Testing Backup and Restore

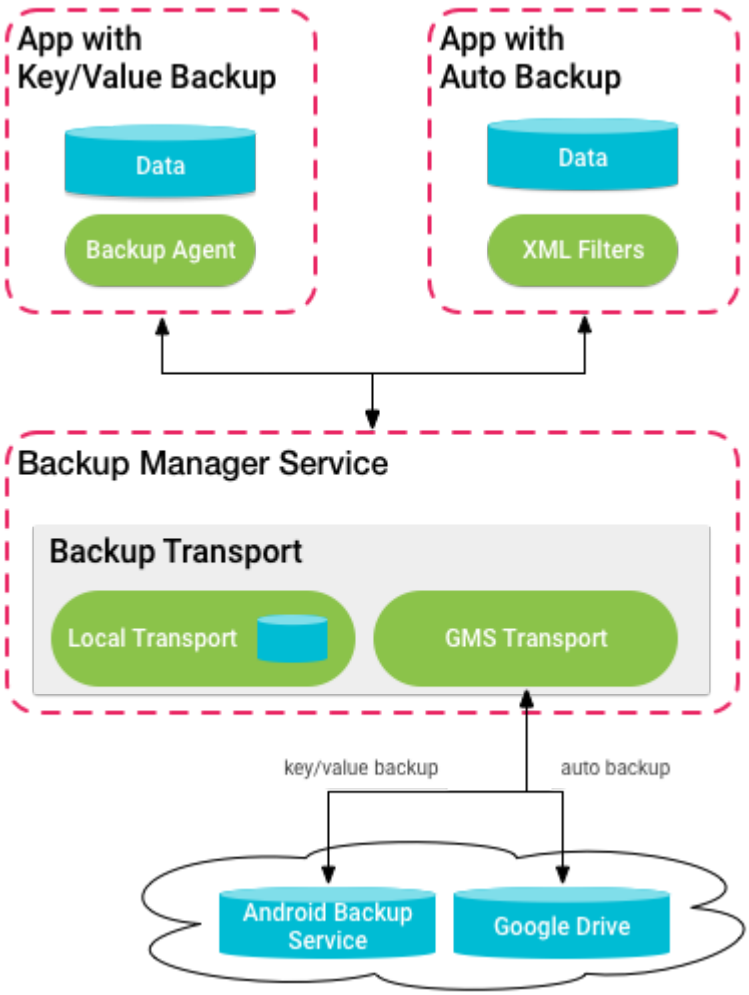
In this document

This page shows you how to manually trigger Auto Backup (https://developer.android.com/guide/topics/data/autobackup.html), Key/Value Backup (https://developer.android.com/guide/topics/data/keyvaluebackup.html), and restore operations to ensure your app saves and restores data properly.

How backup works

The section describes various pieces in the Android backup framework and how they interact with apps that support Auto Backup and Key/Value Backup. During the app development phase, most of the inner working of the framework were abstracted away, so you didn't need to know this information. However, during the testing phase, an understanding of these concepts is important.

The following diagram illustrates how data flows during backup and restore:



The *Backup Manager Service* is an Android system service which orchestrates and initiates backup and restore operations. The service is accessible through the `BackupManager` (https://developer.android.com/reference/android/app/backup/BackupManager.html) API. During a backup operation, the service queries your app for backup data, then hands it to the *backup transport*, which then archives the data. During a restore operation, the backup manager service retrieves the backup data from the backup transport and restores the data to the device.

Backup Transports are Android components that are responsible for storing and retrieving backups. An Android device can have zero or more backup transports, but only one of those transports can be marked active. The available backup transports may differ from device to device (due to customizations by device manufacturers and service providers), but most Google Play enabled devices ship with the following transports:

- **Google Transport**(default) - the active backup transport on most devices, part of Google Mobile Services (https://www.android.com/gms/). This documentation assumes that users are using the Google transport. This transport stores Auto Backup data in a private folder in the user's Google Drive account. Key/Value Backup data is stored in the Android Backup Service (https://developer.android.com/google/backup/index.html).
- **Local Transport** - stores backup data locally on the device. This transport is typically used for development/debugging purposes and is not useful in the real world.

If a device does not have any backup transports, then the data cannot be backed up. Your app is not adversely affected.

> **Caution:** Because the backup transport can differ from device to device, Android cannot guarantee the security of your data while using backup. Be cautious about using backup to store sensitive data, such as usernames and passwords.

Prerequisites

You need to know a bit about the following tools:

- adb (https://developer.android.com/studio/command-line/adb.html) - to run commands on the device or emulator
- bmgr (https://developer.android.com/studio/command-line/bmgr.html) - to perform various backup and restore operations
- logcat (https://developer.android.com/studio/command-line/logcat.html) - to see the output of backup and restore operations.

Preparing your device or emulator

Prepare your device or emulator for backup testing by working through the following checklist:

- For Auto Backup, check that you are using a device or emulator running Android 6.0 (API level 23) or higher.
- For Key/Value Backup, check that you are using a device or emulator running Android 2.2 (API level 8) or higher.
- Check that backup and restore is enabled on the device or emulator and that a Google account has been added. There are two ways to check:
  - On the device, go to **Settings -> Backup & Restore**.
  - From adb shell, run `bmgr enabled`

  On physical devices, backup and restore is typically enabled during the initial setup wizard. Emulators do not run the setup wizard, so don't forget to enable backup and specify a backup account in device settings.
- Make sure the Google Backup Transport is available and active by running the command:

  ```
  $ adb shell bmgr list transports
  ```

  Then, check the console for the following output:

  ```
  android/com.android.internal.backup.LocalTransport
  * com.google.android.gms/.backup.BackupTransportService
  ```

Physical devices without Google Play and emulators without Google APIs might not include the Google Backup Transport. This article assumes you are using the Google Backup Transport. You can test backup and restore with other backup transports, but the procedure and output can differ.

Testing backup

To initiate a backup of your app, run the following command:

```
$ adb shell bmgr backupnow <PACKAGE>
```

The `backupnow` command is available on devices and emulator running Android 7.0 or later. It runs either a Key/Value Backup or Auto Backup depending on the package's manifest declarations. Check logcat to see the output of the backup procedure. For example:

```
D/BackupManagerService: fullTransportBackup()
I/GmsBackupTransport: Attempt to do full backup on <PACKAGE>

---- or ----

V/BackupManagerService: Scheduling immediate backup pass
D/PerformBackupTask: starting key/value Backup of BackupRequest{pkg=<PACKAGE>}
```

If the `backupnow` command is not available on your device, complete the steps below for either Auto Backups or Key/Value Backups.

For Auto Backups, complete the following steps:

1. Run the following command:

   ```
   $ adb shell bmgr backup @pm@ && adb shell bmgr run
   ```

2. Wait until the command in the previous step finishes by monitoring `adb logcat` for the following output:

   ```
   I/BackupManagerService: K/V backup pass finished.
   ```

3. Run the following command to perform a full backup:

   ```
   $ adb shell bmgr fullbackup <PACKAGE>
   ```

> **Note:** The `fullbackup` command forces your app to perform a full backup, even if your app implements key/value backup. The system ignores your app's backup configuration and acts as if the `android:fullBackupOnly` (https://developer.android.com/guide/topics/manifest/application-element.html#fullBackupOnly) attribute were set to true.

For Key/Value Backups, schedule and run your backup with the following steps:

1. If your app hasn't called `BackupManager.dataChanged()` (https://developer.android.com/reference/android/app/backup/BackupManager.html#dataChanged()) since the last backup, you can include your app in the backup operation for testing purposes by running the following command:

   ```
   $ adb shell bmgr backup <PACKAGE>
   ```

2. You can then trigger a backup by running the following command:

   ```
   $ adb shell bmgr run
   ```

`bmgr backup` adds your app to the Backup Manager's queue. `bmgr run` initiates the backup operation, which forces the Backup Manager to perform all backup requests that are in its queue.

When testing Key/Value Backups, you must verify that every preference change schedules a backup. You can verify that a backup is scheduled using one of the following methods:

- Run `adb shell dumpsys backup` and check that your app is listed in the output of the command under `Pending key/value backup`.
- Log a message when you schedule a backup. You can then run `adb logcat`, and check the output of the command to verify that a backup was scheduled.

Testing restore

To manually initiate a restore, run the following command:

```
$ adb shell bmgr restore <TOKEN> <PACKAGE>
```

Warning: This action stops your app and wipes its data before performing the restore operation.

To look up backup tokens, run `adb shell dumpsys backup`. The token is the hexidecimal string following the labels `Ancestral:` and `Current:`. The *ancestral* token refers to the backup dataset that was used to restore the device when it was initially setup (with the device-setup wizard). The *current* token refers to the device's current backup dataset (the dataset that the device is currently sending its backup data to).

Then, check logcat to see the output of the restore procedure. For example:

```
V/BackupManagerService: beginRestoreSession: pkg=<PACKAGE> transport=null
V/RestoreSession: restorePackage pkg=<PACKAGE> token=368abb4465c5c683
...
I/BackupManagerService: Restore complete.
```

You also can test automatic restore for your app by uninstalling and reinstalling your app either with `adb` or through the Google Play Store app.

Troubleshooting

This section helps you troubleshoot some common issues.

Transport quota exceeded

If you see the following messages in logcat:

```
I/PFTBT: Transport rejected backup of <PACKAGE>, skipping

--- or ---

I/PFTBT: Transport quota exceeded for package: <PACKAGE>
```

Your app has exceeded the quota. Reduce the amount of backup data and try again. For example, verify that you are caching data only in the cache directory of your app. The cache directory isn't included in backups.

Full backup not possible

If you see the following message in logcat:

```
I/BackupManagerService: Full backup not currently possible -- key/value backup not yet run?
```

The fullbackup operation failed because no Key/Value Backup operation has yet occurred on the device. Trigger a Key/Value Backup with the command `bmgr run` and then try again.

Timeout waiting for agent

If you see the following message in logcat:

```
12-05 18:59:02.033  1910  2251 D BackupManagerService:
    awaiting agent for ApplicationInfo{5c7cde0 com.your.app.package}
12-05 18:59:12.117  1910  2251 W BackupManagerService:
    Timeout waiting for agent ApplicationInfo{5c7cde0 com.your.app.package}
12-05 18:59:12.117  1910  2251 W BackupManagerService:
    Can't find backup agent for com.your.app.package
```

Your app is taking more than 10 seconds to launch for backup. Notice the timestamp difference in the log output. This error typically occurs when your app makes use of a multidex configuration without ProGuard, and may be caused by Instant Run (https://developer.android.com/studio/run/index.html#instant-run) in Android Studio. To workaround the error, use ProGuard and don't use Instant Run.

Uninitialized backup account

If you see the following messages in logcat:

```
01-31 14:32:45.698 17280 17292 I Backup: [GmsBackupTransport] Try to backup for an uninitiali
01-31 14:32:45.699  1043 18255 W PFTBT: Transport failed; aborting backup: -1001
01-31 14:32:45.699  1043 18255 I PFTBT: Full backup completed with status: -1000
```

The backup was aborted because the backup dataset was not initialized. Run the backup manager with the command `adb shell bmgr run` and then try to perform the backup again.