Android Security Lab

Kick-off Meeting (06.10.2014)

Sven Bugiel, M.Sc.
Organizational matters

- Course agenda
- Registration
Organizational matters

Course agenda
Goal: Platform security of modern mobile operating systems at the example of the open-source Android OS

- Basics of secure architectures
- Android’s security principles and architecture
- Modern attack vectors against smartphone operating systems
- Selected security extensions from research
- Implementation of security extensions

Credits: 6 ECTS
**ORGANIZATIONAL: LAB PROCEDURE**

- **First half: Lecture Period** (06.10.14 – 10.10.14)
  - Room E1.1 2.06, 09:30-16:30 s.t. (lunch break 12:00-14:00)
  - Morning session: Lecture-style
  - Afternoon session: Supervised Exercise
    - Tutor: Tobias Theobald

<table>
<thead>
<tr>
<th>Date</th>
<th>09:30 – 11:30</th>
<th>13:30 – 16:30</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/11/2014</td>
<td><strong>Lecture</strong>: Motivation</td>
<td><strong>Lecture</strong>: Secure architecture principles and Android security architecture</td>
</tr>
<tr>
<td></td>
<td><strong>Lecture</strong>: Applications and application Layer</td>
<td></td>
</tr>
<tr>
<td>07/10/2014</td>
<td><strong>Exercise</strong>: Basic application programming</td>
<td><strong>Exercise</strong>: Data sharing and Android security primitives</td>
</tr>
<tr>
<td>08/10/2014</td>
<td><strong>Lecture</strong>: Attacks on Android</td>
<td><strong>Exercise</strong>: Data sharing and Android security primitives (cont.)</td>
</tr>
<tr>
<td>09/10/2014</td>
<td><strong>Lecture</strong>: Selected research works</td>
<td><strong>Exercise</strong>: Extending Android’s middleware and creating a custom ROM</td>
</tr>
<tr>
<td>10/10/2014</td>
<td><strong>Exercise</strong>: Extending Android’s middleware and creating a custom ROM</td>
<td>Optional slot for exercises</td>
</tr>
</tbody>
</table>
**First half: Supervised project** (13.10.2014 – 17.10.2014)

- Room E1.1 2.06, 09:30-16:30 s.t. (lunch break 11:30-13:30)
- Development of an exemplary Android security extension: Access control based domain isolation (private vs. business)
- Introduction of the project: 13.10.2014, 09:30 - 10:30
ORGANIZATIONAL: LAB PROCEDURE

- **Second half: Project period** (20.10.2014 – 14.11.2014)
  - Teams of two students
    - No fixed classes
    - Team building and topic assignment: **Friday 17.10.2014**
    - But: Talk to your TA (me!) and get feedback!
      No meetings = No complaints about grade 😊
      No last minute meetings!
  - Equipment in room E1.1 2.06 can be used or private PC/laptop
    - UdS card required to get access to E1.1 2.06
Grade based on:

- Final report and **code**
  - About 10 pages report + zipped code (or patch file)
  - Clearly stating for which part which team member was responsible
  - General structure of report explained in project proposals sheet
  - **Firm deadline for report: 14.11.2014 23:59**
Separate PDF file with references available on the course website

Recommended literature:


ONLINE REFERENCES

- http://developer.android.com
- http://source.android.com
- Android tutorials by MarakanaTech on Youtube
- http://www.malgenomeproject.org/
- http://stackoverflow.com
Organizational matters

Registration
Motivation
“APPIFICATION”
Nomophobia

- “no-mobile-phone phobia”: Fear of being out of mobile phone contact
- Mobile phone users tend to be anxious when they “lose their mobile phone, run out of battery or credit, or have no network coverage”
- Stress levels induced by the average case of nomophobia to be on-par with those of "wedding day jitters" and trips to the dentists
- More than one in two nomophobes never switch off their mobile phones

Source: http://en.wikipedia.org/wiki/Nomophobia

Source: Sven’s smartphone
Mobile Users Reach to Phone ~150x a Day
Could be Hands-Free with Wearables

# of Times Typical User Checks Phone per Day

- Messaging: 23 times
- Voice Call: 22 times
- Checking Time: 18 times
- Music: 13 times
- Gaming: 12 times
- Social Media: 9 times
- Camera: 8 times
- Alarm: 8 times
- News & Alerts: 6 times
- Calendar: 5 times
- Web: 3 times
- Search: 3 times
- Other: 14 times

Source: TomiAhonen Almanac 2013, LINK. ‘Other’ includes voicemail, charging and miscellaneous activities. We cross-checked Tomi’s analysis to gain context.

Our references include: 1) Motorola Mobility / Google (consumers interact with their phones more than 100x per day, mid-2012); 2) Leading 3G Carrier with Operations in Europe & Asia (smartphone users interact with mobiles ~150x per day); 3) IDC (51 blended average of social sessions per smartphone user per day in USA, 3/13 excluded services like checking time, alarm and calendar events, web browsing, gaming, using camera, listening to music, searching, using maps, charging and other activities that require checking the phone) and 4) other third parties, including app providers.

Source: Mary Meeker. INTERNET TRENDS 2013
Tablet Units = Growing Faster Than PCs Ever Did... +52%, 2013

Global PC (Desktop / Notebook) and Tablet Shipments by Quarter 
Q1:95 – Q4:13

Source: Morgan Stanley Research. Note: Notebook PCs include Netbooks.

Source: Mary Meeker. INTERNET TRENDS 2014
SMARTPHONE OSes IN NEW CONTEXTS

http://upload.wikimedia.org/wikipedia/commons/7/76/Google_Glass_detail.jpg


http://1.bp.blogspot.com/-wx6nEY5m_kM/TVWlX3vguzI/AAAAAAADWi/sr8-slRN9Vc/s1600/Android+in+car+auto+vehicle.jpg
SELECTED SMARTPHONE OPERATING SYSTEMS

- Android
- iOS
- Blackberry
- Windows Phone
- Bada
SELECTED SMARTPHONE OPERATING SYSTEMS

Android
Why most research done on Android?
1. Almost completely Open Source
**Android Software Stack**

- **Application layer**
  - Default (core) apps
    - Browser
    - Contacts
  - Third party apps
    - Facebook
    - Skype

- **Middleware layer**
  - Application Framework

- **Kernel layer**
  - Native libs (C/C++)
  - Android Runtime
  - (Modified) Linux Kernel
2. The Market
Worldwide Smartphone Sales to End Users by Operating System Sold Units and Market Share Q2/2013 and Q3/2013

Based on Gartner Statistics
GLOBAL SMARTPHONE MARKET LANDSCAPE

Based on Gartner Statistics
Smartphones as Target of Attacks
SMARTPHONES: HIGH-VALUE TARGETS

We always carry them with us... most data can be captured LIVE ...

Who we know:
People, Addresses, Phone numbers

What we do next:
Appointments (... with whom?)

Mobile Banking
(e.g., mTAN)

Current thoughts?
ideas, memories, notes, working documents

Premium calls

Photos, videos
(incl. sound, devices have a mic)

GPS: Where are we, where are we going to, Accelerometers

Premium SMS

E-mails, chatting, Skype: our full communication (generally unencrypted)
THREAT CLASSIFICATION

Hardware Attacks

Privacy Violations

Malware

Runtime Attacks
MALICIOUS CHARGER/Docking Station [19]

- Micro-computer (e.g., BeagleBoard or Raspberry Pi) hidden in charger/docking station
- When user plugs in his iPhone, the software on the micro-computer attacks and compromises the phone’s operating system and achieves root privileges
  - Demo attack against iOS: Silently replaced facebook app with a malicious version that spied on the user
- Android: USB has been shown to be a potential attack surface
  - Zero-day exploits? (e.g., error in file system driver or media manager when mounting external harddrive)
Rogue BaseStation [20]

- Fake base station can be easily build
  - OpenBTS/OpenBSC software
  - Dedicated hardware ($1000) or a Motorola C123
- No mutual authentication between phone and cellular network
  - Network does not have to authenticate to phone
- Fallback to GPRS/EDGE when UMTS/HSPA unavailable
- Man-in-the-middle attacks
  - IMSI catcher, downgrade encryption, redirect/alter traffic, ...

![Diagram showing cellular communication between phone, attacker station, and benign station.](image-url)
Baseband attacks [21,22]

- Baseband processor runs dedicated separate OS (e.g., OKL4)
- Baseband OS usually less hardened against software exploits
  - E.g. “software unlocks” for iPhone’s network locks
- Fuzzing the baseband with specially crafted SMS from a rogue basestation revealed software vulnerabilities
  - Compromise the baseband OS
THREAT CLASSIFICATION
SOFTWARE EXPLOITS

- iPhone rooting:
  Dedicated exploit kits
    - E.g. RedSn0w

- All modern exploits use *return-oriented programming* (code reuse) techniques to circumvent defenses such as *Data Execution Prevention (DEP)* or *W^X* (write-xor-execute)
EXKURSION: CONCEPT OF ROP

- “Classical” attack prevented by DEP

I’m the most lovely and non-harmful program you can imagine. I embody functionality like document analysis of different types. I’m not bad in any way.
**EXKURSION: CONCEPT OF ROP**

- “Classical” attack prevented by DEP

I’m the most lovely and non-harmful program you can imagine. I embody functionality like This program now executes an exploit. Not bad in any way.

Attacker

Malicious Input
- Inject Exploit code
- Redirect control-flow
"Classical" attack with DEP

I’m the most lovely and non-harmful program you can imagine. I embody functionality like

This program now executes an exploit not bad in any way.

Malicious Input
- Inject Exploit code
- Redirect control-flow

Attacker
I'm the most lovely and non-harmful program you can imagine. I embody functionality like document analysis of different types. I'm not bad in any way.

- Code reuse attack: No need to “execute data”

**Attacker**

- Inject addresses in known code (Must be known)
- Redirect control-flow to jump to each of those addresses
THREAT CLASSIFICATION

Privacy Violations

Malware

Hardware Attacks

Runtime Attacks
Some Statistics and Examples
Danger of Advertisement Libraries [12]

- 52.1% of 100,000 apps hosted at Google Play and third-party stores include at least one advertisement library.

- For the apps that include an ad library,
  - 79.41% access location data,
  - 33.62% read phone information,
  - 4.98% read contacts/call logs, and
  - 0.78% read installed packages.

- Some: Hidden update functionality.
CLASSIFICATION OF ANDROID MALWARE [9]

- Exfiltrates user information
- Premium calls or SMS
- Sends SMS spam
- Novelty and amusement
- Exfiltrates user credentials
- Search engine optimization
- Ransom

Pie chart showing the distribution of Android malware categories.
CLASSIFICATION OF ANDROID MALWARE (INDUSTRIAL STATS)

- Premium Service Abuser: 47.72%
- Adware: 31.99%
- Data/Information Stealer: 11.34%
- Malicious Downloader: 6.41%
- Hacktool: 2.09%
- Backdoor/Remote Control: 2.58%
- Others: 1.08%
GRAY ZONE

- WhatsApp messaging app
  - Uploads the entire address book and stores it on the servers
  - Creating fake contacts to enumerate range of phone numbers allows you to detect which numbers belong to real WhatsApp users (no mutual authentication required between users)
  - Several incidents with insufficient security of the communication between the app and the servers

- Tinder dating app
  - Revealed exact user location and allowed stalking of members

- Facebook app
  - Requests permission to read SMS messages for authentication purposes (automatically detect SMS with confirmation code for two-factor authentication)
**Motivation**

- Mobile security a very active research area
  - Feature-rich smartphones and “appification” have induced security research on various (new) aspects

- Android’s open-source nature has made Android very attractive to security researchers

- Android’s market share has made Android the #1 target for malware authors and makes improved security & privacy mechanisms imperative