Advanced computer networking
(IN2097)

Project proposal

Advisor: Johann Schlamp
schlamp@in.tum.de

October 29, 2013
Introduction to **MEASR**DROID
We built MeasrDroid

- MeasrDroid
  - Free measurement client with focus on Networks
  - Utilization of integrated sensors and API functions
  - Incorporation of native C code where beneficial
  - Designed for interdisciplinary research

- Progress so far:
  - >10 students (BA/MA/HiWi) worked in the project
  - 32,000 LOC in 35 packages
  - Periodic measurements of 335 distinct data points:
    - hardware-environment-location-network-telephony
CURRENT STATE OF DEVELOPMENT

- **Simple GUI**
  - Focus on research ("donate data")
  - Achievement system to keep users using the App

- **Modular design:**
  - Measurement core API can be used as a library
  - Designed for multiple GUIs

- **Backend**
  - 10 services up and running
  - Large set of Python classes available to ease the development of new services

- **Resource Consumption**
  - Battery: only ~1-2%
  - Network: between 25 MB and 300 MB per month (configurable)

- **Security & Privacy**
  - Full encryption of measurements on-device
  - Cryptographic signatures on all downloaded configuration files
  - Privacy statement, no hidden features

- **BETA Test**
  - 4 months of beta test data available
  - 50 participating clients, 150,000 data sets (~6 GB)
**ANALYSIS FRAMEWORKS SO FAR (1)**

- **Anonymized client visualization**
  - How to visualize location data?
  - How to visualize movement data?
  - *while preserving privacy of our users?*

---

Fully anonymized

Not anonymized
Can we reverse-engineer Google’s Wifi localization technique?

- Try to triangulate Wifi access points (APs)
- Map signal strengths to radii and intersect circles
- Compare results to publicly available ground truth (open wifi spots), evaluate *Telekom WLAN-to-go*

triangulation of a single AP

all APs localized within our building
Advanced computer networking (IN2097)

PROJECT PROPOSAL
OVERVIEW

- **Use MeasrDroid**
  - To initiate measurements to your virtual server
  - To analyze delay characteristics over time

- **Listen for measurements**
  - To initiate counter-measurements from your VM
  - To analyze asymmetries in network topologies

- **Learn more about**
  - The Internet’s dynamic topology
  - Environmental influences in mobile networks
  - Asymmetric routing
YOUR TASKS

- **(1) Project Plan**
  - Find team partner
  - Develop a schedule and milestones
  - Anticipate challenges and problems

- **(2) Measurements**
  - Monitor delay and traceroute measurements
  - Plot and display measurements over time (live)
  - Initiate counter-measurements

- **(3) Evaluation**
  - Find correlations between delay variability and environmental conditions (included in MeasrDroid data)
  - Define a metric for path (a)symmetry, analyze bidirectional measurement data and interpret your findings

- **(4) Final Assessment**
  - Write a detailed report about steps (1) – (3)
  - Reflect on the project (what was good/bad?)
**Technical Issues**

- **Hardware requirements**
  - One Android (> 2.1) device per team
  - >50 MB of traffic volume per month
  - Access to one virtual server per team

- **Download and configure MeasrDroid**
  - Go to [http://mccn.droid.net.in.tum.de/](http://mccn.droid.net.in.tum.de/) and download client
  - Start MeasrDroid and accept privacy agreement
  - You may configure measurements to your needs, but you should not deactivate active network measurements
  - Enter your team name (*important*)

- **Obtain your MeasrDroid data**
  - Your measurement data will be continuously transferred to your virtual server
  - Details can be found on the project sheet (soon)
Practical work
- Programming language is Python
- Auto-generate diagrams with Python/matplotlib
- Live-view of diagrams with Apache/CGI/Python

Important deadlines
- [Nov 5, 2013] Project starts.
- [Nov 12, 2013] Hand in your project plan.
Thanks.

INTERESTED?