Segurança em dispositivos Android

Priscila P. Apocalypse
Inatel - Eng. da Computação 2011

Dev e Sec - SIDI – Instituto Samsung
(desde 2013)

Visão Geral

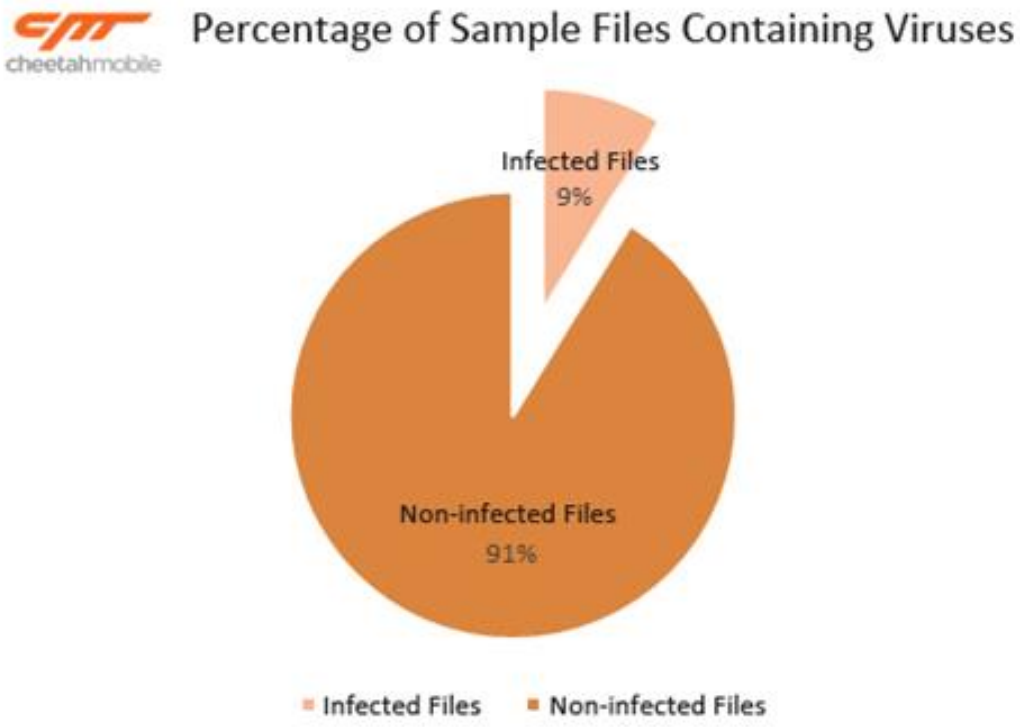
- Porque devemos entender o modelo de segurança?
- Como funcionam os aplicativos Android
- Ciclo de vida dos aplicativos
- Camada de aplicação - Niveis de proteção
- Como podemos “hackear” um app
- Drozer: Demonstração
- Como se proteger

Porque devemos entender o modelo de segurança?

- Os celulares são bem atrativos para ataques:
  - Pessoas armazenam muitas informações pessoais nos celulares : email, contatos, fotos, etc…
  - Informações organizacionais também estão sendo armazenadas
  - Fácil de perder ou ser roubado
  - Aplicações que realizam compras/transações bancárias

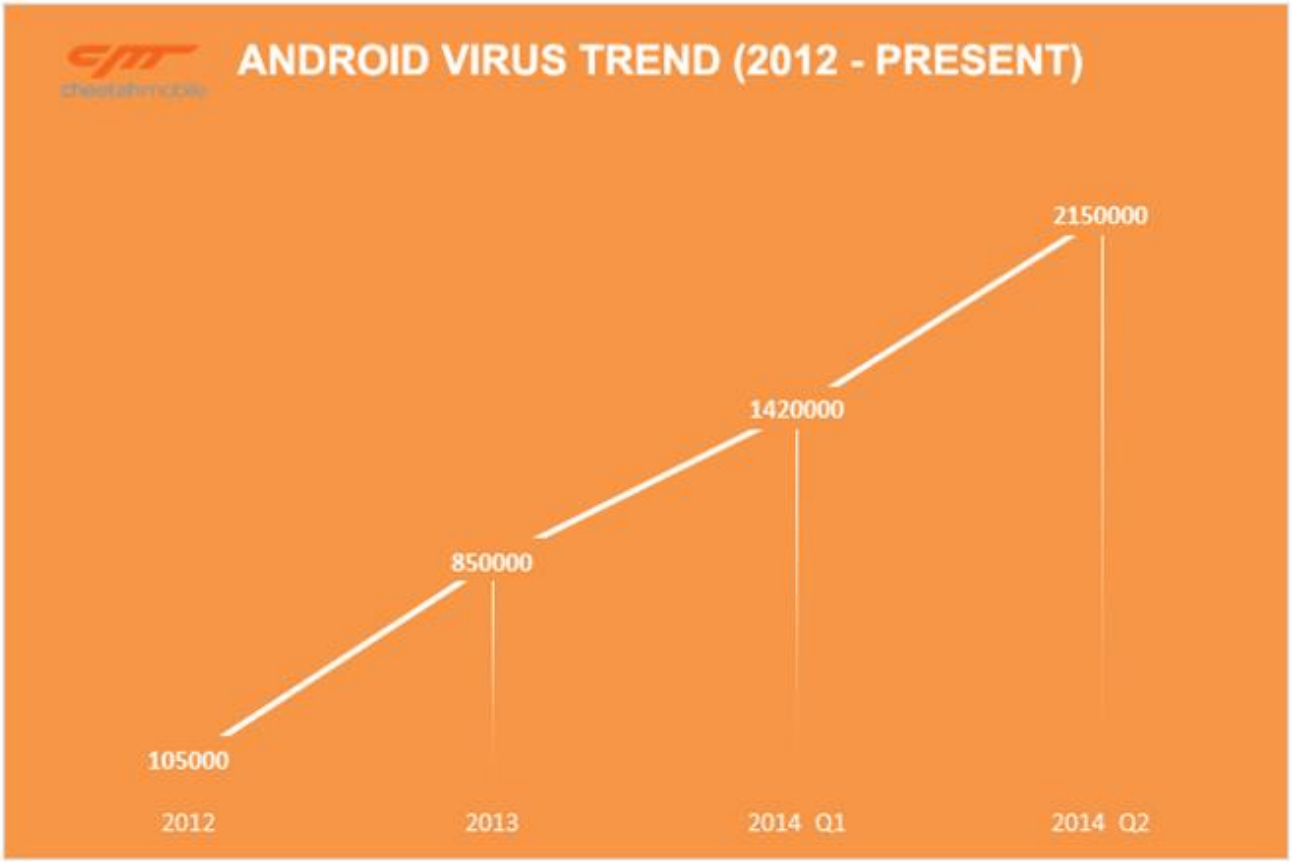
Porque devemos entender o modelo de segurança?

- 1 a cada 10 apps no Android é classificado como malicioso (Total de 24,4 milhões de amostras)



Porque devemos entender o modelo de segurança?

- Aumento de 600% nos últimos anos



Porque devemos entender o modelo de segurança?

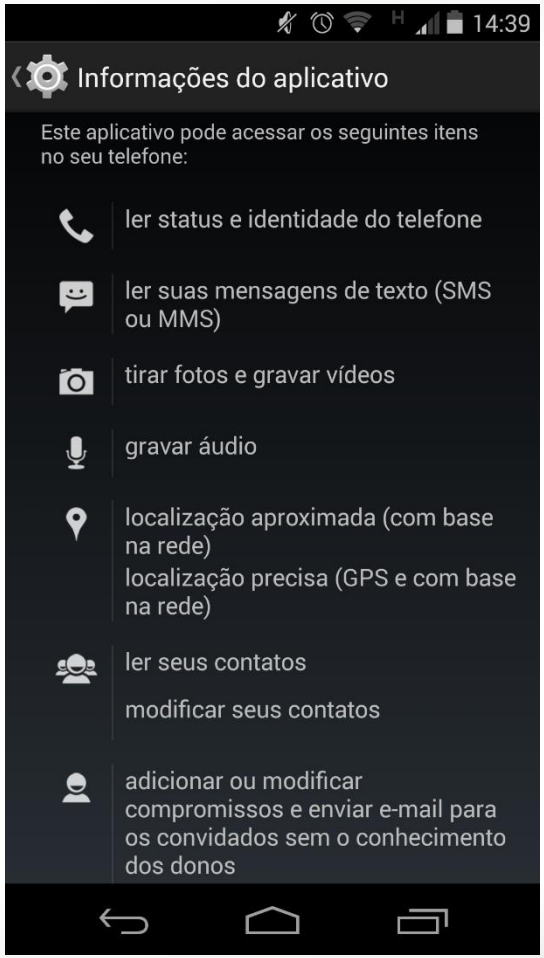
Brecha de teclado SwiftKey da Samsung deixa 600 milhões vulneráveis

A brecha estava no pacote de atualização de idiomas do SwiftKey (teclado pré-instalado em smartphones Samsung)

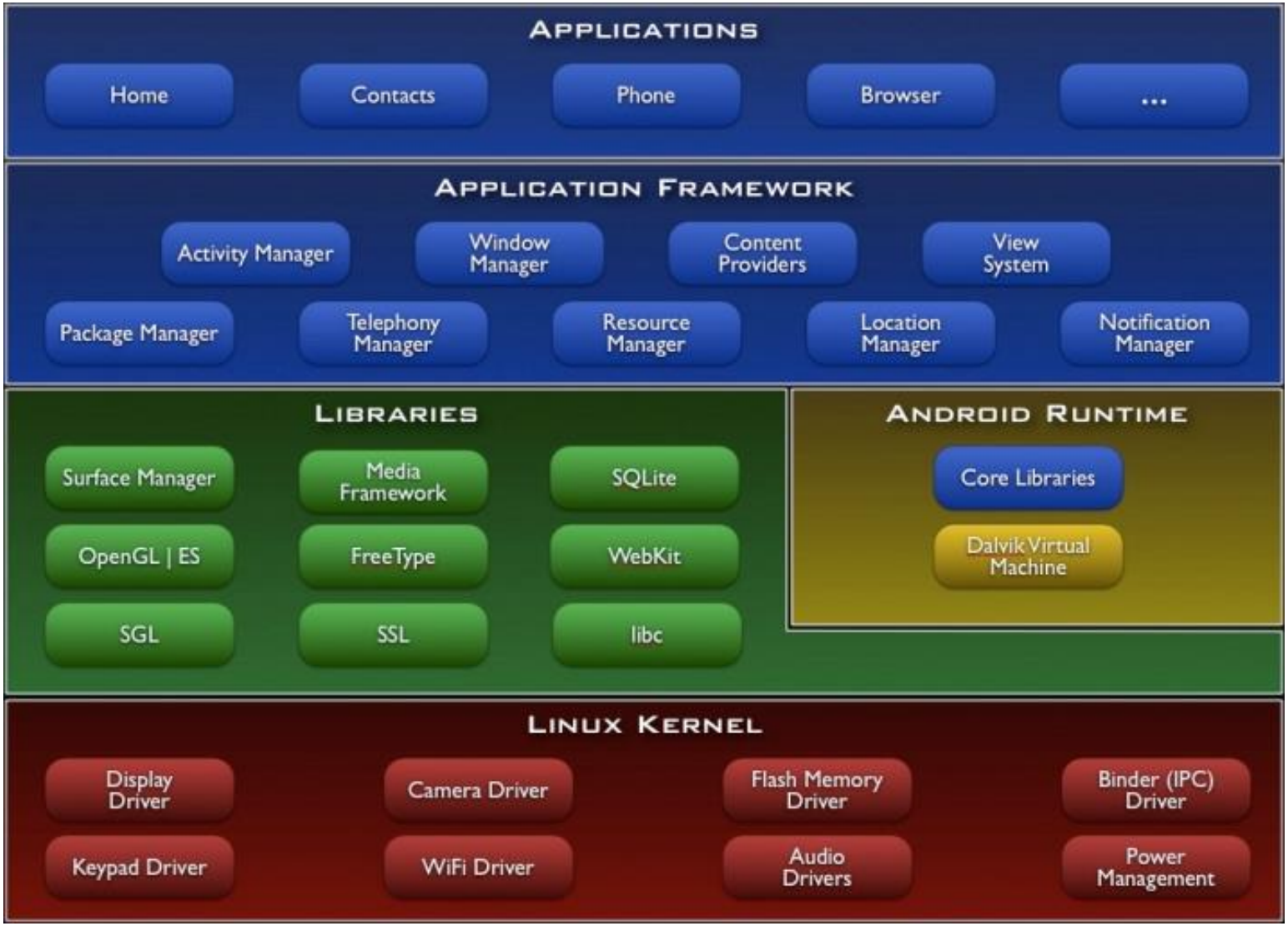
SEGURANÇA

Porque devemos entender o modelo de segurança?

- Plataforma Open Source
- **Você controla as permissões do seu dispositivo!!**
- **Facebook** app ao lado ->



Arquitetura do SO Android



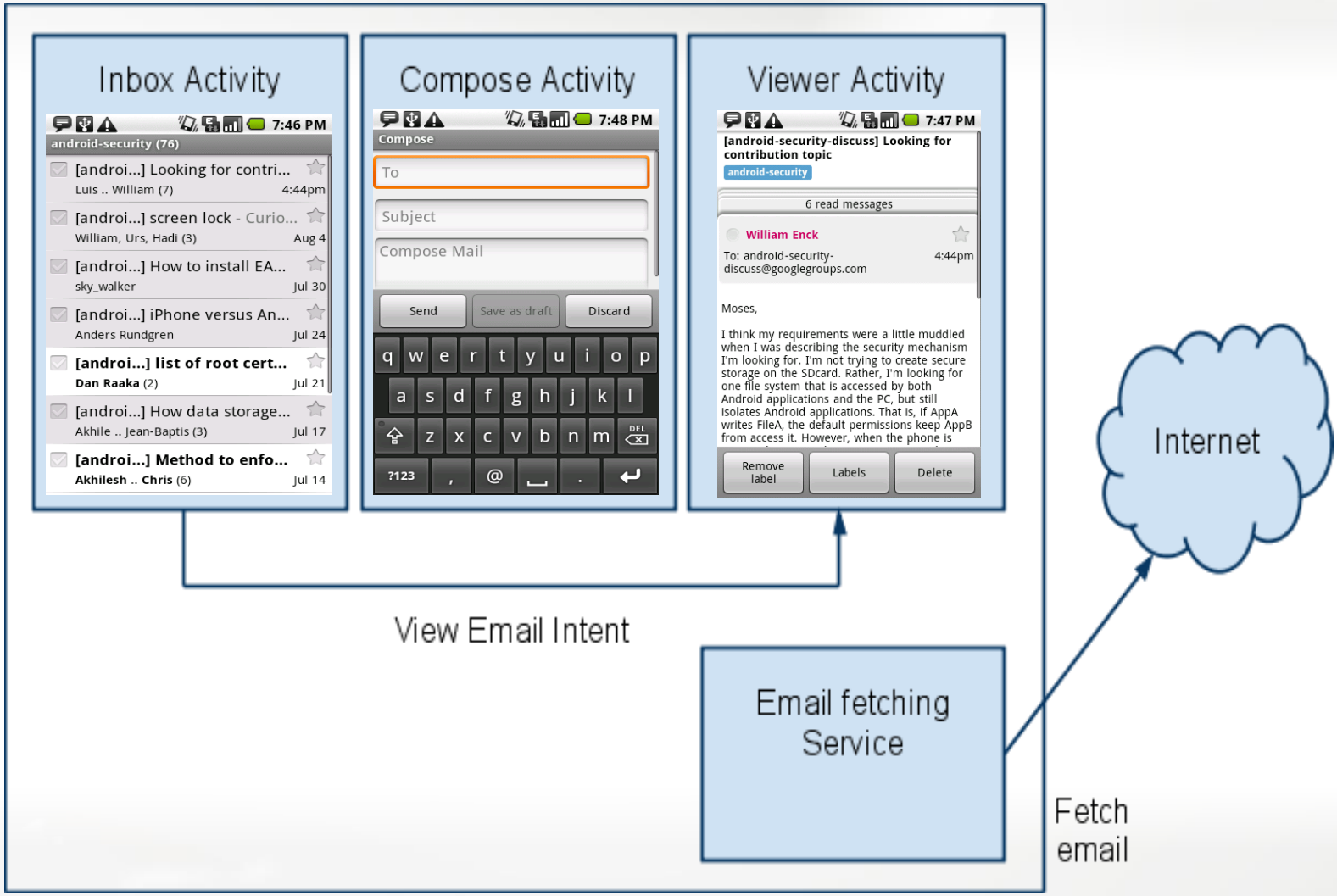
http://developer.android.com/guide/basics/what-is-android.html

Como funcionam os aplicativos Android?

- **Activity**: Define as telas, interface visual
- **Service**: Processo em background
- **Broadcast Receiver**: Recebe “msgs” de outras aplicações
- **Content Provider**: Banco de dados relacional (ou arquivo) para compartilhamento de dados entre os apps
- Comunicação entre processos (IPC):

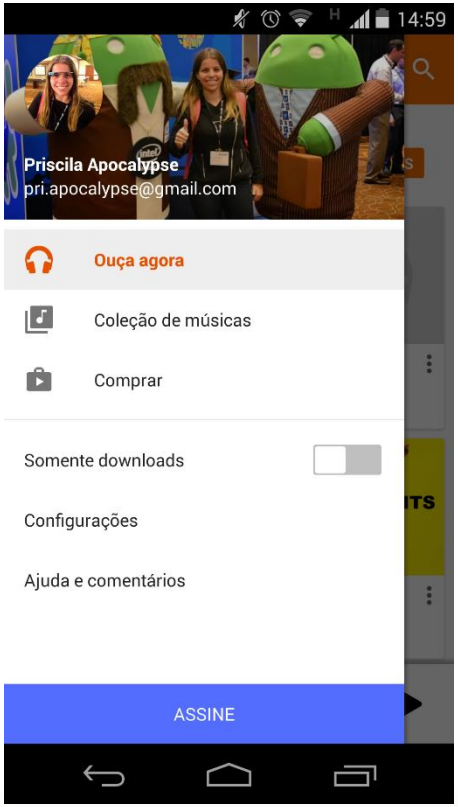
Intent: Um objeto de mensagem que pode ser usado para solicitar uma ação para outro componente do aplicativo e outros aplicativos

Como funcionam os aplicativos Android?



Ciclo de vida dos aplicativos

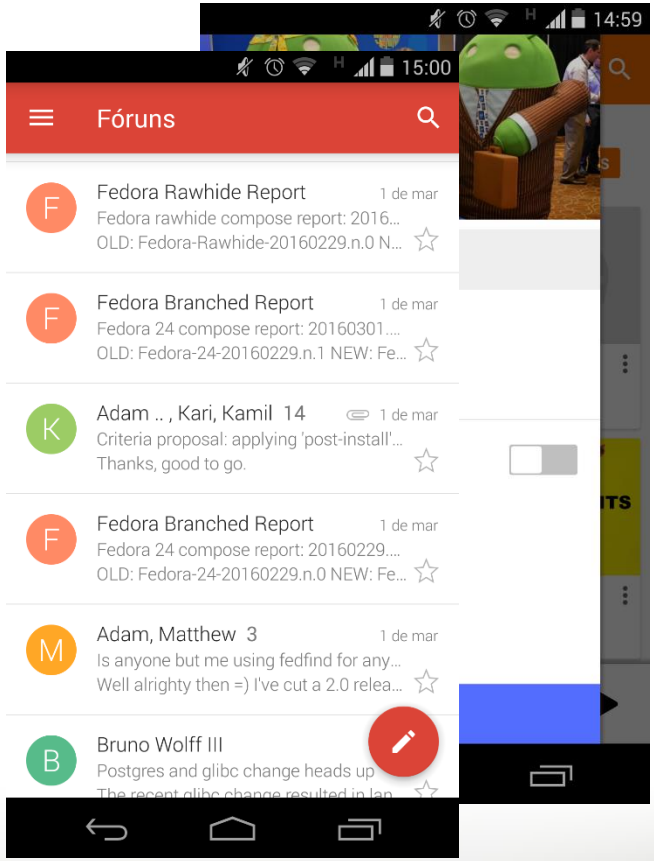
- Desenvolvidos para economizar bateria



Ciclo de vida dos aplicativos

- Desenvolvidos para economizar bateria
- Activities são chamadas,

Tomando prioridades na “pilha”



Ciclo de vida dos aplicativos

- Desenvolvidos para economizar bateria
- Activities são chamadas,

Tomando prioridades na “pilha”

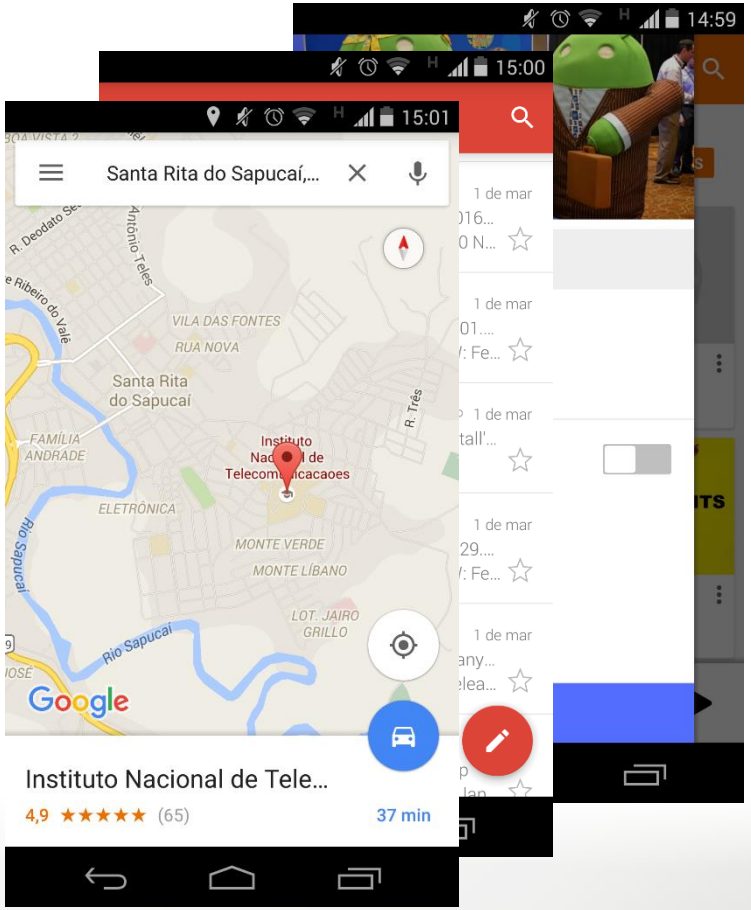
- Activities em background podem

ser encerradas a qualquer momento

- Aqui começam os problemas

de DoS

- DoS = Denial Of Service



Camada de aplicação - Niveis de proteção

- As permissões são definidas em AndroidManifest.xml
- PackageManager e ActivityManager aplicam as permissões
- O usuário aceita estas permissões no ato da instalação

e normalmente ninguém lê! ☹ ☹ ☹

- Agora no Android M e N aparece uma notificação para o usuário, solicitando que aceite ou não

Niveis de proteção

- **NORMAL**

  `android.permission.VIBRATE`

  `com.android.alarm.permission.SET_ALARM`

- **DANGEROUS**

  `android.permission.SEND_SMS`

  `android.permission.CALL_PHONE`

- **SIGNATURE**

  `android.permission.FORCE_STOP_PACKAGES`

  `android.permission.INJECT_EVENTS`

- **SIGNATURE OR SYSTEM**

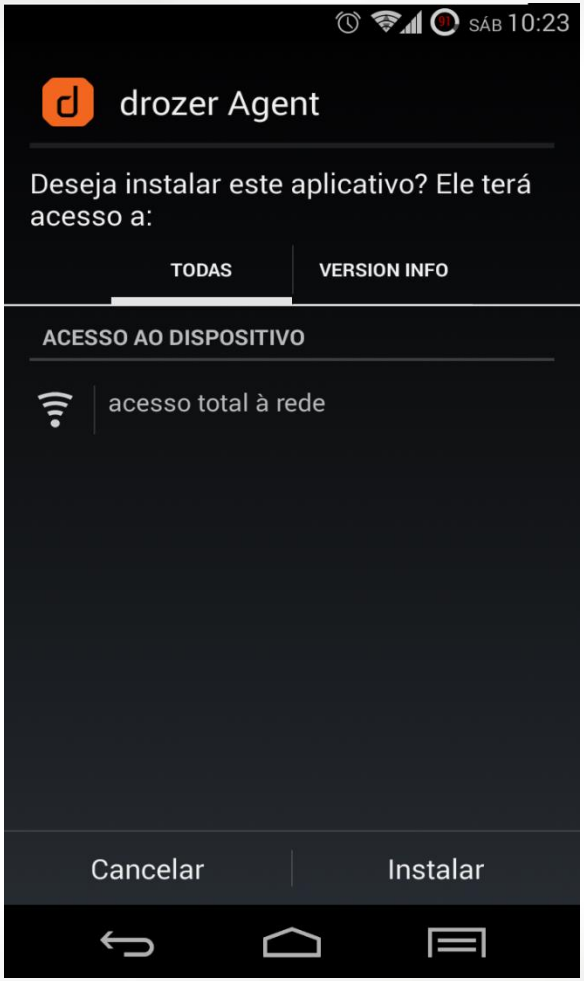
  `android.permission.ACCESS_USB`

  `android.permission.SET_TIME`

Como podemos “hackear” um app

- Existe um conjunto de ferramentas nos ajudam a validar e encontrar estas falhas de segurança
- Introdução ao Drozer https://labs.mwrinfosecurity.com/tools/drozer/
- Cliente/Servidor
- Permissão de acesso total a rede

Demonstração



Como se proteger?

Como se proteger?

- Atualizar sempre o Android
- Evitar fazer root (A não ser que saiba o que está fazendo) ☹☹☹
- Utilizar criptografia
- Preferência para aplicativos que criptografam seus arquivos armazenados na memória interna
- Verificar as permissões antes de instalar um app
- Para o desenvolvedor: Utilizar todas as ferramentas disponíveis para validar e fazer testes de segurança durante o desenvolvimento

Obrigada!

Apresentação,instalação,links e apks em :
https://github.com/priscila225/SemanaDaComputacao