Learn More](#)

 Warning

This App Bundle contains native code, and you've not uploaded debug symbols. We recommend you upload a symbol file to make your crashes and ANRs easier to analyze and debug. [Learn More](#)

The first warning about deobfuscation appears because our project does not use the obfuscation method with proguard, but that doesn't matter, your code and information data will remain safe because they are protected with the url encryption method. The second caveat about native code arises because some library implementations use native C# code in their coding, it is common practice and will not impact app performance.

So, in conclusion you don't need to worry, the warning message can be safely ignored and you can publish your app to Google Play safely.