Towards formal model-based analysis and testing of Android's security mechanisms

Gustavo Betarte, Juan Campo, Maximiliano Cristiá,
Felipe Gorostiaga, Carlos Luna, Camila Sanz

FING-UDELAR, Uruguay; IMDEA Software Institute, Spain; CIFASIS, Argentina.

7 de septiembre de 2017

Resumen

Resumen

Resumen

Resumen

Resumen

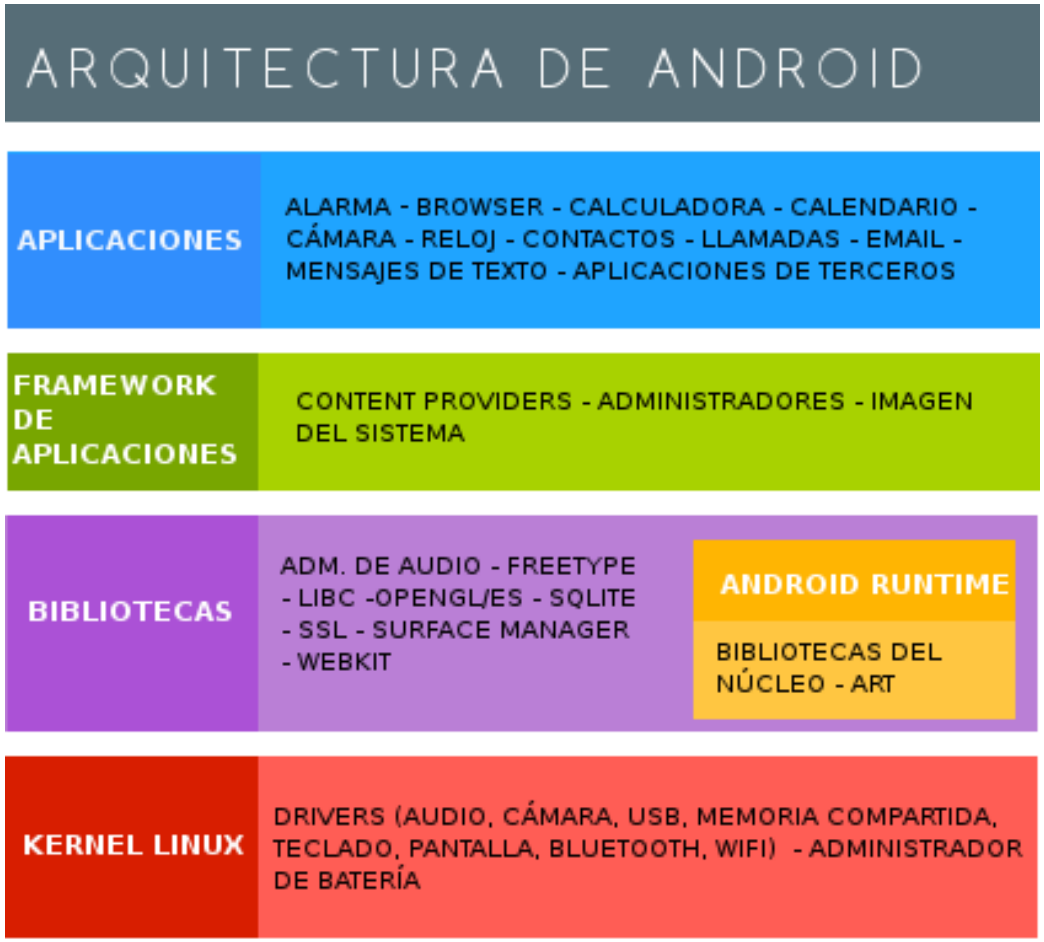
Por qué Android?

- Presente en más de 1000 millones de dispositivos móviles
- Objetivo de numerosos ataques informáticos
- Documentación informal e incompleta

Resumen

Introducción a Android

- Sistema operativo *open-source* para dispositivos móviles
- Desarrollado por Google y la Open Handset Alliance (OHA)



Introducción a Android

- Dos grupos de aplicaciones:
  1. Aplicaciones ya instaladas en la distribución de Android
     Ejemplo. Reloj, Libreta de Contactos
  2. Aplicaciones nuevas creadas por desarrolladores
     Ejemplo. Whatsapp, Facebook

Introducción a Android

- Dos grupos de aplicaciones:
  1. Aplicaciones ya instaladas en la distribución de Android
     Ejemplo. Reloj, Libreta de Contactos
  2. Aplicaciones nuevas creadas por desarrolladores
     Ejemplo. Whatsapp, Facebook
- Ambos tipos de aplicaciones pueden usar los recursos/servicios del teléfono móvil y de otras aplicaciones
- Las aplicaciones nuevas son desarrolladas mayormente en Java mediante el Android Software Development Kit (SDK)

Componentes de una Aplicación

- **Actividades**
  - Pantallas de la aplicación
  - Manejan la interacción con el usuario
- **Content Providers**
  - Comparten datos entre aplicaciones
  - Interfaz entre datos y aplicaciones externas
- **Servicios**
- **Broadcast Receivers**

Componentes de una Aplicación

- **Actividades**
  - Pantallas de la aplicación
  - Manejan la interacción con el usuario
- **Content Providers**
  - Comparten datos entre aplicaciones
  - Interfaz entre datos y aplicaciones externas
- **Servicios**
- **Broadcast Receivers**

Componentes de una Aplicación

- **Actividades**
  - Pantallas de la aplicación
  - Manejan la interacción con el usuario
- **Content Providers**
  - Comparten datos entre aplicaciones
  - Interfaz entre datos y aplicaciones externas
- **Servicios**
- **Broadcast Receivers**

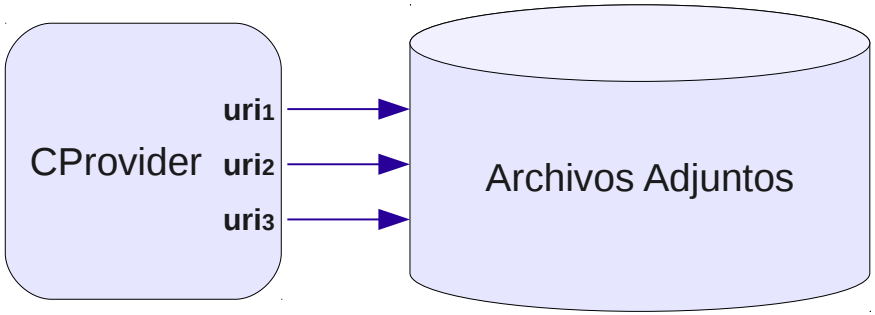
Componentes de una Aplicación: Ejemplo

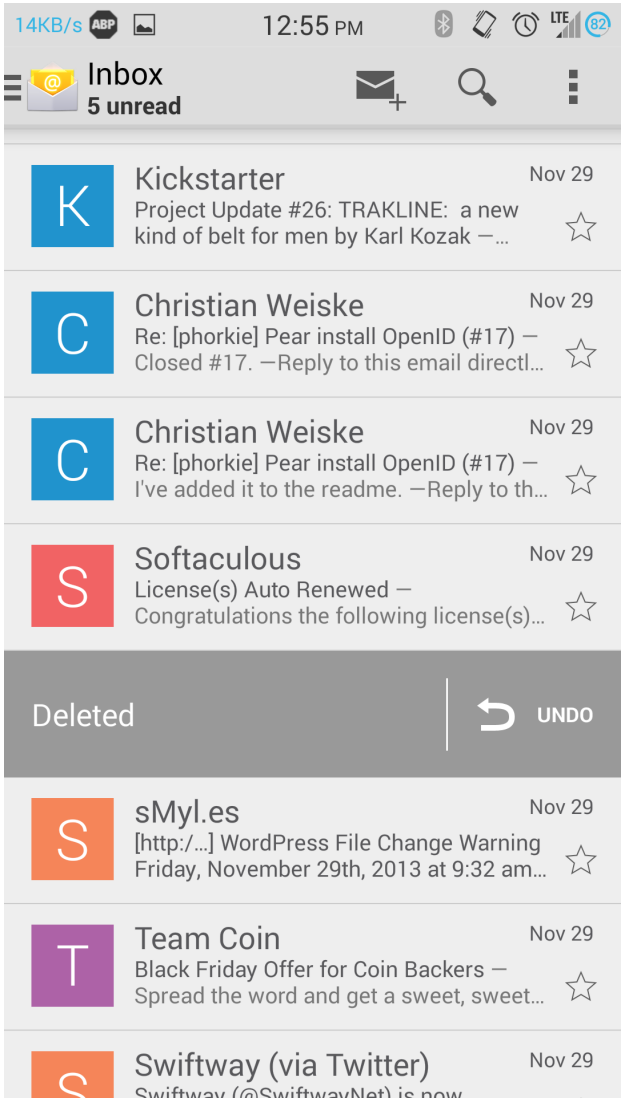
Actividades

- Bandeja de entrada
- Nuevo correo

Content Providers

- Archivos adjuntos





Comunicación entre Componentes

- Acceso a *Content Providers*:
  - Consultas
  - *URI*s
- Acceso a los demás componentes:
  - *Intents*

El Modelo de Seguridad de Android

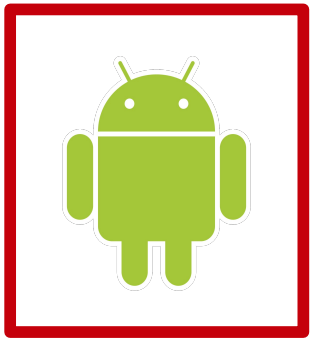
El acceso al dispositivo móvil debe estar regulado para preservar:

- La integridad y confidencialidad de los datos
- El control de costos por parte del usuario
- El correcto funcionamiento del sistema
- ...

Principio de Mínimo Privilegio

Principio de Mínimo Privilegio

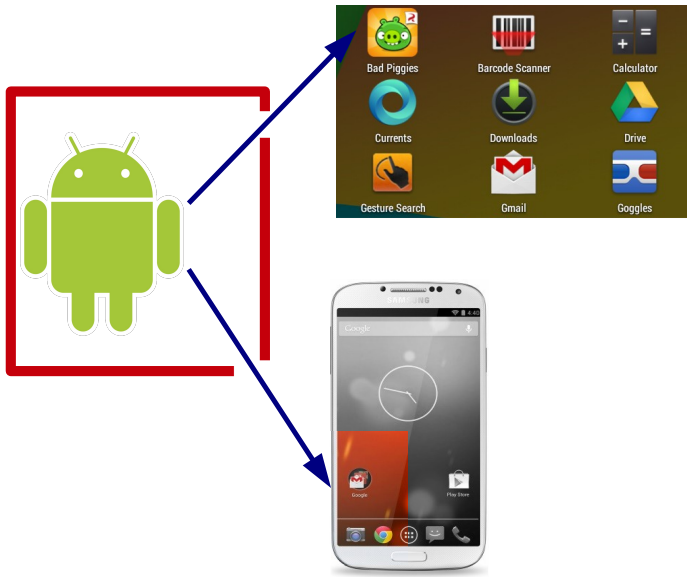
- **Application Sandbox**



Principio de Mínimo Privilegio

- **Application Sandbox**

- **Sistema de Permisos**



AndroidManifest

- Archivo XML que debe incluir toda aplicación Android
- Se declaran *estáticamente*:
  1. Permisos solicitados
  2. Permisos exigidos
  3. ...
- Al instalar una aplicación se decide si se conceden los permisos solicitados (dependiendo del tipo y la versión)

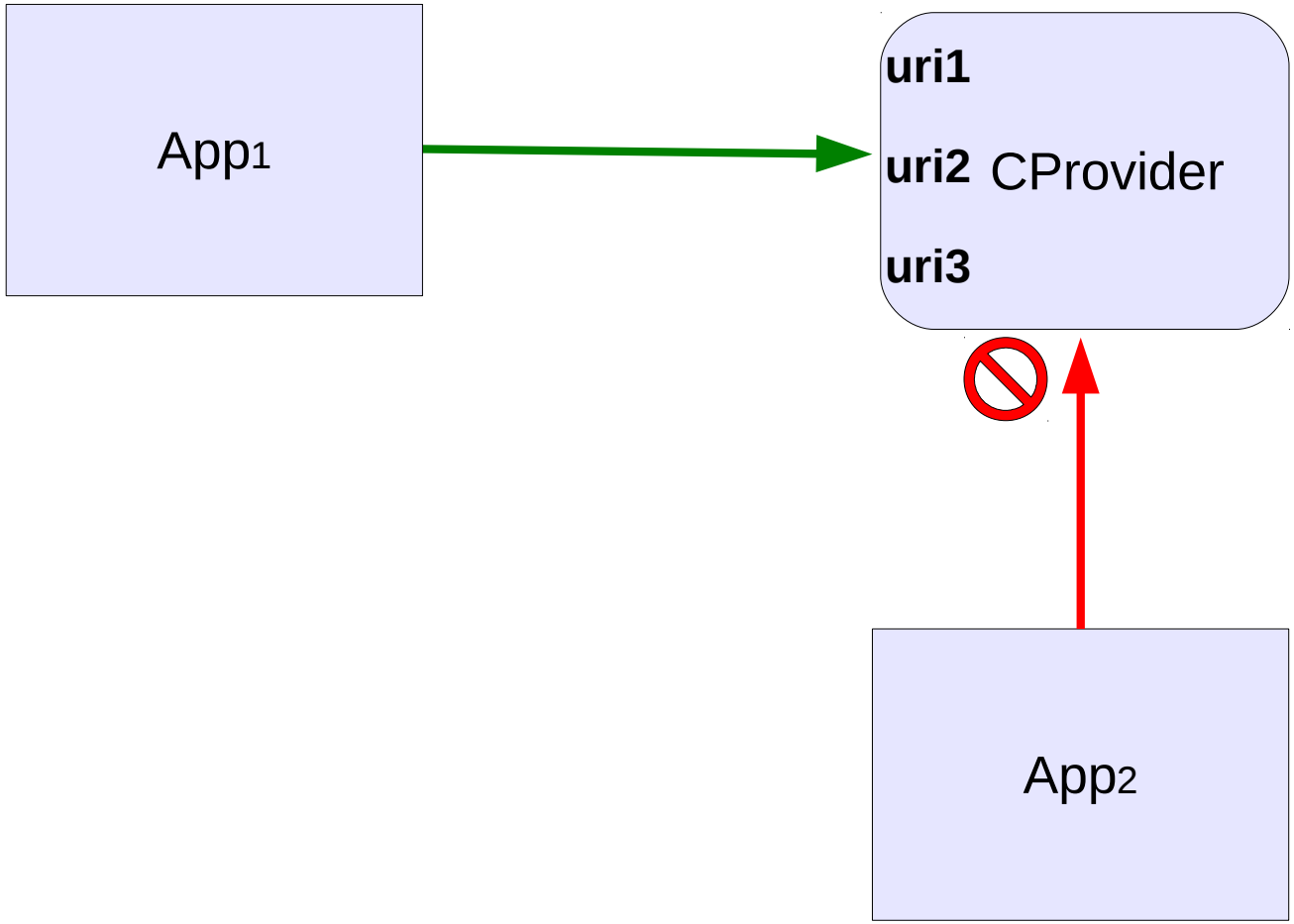
AndroidManifest: Ejemplo

```
<manifest package="com.cpexample" ... >

    :

    <uses-permission android:name="android.permission.SEND_SMS" />

    <application
        android:permission="android.permission.SET_WALLPAPER" ... >

        <activity
            android:name="com.cpexample.MainActivity"
            android:permission="android.permission.CALL_PHONE" ... >
        </activity>

        <provider android:name="com.cpexample.MiProvider"
            android:permission="android.permission.SEND_SMS" ... >

        </provider>

        :

    </application>

</manifest>
```

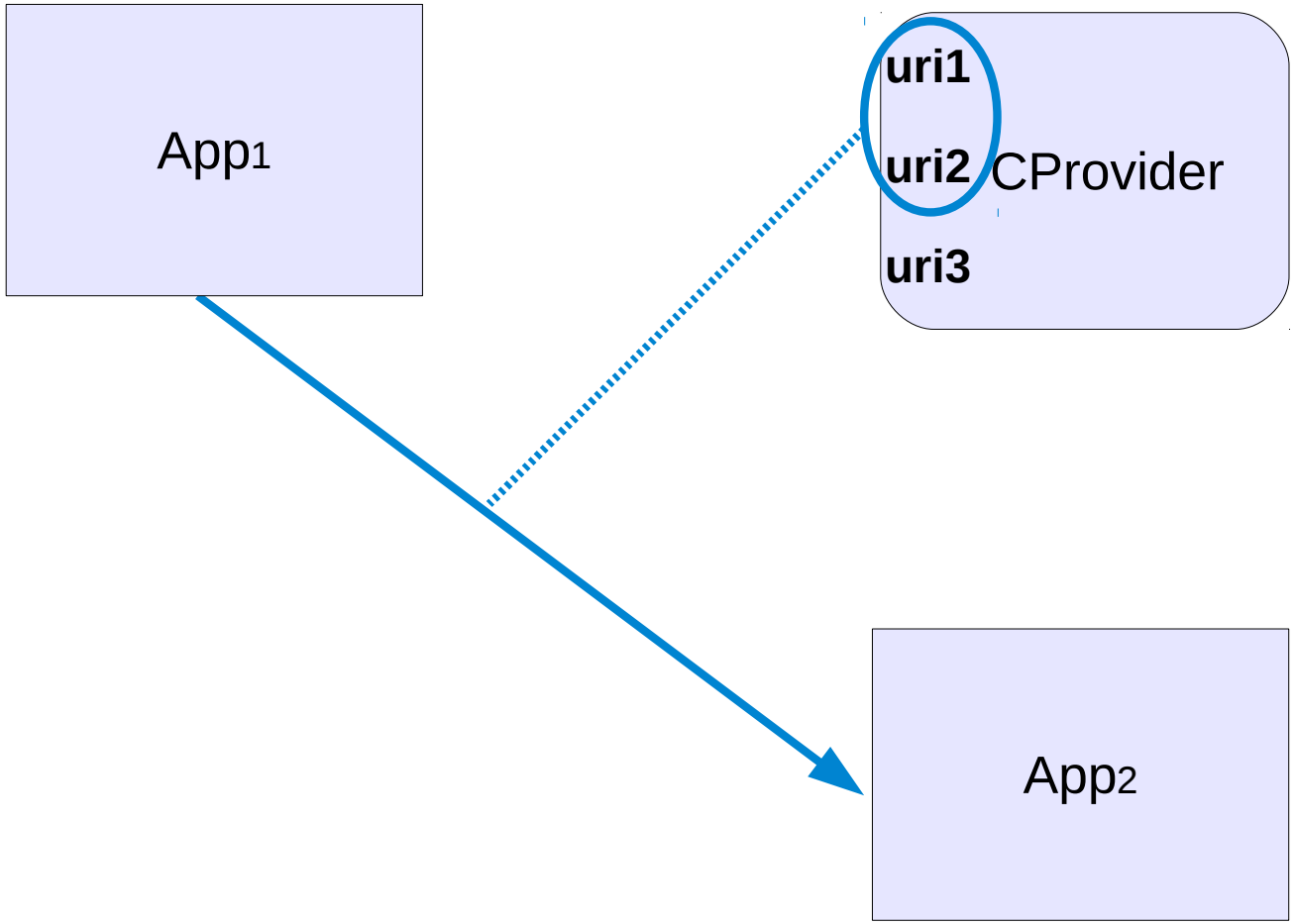
Delegación de Permisos

- Concesión de permisos entre aplicaciones
- Permisos vigentes hasta su revocación
- Dos mecanismos de delegación:
  - *Pending intents*
  - *URI permissions*

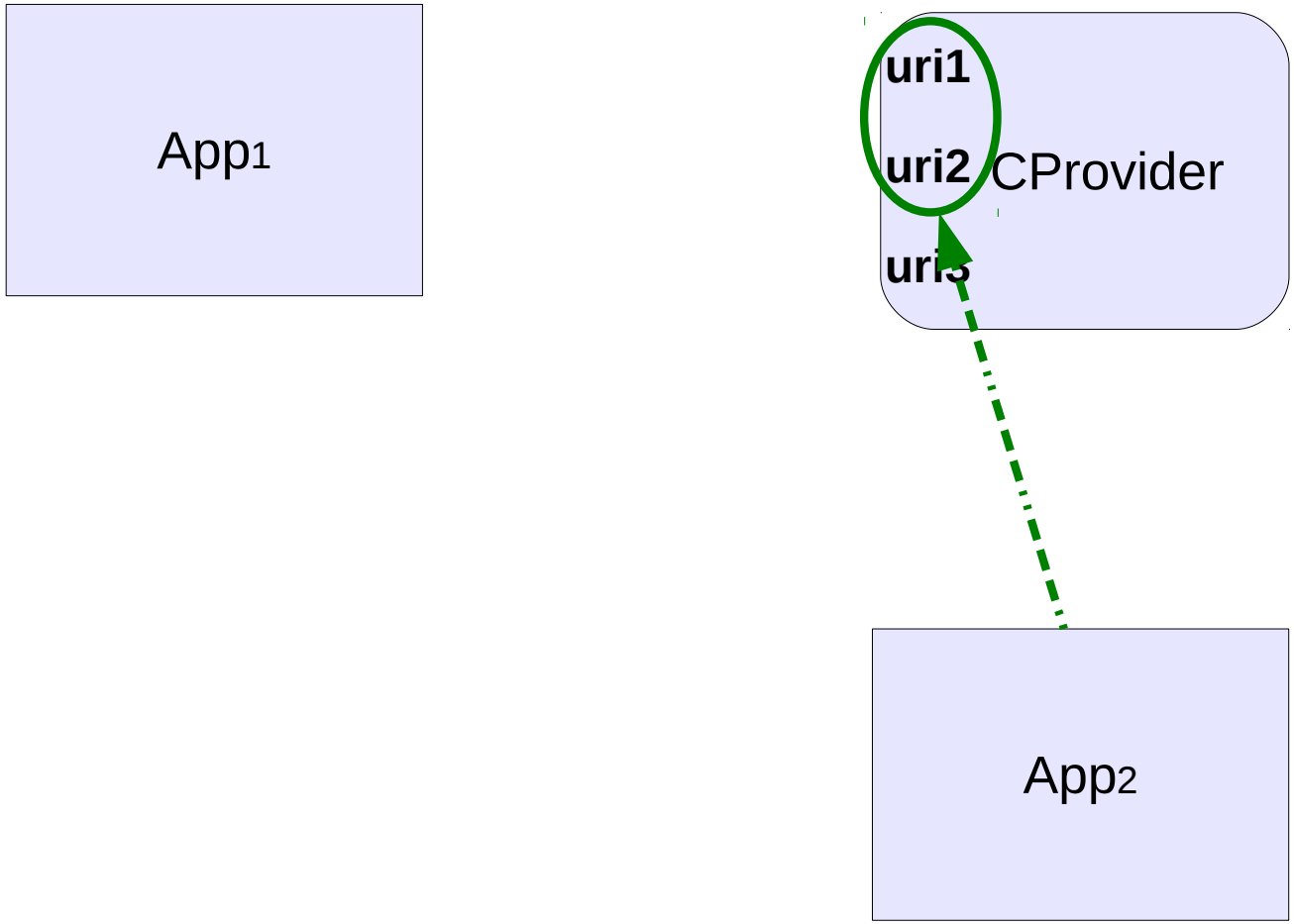
Delegación de Permisos: *URI permissions*



Delegación de Permisos: *URI permissions*



Delegación de Permisos: *URI permissions*



Resumen

Características Generales

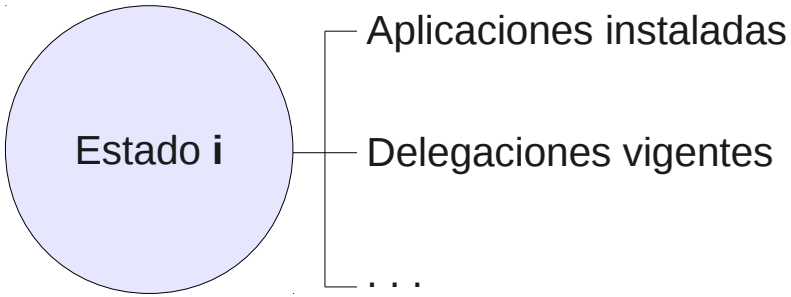
- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados

Características Generales

- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados

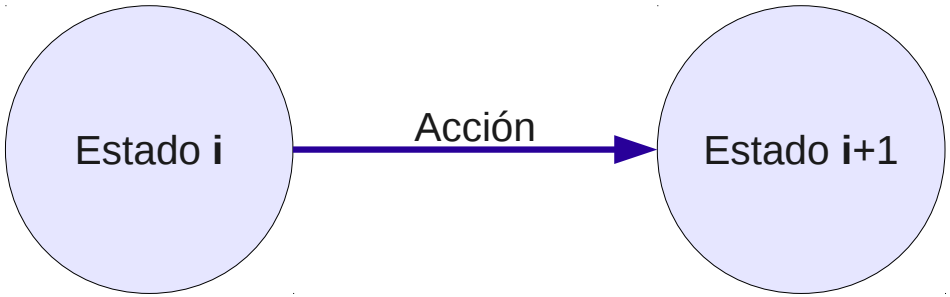
Características Generales

- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados



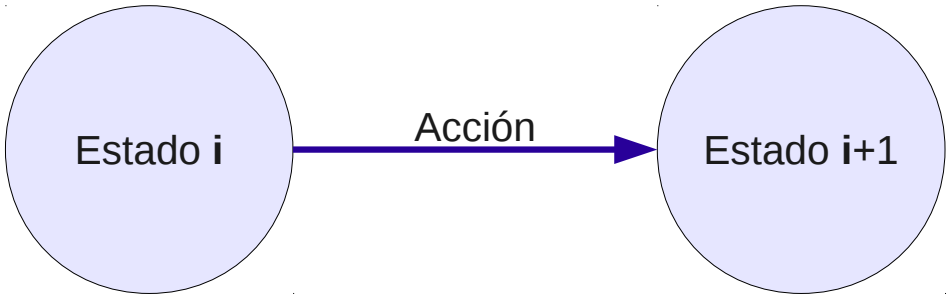
Características Generales

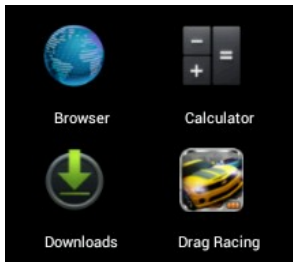
- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados



Características Generales

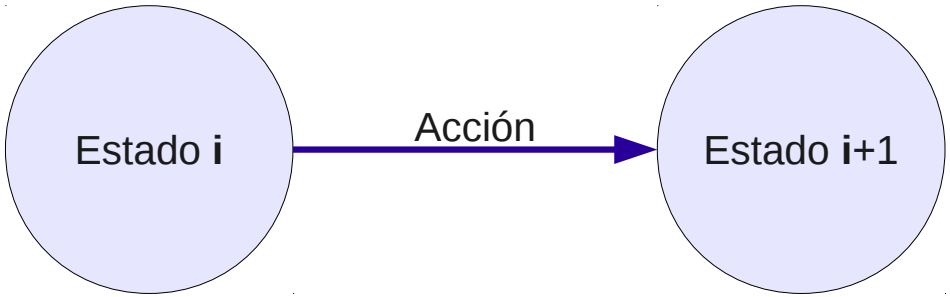
- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados





Características Generales

- Formalización del modelo de seguridad de Android
- Desarrollada en el asistente de pruebas Coq
- Especial atención en:
  - Sistema de permisos
  - Interacción entre aplicaciones y el sistema
- Especificación de alto orden basada en máquinas de estados



Estado del Sistema

$$
\begin{aligned}
\mathit{InstApps} &::= \{\mathit{AppId}\} \\
\mathit{GrantedGroups} &::= \{\mathit{AppId} \times \{\mathit{PermGrp}\}\} \\
\mathit{AppsPerms} &::= \{\mathit{AppId} \times \{\mathit{Perm}\}\} \\
\mathit{CompInsRunning} &::= \{\mathit{CompInstance}\} \\
\mathit{OpType} &::= \mathit{read} \mid \mathit{write} \mid \mathit{rw} \\
\mathit{DelPPerms} &::= \{\mathit{AppId} \times \mathit{ContProv} \times \mathit{Uri} \times \mathit{OpType}\} \\
\mathit{DelTPerms} &::= \{\mathit{iComp} \times \mathit{ContProv} \times \mathit{Uri} \times \mathit{OpType}\} \\
\mathit{AppsResCont} &::= \{\mathit{AppId} \times \mathit{Res} \times \mathit{ResVal}\} \\
\mathit{SentIntents} &::= \{\mathit{iComp} \times \mathit{Intent}\} \\
\mathit{AppsManifest} &::= \{\mathit{AppId} \times \mathit{Manifest}\} \\
\mathit{AppsCert} &::= \{\mathit{AppId} \times \mathit{Cert}\} \\
\mathit{AppsDefPerms} &::= \{\mathit{AppId} \times \{\mathit{Perm}\}\} \\
\mathit{ImageApps} &::= \{\mathit{App}\} \\
\mathit{AndroidState} &::= \mathit{InstApps} \times \mathit{GrantedGroups} \times \mathit{AppsPerms} \times \mathit{DelTPerms} \times \\
&\quad \mathit{CompInsRunning} \times \mathit{DelPPerms} \times \mathit{AppsResCont} \times \mathit{SentIntents} \times \\
&\quad \mathit{AppsManifest} \times \mathit{AppsCert} \times \mathit{AppsDefPerms} \times \mathit{ImageApps}
\end{aligned}
$$

Un estado es *válido* si los identificadores de las aplicaciones instaladas son únicos ...

Algunas Acciones

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)
- **Lectura** a través de un *content provider* (`read`)
- **Escritura** a través de un *content provider* (`write`)

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)
- **Lectura** a través de un *content provider* (`read`)
- **Escritura** a través de un *content provider* (`write`)
- **Delegación temporal** de permisos (`grantT`)
- **Delegación permanente** de permisos (`grantP`)

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)
- **Lectura** a través de un *content provider* (`read`)
- **Escritura** a través de un *content provider* (`write`)
- **Delegación temporal** de permisos (`grantT`)
- **Delegación permanente** de permisos (`grantP`)
- **Revocación** de permisos delegados (`revoke`)

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)
- **Lectura** a través de un *content provider* (`read`)
- **Escritura** a través de un *content provider* (`write`)
- **Delegación temporal** de permisos (`grantT`)
- **Delegación permanente** de permisos (`grantP`)
- **Revocación** de permisos delegados (`revoke`)
- **Llamada** a la API del sistema (`call`)
- .....

Algunas Acciones

- **Instalación** de una aplicación (`install`)
- **Desinstalación** de una aplicación (`uninstall`)
- **Inicio** de ejecución de un componente (`start`)
- **Fin** de ejecución de un componente (`stop`)
- **Lectura** a través de un *content provider* (`read`)
- **Escritura** a través de un *content provider* (`write`)
- **Delegación temporal** de permisos (`grantT`)
- **Delegación permanente** de permisos (`grantP`)
- **Revocación** de permisos delegados (`revoke`)
- **Llamada** a la API del sistema (`call`)
- .....

Semántica expresada utilizando pre y postcondiciones

Ejecución de Acciones

La transición de estados es representada por la relación $\hookrightarrow$:

$$\frac{validState\ s \qquad Pre\ s\ act \qquad Post\ s\ act\ s'}{s \xhookrightarrow{act,ok} s'}$$

$$\frac{validState\ s \qquad ErrorMsg\ s\ act\ err}{s \xhookrightarrow{act,err} s}$$

Ejecución de Acciones

La transición de estados es representada por la relación $\hookrightarrow$:

$$\frac{validState\ s \qquad Pre\ s\ act \qquad Post\ s\ act\ s'}{s \xhookrightarrow{act,ok} s'}$$

$$\frac{validState\ s \qquad ErrorMsg\ s\ act\ err}{s \xhookrightarrow{act,err} s}$$

$$\boxed{s_i \xhookrightarrow{act,r} s_{i+1}}$$

Ejecución de Acciones

La transición de estados es representada por la relación $\hookrightarrow$:

$$\frac{validState\ s \qquad Pre\ s\ act \qquad Post\ s\ act\ s'}{s \xhookrightarrow{act,ok} s'}$$

$$\frac{validState\ s \qquad ErrorMsg\ s\ act\ err}{s \xhookrightarrow{act,err} s}$$

$$\boxed{s_i \xhookrightarrow{act,r} s_{i+1}}$$

$$s_0 \xhookrightarrow{act_1,r_1} s_1 \xhookrightarrow{act_2,r_2} \cdots \xhookrightarrow{act_{n-1},r_{n-1}} s_{n-1} \xhookrightarrow{act_n,r_n} s_n$$

Resumen

Propiedades de Seguridad

Propiedades básicas

Ejemplo: invarianza de la validez de estado

Propiedades de Seguridad

Propiedades básicas

Ejemplo: invarianza de la validez de estado

Propiedades deseables

Ejemplo: principio del mínimo privilegio

Propiedades de Seguridad

Propiedades básicas

Ejemplo: invarianza de la validez de estado

Propiedades deseables

Ejemplo: principio del mínimo privilegio

Propiedades no deseables

Ejemplo: escalada de privilegios

Propiedades de Seguridad

Propiedades básicas

Ejemplo: invarianza de la validez de estado

Propiedades deseables

Ejemplo: principio del mínimo privilegio

Propiedades no deseables

Ejemplo: escalada de privilegios

Propiedades mitigadoras

Ejemplo: para *eavesdropping*,*intent spoofing*

Eavesdropping

- Monitoreo no autorizado de información.
- En Android: cuando se mandan mensajes públicos de tipo broadcast con información sensible.

Eavesdropping

- Monitoreo no autorizado de información.
- En Android: cuando se mandan mensajes públicos de tipo broadcast con información sensible.
- Proteger mensajes dirigidos broadcast receivers que contienen información sensible.

Eavesdropping

- Monitoreo no autorizado de información.
- En Android: cuando se mandan mensajes públicos de tipo broadcast con información sensible.
- Proteger mensajes dirigidos broadcast receivers que contienen información sensible.

Lema

Si un componente c perteneciente a una aplicación a envía un *intent* de tipo *broadcast* protegido por un permiso de tipo *signature* o *signature or system* entonces si a' no tiene el mismo certificado que a, no podrá recibirlo.

Intent Spoofing

- Tomar ventaja de un bug, una falla de diseño o de configuración.

Intent Spoofing

- Tomar ventaja de un bug, una falla de diseño o de configuración.
- En Android: malas configuraciones de las aplicaciones, cuando, por ejemplo, no se explicita el atributo `exported` (del `manifest`).

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          package="com.example.android.wearable.timer" >

    <application>

        <!-- Timer components -->
        <activity android:name=".SetTimerActivity">

            <intent-filter>
                <action android:name="com.android.example.clockwork.timer.TIMER"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>

        </activity>


    </application>
</manifest>
```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          package="com.example.android.wearable.timer" >

    <application>

        <!-- Timer components -->
        <activity android:name=".SetTimerActivity">

        </activity>


    </application>
</manifest>
```

Intent Spoofing

- Chequeo estático del *manifest*. Si el atributo `exported` de una aplicación es falso o si el atributo `exported` no está presente y no se declara ningún elemento de tipo `<intent-filter>` en su *manifest*, entonces la aplicación no podrá ser iniciada por terceros.

Lema

Si un componente c no puede ser iniciado por terceros, entonces la aplicación que lo contiene no podrá recibir un *intent* dirigido a c.

Desarrollo y uso de una implementación certificada

Implementación certificada

- Implementación de funciones Coq para las acciones especificadas.

Desarrollo y uso de una implementación certificada

Implementación certificada

- Implementación de funciones Coq para las acciones especificadas.
- Prueba de corrección: las funciones implementan las relaciones de ejecución para cada acción.

Desarrollo y uso de una implementación certificada

Implementación certificada

- Implementación de funciones Coq para las acciones especificadas.
- Prueba de corrección: las funciones implementan las relaciones de ejecución para cada acción.
- Extracción de un programa Haskell (*dispatcher* de comandos) correcto por construcción.

Desarrollo y uso de una implementación certificada

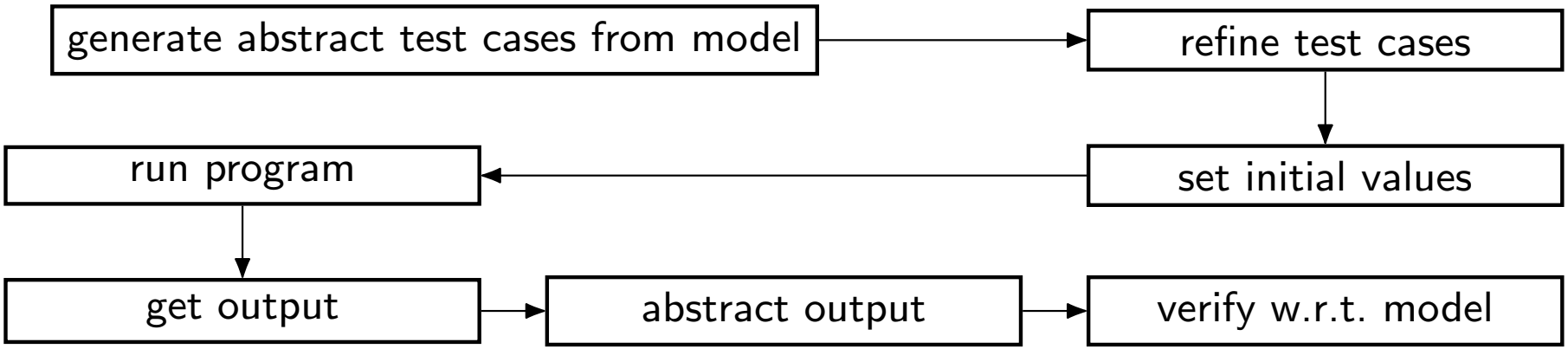
Implementación certificada

- Implementación de funciones Coq para las acciones especificadas.
- Prueba de corrección: las funciones implementan las relaciones de ejecución para cada acción.
- Extracción de un programa Haskell (*dispatcher* de comandos) correcto por construcción.

Uso de la implementación certificada como un *oráculo*

Generación de casos de test para un sistema Android real a partir del modelo usando la técnica *model-based testing*, incorporando el uso del *oráculo*.

Un proceso de testing basado en modelos



Conclusiones

- Especificación formal exhaustiva de la versión actual del modelo de seguridad de Android
- + Formulación y demostración de diferentes tipos de propiedades de seguridad
- + Desarrollo y uso de un prototipo certificado del modelo
- = 25k LOC of Coq

¿Preguntas?

¿Preguntas?

¡Gracias!

¡Gracias!