ReCon: Revealing and Controlling PII Leaks in Mobile Network Systems

David Choffnes

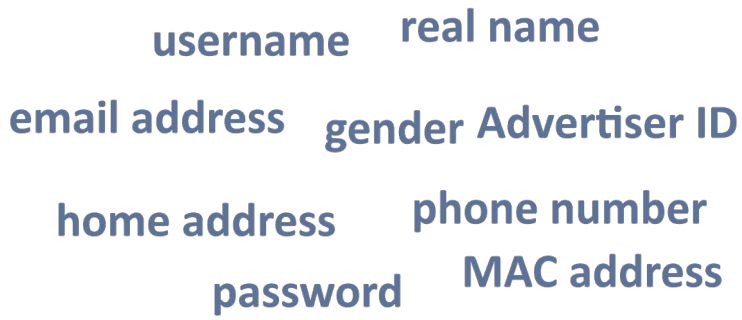
Associate Professor

Khoury College of Computer Sciences

Privacy in the Mobile Internet

Mobile devices

- Rich sensors
- Ubiquitous connectivity



Key questions

- What personal information is transmitted?
- To whom does it go?
- What can average users do about it?

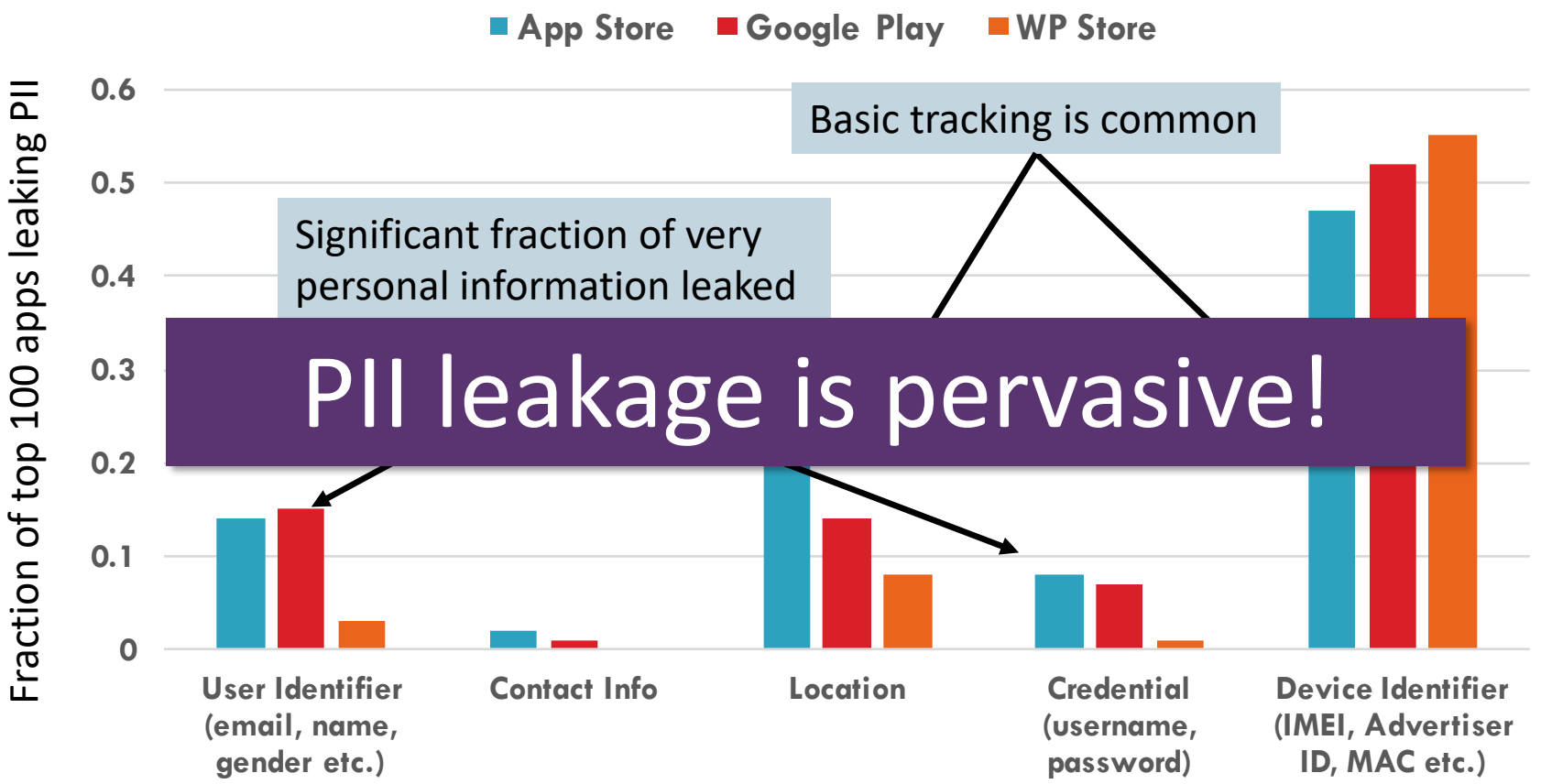
Initial Analysis of PII Leaks

What are our apps sharing about us?

Controlled experiments

- Seed devices with conspicuous PII
- **Manual tests** of top 100 apps for each OS
  - **iOS**, Android, **Windows Phone**
  - (Note results have **significantly better coverage** than automated tests.)

How Frequently Is PII Leaked?



(Tested in September, 2015)

Why do these issues persist?

Researchers, operators and end-users lack good tools for **understanding** and **controlling** *network activity* from their mobile systems

Visibility

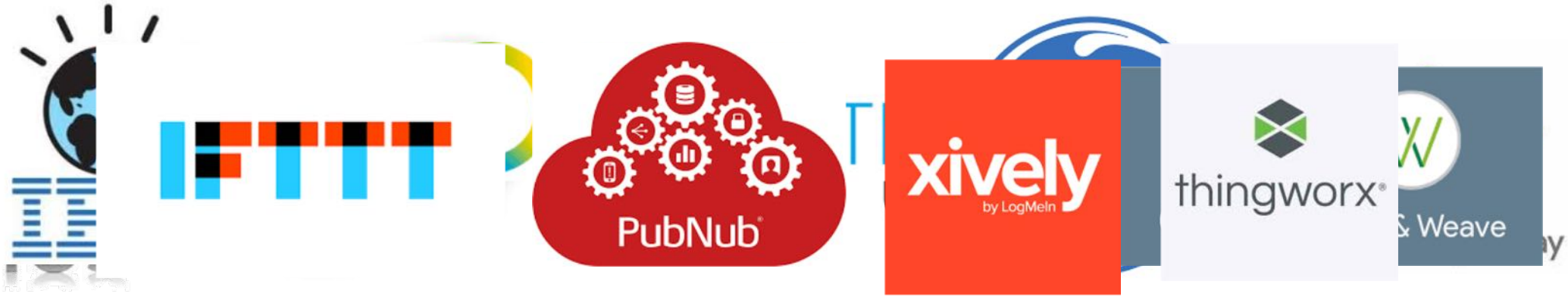
- What apps are gathering our data?
- Where are they sending it?
- How are they protecting it?

Control

- What can we do about any of these problems?
- How do we deploy a solution?

Addressing information leaks

- How do we improve privacy for mobile/IoT devices?



- Our solution
  - Look for PII leaks *in the network*
    - Analysis can run anywhere
    - Trivially easy if you know what PII to search for…

Automatically Identifying PII Leaks

Hypothesis: PII leaks have distinguishing characteristics

- Is it just simple key/value pairs (e.g., "`user=R3C0N`")?
  - Nope, this leads to high FP/FN rates
- Need to ***learn*** the structure of PII leaks

Approach: Build ML classifiers to reliably detect leaks

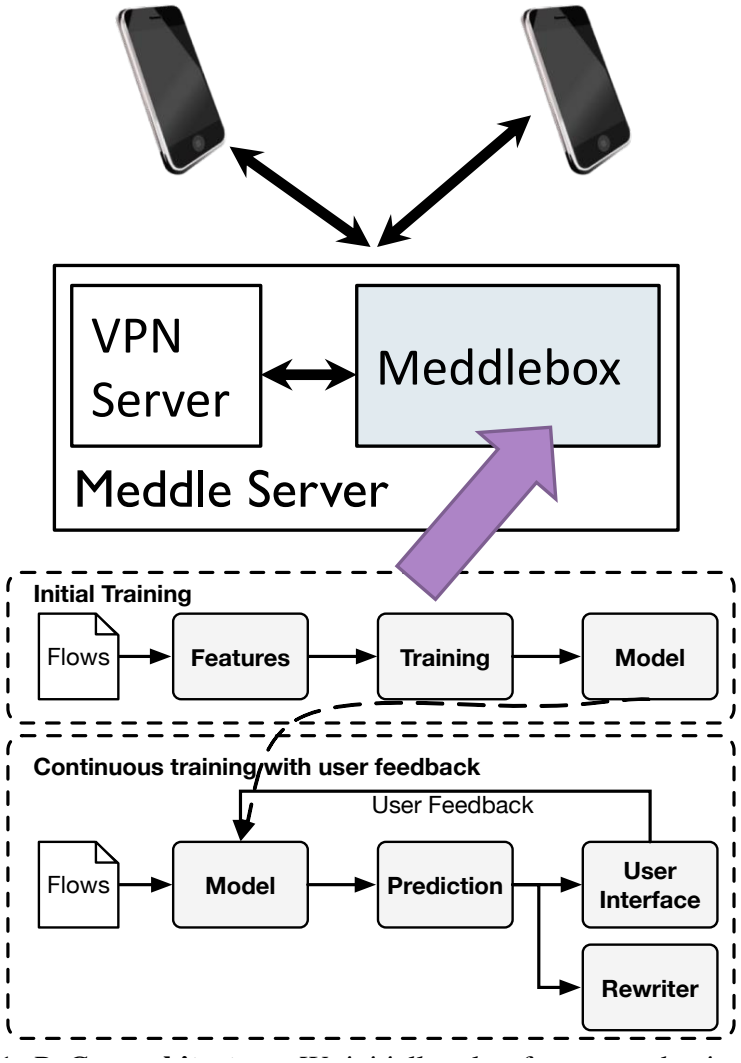
- Does not require knowing PII in advance
- Resilient to changes in PII leak formats over time

ReCon Components

Machine learning to reveal PII leaks from mobile devices

Software middleboxes to intercept and control leaks

Web-based UI, works on all major platforms (**iOS, Android, Windows Phone**)



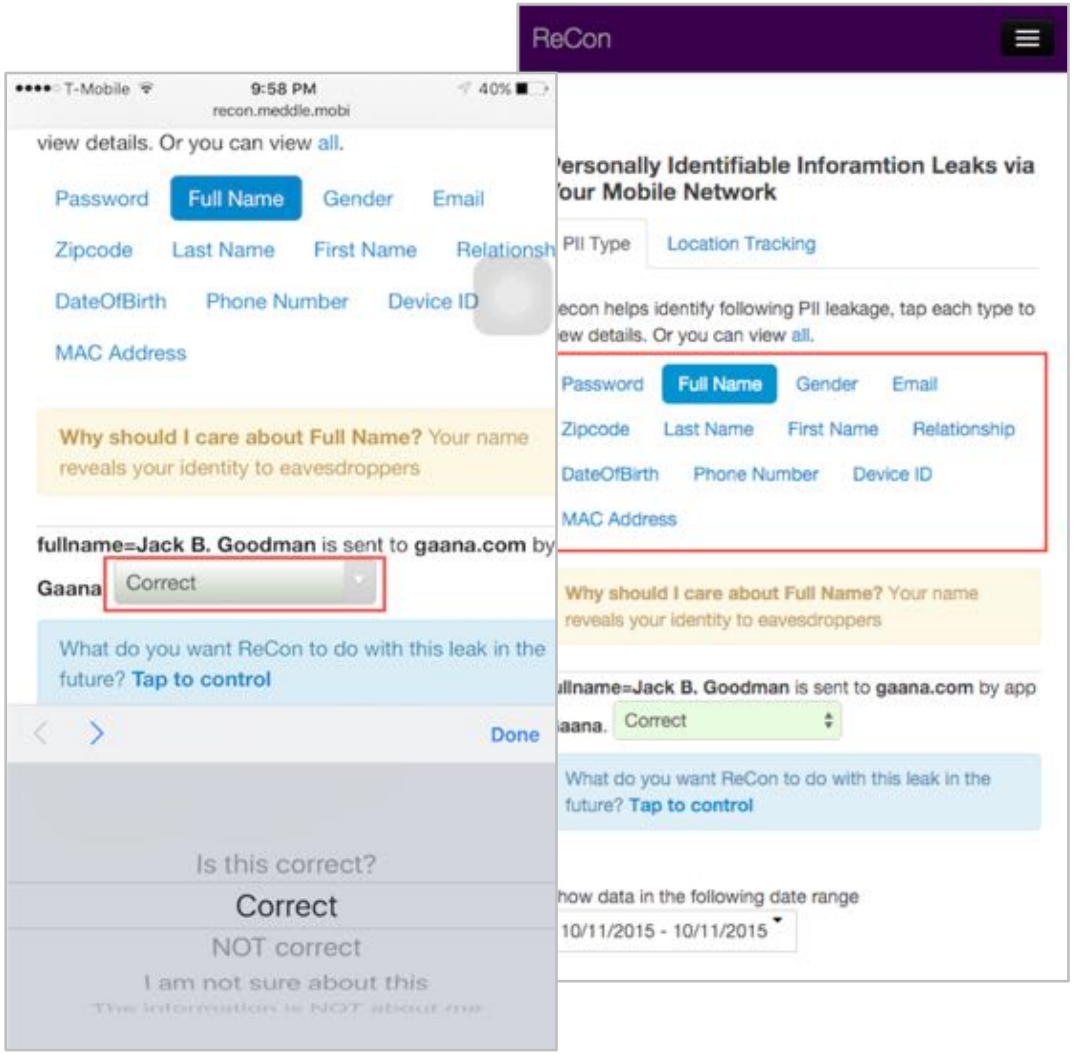
ReCon: Viewing Detected Leaks

PII Category

- Device Identifiers
- Contact Information
- User Identifiers
- Credentials

User Feedback

- Correct
- Incorrect
- Not sure
- Not about me



Outline

Overview

Design and Implementation
- Machine learning meets PII
- Mitigating PII leaks

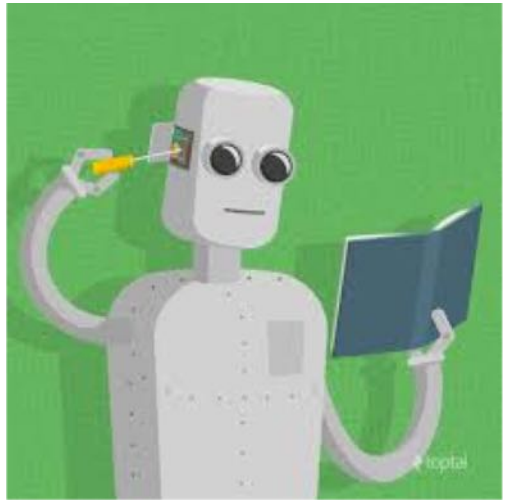
Evaluation

Results from user study

Learning When PII is Leaked

Key challenges for ML-based PII detection

- Which classifier do we use?
- How do we train the classifier?
  - Use traces from real users and controlled experiments
  - Feature selection for scalability
- How well are we doing?
  - Comparison with TaintDroid/Andrubis
  - ***Only the users themselves know for sure!***
  - Crowdsourced reinforcement



ML Approach

Controlled experiments as ground truth

Text classification approaches

- Problem: Given a network flow, does it contain PII?

```
GET /index.html?id=1234567890;foo=bar;name=Choffnes;pass=somepassword
```

ML Approach

Controlled experiments as ground truth

Text classification approaches

- Problem: Given a network flow, does it contain PII?

```
GET /index.html?id=1234567890;foo=bar;name=Choffnes;pass=somepassword
```

ML Approach

Controlled experiments as ground truth

Text classification approaches

- Problem: Given a network flow, does it contain PII?

```
GET /index.html?id=1234567890;foo=bar;name=Choffnes;pass=somepassword
```

ML Approach

Controlled experiments as ground truth

Text classification approaches

- Problem: Given a network flow, does it contain PII?

```
GET /index.html?id=1234567890;foo=bar;name=Choffnes;pass=somepassword
```

- Feature Extraction: Bag-of-words model
- Per-tracker classifiers (e.g. Google-Analytics)
  - Faster (compared to one-size-fits-all)
  - More accurate
- Throw a whole bunch of classifiers at the problem

How Does ReCon Work?

Which classifier do we use?

- **C4.5 Decision Tree** is best trade-off between speed and accuracy (Added bonus: We can understand its outputs!)

How do we train the classifier?

- Use traces from **real users** and **controlled experiments**
- Break flows into separate **words** that may indicate a leak
- **Feature selection** for scalability

How well are we doing?

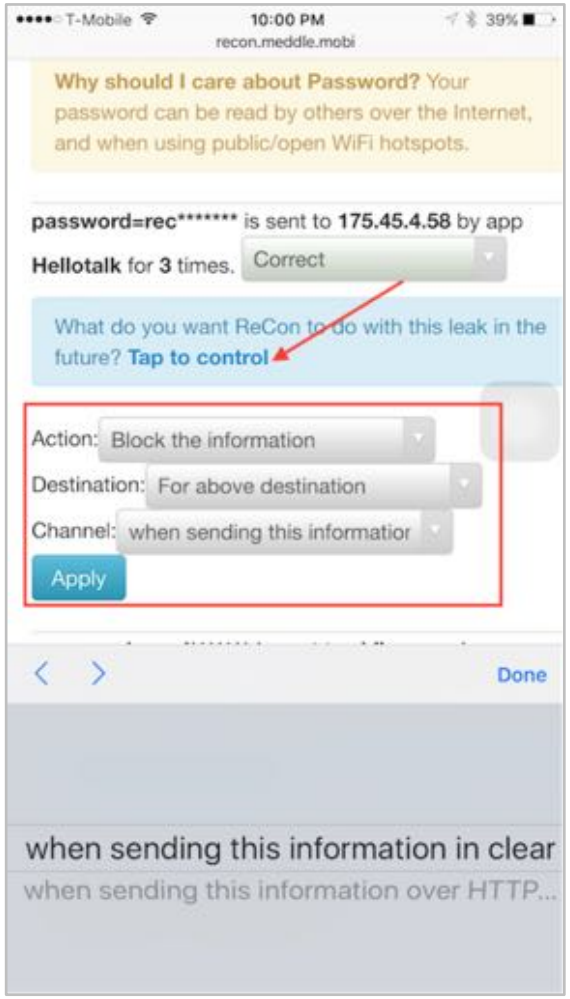
- Controlled experiments
- *In the wild:* ***Only the users themselves know for sure!***
  - Crowdsourced reinforcement

Mitigating PII Leaks

How should ReCon block/modify PII?

- Block all flows with leaks
- Replace/modify all leaked data
- Act on only *some* data

Let the user decide!



Outline

Overview

Design and Implementation

Evaluation

- ML accuracy
- Comparison with alternative approaches

Results from user study

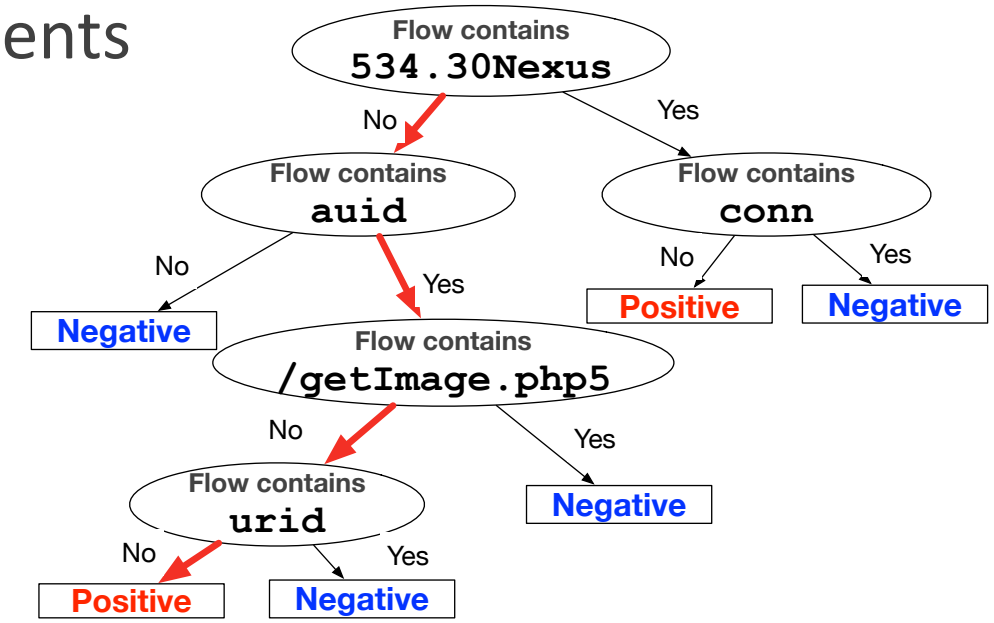
Key Results: ReCon accuracy

Manual test dataset

- 10-fold cross validation

How accurate is ReCon?

- **99% overall accuracy** from controlled experiments
- FPR: 2.2%, FNR: 3.5%
- Why?
  - Per-domain classifiers
  - Decision tree captures non-trivial cases



Outline

Overview

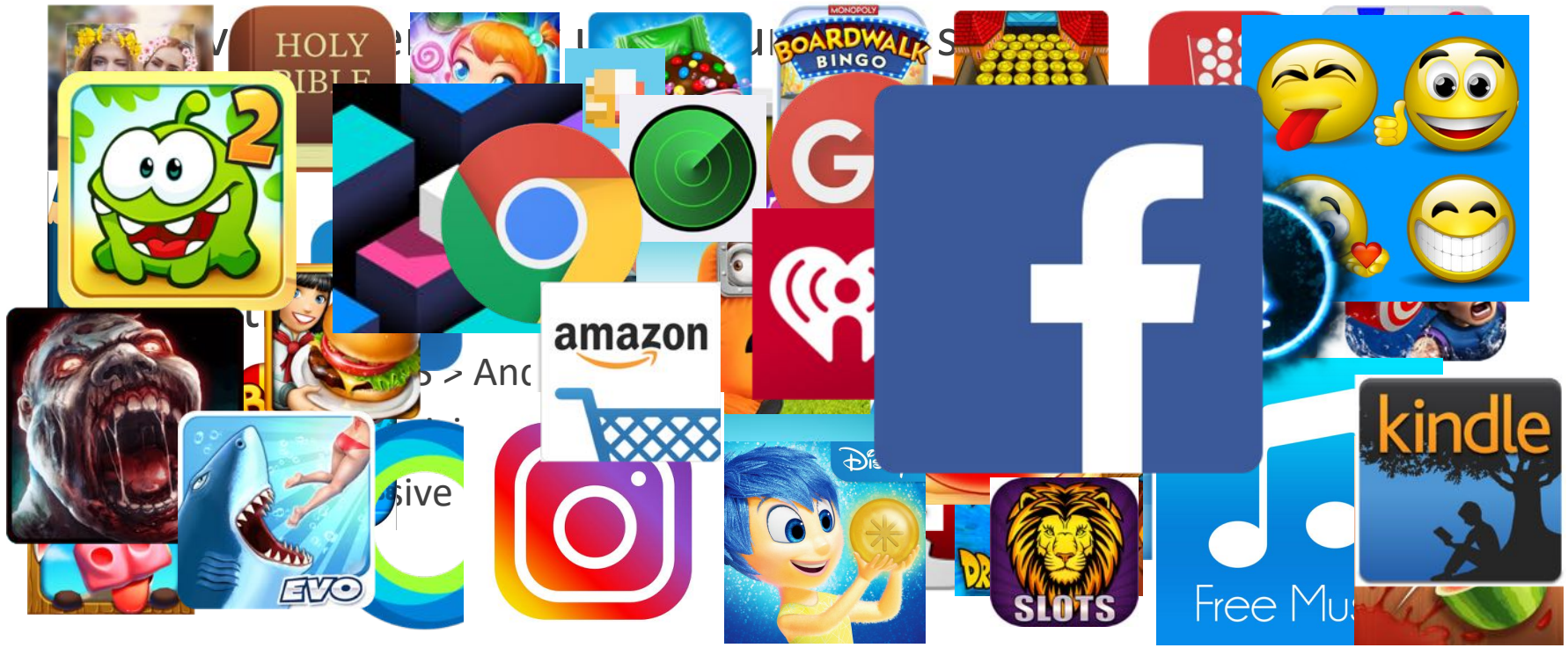
Design and Implementation

Evaluation

Results from user study

- Aggregate results
- Interesting cases

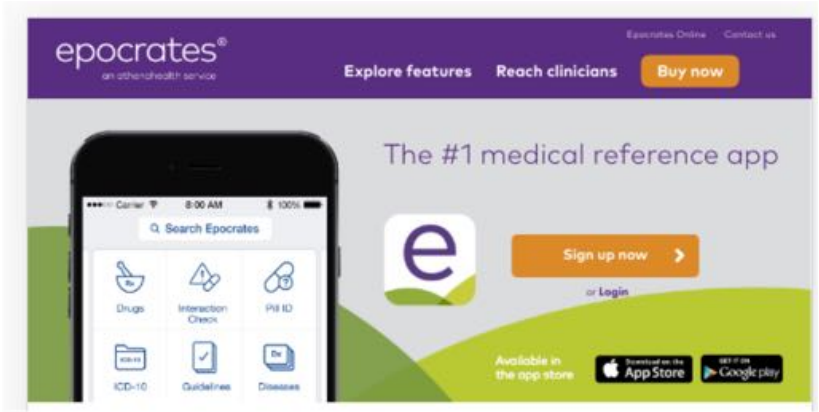
Just some leaks found by ReCon



Impact: security and transparency

*Identified 25 apps exposing **passwords** in **plaintext***

- Used by **millions** (Match.com, Epocrates.com)
- Responsibly disclosed
- Gave 3 months to remediate



Impact: security and transparency

*Identified 25 apps exposing **passwords** in **plaintext***

- Used by **millions** (Match.com, Epocrates.com)
- Responsibly disclosed
- Gave 3 months to remediate

*5 apps sending encrypted passwords to **3rd parties***

- Includes Pinterest, GrubHub, JetBlue

Interesting responses from developers

“Thank you for responsibly disclosing this”

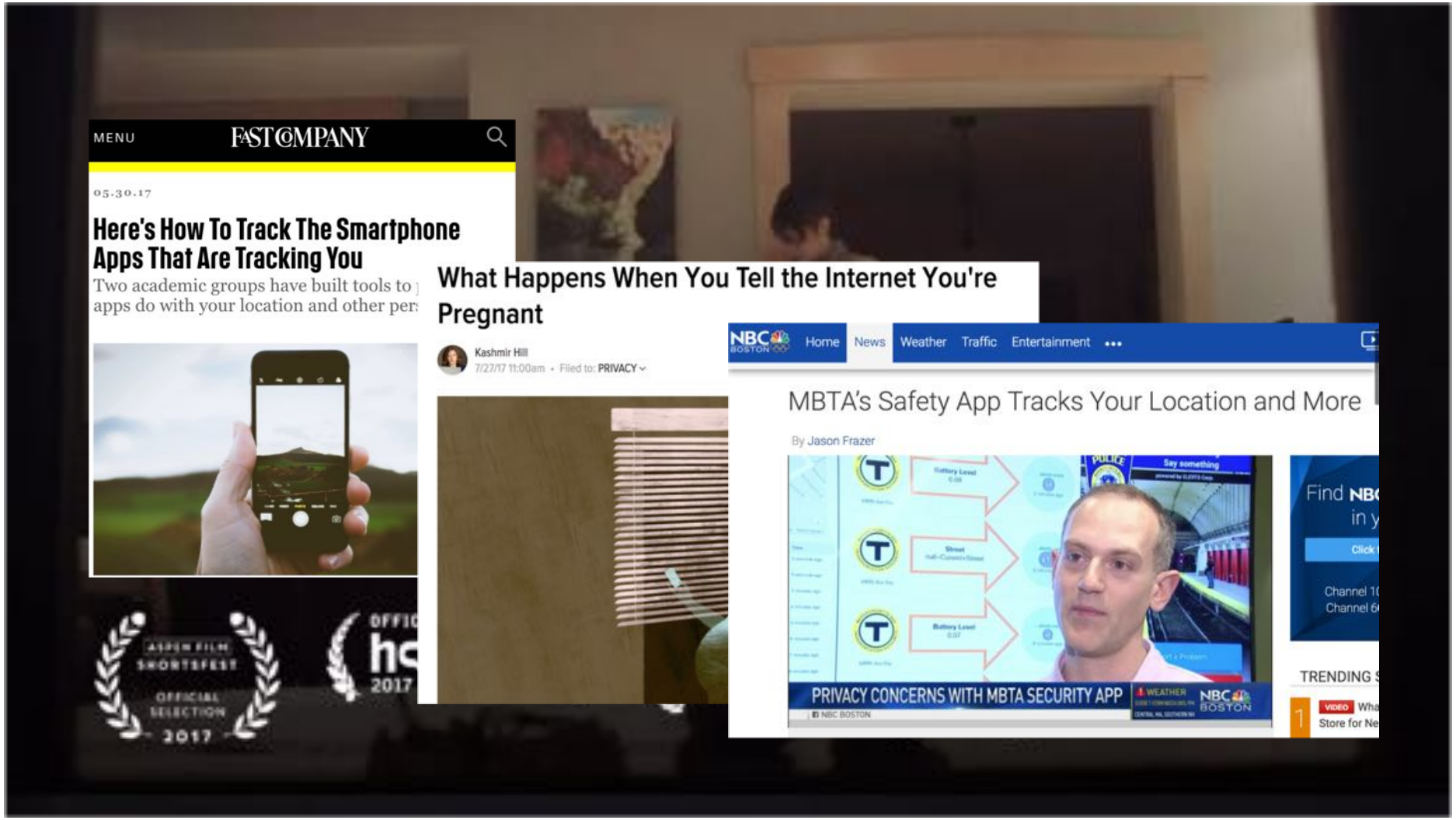
“We do not claim to be a secure messaging app”

“Sending passwords in plaintext is intentional”

“We can’t fix this because our vendor went out of business”

“…”

Impact: Public awareness and discourse



http://www.harvest-documentary.com

Beyond ReCon

Which is worse: app or website? [IMC 2016]

- Answer surprisingly depends
- Even differences between Android and iOS

How are privacy leaks changing over time? [NDSS 2018]

- Getting worse overall
- High levels of variability between versions

Beyond text: Are our apps spying on us? [PETS 2018]

Put on your tinfoil hats

 + internet connectivity … 👀

SilverPush ultrasonic beacons for cross-device linking

patents for recognizing user emotion

⚽ listening for unlicensed broadcasting

📷 photos taken surreptitiously by shrinking preview to 1x1 pixel

Results

We looked at 17,260 Android apps

- Static analysis
- Dynamic analysis
- Manual validation

What did we find?

- 21 cases of detected media – 12 considered **leaks**
  - Unexpected or unencrypted
- 9 shared with third parties

Case study: goPuff + Appsee

goPuff

Screen recording of user interaction, where PII was exposed

- Leaked to an Appsee domain

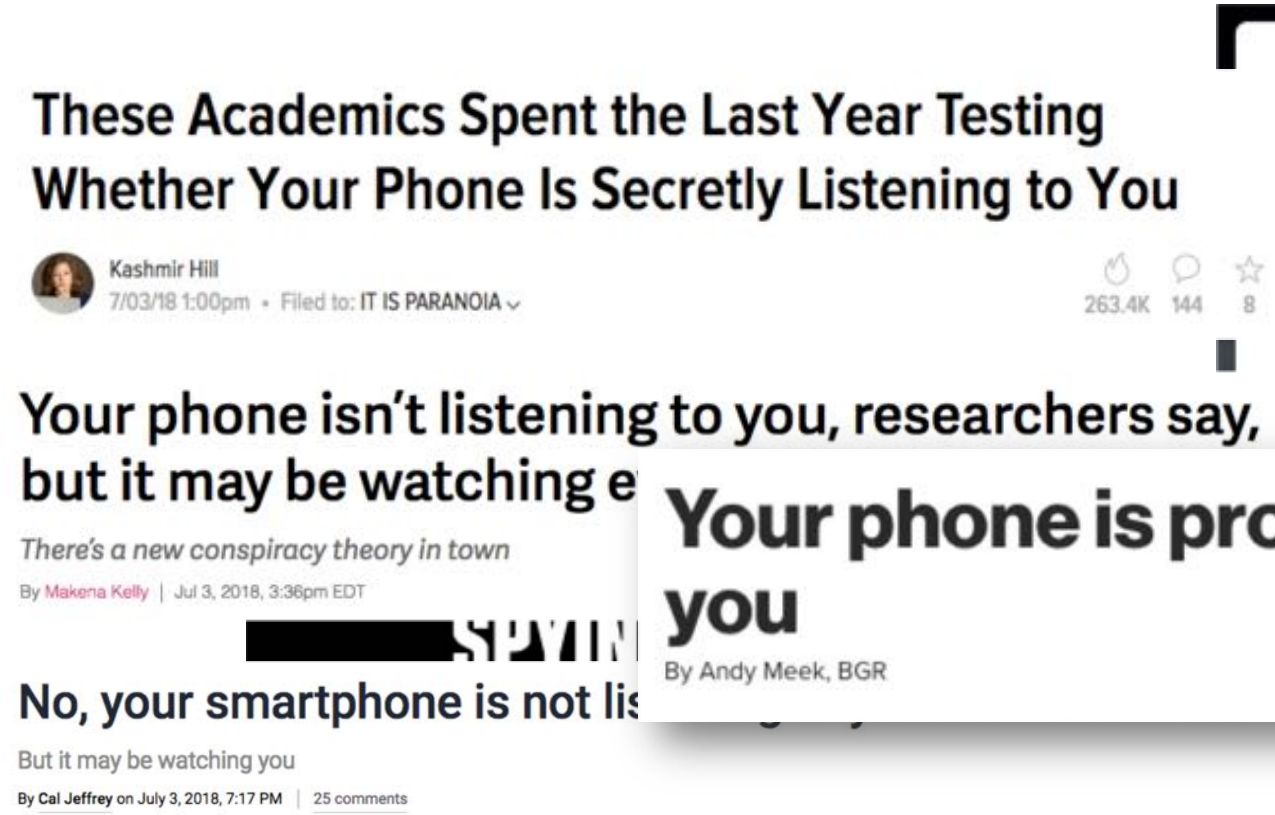
appsee

Screen recording as a feature

Developers are responsible for hiding sensitive screens

Few apps use the API method to do so – 5/33 apps

- Server-side way exists, unknown how many apps use it



These Academics Spent the Last Year Testing Whether Your Phone Is Secretly Listening to You

Kashmir Hill
7/03/18 1:00pm • Filed to: IT IS PARANOIA

263.4K 144 8

Your phone isn't listening to you, researchers say, but it may be watching e

There's a new conspiracy theory in town

By Makena Kelly | Jul 3, 2018, 3:36pm EDT

[redacted]

No, your smartphone is not lis

But it may be watching you

By Cal Jeffrey on July 3, 2018, 7:17 PM | 25 comments

Smartphone apps don't listen to your conversations, but they do something equally creepy

The researchers found that while smartphone applications did not send audio clippings to third-party domains, they did send screenshots or screen recordings to them.

BusinessToday.In New Delhi Last Updated: July 4, 2018 | 22:14 IST

Your phone is probably spying on you

By Andy Meek, BGR

July 5, 2018 | 10:25am | Updated

20/20

Follow

Uh-oh. Boffins say most Android apps can slurp your screen – and you wouldn't even know it

Fancy that

59 SHARE

al airs Friday at

may be spying on you

pect

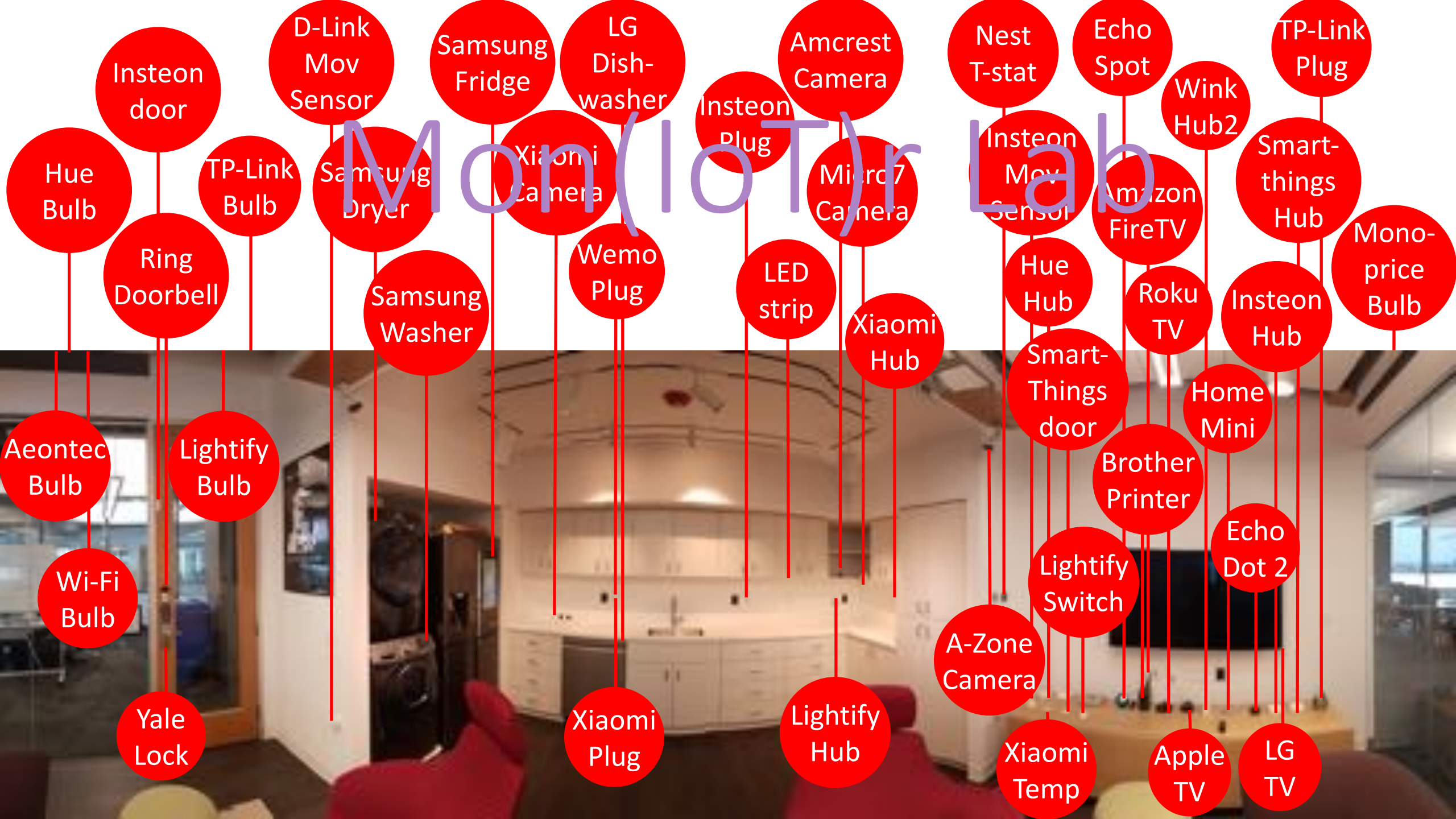
Elizabeth Weise, USA TODAY Published 12:04 p.m. ET July 5, 2018 | Updated 4:21 p.m. ET July 6, 2018

Yes, your phone is spying on you...but not how you think it is

Yahoo Finance Video • July 5, 2018

'ScreenTime: Diane Sawyer Reporting' - Watch Friday at 8|7c on ABC

Privacy in a world without walls


Mon(IoT)r Lab
Insteon door
D-Link Mov Sensor
Samsung Fridge
LG Dish-washer
Amcrest Camera
Nest T-stat
Echo Spot
TP-Link Plug
Wink Hub2
Hue Bulb
TP-Link Bulb
Samsung Dryer
Xiaomi Camera
Insteon Plug
Insteon Mov Sensor
Smart-things Hub
Ring Doorbell
Samsung Washer
Wemo Plug
LED strip
Xiaomi Hub
Hue Hub
Amazon FireTV
Roku TV
Insteon Hub
Mono-price Bulb
Aeontec Bulb
Lightify Bulb
Smart-Things door
Home Mini
Brother Printer
Wi-Fi Bulb
Lightify Switch
Echo Dot 2
A-Zone Camera
Yale Lock
Xiaomi Plug
Lightify Hub
Xiaomi Temp
Apple TV
LG TV

Key challenges

Often no user-configurable operating system

- Makes analysis on the device impractical

Everything is encrypted

- Often **strongly encrypted**
- Can’t MITM

Leaks are not just text

- ReCon won’t work

Ongoing work

Can we reliably **infer contents** of encrypted network traffic?

Can we define notions of **normal** and **abnormal** behavior?

Can we automatically flag **suspicious**, **malicious** behavior?

How does IoT behavior vary over **time** and **location**?

Questions / Acknowledgments

Jingjing Ren, ***Chris Leung, Elleen Pan***, Daniel Dubois, Christo Wilson
Ashwin Rao, *Martina Lindorfer*, Arnaud Legout, *Narseo Vallina-Rodriguez*

DATA TRANSPARENCY LAB

COMCAST | INNOVATION FUND

Homeland Security
Science and Technology