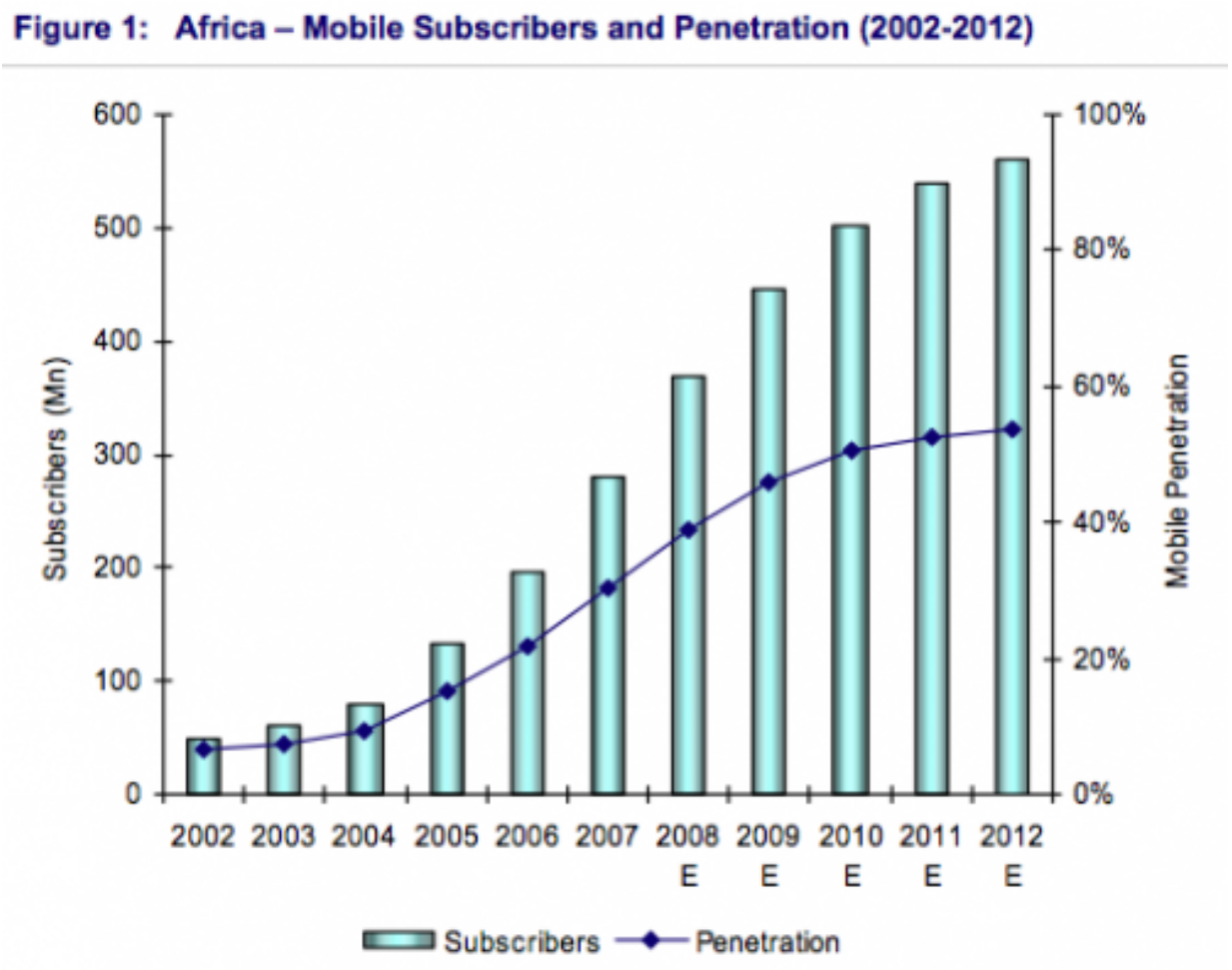
Crime: an infamous thing...

- 2011: 3.3 million crimes
  - > 50% unreported
  - petty crimes? No - burglaries, hijackings, rapes...
- Lasley (1995): anonymity influences a person's decision to report a crime
  - less repercussions

Mobile phones: a popular thing!



Retrieved from: *http://whiteafrican.com/wp-content/uploads/2008/08/african_mobile_subscribers-500x402.png*

Cry-Help

A mobile crime reporting application

Problem statement

There currently does not exist a **cost-effective** way to report crime to the police covertly and **securely**, *and* have the information kept safe efficiently.

Work division

- Thabo: supporting covertness

  interface design task...

- Nina: secure transmission of data

- Tami: performance and security of the backend system

- overarching principle: performance

  (=> cost-effective)

Whither do we go from here?

Motivation for *Cry-Help*

Project description

Problem statement

Work division

A closer look at each component

- Interface design
- Secure transmission
- Backend security and performance

Conclusion

Interface design

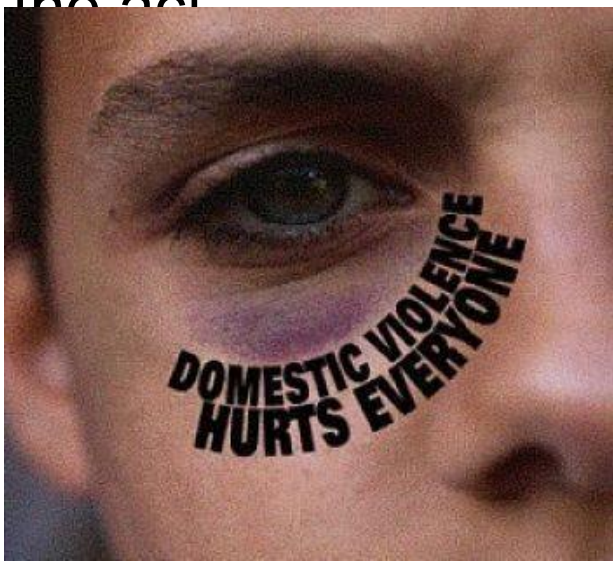
Covertness in *Cry-Help*

The Problem

The Problem

Often individuals are unable to report their circumstance due to:

- Fear of retaliation from the criminal oppressor
  - Do not want to be seen reporting the crime
  - Want the oppressor to be caught in the act
- Identification by the public
  - Public opinion
  - Credibility



The Question then?

- conceal the reporting process
- What means should be used to gather data
- how much data can be pre-collected?
- What kind of interface
- what of time constraints in emergency situations?
- .....

The Question then?

Is it possible to develop a system that allows effective covert message passing in an oppressive environment on mobile devices?

Our Answer

An mobile interactive prototype that allows data collection and crime reporting in a potentially hostile environment.

Data fields necessary for an effective crime report have been collected from authorities such as the CPS.

Design Considerations

- What information is important for an effective report
- Who is making the report
- When can they supply that information

Design Considerations

- From related works 2 interface concepts were conceived.
- Essentially these interfaces are:

Design Considerations Perspective: 1st Person Report

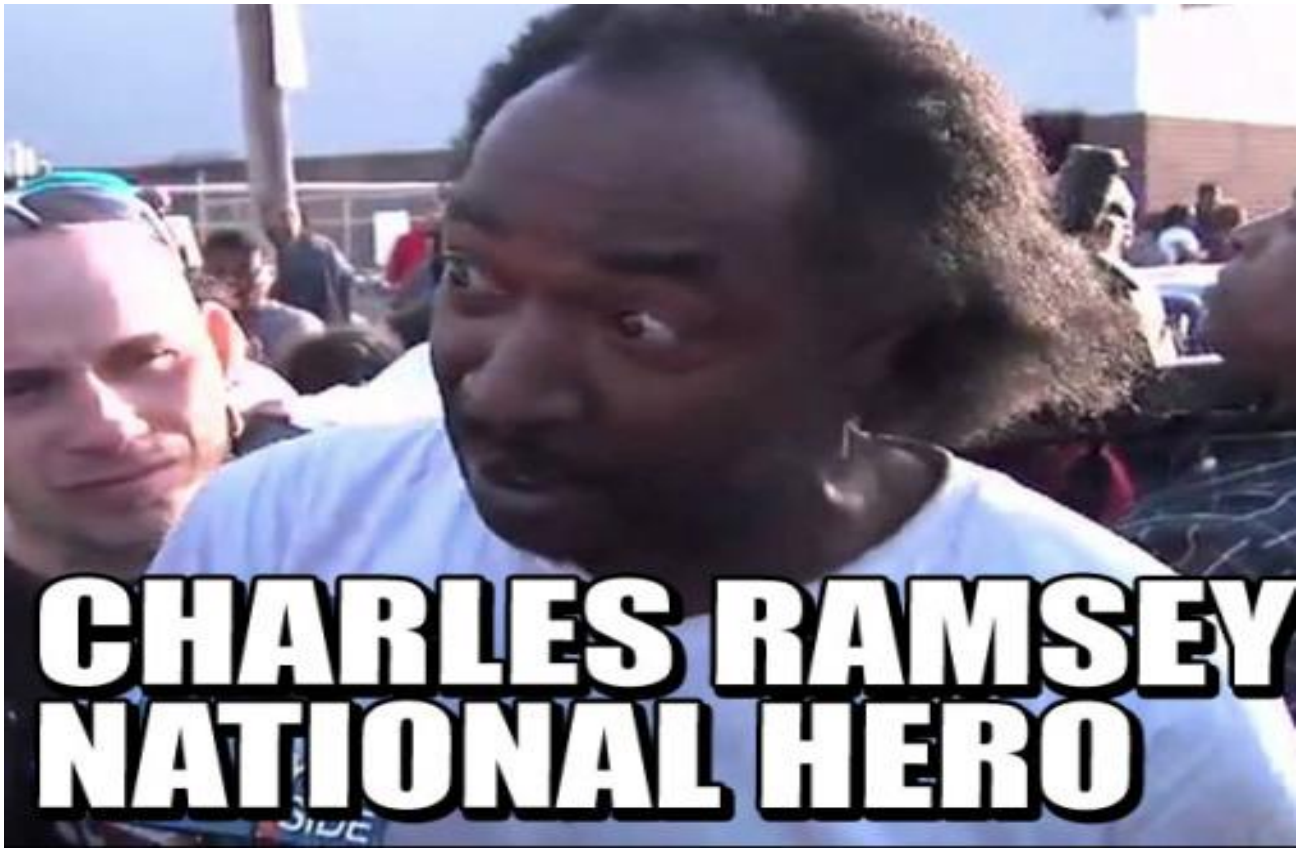
Design Considerations Perspective: 1st Person Report

Interface 1(1st person)- data accumulated is from the victim

- **Panic Button** - report is immediately compiled and sent to authorities
- No/very **little time for conventional input**
- Potentially **hostile environment**, must be sent without second party knowing
- There must be **no trace of communication** eg sent mail, beeps or vibration
- There must still be a way to **confirm a sent message**

Design Considerations Perspective: 3rd Person Report

But what if you see a crime, happen to someone else? #dead_give_away



Design Considerations Perspective: 3rd Person Report

Interface 2(3rd person) -data collected is not necessarily from a third person

- Report may/may not be sent during the crime
- Data accumulated is potentially **more substantial** such as details of the crime an observer can give
- Data sent in a **uniform format**
- Use of **visualisations** and other cues to extract as much data

Methodology

- A design will be devised iteratively with the aid of volunteers to try capture what kinds of input are the most suitable to send data covertly
  - gestures
  - typing
  - speed dial
- 1st person interface
  - prototyping
- 3rd person interface
  - low fidelity
  - high fidelity prototyping

Development

- The application will be developed for an Android target device with the Android SDK on Java
  - Growing smart phone presence
  - Android has largest market share
- Use of existing libraries for gesture recognition

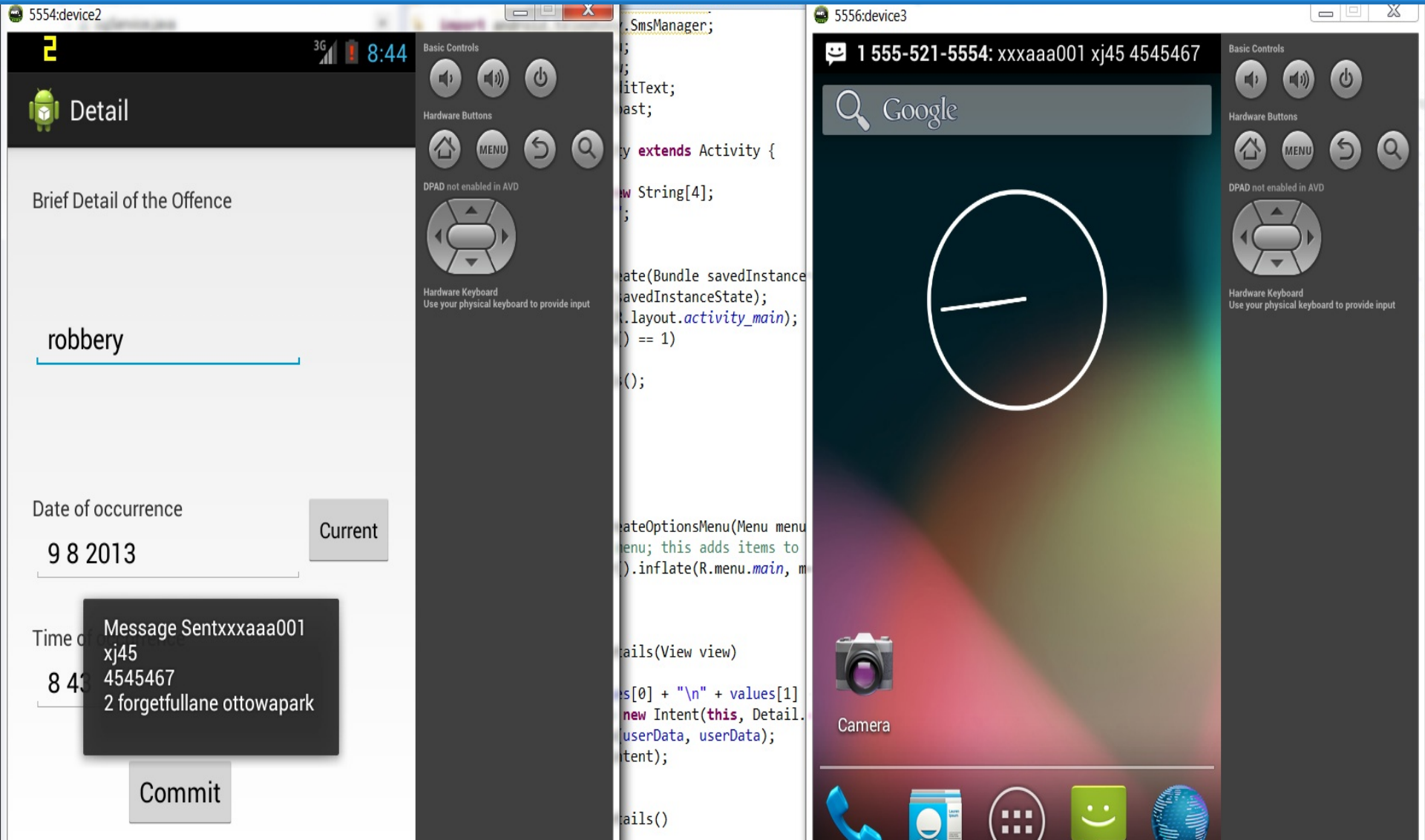
Current Work

Currently the application is in the design stage. I am working with

- Paper prototypes
- Questionnaires
- high fidelity prototype

Will refine the prototype (incorporate 1st person reporting) for more usability testing

Prototype


5554:device2
8:44
Detail
Brief Detail of the Offence
robbery
Date of occurrence
9 8 2013
Current
Time
8 43
Message Sentxxxaaa001
xj45
4545467
2 forgetfullane ottowapark
Commit
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input
.SmsManager;
itText;
ast;
y extends Activity {
w String[4];
ate(Bundle savedInstance
avedInstanceState);
.layout.activity_main);
) == 1)
();
ateOptionsMenu(Menu menu
enu; this adds items to
).inflate(R.menu.main, m
ails(View view)
s[0] + "\n" + values[1]
new Intent(this, Detail.
userData, userData);
tent);
ails()
5556:device3
1 555-521-5554: xxxaaa001 xj45 4545467
Google
Camera

Secure transfer protocol

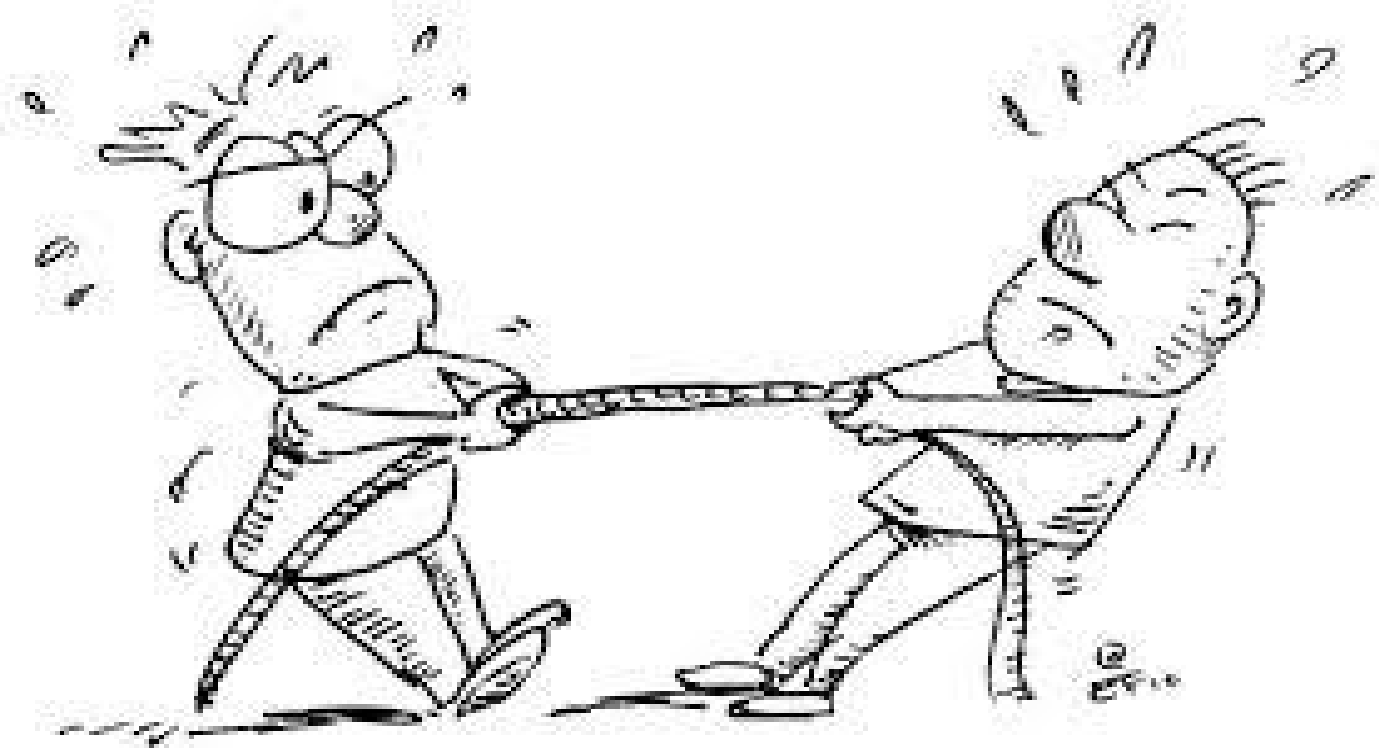
Secure transmission of sensitive data

Problem Statement (1/2)

- Mobile communication security entails
  - **Confidentiality**
  - Authenticity
  - Non repudiation
  - Integrity
  - **Privacy**
- Most cryptographic algorithms tend to be too costly
- Tradeoff: security vs performance
- Mobile data usually unencrypted yet **crime report data is highly sensitive**
- Many protocols have been suggested….

End-to-end security for mobile communication

The tug of war



It's possible!

No! Too costly

Related Work

- Mynttien (2000) on proxy-based security solutions such as WTLS for WAP
- Arapinis et al(2012) points out 3G weakness of linkability and discusses improvements
- Kayayurt & Tuglular (2006) on end-to-end security on mobile devices using TLS and SSL

Research question

Can we implement **end-to-end security** on mobile devices that **performs well**? *Specifically for mobile crime reporting*

Aim

- Implement Kayayurt and Tuglular's protocol
- Test

Why?

- One of few that deals with mobile to computer communication
- Use of TLS which is already well known and trusted

Design & Methodology

- Protocol based on TLS specifications
- Client/server architecture
- Class diagram defined and paper will be used as guide for implementation
- Implementation to be done using Java and Bouncy Castle Crypto Package
- Iterative implementation

Test Plan

- Want to test for both security and performance
- Test for MITM attacks
- Test to ensure that there won’t be any replay message attacks

What has been done so far

Demo

- Got the client/server connection running
- Kayayurt and Tuglular's proposed protocol proved to be too complicated
- SSL was used instead
- SSL easier and supported by android
- Looked into using NS-2 for testing

  -supports c++ and we are using java :(

- Looked into public key cryptography (costly but could be optimised)

Future

- Finalise implementation
- More merging with other team mates parts
- Test (compare protocols)
- Write up
- Celebrate!!!!

Backend security & performance

Excellent safekeeping *and* excellent performance

Recent news...

Source: *http://www.enca.com/south-africa/thousands-vulnerable-after-saps-website-hijacked*

DB Encryption: Motivation

- Protecting data has been an important theme of research
  - more and more sensitive data
  - more and more sophisticated attacks
- Main result so far?
  - Access control systems... various!
- But what happens if those are bypassed?
  - DB is completely readable and otherwise unprotected.
- Enter: DB encryption!

What has been done

- Basic approach: encrypt data-at-rest, and decrypt when needed
  - not too useful
- Growing interest in querying *encrypted* data
- But before we get there...

Access control + Cryptography

- Related work: most use ‘dependent’ keys…
- … but they too can have their flaws
  - e.g. DB availability issues, performance (derive, derive, derive…)

Research question

Crime data need to be kept with utmost security, but also need to be available to officers promptly. In this context:

Can we design a role-based access control *scheme that works with cryptography - and particularly* 'independent' keys *- to add a deeper layer of security to the data?*

Progress and the future

- Developing a basic framework:
  - password-based key generation
  - public-private keys for each user
  - encrypt each user's data with their key
- *Next up:* implementing an efficient mechanism to communicate keys to users higher up in the hierarchy
- *Then:* testing – scalability, latency, security, performance

Conclusion

In a nutshell...

We want *Cry-Help* to be an improvement of its ancestors (E9, CrimePush…)

- lightweight

  use tech that people already own

- highly secure

  transmission + safekeeping

- ‘anonymous’

  covert + secure handling

Many thanks; questions?

References

Arapinis, M., Mancini, L., Ritter, E., Ryan, M., Golde, N., Redon, K., & Borgaonkar, R. (2012, October). New privacy issues in mobile telephony: fix and verification. In Proceedings of the 2012 ACM conference on Computer and communications security (pp. 205-216). ACM.

Blom, J., Viswanathan, D., Spasojevic, M., Go, J., Acharya, K. & Ahonius, R. 2010. Fear and the city: role of mobile services in harnessing safety and security in urban use contexts

Kayayurt, B., & Tuglular, T. (2006). End-to-end security implementation for mobile devices using TLS

protocol. Journal in Computer Virology, 2(1), 87-97.

Lasley, J.R. & Palombo, B.J. 1995. When crime reporting goes high-tech: An experimental test of computerized citizen response to crime. *Journal of criminal justice*

Mynttinen, J. (2000, November). End-to-end security of mobile data in GSM. InTik-110.501 Seminar on Network Security. Helsinki University of Technology.

Satchell, C. & Foth, M. 2011. Welcome to the jungle: Hci after dark. *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*