

# Distribución de builds Android con EAS — VeriX

A diferencia de iOS, Android **no requiere registrar dispositivos** ni certificados de por vida — un APK se puede instalar en cualquier teléfono con "orígenes desconocidos" habilitado. Esto hace que el flujo de pruebas sea más simple.

“ Para generar el build en sí ( `.apk` / `.aab` ) con versión y nombre automáticos, ver `BUILD.md` ( `scripts\build.ps1` ). Esta guía asume que ya tienes ese archivo listo.

## Opción A: Test / instalación directa (APK)

Ideal para pruebas rápidas en cualquier dispositivo del equipo, sin pasar por Google Play.

### 1. Verificar el perfil en `eas.json`

Para pruebas, el perfil `test` debe generar un **APK** (instalable directo), no un AAB (que es exclusivo para Play Store):

```
"test": {  
  "distribution": "internal",  
  "android": {  
    "buildType": "apk"  
  }  
}
```

### 2. Correr el build

```
.\scripts\build.ps1 -Env test -Platform android
```

El resultado es un `.apk` en la carpeta `builds\`.

## 3. Instalar en el dispositivo

Dos formas:

### a) Link directo (más fácil para compartir)

- El build queda disponible en [expo.dev](https://expo.dev) con un link de descarga.
- Abre ese link **desde el navegador del teléfono Android** y descarga el APK.
- Si es la primera vez, Android pedirá habilitar "Instalar apps desconocidas" para ese navegador — se acepta una sola vez.

### b) Cable USB (si tienes el .apk en la PC)

```
adb install builds\VeriX-test-<version>.apk
```

(requiere `adb` — viene con Android Studio / platform-tools, y el dispositivo con depuración USB activada)

## Notas

- No hay expiración de certificado ni límite de dispositivos, a diferencia de iOS ad hoc.
- Cada instalación reemplaza la anterior si usas el mismo `applicationId` — no necesitas desinstalar manualmente entre builds.

## Opción B: Producción (publicación en Google Play)

### 1. Requisitos previos

- Cuenta de Google Play Console (pago único de \$25, una sola vez).
- App creada en Play Console con su ficha (descripción, screenshots, política de privacidad, clasificación de contenido).
- Una **cuenta de servicio (service account)** de Google Cloud con permisos en Play Console, y su archivo JSON de credenciales — necesario para que `eas submit` pueda subir builds automáticamente.
  - Se configura una vez en `eas.json` → `submit.production.android.serviceAccountKeyPath`, apuntando al archivo `.json` (no se sube a git — agregarlo a `.gitignore`).

### 2. Ajustar el perfil de producción en `eas.json`

Play Store requiere formato **AAB** (Android App Bundle), no APK:

```
"production": {  
  "distribution": "store",  
  "android": {  
    "buildType": "app-bundle"  
  }  
}
```

### 3. Generar el build

```
.\scripts\build.ps1 -Env prod -Platform android
```

### 4. Subir el build a Play Console

```
npx eas-cli submit -p android --latest
```

Esto sube el `.aab` automáticamente al track que defines (por defecto, `internal` — se puede cambiar a `alpha`, `beta` o `production` en `eas.json` bajo `submit.production.android.track`).

### 5. Testing interno en Play Console (recomendado antes de producción)

- En Play Console → tu app → **Testing** → **Internal testing**.
- Agrega testers por email o crea un link de opt-in.
- Los testers instalan la app **desde el Play Store** (no requiere APK manual) una vez aceptan la invitación.
- Esto evita mandar builds directo a producción sin pasar por revisión previa de tu equipo.

### 6. Promover a producción

- Una vez validado en internal testing, en Play Console → **Production** → **Create new release** → selecciona el build ya subido (o promuévelo directo desde el track de testing con "Promote release").
- Completa el rollout (puede ser gradual, ej. 20% → 50% → 100%, para reducir riesgo).
- Google revisa el release (usualmente horas, a veces hasta un par de días en la primera publicación de una app nueva).

## Notas

- El `versionCode` de Android debe ser mayor en cada release nuevo — si usas `"autoIncrement": true` en `eas.json` (como ya está configurado en VeriX), EAS lo maneja automáticamente.
- No puedes volver a subir el mismo `versionCode` una vez publicado, ni siquiera si cancelas el rollout.

## ¿Cuál usar?

|                                | Test (APK directo)                 | Producción (Play Store)                      |
|--------------------------------|------------------------------------|----------------------------------------------|
| Setup inicial                  | Ninguno                            | Cuenta de Play Console + service account     |
| Requiere registrar dispositivo | No                                 | No                                           |
| Formato de build               | APK                                | AAB                                          |
| Pasa por revisión de Google    | No                                 | Sí (horas a días)                            |
| Ideal para                     | Prueba rápida en cualquier Android | Lanzamiento público / testers vía Play Store |

## Comparación rápida con iOS

|                           | Android                             | iOS                                   |
|---------------------------|-------------------------------------|---------------------------------------|
| Pruebas rápidas           | APK directo, sin registrar nada     | Requiere registrar UDID (ad hoc)      |
| Testing con más gente     | Internal testing track (Play Store) | TestFlight                            |
| Producción                | Play Console (AAB)                  | App Store Connect (revisión completa) |
| Costo de cuenta developer | \$25 único                          | \$99/año                              |