

cant download feet app in google play says login



Cant download feet app in google play says login.

Completing the CAPTCHA proves you are a human and gives you temporary access to the web property.

What can I do to prevent this in the future?

If you are on a personal connection, like at home, you can run an anti-virus scan on your device to make sure it is not infected with malware.

If you are at an office or shared network, you can ask the network administrator to run a scan across the network looking for misconfigured or infected devices.

Another way to prevent getting this page in the future is to use Privacy Pass. You may need to download version 2.0 now from the Chrome Web Store.

Cloudflare Ray ID: 669a374ae8a2c442 • Your IP : 188.246.226.140 • Performance & security by Cloudflare.

Setting Up Google Play Games Services.

This document covers how to use the Google Play Console to set up Google Play games services for your Android game. The Google Play Console provides a centralized place for you to manage game services and configure metadata for authorizing and authenticating your game.

To add your game to the Google Play Console, follow these general steps:

Create a game project for your game and specify details such as the name and description of the game. Create and link the necessary credentials to authorize and authenticate your game to Google Play Games Services.

These steps are covered in more detail in the sections below.

Step 1. Sign in to the Google Play Console.

To sign in, go to the Google Play Console. If you haven't registered for the Google Play Console before, you will be prompted to do so.

Step 2. Add your game to the Google Play Console.

To add your game, follow these steps:

Create a game in Play Console if you haven't already. See here for more details. Navigate to Grow > Play Games Services > Setup and management > Configuration .

Specify whether your game already uses Google APIs (such as Firebase). Choosing the correct option is very important.

If this is a game you are creating from scratch, or you have never set up Google APIs for it previously, choose No, my game doesn't use Google APIs . Enter your game's name and then click Create . If this is a game for which you have already set up one or more Google APIs, choose Yes, my game already uses Google APIs . You will see a list of projects from the Google Cloud Console for which you are listed as a developer. Select your project from the list and then click Use . If you want to use an existing Play Games Services project, choose Use an existing Play Games Services project . You will see a list of existing Play Games Services projects for your account. Select the correct game project from the list and then click Use . This is not a typical choice, but you might do this if you are creating a new game in Play Console to change the package name, or if you have free and paid versions of your game with different package names.

A Play Games Services game project is created, and a corresponding entry is created for you in the Google Cloud Console.

In the Properties section, you can select Edit Properties and add the description, category, and graphic assets for your game.

Only the display name is required for testing. The other fields must be filled out before you can publish your game. The display name and description for your game should match what you have set up for your game's Play Store listing. For guidelines on creating the graphic assets, see the Google Play for Developers guide and the Google Play Featured-Image Guidelines.

Step 3. Generate an OAuth 2.0 client ID.

Your game must have an OAuth 2.0 client ID in order to be authenticated and authorized to call the Google Play games services. To set up a credential for Play Games Services, which is the association between a client ID and your game, use Google Cloud Platform to create the client ID. Then, use Google Play Console to add a credential, linking the client ID to your game.

For more detailed instructions, see the following steps:

a. Configure the OAuth consent screen.

You may have already configured the OAuth consent screen, but if you haven't then the Credentials section will display a message notifying you that you have not done so.

Click **Configure OAuth consent screen** . This opens a dialog with further instructions and a deep link to the Google Cloud Platform. Note the required information described in step 2 of the instructions in the dialog.

If you have completed the setup of the OAuth consent screen, click **Done** . Google Play Console refreshes automatically, and if configuration was successful you will be able to create a credential:

b. Create a credential.

In order to authorize your game to communicate with Google Play games services, you must create a credential with an authorized OAuth2 client ID.

In the **Credentials** section, click **Add credential** .

In the wizard, choose whether you want to create an Android credential (if your game APK will authenticate the user and use PGS APIs) or a game server credential (if your game server will use PGS APIs). Follow the instructions specific to your desired credential type.

Android.

Set up credential details.

Ensure that the name in the **Name** field matches the name of your game. Choose whether to enable **Anti-Piracy**.

Set up authorization.

Next, choose an OAuth client ID to use for this game project. If you already have OAuth2 client IDs, you can choose one. However, you will usually create a new one. Click **Create OAuth client** . This opens a dialog with deep links and instructions for creating an OAuth Client ID in Google Cloud Platform.

Select **Android** as the application type. Enter your game's name in the **Name** field. Enter your Android application's package name in the **Package name** field.

Open a terminal and run the **Keytool** utility to get the SHA1 fingerprints of the release and debug certificates.

To get the release certificate fingerprint, run the following command:

```
keytool -list -keystore <path-to-production-keystore> -v.
```

To get the debug certificate fingerprint, run the following command:

```
keytool -list -keystore <path-to-debug-keystore> -v.
```

The **keytool** utility prompts you to enter a password for the keystore. The default password for the debug keystore is **android** . The **keytool** then prints the fingerprint to the terminal.

Paste the SHA1 fingerprint into the **Signing certificate fingerprint (SHA1)** field.

Click **Create** .

For more information about OAuth 2.0 on Android, see [Authenticating to OAuth2 Services](#).

After you click **Done** in the dialog, the available Client IDs will refresh. Choose the credential you created from the drop-down menu and then click **Save Changes** . This creates the credential as a draft, enabling you to authenticate to Play Games Services in your game.

You may want to create two credentials: one with the release certificate fingerprint, and one with the debug certificate fingerprint. Make sure to use the same package name for both. This allows Google Play Games Services to recognize calls from your linked APKs that are signed with either certificate. For more information about certificate signing for Android, see [Sign your app](#).

Game server.

Set up credential details.

Ensure that the name in the **Name** field matches the name of your game.

Set up authorization.

Next, choose an OAuth client ID to use for this game project. If you already have OAuth2 client IDs, you can choose one. However, you will usually create a new one. Click **Create OAuth client** . This opens a dialog with deep links and instructions for creating an OAuth Client ID in Google Cloud Platform.

Select **Web application** as the application type. Enter your game's name in the **Name** field. Click **Create** .

For more information about OAuth 2.0 on Android, see [Authenticating to OAuth2 Services](#).

After you click Done in the dialog, the available Client IDs will refresh. Choose the credential you created from the drop-down menu and then click Save Changes . This creates the credential as a draft, enabling you to authenticate to Play Games Services from your game server. For more information about using Play Games Services with your game server, see [Enabling Server-Side Access to Google Play Games Services](#).

Avoiding common setup problems.

To avoid common setup mistakes, make sure to follow these recommendations when setting up your game to use Google Play games services.

1. Set up your game with the Google Play Console If you created an OAuth 2.0 client ID for your app in the Google Cloud Console, Google Play games services will not know about the association between the game's achievement and leaderboards and the client ID. To create this association, you must create a credential using the OAuth 2.0 client ID as described in [Create a credential](#).
2. Use the correct application ID in Android The application ID is a required string resource that you must reference in your Android manifest. The application ID string consists only of the digits (typically 12 or more) at the beginning of the client ID provided by the Google Play Console. The application ID can be found at the top of the Configuration page and is labeled as Project ID below the name of your game.
3. Sign your APK with the correct certificate When linking your Android app to your game in the Google Play Console, you must use exactly the same package name and certificate fingerprint that you used to publish your app. If there is a mismatch, calls to Google Play games services will fail. You should create two client IDs, one with the release certificate fingerprint and another with the debug certificate fingerprint, and use the same package name for both. To learn more about how to specify the signing certificate in the Google Play Console, see [Signing Your Applications](#).
4. When developing for Android, include the Play Games SDK as a library project, not as a standalone JAR Make sure that the Google Play services SDK is referenced as a library project in your Android project, otherwise this could lead to errors when your app is unable to find Google Play services resources. To learn how to set up your Android project to use Google Play services, see [Setting Up Google Play Services](#).
5. Sign in with a tester account during development If you have not published your game setting changes in the Google Play Console, you might encounter errors during testing if you are not signed in with a whitelisted tester account. You should always enable your Google Play Console publisher account for testing. To learn how to manage tester accounts, see [Enabling accounts for testing](#).
6. At release, publish the Play Games Services settings first before you publish your game Developers might accidentally publish their app without publishing the corresponding Google Play games services settings for their app. This might cause players who are signing in with non-tester accounts to encounter errors since the app cannot reference the correct game settings. When releasing your game, remember to first publish your game settings by using the Publish Game option in the Google Play Console. To learn how to publish your changes, see [Publishing your game changes](#).

For additional tips, refer to these related resources:

- Video that describes common Google Play games services setup pitfalls and scenarios.
- Developer documentation that describes how to troubleshoot issues while developing Android games.

Next steps.

Before proceeding further, you should add test accounts to your game as described in [Enabling accounts for testing](#). Users with authorized test accounts will have access to your unpublished Play Games Services game project and can test that your configured Google Play games services are working correctly.

Once you have completed the initial setup tasks described above, you can proceed to configure features for your game, such as leaderboards and achievements. To learn more, see [Enable features](#).

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 4.0 License, and code samples are licensed under the Apache 2.0 License. For details, see the [Google Developers Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Unable to download an app from Google Play Store, this is how you can fix it.

Not being able to download an app or game from the Google Play Store or the download screen shows some random error message the moment we hit that download button? This is probably the most common problem faced by Android users when they find an exciting app or a game on the Google Play Store and get stuck with the 'downloading' or 'pending' message below the download bar. At times even after waiting for several minutes or probably hours the download screen keeps on showing the same message. Pretty annoying, right? Here are a few methods that can help users to resolve such issues related to the Google Play Store.

First things first Before talking about the solution, there are a few things that users need to check. Google Play Store downloads only one app at a time, so in case the screen show pending check is there any other app being downloaded or updated at the moment. If yes, either wait for the download to finish or manually stop the download by heading to the My apps.

Apart from this, users also need to ensure that their internet connectivity is working properly. One more thing, Google Play Store download apps and games with a bigger file size over Wi-Fi. So, make sure to uncheck the 'Download over Wi-Fi' option after pressing the download button.

Now, in case everything seems fine, the internet is working, you have enough space on your phone and still unable to download, then here's what one should do.

The error is commonly known as 'Error-20' and to resolve the issue users need to clear the cache and re-sync their Google Account and here's

how to do it.

How to fix "Google Play authentication is required" error.

The "Google Play authentication is required" error is a surprisingly common problem on Android devices. It makes you realize just how much you rely on Google products for your smartphone activity, especially the Play Store. The good news is that it's a super easy problem to fix. Let us show you how to fix this Google Play authentication error in just a few simple steps.

There are several different ways to go about solving this problem, from a simple data clearing to the more drastic factory reset as a last resort.

Uninstall Play Store updates.

Sometimes the problem is not with your credentials but the Play Store itself. Go to the Play Store in Settings > Applications and hit Uninstall updates (usually found by tapping the 3 dots menu in the top right). This will take you back to the original version of the Play Store that was installed on your device.

Then, all you need to do to fix the error is install the latest version of the Google Play Store and sign in once again.

Simply remove your Google account.

The error may simply be a sign-in issue, which sometimes occurs when the Play Store is updated. The first trick is to go into your phone's main Settings menu and then Accounts and simply remove the Google account you have registered (the one that's getting the "authentication is required" message).

Once you've done this, you can re-add the account and it should work just fine. However, you may need to do this step along with the second step below.

Try a fresh sync.

It's possible that things aren't syncing for some reason, and just need a little push. Go to your Settings, tap Accounts, then Google, your email address, and Sync all. Then you can go back to the Play Store and see if the authentication message is gone.

Clear data from the Google Play Store.

To clear data from the Play Store, you need to go into Settings and then Applications and find the Google Play Store. Tap it and then select Clear data. (You can try just clearing the cache first, if you like, but clearing data will clear the cache as well).

If the previous step (the fresh sync) didn't work for you, you may find that it does if you repeat it after carrying out this step.

Download an app using your mobile browser.

This workaround bypasses the malfunctioning Play Store app completely. Go to your web browser and navigate to the Google Play Store website. Try downloading an app directly from there rather than through the Google Play Store app. Just sign into your Google account through your browser and then install the chosen app.

If you're still having problems after trying all these steps, run through them again in this order, rebooting your phone between the steps: remove Google account, reboot, uninstall updates, install new Play Store, reboot, add account, and so on. Just keep trying until the error goes away.

The last resort: factory reset.

One user tried everything and still couldn't get rid of the error message. But, the last resort worked: a factory reset. Here's how to do it.

Fix: Can't login into Google Play Games in any game.

Even in the growing, yet smaller mobile gaming industry, the element of social gaming is gaining a lot of traction. To help you keep a track of your scoreboard, connect with other gamers and compete with friends online, Android offers the Google Play Games app.

Similar to how Steam works on your computer, Google Play Games acts as your gaming profile to not only access games, but compile your gaming history as well. But just like any other app, Google Play Games can malfunction, causing you to be cut-off from your favorite games altogether.

One of the issues with the Google Play games is that you are unable to sign into any game on your Android device. That is a weird issue, but we do have solutions to fix this one, so that you are able to log into your games easily.

Related:

How to fix Google Play games cannot log in issue Step 1: Clear app cache and app data Step 2: Reboot and reinstall Step 3: Complete your Google+ profile.

How to fix Google Play games cannot log in issue.

Step 1: Clear app cache and app data.

The first step to fixing the issue with Google Play Games is by clearing out the app cache, which could've left behind some broken files that you could be the culprit behind the issue.

From the home screen of your device, navigate to Settings – Apps . Scroll down to find the Google Play Games app and open the Storage Here you will be able to see the Clear Data and Clear Cache options, so use both of them to refresh the app. Follow the steps above to find the Google Play Store app and use the Clear Data and Clear Cache options once again.

Just in case if you see that the Google Play Games app is disabled, make sure to press the Enable button to activate it.

Step 2: Reboot and reinstall.

The simplest fix is sometimes the only one you need when it comes to fixing broken apps, so here's what you need to do.

Press and hold the Power button to reboot the Android device. Wait till the device has restarted and is back to Android OS. Head over to the Google Play Store, and uninstall the Google Play Games app and reboot your device again. Now go back to the Google Play Store and download the Google Play Games app and install it.

Check to see if you are now about to sign into your Google Play Games account, and head over to the next step if you are still facing issues.

Step 3: Complete your Google+ profile.

As absurd as it may seem, users have reportedly been able to fix the Google Play Games signing in error by completing their Google+ profiles.

Using the Chrome browser app, head over to your Gmail and use the tabs to open your Google+ A pop-up window should take you to your Profile , asking you to add more details and complete your profile. Add the remaining details and press the Complete button, and enter your Google account email address that you're trying to use the Google Play Games app.

Once the process is completed, you will most likely be able to access the Google Play Games service in other apps as well.

Be sure to let us know in the comments section about which steps worked out for you.

Husain.

Sharing a love-hate relationship with Biotechnology in college, Husain moved on to his true calling of being a tech geek through and through. While briefly being persuaded by the iPhone, he made it back to the Android camp with the OnePlus One and has been a loyal comrade since.