

# Play Integrity Integration

The SDK is using the play integrity to utilize built in android security features. To use those features you have to configure the Play Integrity on the platform, which would include sharing the cloud project number and creating a key.

To be able to use the feature, you need to add the following dependency to your `build.gradle` file:

```
dependencies {  
    // ...  
    implementation 'com.google.android.play:integrity:1.4.0'  
}
```

## Enabling the feature in Google Play Console

Also the feature has to be enabled in the Google Play Console for your project.



### Google Play Integrity API

[Google](#)

Check that interactions are coming from your genuine app running on a genuine Android device.

ENABLE

TRY THIS API [↗](#)

## Using Firebase


Firebase is a convenient way to ensure we have all the data to perform Play Integrity tests.


To retrieve the necessary info you have to go to your firebase console and open the project settings. <https://console.firebase.google.com/> The project number will be under the `General` tab


# Project settings


General Cloud Messaging Integrations Service accounts Data privacy Users and permissions

## Your project

Project name [redacted] 

Project ID  [redacted]

Project number  [redacted]

Web API Key  No Web API Key for this project

## Environment


This setting customizes your project for different stages of the app lifecycle

Environment type Unspecified 

We also need a service account private key to make requests to google on your behalf for retrieving additional information regarding suspicious activity on the device. You can generate the private key from firebase.

# Project settings

General Cloud Messaging Integrations Service accounts Data privacy Users and permissions

[Manage service account permissions](#) 

## Firebase Admin SDK


Legacy credentials

## Database secrets

All service accounts

## [5 service accounts](#)

## Firebase Admin SDK

Your Firebase service account can be used to authenticate multiple Firebase features, such as Database, Storage and Auth, programmatically via the unified Admin SDK. [Learn more](#) 

Firebase service account [redacted]

Admin SDK configuration snippet

☒ Node.js ☐ Java ☐ Python ☐ Go

```
var admin = require("firebase-admin");

var serviceAccount = require("path/to/serviceAccountKey.json");

admin.initializeApp({
  credential: admin.credential.cert(serviceAccount)
});
```



[Generate new private key](#)

Then you'll have to enable Play Integrity from the google console

<https://console.cloud.google.com/marketplace/product/google/playintegrity.googleapis.com> for your selected project.

# Using Google Cloud Console

You can also get all the necessary info from the Google Cloud Console.

<https://console.cloud.google.com>

The project number will be right on the home page with the name of your project.



You're working in [REDACTED]

Project number: [REDACTED]

Project ID: [REDACTED]

[Dashboard](#)

[Recommendations](#)

To generate the key there is some navigation to perform.

1. First go to the IAM & Admin page by clicking this navigation button



You're working in [REDACTED]

Project number: [REDACTED]

Project ID: [REDACTED]

[Dashboard](#)

[Recommendations](#)

2. Then select **Service Accounts** from the navigation bar.

DETAILS	PERMISSIONS	<b>KEYS</b>	METRICS	LOGS
---------	-------------	-------------	---------	------

## Keys



Service account keys could pose a security risk if compromised. We recommend you avoid downloading service a [about the best way to authenticate service accounts on Google Cloud](#).



Google automatically disables service account keys detected in public repositories. You can customize this behav [Learn more](#)

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies](#).

[Learn more about setting organization policies for service accounts](#)

ADD KEY ▾

Create new key

Upload existing key

Key	Creation date	Expiration date	
11c69fa9e3016a13bc637103e3052c458106dec7	Aug 5, 2024	Jan 1, 10000	

3. Inside your service account, go to the Keys tab and create a new Key

## Keys



Service account keys could pose a security risk if compromised. We recommend you avoid downloading service a [about the best way to authenticate service accounts on Google Cloud](#).



Google automatically disables service account keys detected in public repositories. You can customize this behav [Learn more](#)

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies](#).  
[Learn more about setting organization policies for service accounts](#)

ADD KEY ▾

Create new key

Upload existing key

Key	Creation date	Expiration date	
11c69fa9e3016a13bc637103e3052c458106dec7	Aug 5, 2024	Jan 1, 10000	

4.Select JSON as the key type

## Create private key for [REDACTED]

Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

### Key type



JSON

Recommended



P12

For backward compatibility with code using the P12 format

CANCEL

CREATE

