World App: El Secreto de tus Ojos está en los permisos Android

Amador Aparicio[1], M. Mercedes Martínez-González[1], Pablo A. Criado-Lozano[2]
Grupo de Investigación en Ingeniería de la Privacidad de la Universidad de Valladolid
{amador,mercedes}@infor.uva.es[1]
pacriado@uemc.es[2]

***Resumen*—A medida que se extiende el uso de dispositivos móviles, la privacidad en las aplicaciones móviles adquiere cada vez más importancia. Este artículo investiga el nivel de control que tienen los usuarios sobre su privacidad cuando utilizan aplicaciones móviles. Para ello, se propone un análisis estático de las aplicaciones. Se eligió como caso de estudio la aplicación "World App", conocida por su posible impacto en la privacidad. Se descubrió que World App solicita un total de 23 permisos, incluidos 10 clasificados como "peligrosos"que podrían afectar significativamente a la privacidad y seguridad de los usuarios. Además, hay otros permisos especiales que los usuarios no pueden controlar. Se concluye que los usuarios no tienen el nivel deseado de control sobre su privacidad: Los permisos especiales están fuera de su control, y la granularidad del control de acceso basado en permisos que utiliza Android no permite el control detallado sobre los datos que los usuarios necesitan para proteger su privacidad de forma eficaz.**

***Index Terms*—Privacidad, WorldCoin, WorldApp, Permisos Android, Aplicaciones móviles.**

Tipo de contribución: *Investigación original (límite 8 páginas).*

I. Introducción

La creciente popularidad de las aplicaciones móviles ha supuesto un aumento de la cantidad de datos que se almacenan en los dispositivos móviles, y de los datos personales a los que se accede a través de las aplicaciones instaladas en ellos, con el consiguiente riesgo para la privacidad de sus usuarios. En este contexto cabe preguntarse: ¿De qué herramientas disponen los usuarios para ser parte activa en la protección de su privacidad? ¿Conocen y saben utilizar estas herramientas los usuarios ¿Son suficientes para garantizarles un control total de su privacidad y una protección completa si es lo que desean?

Para responder a esta última pregunta, hemos elegido un caso de estudio cuya popularidad reciente se debe precisamente a la prevención que ha despertado en los especialistas en privacidad y en los organismos supervisores como la Agencia Española de Protección de Datos (AEPD)[1]: el proyecto World Coin. En concreto, nos centramos en la aplicación móvil asociada, *World App*.

Según figura en su presentación, el proyecto *Worldcoin*[2] integra tecnología de inteligencia artificial (IA) y blockchain para ofrecer acceso económico global [1]. *Worldcoin* consta de tres componentes clave: una identidad digital universal única basada en rasgos biométricos, una moneda global en forma de *tokens Worldcoin* y una aplicación[3], *World App*, para pagos en criptomoneda o activos tradicionales, transferencias y compras [2].

El método de trabajo que utilizamos se apoya en el estudio de la única herramienta que Android proporciona a los usuarios para gestionar su privacidad: los permisos. Sin embargo, los usuarios no pueden controlar todos los permisos que utilizan las aplicaciones móviles que instalan. Deseablemente, deberían ser precisamente estos, los que sí pueden controlar, los que gestionan el acceso a sus datos personales, lo cual les daría mayor control sobre su privacidad.

Para conocer en qué grado una app pone en riesgo la privacidad de sus usuarios, y la capacidad de acción que tienen éstos, realizamos un análisis estático. En este análisis, contrastamos las declaraciones que las apps hacen en su *AndroidManifest.xml* y clasificamos los permisos que solicita en función de su *tipo*. Posteriormente, filtramos en base a las evidencias que encontramos en su código aquellos que realmente utiliza. A partir de estos datos, podemos conocer qué permisos utiliza una app, y cuáles de ellos podrían controlar los usuarios en tiempo de ejecución.

En el caso de la app World App se han encontrado 23 permisos, de los cuales los usuarios pueden controlar 10, aunque son más los que pueden facilitar acceso a datos privados de los usuarios. Esto evidencia que no todos los permisos cuya concesión puede suponer un riesgo para la privacidad, son susceptibles de control por parte de los usuarios. De lo cual se deriva una reflexión sobre la necesidad de mejorar el nivel de granularidad del sistema de permisos de Android, para mejorar el control de los usuarios sobre su privacidad.

El resto del artículo se estructura como sigue. En la Sección II se discute el trabajo relacionado. En la Sección III se presenta el método de trabajo utilizado para analizar la app objeto de este estudio. Los resultados obtenidos cuando se aplica a la app *World App* se muestran en la sección IV. Finalmente, se presentan las conclusiones en la sección V.

II. Trabajo relacionado

II-A. Worldcoin

Según *Worldcoin*, su misión es crear una red mundial que ofrezca identidad inclusiva y acceso financiero a la mayoría de la humanidad [2]. *Worldcoin* comprende una red de identidad digital, conocida como *World ID*[4], basada en el concepto de *"prueba de personalidad"*. El *World ID* se complementa

[1]"La Agencia ordena una medida cautelar que impide a Worldcoin seguir tratando datos personales en España". 6 de Marzo de 2024. https://www.aepd.es/prensa-y-comunicacion/notas-de-prensa/la-agencia-ordena-medida-cautelar-que-impide-a-worldcoin-seguir-tratando-datos-personales-en-espana

[2]https://es-es.worldcoin.org/world-app

[3]https://play.google.com/store/apps/details?id=com.worldcoin&hl=es_AR

[4]https://es-es.worldcoin.org/world-id

con una moneda digital denominada WLD[5], que otorga a los usuarios una participación simplemente por ser humanos. La *World App*[6], sirve como la primera interfaz para el *World ID* y el protocolo *Worldcoin*. *Worldcoin* se distribuye gratuitamente a las personas que se registran y verifican su identidad a través del sistema *Worldcoin* [3]. Esto puede hacerse descargando la aplicación *Worldcoin App*. A continuación, los usuarios deben verificar su identidad escaneando su iris con un dispositivo llamado *The Orb*[7].

La razón por la que *Worldcoin* eligió el escáner de iris para confirmar la identidad es que nuestros iris, al igual que las huellas dactilares, son únicos para cada persona y más difíciles de falsificar.

Sin embargo, la recopilación, transmisión y almacenamiento de datos biométricos tan sensibles como los patrones del iris plantea interrogantes fundamentales sobre la seguridad de estos datos. La existencia de vulnerabilidades dentro del ecosistema *Worldcoin* que exponga información biométrica irremplazable podría tener consecuencias duraderas para los individuos afectados [4]. A diferencia de las contraseñas o incluso los números de teléfono, los datos biométricos no pueden modificarse una vez que han sido comprometidos.

II-B. Modelo de permisos Android

Android implementa un mecanismo de permisos que protegen el acceso a datos sensibles del usuario y recursos críticos del sistema [5]. Para obtener acceso, las aplicaciones deben solicitar permisos específicos. Estos permisos pueden otorgarse automáticamente durante la instalación de la app, sin la intervención del usuario, o requerir la aprobación explícita del usuario. Para distinguirlos, Android establece tres categorías, o *niveles de protección*. Cada permiso tiene asociado un nivel de protección: todos los permisos tienen asociado un nivel de protección, y solo pueden pertenecer a una de estas categorías.

II-B1. Niveles de protección: Son tres: *normal*, *dangerous* y *signature* [6]. Existe además una cuarta categoría de permisos *especiales*.

- *Normal*. Permiso de bajo riesgo en cuanto a los recursos del dispositivo y los datos de los usuarios. Se otorga automáticamente durante la instalación de una aplicación, sin requerir la aprobación explícita del usuario.
- *Dangerous*. Permiso crítico debido a su capacidad para introducir riesgos potenciales en la privacidad y seguridad del usuario. Debido al riesgo inherente que conlleva, el sistema requiere una aprobación explícita por parte del usuario para conceder estos permisos. Durante la ejecución de una aplicación, los usuarios pueden conceder o denegar estos permisos de forma dinámica.
- *Signature*. Permiso que se otorga únicamente a aplicaciones que están firmadas con un certificado. Si la aplicación solicitante presenta el certificado adecuado, el sistema otorga automáticamente el permiso, sin notificar al usuario ni requerir su aprobación explícita.
- *Especiales*. Permisos que se conceden durante la instalación de la app. Los usuarios no tienen capacidad para concederlos o denegarlos durante la ejecución de la app en el dispositivo. Estos permisos se solicitan para realizar operaciones especiales dentro de una app. Tanto la plataforma Android como los fabricantes OEM[8] los utilizan cuando quieren proteger el acceso a acciones particularmente importantes, como actuar sobre otras apps [7].

II-C. Apps, permisos y privacidad: ¿control de los usuarios?

La privacidad de los usuarios cuando utilizan las apps ha sido una preocupación constante desde que su uso se popularizó [8], [9], [10], [11]. Existen estudios que demuestran que el nivel de estudios u otros factores sociales no son determinantes en la actitud que los usuarios tienen respecto a su privacidad cuando gestionan una aplicación móvil [12], [13], [14].

Esta gestión, más despreocupada de lo que los propios usuarios manifestaban, dio lugar a lo que se conoce como la *Paradoja de la Privacidad* [15]. Esta consiste en que los usuarios manifiestan estar muy preocupados por su privacidad, pero su gestión contradice esta afirmación, mostrando una cierta despreocupación sobre cómo protegerla. No obstante, la evolución histórica muestra un interés creciente por parte de los usuarios en hacer una gestión eficaz para proteger su privacidad. El problema es que en muchos casos se encuentran desorientados sobre el mejor modo de hacerlo.

Para ayudar a los usuarios, los mercados (*markets*) para descarga de apps de los proveedores más importantes, Google y Apple, han introducido en los últimos tiempos información sobre los datos privados a los que las apps acceden. Esta información, que los usuarios pueden consultar en el propio market, se basa en declaraciones que los desarrolladores aportan voluntariamente. Google ha empezado a potenciar estas prácticas más recientemente que Apple, desde 2021 [16].

La información que ofrecen en ambos markets es información sobre el posible acceso de las apps a un conjunto de categorías de datos. Estas categorías guardan relación con las agrupaciones lógicas de los permisos que los usuarios pueden encontrar en los Ajustes de sus dispositivos. Los permisos son el mecanismo que Android utiliza para controlar el acceso a los datos personales que manejan las aplicaciones [17]. De hecho, este es el único mecanismo que Android ofrece a los usuarios para empoderarse en la protección de su privacidad. No tienen más capacidad de acción, salvo la decisión de instalar o no una app.

La relación (*dato personal*, *permiso*) ha motivado diversos trabajos que investigan si es posible determinar el nivel de riesgo para la seguridad y/o privacidad de los usuarios derivada de las aplicaciones móviles en base a los permisos que requieren. Entre las propuestas se encuentran medidas para analizar el impacto sobre la privacidad de las aplicaciones móviles [18], [19], [20], [21]. Shrivastava y col. ofrecen en [22] un compendio de las investigaciones realizadas entre los años 2010 y 2020. Una de las conclusiones más relevantes de

[5]https://es-es.worldcoin.org/worldcoin-token
[6]https://es-es.worldcoin.org/world-app
[7]https://es-es.worldcoin.org/find-orb

[8]En el contexto de Android, OEM se refiere a *Original Equipment Manufacturer*, o Fabricante de Equipos Originales en el contexto de Android). Son los fabricantes de hardware que construyen dispositivos como teléfonos inteligentes y tabletas que ejecutan el sistema operativo Android. Los OEMs a menudo agregan su propia interfaz de usuario y aplicaciones sobre la base del sistema operativo Android proporcionado por Google.

estos estudios es que el malware suele requerir mayor cantidad de permisos que las aplicaciones benignas. Razón por la cual se ha propuesto utilizar este aspecto como indicador para reconocer malware [23], [24], [25]. Otra conclusión relevante es que a medida que avanzan las versiones de las apps, estas requieren más permisos, incluso aunque sean benignas, lo cual aumenta su capacidad potencial para impactar sobre la privacidad de sus usuarios [26]. En general, cuanto mayor es el número de permisos que una app solicita, mayor es su impacto en la privacidad de sus usuarios [27], [14], [13], [28].

III. Metodología

La figura 1 describe el proceso de análisis particularizado en la app *World App*, comenzando desde la obtención de la aplicación hasta la clasificación de los permisos que utiliza. Las elipses representan procesos y las fechas los datos en cada una de las etapas, que sirve como entrada a la siguiente.

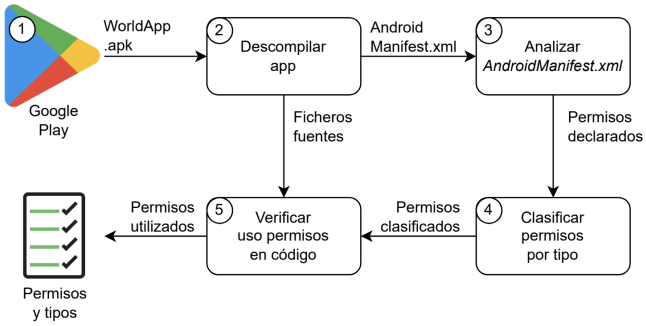

Figura 1. Representación gráfica de la metodología empleada.

Los pasos de la metodología empleada y que se seguirán de manera secuencial para responder a la pregunta de investigación planteada son los siguientes:

1. **Obtención de la aplicación**. Se descarga la app de estudio desde su fuente correspondiente. En nuestro caso, la app de estudio será *World App*[9] y se descargará desde el *market* oficial *Google Play*[10].
2. **Descompilación de la aplicación**. Esta estapa recibe como entrada la app. Se descompila para obtener acceso al fichero *AndroidManifest.xml* y al resto del código fuente de la app. Como resultado de esta etapa se tienen los ficheros de código fuente que forman la app.
3. **Análisis del archivo *AndroidManifest.xml***. Se examina el fichero *AndroidManifest.xml* para identificar y extraer todos los permisos declarados en la app objeto de estudio. El resultado de esta etapa son todos los permisos declarados en el fichero *AndroidManifest.xml*
4. **Clasificación de los permisos por tipo**. Esta etapa recibe como entrada todos los permisos extraídos del fichero *AndroidManifest.xml*. Los permisos extraídos se clasifican según su tipo utilizando la definición de permisos y tipos proporcionada por Google, según se describe en la documentación oficial [29], [6]. Como resultado de esta etapa se tienen los permisos declarados en el fichero *AndroidManifest.xml* clasificados en función del tipo. Esta fase es necesaria porque en el fichero *AndroidManifest.xml* no figura el tipo de cada permiso, únicamente figura el nombre del permiso.
5. **Verificar uso de permisos en código**. Para confirmar que los permisos declarados en el archivo *AndroidManifest.xml* son efectivamente utilizados por la aplicación, se analiza el código fuente en busca de llamadas a la activación de los permisos. Se filtran los permisos realmente utilizados por la aplicación en su funcionamiento. El resultado de esta etapa son los permisos que realmente está utilizando la app.

Disponer de la relación de los permisos y tipo que utiliza una app, nos permite conocer cuáles son los permisos que el usuario puede controlar. En consecuencia, disponer de indicadores de la capacidad real de los usuarios para tomar el control de su privacidad cuando utiliza esa app.

IV. Resultados

Se presenta el análisis de la app *World App*[11] , vinculada al proyecto *World Coin*[12]. Se han extraído los permisos declarados en el fichero *AndroidManifest.xml* de la app y clasificados en función del tipo de permiso. Los resultados se muestran en la tabla I.

Se han categorizado en tres tipos: *dangerous*, *normal*, y *otros*. Se observa que la categoría con el mayor número de permisos es la de permisos *dangerous* con un total de 10 permisos. Esto indica que la aplicación necesita acceder a funciones y datos del dispositivo que requieren una atención especial por parte del usuario en el momento de otorgarlos. Estos son los permisos que los usuarios pueden controlar.

Tabla I
Cantidad de permisos declarados en la app.

Categorías de Permisos	Número
Permisos *dangerous*	10
Permisos *normal*	9
Permisos especiales	4
Total permisos	23

Los permisos de tipo *normal* están ligeramente por debajo, con un total de 9. Estos permisos suelen ser menos críticos y generalmente no se consideran un riesgo para la privacidad o seguridad del usuario. El usuario no puede controlarlos.

Por último, hay 4 permisos especiales, una categoría que incluye permisos definidos por los fabricantes de dispositivos sobre los cuales los usuarios no tienen ningún poder de decisión una vez instalada la app en el dispositivo.

IV-A. Permisos de tipo dangerous declarados en la app

La tabla II proporciona un desglose detallado de los permisos de tipo *dangerous* que están declarados en la app. Los permisos que solicita *World App* le permiten conocer la ubicación de los usuarios, acceder a la cámara del dispositivo, a los contactos, al estado del teléfono, acceder a los ficheros de audio, vídeo e imágenes almacenadas en los dispositivos externos conectados al teléfono. Todas estas categorías están clasificadas por Google como potencialmente intrusivas para la privacidad [30].

[9]https://play.google.com/store/apps/details?id=com.worldcoin&hl=es&gl=US

[10]https://play.google.com/store/apps?hl=es&gl=US

[11]https://play.google.com/store/apps/details?id=com.worldcoin&hl=es&gl=US

[12]https://es-es.worldcoin.org/

Tabla II
PERMISOS DE TIPO *dangerous* DECLARADOS EN LA APP.

Permisos *dangerous*	**Descripción**
ACCESS_COARSE_LOCATION	Permite que una aplicación acceda a la ubicación aproximada del dispositivo.
CAMERA	Permite a una aplicación acceder al *hardware* de la cámara del dispositivo.
ACCESS_FINE_LOCATION	Permite que una aplicación acceda a una ubicación precisa.
READ_EXTERNAL_STORAGE	Permite a una app acceder a los archivos almacenados en el almacenamiento externo del dispositivo.
READ_PHONE_STATE	Permite a una app acceder al estado del teléfono.
READ_CONTACTS	Permite a una app acceder a los contactos almacenados en el dispositivo.
READ_MEDIA_VIDEO	Permite a una app acceder a los archivos de medios almacenados en el dispositivo, incluidos los videos.
POST_NOTIFICATIONS	Permite a una app publicar notificaciones.
READ_MEDIA_IMAGES	Permite a una aplicación acceder a los archivos almacenados en el almacenamiento externo del dispositivo, incluidas las imágenes.
READ_MEDIA_AUDIO	Permite a una aplicación acceder a los archivos almacenados en el almacenamiento externo del dispositivo, incluidos los archivos de audio.

IV-B. Permisos de tipo normal declarados en la app

La tabla III muestra los permisos de tipo *normal* que la app solicita. El conjunto de permisos listados se asocian a operaciones estándar, como la conexión a internet, la utilización de características de *hardware* para la autenticación biométrica, y el manejo de estados de conexión y servicios en primer plano.

La autorización de estos permisos es generalmente menos invasiva. Aún así, es esencial que los desarrolladores y las plataformas expliquen su uso.

IV-C. Permisos especiales declarados en la app

La tabla IV muestra un conjunto de permisos categorizados como "permisos especiales" en *World App*. Estos permisos, definidos por los fabricantes de dispositivos, están vinculados con la integración de la app en servicios específicos, principalmente los proporcionados por Google. A primera vista, estos permisos pueden parecer técnicos y altamente especializados, no relacionados con datos personales, pero tienen implicaciones significativas en términos de privacidad y seguridad.

Resulta particularmente interesante el permiso `READ_GSERVICES`, que permite a las aplicaciones acceder a los servicios proporcionados por Google en dispositivos Android, como *Google Maps* y *Google Location Services*. Este permiso es esencial para aplicaciones que dependen de la funcionalidad de ubicación proporcionada por Google para ofrecer servicios como navegación, búsqueda de lugares y seguimiento de ubicación [31]. El hecho de que la aplicación requiera permisos para leer datos de configuración de los servicios de Google y para vincularse con el servicio de referencias de Google Play indica que la aplicación se comunica con servicios externos, lo que puede incluir la transmisión de datos del usuario. Aunque estos permisos pueden ser necesarios para funcionalidades legítimas, también abren la posibilidad de que la información del usuario sea recopilada de forma que el usuario final desconoce.

El permiso `RECEIVE` es utilizado para recibir mensajes del servicio de mensajería en la nube de Google. Es un permiso común para las aplicaciones que necesitan recibir notificaciones push. Desde la perspectiva de la privacidad, este permiso no resulta intrusivo, ya que solo permite a la aplicación recibir mensajes, y no le da acceso a ninguna otra funcionalidad o dato en el dispositivo.

El permiso `DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION` permite a la aplicación recibir notificaciones *push*. Su presencia en la lista de permisos sugiere que la aplicación podría estar protegiendo sus propios procesos de intervenciones o accesos externos, para lo cual no necesita acceder a los datos del usuario. En consecuencia, no parece un permiso que pueda llegar a comprometer la privacidad de los usuarios.

El permiso `BIND_GET_INSTALL_REFERRER_SERVICE` se utiliza para recoger estadísticas sobre cómo instalan los usuarios las aplicaciones, lo que puede incluir, por ejemplo, desde qué anuncio o medio se hizo clic para llegar a la instalación. Estos datos pueden ayudar a los desarrolladores a comprender la efectividad de sus campañas de marketing y mejorar la distribución de sus aplicaciones. Para el usuario, podría significar que la información sobre su comportamiento de instalación está siendo compartida con los desarrolladores de aplicaciones.

IV-D. Invocación de los permisos declarados

La figura 2 muestra parte del código fuente de la app *World App* que se obtiene tras descompilarla[13]. Aunque el código fuente aparece ofuscado, se observa la creación del objeto *context*: `Context context = l0Var.f1654a;` Este objeto proporciona acceso a recursos y servicios específicos a nivel de aplicación [32].

```
Context context = l0Var.f1654a;
int f11 = cg.d.f(context, "android.permission.ACCESS_COARSE_LOCATION");
Location location3 = null;
LocationManager locationManager = l0Var.f1655b;
if (f11 == 0) {
    try {
        if (locationManager.isProviderEnabled("network")) {
            location2 = locationManager.getLastKnownLocation("network");
            location = location2;
        }
    } catch (Exception e11) {
        Log.d("TwilightManager", "Failed to get last known location", e11);
    }
    location2 = null;
    location = location2;
} else {
    location = null;
}
```

Figura 2. Invocación al permiso `ACCESS_COARSE_LOCATION`.

Posteriormente se comprueba si el permiso `ACCESS_COARSE_LOCATION` ha sido otorgado a la

[13]Para descompilar la app y buscar la invocación a los permisos dentro del código hemos utilizado la herramienta *jaxd-gui*: https://github.com/skylot/jadx.

Tabla III
PERMISOS DE TIPO *normal* DECLARADOS EN LA APP.

Permisos *normal*	**Descripción**
INTERNET	Permite que la aplicación se conecte a internet.
USE_FINGERPRINT	Permite a una app utilizar el *hardware* del sensor de huellas dactilares en un dispositivo.
ACCESS_NETWORK_STATE	Permite que una aplicación acceda al estado de la red del dispositivo.
USE_BIOMETRIC	Una aplicación tiene permiso para utilizar la autenticación biométrica en el dispositivo.
RECEIVE_BOOT_COMPLETED	Permite que una aplicación reciba una notificación cuando el dispositivo se haya iniciado completamente después de un arranque.
FOREGROUND_SERVICE	Permite a una aplicación ejecutar servicios en primer plano.
VIBRATE	Permite a una aplicación controlar la vibración del dispositivo.
ACCESS_WIFI_STATE	Permite a una aplicación acceder al estado del Wi-Fi en el dispositivo.
WAKE_LOCK	Permite a una aplicación mantener el dispositivo despierto cuando está en modo de suspensión o bloqueo automático.

Tabla IV
OTROS PERMISOS DECLARADOS EN LA APP.

Otros permisos	**Descripción**
READ_GSERVICES	Permite que una aplicación lea los datos de configuración y servicios proporcionados por los Servicios de Google. Se otorga en la instalación de la app.
RECEIVE	Permite a la aplicación recibir notificaciones push, no proporciona acceso a ninguna otra funcionalidad o datos en el dispositivo.
DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION	Permiso utilizado por las apps para impedir que otras aplicaciones se conecten a receptores dinámicos sin permiso explícito.
BIND_GET_INSTALL_REFERRER_SERVICE	Permite a una aplicación vincularse con el servicio de obtención de referencias de instalación proporcionado por Google Play Store.

app. En caso de que aún no haya sido otorgado, se obtiene una instancia de *LocationManager* (`LocationManager locationManager = l0Var.f1655b;`), que es el servicio de Android que permite a las aplicaciones acceder a la ubicación del dispositivo[14]. Con esto tenemos la evidencia de que el permiso se invoca realmente.

Para comprobar la invocación del resto de permisos basta con buscar el permiso vinculado al contexto (objeto *context*), es decir, la cadena `(context,NOMBRE_PERMISO)`, que comprueba si el permiso ha sido invocado o no.

La tabla V muestra los permisos de tipo *dangerous* (columna Permisos *dangerous*) que se invocan en el código de la app (columna Invocación). La primera columna muestra el permiso. La segunda, la invocación encontrada en el código de la app.

La tabla VI muestra los permisos *especiales* (columna Permisos *especiales*) que son invocados en el código de la app (columna Invocación). Se ha conseguido encontrar la invocación de dos de ellos, `READ_GSERVICES` y `DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION`. Se confirma que la app podría acceder a datos personales a través del servicio de Google, puesto que dispone del permiso necesario.

V. CONCLUSIONES

Tras el análisis de la app World App se ha encontrado que tiene declarados un total de 23 permisos, de los cuales 10 son de tipo *dangerous* (permisos que impactan en la privacidad y seguridad de los usuarios), 9 de tipo *normal* y 4 permisos especiales. Tanto los permisos de tipo *normal* como los permisos especiales son permisos que se conceden durante la instalación de la app y posteriormente los usuarios no tienen capacidad de decisión sobre ellos.

[14]https://developer.android.com/reference/android/content/Context

Resulta particularmente interesante en relación con la privacidad el permiso especial `READ_GSERVICES`. Al tratarse de un permiso especial, los usuarios no pueden controlarlo en tiempo de ejecución. Una vez se instala la app, el permiso se otorga y los usuarios no pueden denegarlo. En nuestra opinión, este tipo de permisos, que abren la posibilidad de acceder por vía indirecta a datos personales, restan capacidad de control a los usuarios sobre su privacidad. Su uso debería restringirse al mínimo imprescindible, y los usuarios deberían estar informados del modo más preciso sobre la finalidad para la que se solicitan.

El análisis de los permisos solicitados por esta app muestra que hay razones suficientes para cuestionarse si los usuarios tienen suficiente información y control sobre su privacidad. Lo cual nos lleva a la conclusión, compartida con investigaciones previas en el campo, de que la granularidad del control de acceso basado en permisos que utiliza Android es demasiado *grueso* para facilitar a los usuarios un control real de su privacidad. Del mismo modo que en una base de datos el control de acceso se diseña en *grano fino*, llegando a diferenciar los permisos otorgados para cada atributo, en los dispositivos móviles, cada vez más cercanos a contenedores o repositorios de datos personales, esta parece una evolución necesaria para conseguir un empoderamiento real de los usuarios.

AGRADECIMIENTOS

Este trabajo se incluye en las actividades del proyecto estratégico de Ciberseguridad “App-PI (*App Privacy Impact*): Un ecosistema para la evaluación del impacto de apps para dispositivos móviles sobre la privacidad y seguridad de sus usuarios”, el cual se realiza al amparo de un convenio de colaboración entre la Universidad de Valladolid y la S.M.E. Instituto Nacional de Ciberseguridad de España M.P., S.A. para la promoción de proyectos estratégicos de Ciberseguridad en España, en el marco de los fondos del Plan de

Tabla V
PERMISOS DE TIPO *dangerous* INVOCADOS POR LA APP.

Permisos *dangerous*	**Invocación**
ACCESS_COARSE_LOCATION	`int f11 = cg.d.f(context, ''android.permission.ACCESS_COARSE_LOCATION");`
CAMERA	`if (a4.a.a(context, ''android.permission.CAMERA'') == 0)`
ACCESS_FINE_LOCATION	`if (cg.d.f(context, ''android.permission.ACCESS_FINE_LOCATION'') == 0)`
READ_EXTERNAL_STORAGE	`if (a4.a.a(context, ''android.permission.READ_EXTERNAL_STORAGE'') == 0)`
READ_PHONE_STATE	-
READ_CONTACTS	`if (a4.a.a(context, ''android.permission.READ_CONTACTS'') == 0)`
READ_MEDIA_VIDEO	`if (a4.a.a(context, ''android.permission.READ_MEDIA_VIDEO'') == 0).`
POST_NOTIFICATIONS	`if (f53738b(context, ''android.permission.POST_NOTIFICATIONS'') == 0)`
READ_MEDIA_IMAGES	`if (a4.a.a(context, ''android.permission.READ_MEDIA_IMAGES'') == 0)`
READ_MEDIA_AUDIO	`if (a4.a.a(context, ''android.permission.READ_MEDIA_AUDIO'') == 0)`

Tabla VI
PERMISOS ESPECIALES INVOCADOS EN LA APP.

Permisos especiales	**Invocación**
READ_GSERVICES	`if (d.f(context, ''com.google.android.providers. gsf.permission.READ_GSERVICES'') == 0)`
RECEIVE	-
DYNAMIC_RECEIVER_NOT_EXPORTED_ PERMISSION	`if (d.f(context, context.getPackageName().+ ''.DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION'') == 0)`
BIND_GET_INSTALL_REFERRER_SERVICE	-

Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (*Next Generation*), el proyecto del Gobierno de España que traza la hoja de ruta para la modernización de la economía española, la recuperación del crecimiento económico y la creación de empleo, para la reconstrucción económica sólida, inclusiva y resiliente tras la crisis de la COVID19, y para responder a los retos de la próxima década.

REFERENCIAS

[1] T. Kraiwanit, P. Limna, P. Wattanasin, S. Asanprakit, and R. Thetlek, "Research in globalization," 2023.

[2] Worldcoin, "A new identity and financial network," 2023. [Online]. Available: https://whitepaper.worldcoin.org/

[3] E. Gent, "A cryptocurrency for the masses or a universal id?: Worldcoin aims to scan all the world's eyeballs," *IEEE Spectrum*, vol. 60, no. 1, pp. 42–57, 2023.

[4] D. Xiaotong and Z. Peng, "Exploring the intersection of data and ethics: Seeking a societal role for artificial general intelligence," *J Huma Soci Scie*, vol. 7, no. 3, pp. 1–11, 2024.

[5] S. Kumar and S. K. Shukla, "The state of android security," *Cyber Security in India: Education, Research and Training*, pp. 17–22, 2020.

[6] A. para desarrolladores, "Tipos de permisos básicos en android." [Online]. Available: https://developer.android.com/guide/topics/manifest/permission-element

[7] ——, "Permisos en android." [Online]. Available: https://developer.android.com/guide/topics/permissions/overview?hl=es-419

[8] Y.-A. de Montjoye, S. Gambs, V. D. Blondel, G. S. J. Canright, N. de Cordes, S. Deletaille, K. Engø-Monsen, M. García-Herranz, J. Kendall, C. F. Kerry, G. Krings, E. Letouzé, M. A. Luengo-Oroz, N. Oliver, L. Rocher, A. Rutherford, Z. Smoreda, J. E. Steele, E. Wetter, A. S. Pentland, and L. Bengtsson, "On the privacy-conscientious use of mobile phone data," *Scientific Data*, vol. 5, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:54472286

[9] B. Liu, M. S. Andersen, F. Schaub, H. Almuhimedi, S. Zhang, N. M. Sadeh, Y. Agarwal, and A. Acquisti, "Follow my recommendations: A personalized privacy assistant for mobile app permissions," in *Twelfth Symposium on Usable Privacy and Security, SOUPS 2016, Denver, CO, USA, June 22-24, 2016*. USENIX Association, 2016, pp. 27–41. [Online]. Available: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/liu

[10] E. Gashi and Z. Tafa, "Permission-based privacy analysis for android applications," *International Journal of Business and Technology*, vol. 6, no. 3, 2018. [Online]. Available: https://knowledgecenter.ubt-uni.net/ijbte/vol6/iss3/2

[11] J. Arbanas, P. H. Silverglate, S. Hupfer, J. Loucks, P. Raman, and M. Steinhart, "Data privacy and security worries are on the rise, while trust is down. deloitte's connected consumer survey 2023," Deloitte Center for Technology, Media & Telecommunications, Tech. Rep. [Online]. Available: https://www2.deloitte.com/us/en/insights/industry/telecommunications/connectivity-mobile-trends-survey/2023/data-privacy-and-security.html

[12] V. M. Wottrich, E. A. [van Reijmersdal], and E. G. Smit, "The privacy trade-off for mobile app downloads: The roles of app value, intrusiveness, and privacy concerns," *Decision Support Systems*, vol. 106, pp. 44 – 52, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923617302221

[13] K. Degirmenci, "Mobile users' information privacy concerns and the role of app permission requests," *International Journal of Information Management*, vol. 50, pp. 261 – 272, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0268401218307965

[14] S. Hudson and Y. Liu, "Mobile app users' privacy concerns: different heuristics for privacy assurance statements in the EU and china," *Inf. Technol. People*, vol. 36, no. 1, pp. 245–262, 2023. [Online]. Available: https://doi.org/10.1108/ITP-06-2021-0478

[15] S. Barth, M. D. de Jong, M. Junger, P. H. Hartel, and J. C. Roppelt, "Putting the privacy paradox to the test: Online privacy and security behaviors among users with technical knowledge, privacy awareness, and financial resources," *Telematics and Informatics*, vol. 41, pp. 55–69, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736585317307724

[16] Google, "Desarrolladores de android. documentos. guías. prácticas recomendadas sobre privacidad," Desarrolladores de Android. Documentos. Guías., disponible en https://developer.android.com/training/articles/security-tips?hl=es-419#UserData (última visita: 9/3/2023).

[17] R. Mayrhofer, J. V. Stoep, C. Brubaker, and N. Kralevich, "The android platform security model," *ACM Trans. Priv. Secur.*, vol. 24, no. 3, apr 2021. [Online]. Available: https://doi.org/10.1145/3448609

[18] Chen, Kuan-Lin, Yang, and Chung-Huang, "Design and implementation of privacy impact assessment for android mobile devices," 2016.

[19] Y. C.-H. CHEN Kuan-Lin, "Design and implementation of privacy impact assessment for android mobile devices," *ZTE Communications*, vol. 14, no. S0, pp. 37–43, 2016.

[20] K. E. Orjiude and C. O. Yinka-Banjo, "A multilateral privacy impact analysis method for android applications," *Annals of Science and Technology*, vol. 7, no. 2, pp. 1–20, 2022. [Online]. Available: https://doi.org/10.2478/ast-2022-0005

[21] A. Aparicio, M. M. M. González, and V. Cardeñoso, "Métrica basada en grupos de permisos para entender el impacto de las aplicaciones android sobre la privacidad," in *2022 17th Iberian Conference on Information Systems and Technologies (CISTI)*, 2022, pp. 1–5.

[22] G. Shrivastava, P. Kumar, D. Gupta, and J. J. P. C. Rodrigues, "Privacy issues of android application permissions: A literature review,"

Trans. Emerg. Telecommun. Technol., vol. 31, no. 12, 2020. [Online]. Available: https://doi.org/10.1002/ett.3773

[23] M. S. Saleem, J. Mišić, and V. B. Mišić, "Android malware detection using feature ranking of permissions," 2022.

[24] R. Li, W. Diao, Z. Li, J. Du, and S. Guo, "Android custom permissions demystified: From privilege escalation to design shortcomings," in *2021 2021 IEEE Symposium on Security and Privacy (SP)*. Los Alamitos, CA, USA: IEEE Computer Society, may 2021, pp. 70–86. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/SP40001.2021.00070

[25] Y. Wang, J. Zheng, C. Sun, and S. Mukkamala, "Quantitative security risk assessment of android permissions and applications," in *Data and Applications Security and Privacy XXVII*, L. Wang and B. Shafiq, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 226–241.

[26] Á. Feal, P. Calciati, N. Vallina-Rodriguez, C. Troncoso, and A. Gorla, "Angel or devil? A privacy study of mobile parental control apps," *Proc. Priv. Enhancing Technol.*, vol. 2020, no. 2, pp. 314–335, 2020. [Online]. Available: https://doi.org/10.2478/popets-2020-0029

[27] A. Khatoon and P. M. Corcoran, "Android permission system and user privacy — a review of concept and approaches," *2017 IEEE 7th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, pp. 153–158, 2017.

[28] Y. Wang, J. Zheng, C. Sun, and S. Mukkamala, "Quantitative security risk assessment of android permissions and applications," in *Database Security*, 2013.

[29] A. para desarrolladores, "Permisos android." [Online]. Available: https://developer.android.com/reference/android/Manifest.permission

[30] A. Developers, "Declara el uso de datos de tu app." [Online]. Available: https://developer.android.com/privacy-and-security/declare-data-use?hl=es-419

[31] "Android permissions. all you ever wanted to know about android permissions." [Online]. Available: http://androidpermissions.com/

[32] A. Developers, "Context." [Online]. Available: https://developer.android.com/reference/android/content/Context