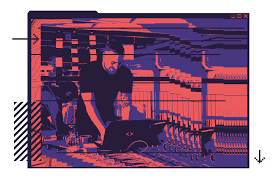
Puesta en producción segura

## 



| Vulnerabilidades |
| --- |
| Fuentes de información |
| [Incibe](https://www.incibe.es/incibe-cert/tags/Android) ([último reporte](https://www.incibe.es/incibe-cert/alerta-temprana/avisos/boletin-de-seguridad-de-android-de-noviembre-de-2023))  * [Android](https://source.android.com/docs/security/bulletin?hl=es) |
| Ejemplos |

| **CVE** | **Descripción** | **Tipo** | **Versión** |
| --- | --- | --- | --- |
| CVE-2021-0701 | Posible actividad en segundo plano de servicio de impresión | Overflow | cpe:2.3:o:google:android:14.0:\*:\*:\*:\*:\*:\*:\* |
| CVE-2021-0877 | El código que usa FILTER\_VALIDATE\_FLOAT con límites mín/máx. | Overflow | PHP  7.4.x < 7.4.28, versiones 8.0.x < 8.0.16 y versiones 8.1.x < 8.1.3, |
| CVE-2022-42536 |  | Ejecución remota de código | cpe:2.3:o:google:android:-:\*:\*:\*:\*:\*:\*:\* |
| CVE-2023-21162 |  | Escalada de privilegios | cpe:2.3:o:google:android:-:\*:\*:\*:\*:\*:\*:\* |
| CVE-2023-21361 | Ejecución de còdigo tras el uso de un dispositivo bluetooth | Ejecución remota de código | cpe:2.3:o:google:android:14.0:\*:\*:\*:\*:\*:\*:\* |

| Bases |
| --- |
| Android es un sistema operativo para móviles desarrollado por la Open Handset Alliance y lanzado en septiembre.  Está disponible para distintos tipos de arquitectura, como x86, x86\_64 y ARM. Este sistema operativo está basado en el kernel de Linux y ejecuta aplicaciones a través del Android Runtime (ART).  Además, Android soporta varios sistemas de archivos, incluyendo EXT4, JFFS2, YAFFS2, entre otros. También ofrece la capacidad de cifrado de disco, proporcionando una capa adicional de seguridad para los datos almacenados en dispositivos móviles.  Una característica destacada de Android es que es de código libre, basado en una licencia Apache 2.0, lo que permite a los desarrolladores acceder y modificar el código fuente según sus necesidades. Cabe destacar que Android pertenece a Google, lo que ha contribuido a su integración con servicios y productos de la empresa. |
| Componentes |
|  |
| Particiones |
| [Link sobre particiones](https://source.android.com/docs/core/architecture/partitions?hl=es) |
| > Android 11 |
|  |
| * **Imagen de sistema único (SSI).** Una nueva imagen conceptual que contiene las imágenes system y system\_ext . Cuando estas particiones son comunes para un conjunto de dispositivos de destino, esos dispositivos pueden compartir el SSI y omitir la creación del system y de las imágenes system\_ext . * **partición system\_ext .** Una nueva partición que puede utilizar recursos system y puede incluir módulos del sistema que:   + *Amplíe los módulos del sistema AOSP en la partición system .* Recomendamos actualizar dichos módulos a AOSP para que puedan instalarse en la partición system más adelante.   + *Agrupe módulos OEM o específicos de SoC.* Recomendamos desagregar dichos módulos para que puedan instalarse en la partición del product o vendor . * **partición system .** Imagen de sistema común utilizada para productos OEM. Recomendamos sacar los módulos propietarios de la partición system , ya sea transfiriéndolos a AOSP o moviéndolos a la partición system\_ext . * **partición product .** Esta partición ahora puede usar interfaces permitidas para instalar módulos específicos del producto que no están incluidos con ninguna otra partición. |
| Estructura de proyecto android |
| 1. **Directorio "app":**    * **src/main/java:** Contiene el código fuente de la aplicación en el lenguaje de programación Java o Kotlin. Dentro de este directorio, se organiza el código en paquetes según la convención de nombres de dominio invertido (por ejemplo, "com.ejemplo.miapp").    * **res:** Contiene recursos como diseños de interfaz de usuario, imágenes, archivos de animación, etc.      + **drawable:** Recursos gráficos (imágenes, iconos, etc.).      + **layout:** Archivos XML que definen la estructura y apariencia de las interfaces de usuario.      + **values:** Archivos XML para definir valores constantes, estilos y otros recursos.    * **AndroidManifest.xml:** Este archivo XML describe la configuración fundamental de la aplicación, como actividades, servicios, permisos, etc.    * **assets:** Contiene archivos estáticos que se incluirán tal cual en el paquete de la aplicación. 2. **Directorio "res":**    * **mipmap:** Contiene íconos de la aplicación en diferentes densidades.    * **raw:** Contiene archivos de recursos que deben conservarse en su forma original.    * **anim:** Almacena archivos de animación en formato XML.    * **xml:** Puede contener archivos XML adicionales utilizados por la aplicación. 3. **Directorio "src":**    * **androidTest:** Contiene pruebas de instrumentación para la aplicación.    * **test:** Contiene pruebas unitarias para la aplicación. 4. **Otros archivos y directorios importantes:**    * **build.gradle (Module: app):** Archivo de configuración del proyecto que especifica las dependencias, versiones de compilador, y otros ajustes.    * **build.gradle (Project: nombre\_del\_proyecto):** Archivo de configuración del proyecto a nivel de proyecto.    * **gradle:** Directorio que contiene archivos y carpetas relacionados con la configuración de Gradle, el sistema de construcción utilizado por Android Studio.    * **gradle w, gradlew.bat, gradle/ :** Archivos y directorios relacionados con Gradle Wrapper, que permite ejecutar Gradle sin necesidad de tenerlo instalado globalmente.    * **proguard-rules.pro:** Archivo de configuración para ProGuard, una herramienta de optimización y ofuscación de código. |
| Guía de Seguridad de Android |
| Guia de uso seguro de dispositivos móviles android → [Link](https://www.ccn-cert.cni.es/es/pdf/guias/series-ccn-stic/400-guias-generales/3588-ccn-stic-453g-guia-practica-de-seguridad-en-dispositvos-moviles-android-9/file?format=html) |
| **Principales medidas a tener en cuenta a la hora de utilizar tanto los dispositivos como las comunicaciones móviles**.   * El dispositivo móvil debe estar protegido mediante un código de acceso robusto asociado a la pantalla de bloqueo (o una huella digital). * Se debe hacer uso de las capacidades nativas de cifrado del dispositivo móvil con el objetivo de proteger todos los datos e información almacenadas en el mismo. * El sistema operativo del dispositivo móvil debe estar siempre actualizado. * No conectar el dispositivo móvil a puertos USB desconocidos. * Deshabilitar todos los interfaces de comunicaciones inalámbricas del dispositivo que no vayan a ser utilizadas de forma permanente por parte del usuario. * No conectar el dispositivo a redes wifi públicas abiertas que no implementan ningún tipo de seguridad. * No instalar ninguna aplicación móvil que no provenga de una fuente de confianza como los mercados oficiales. * Se recomienda no otorgar permisos innecesarios o excesivos a las apps, limitando así los datos y la funcionalidad a la que éstos tendrán acceso. * Siempre que sea posible se debe hacer uso del protocolo HTTPS (mediante la inserción del texto "https://" antes de la dirección web del servicio a contactar). * Se deben realizar copias de seguridad periódicas y preferiblemente automáticas, de todos los contenidos del dispositivo móvil que se desea proteger y conservar. |
| [Permisos de aplicaciones android](https://developer.android.com/guide/topics/permissions/overview?hl=es-419) |
| Tipos |
| De instalaciónNormalesDe Firma El sistema le otorga un permiso de firma a una app solo cuando está firmada por el mismo certificado que la app o el SO que define el permiso.  Las aplicaciones que implementan servicios con privilegios, como los servicios Autocompletar o VPN, también usan permisos de firma. Estas apps requieren permisos de firma de vinculación del servicio para que solo el sistema pueda vincularse a los servicios. De tiempo de ejecución Los permisos de tiempo de ejecución, también conocidos como permisos peligrosos, le brindan a la app acceso adicional a datos restringidos o permiten que realice acciones restringidas que afectan de forma más significativa el sistema y a otras apps. Por lo tanto, debes [solicitar permisos de tiempo de ejecución](https://developer.android.com/training/permissions/requesting?hl=es-419) en la app antes de poder acceder a los datos restringidos o realizar acciones restringidas. Grupos Constan de un conjunto de permisos relacionados de forma lógica. Por ejemplo, los permisos para enviar y recibir mensajes SMS pueden pertenecer al mismo grupo, ya que ambos se relacionan con la interacción de la aplicación con SMS. |
| Declaración |
| Proceso de solicitud de permisos:  **Declaración en el manifiesto:** El desarrollador debe declarar los permisos necesarios en el archivo de manifiesto de la aplicación. |
| **Solicitud en tiempo de ejecución:** Para permisos peligrosos, la aplicación debe solicitarlos explícitamente en tiempo de ejecución. Esto se hace mediante el código en el que se comprueba si el permiso está concedido y, si no lo está, se solicita al usuario. |
| **if (ContextCompat.checkSelfPermission(this,Manifest.permission.CAMERA) == PackageManager.PERMISSION\_GRANTED) {**  **// El permiso ya está concedido**  **} else {**  **// Solicitar el permiso**  **ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.CAMERA}, REQUEST\_CAMERA\_PERMISSION);**  **}** |
| **Respuesta del usuario:** El usuario puede conceder o denegar el permiso. La aplicación debe manejar esta respuesta en el método onRequestPermissionsResult. |
| **@Override**  **public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {**  **if (requestCode == REQUEST\_CAMERA\_PERMISSION) {**  **if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {**  **// Permiso concedido**  **} else {**  **// Permiso denegado**  **}**  **}**  **}** |
| Consideraciones adicionales:  * **Solicitudes múltiples:** Una aplicación puede necesitar solicitar varios permisos, y el usuario tiene la opción de conceder o denegar cada uno individualmente. * **Versiones de Android:** A partir de Android 6.0 (API nivel 23), el sistema operativo introdujo el modelo de permisos en tiempo de ejecución. * **Revocación de permisos:** Los usuarios pueden revocar permisos en cualquier momento desde la configuración del dispositivo. * **Permisos personalizados:** Los desarrolladores pueden definir permisos personalizados para requerir acceso a características específicas de sus aplicaciones. |