

React Native — APK Build Commands

A quick-reference cheat sheet for building, signing, and installing Android APKs/AABs from a React Native project.

1. Install Dependencies & Start Metro

Run these from the root of your project before doing anything else.

```
npm install
# or
yarn install

npx react-native start
```

2. Build a Debug APK

Fastest way to get an installable APK for testing. No signing setup required (uses the default debug keystore).

```
cd android
./gradlew assembleDebug
```

Output file:

```
android/app/build/outputs/apk/debug/app-debug.apk
```

Or run and install directly on a connected device/emulator in one step:

```
npx react-native run-android
```

3. Build a Signed Release APK

Step 1 — Generate a keystore (only once per app)

```
keytool -genkeypair -v -storetype PKCS12 \
-keystore my-release-key.keystore \
-alias my-key-alias \
-keyalg RSA -keysize 2048 -validity 10000
```

Move the generated .keystore file into android/app.

Step 2 — Add credentials to android/gradle.properties

```
MYAPP_RELEASE_STORE_FILE=my-release-key.keystore
MYAPP_RELEASE_KEY_ALIAS=my-key-alias
MYAPP_RELEASE_STORE_PASSWORD=*****
MYAPP_RELEASE_KEY_PASSWORD=*****
```

Step 3 — Reference it in android/app/build.gradle

```
signingConfigs {  
    release {  
        storeFile file(MYAPP_RELEASE_STORE_FILE)  
        storePassword MYAPP_RELEASE_STORE_PASSWORD  
        keyAlias MYAPP_RELEASE_KEY_ALIAS  
        keyPassword MYAPP_RELEASE_KEY_PASSWORD  
    }  
}
```

Step 4 — Build the release APK

```
cd android  
./gradlew assembleRelease
```

Output file:

android/app/build/outputs/apk/release/app-release.apk

4. Build an Android App Bundle (for Play Store)

Google Play requires an .aab bundle rather than a raw APK for new submissions.

```
cd android  
./gradlew bundleRelease
```

Output file:

android/app/build/outputs/bundle/release/app-release.aab

5. Clean & Troubleshooting Commands

Use these when a build fails, behaves oddly, or after switching branches with dependency changes.

```
# Clean Gradle build cache
cd android && ./gradlew clean

# Clear Metro bundler cache
npx react-native start --reset-cache

# Reinstall node modules from scratch
rm -rf node_modules && npm install

# Check your environment for common setup issues
npx react-native doctor

# Print environment info (useful for bug reports)
npx react-native info
```

6. Install an APK on a Device / Emulator

```
adb install path/to/app-release.apk

# Reinstall, keeping existing app data
adb install -r path/to/app-release.apk

# List connected devices
adb devices
```

7. Quick Reference

Goal	Command
Start Metro bundler	<code>npx react-native start</code>
Run on Android (debug)	<code>npx react-native run-android</code>
Build debug APK	<code>./gradlew assembleDebug</code>
Build release APK	<code>./gradlew assembleRelease</code>
Build release AAB	<code>./gradlew bundleRelease</code>
Clean Gradle cache	<code>./gradlew clean</code>
Install APK via adb	<code>adb install app-release.apk</code>
Diagnose setup issues	<code>npx react-native doctor</code>

Note: All 'gradlew' commands assume you are inside the android/ folder, or you can run them from the project root by prefixing with android/ (e.g. android/gradlew assembleDebug) or by using ./android/gradlew.