http://defdroid.org

DefDroid: Towards a More Defensive Mobile OS Against Disruptive App Behavior

Peng (Ryan) Huang, Tianyin Xu, Xinxin Jin, Yuanyuan Zhou

UC San Diego

Growing number of (novice) app developers

App Developers Who Are Too Young to Drive

By JESSICA E. VASCELLARO
Updated June 18, 2012 2:16 p.m. ET

Paul Dunahoo went on a business trip to San Francisco last week, where he attended technical sessions at Apple Inc.'s developer conference, networked with other programmers and received feedback from Apple engineers on his six productivity apps.

CONNECTED WORLD DEVELOPER CORNER *July 6th, 2015*

There Are 800,000 New App Developers Every Year

written by David Bolton

Product Development Web Development

Growing As A Developer With No Formal Computer Science Background

Bruce Williams is the CTO of CargoSense, a company that helps companies monitor shipments that require special care in handling or are environmentally sensitive. He

Mobile apps often less mature

Top 1,000 mobile apps

Platform	Avg. age	% of apps by individual developers
App Store	2.5 years	> 12%
Google Play	1.5 years	> 5%

Popular desktop applications

Application	History
MySQL	19 years
Firefox	12 years
Chrome	7 years
…	

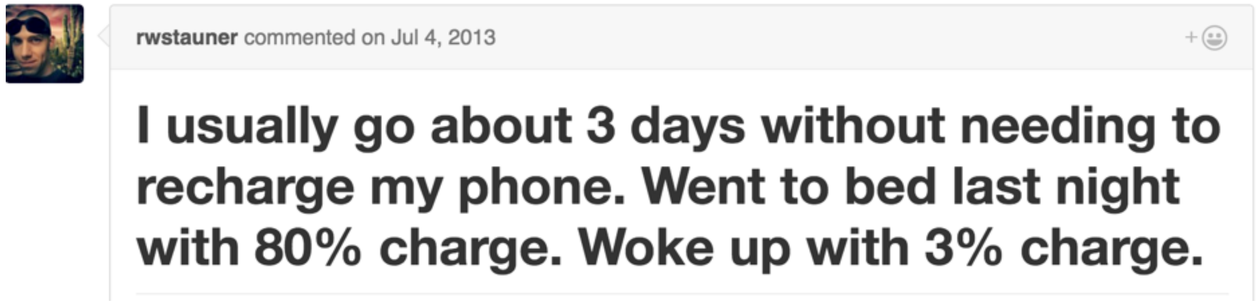
Rise of disruptive app behavior (DAB)

App acts in a way that hurts the ecosystem and other apps

Rise of disruptive app behavior (DAB)

App acts in a way that hurts the ecosystem and other apps

Battery drain



Rise of disruptive app behavior (DAB)

App acts in a way that hurts the ecosystem and other apps

Battery drain

rwstauner commented on Jul 4, 2013

I usually go about 3 days without needing to recharge my phone. Went to bed last night with 80% charge. Woke up with 3% charge.

Excessive storage

Bugsbane commented on Jan 24, 2015

Not sure why, but my app data for focal has ballooned to 430mb!

Rise of disruptive app behavior (DAB)

App acts in a way that hurts the ecosystem and other apps

Battery drain

rwstauner commented on Jul 4, 2013

I usually go about 3 days without needing to recharge my phone. Went to bed last night with 80% charge. Woke up with 3% charge.

Excessive storage

Bugsbane commented on Jan 24, 2015

Not sure why, but my app data for focal has ballooned to 430mb!

Data hog

yamatt commented on Nov 22, 2011

I just installed Podax on my new phone and it used up nearly my entire month of bandwidth in an hour!

Rise of disruptive app behavior (DAB)

App acts in a way that hurts the ecosystem and other apps

Battery drain

rwstauner commented on Jul 4, 2013

I usually go about 3 days without needing to recharge my phone. Went to bed last night with 80% charge. Woke up with 3% charge.

Excessive storage

Bugsbane commented on Jan 24, 2015

Not sure why, but my app data for focal has ballooned to 430mb!

Data hog

yamatt commented on Nov 22, 2011

I just installed Podax on my new phone and it used up nearly my entire month of bandwidth in an hour!

Noisy notifications

valerio-bozzolan commented on Jan 15, 2015

My phone has a LED status for every app notitication, and so it's very annoying to see every 2 second this notification when I am under stupid firewall °^°

Our main contributions

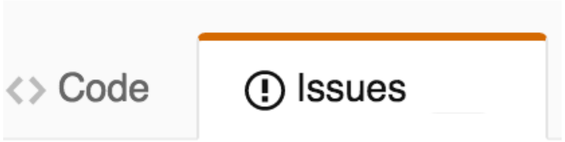
- A characteristics study on disruptive app behavior problem
- An OS-level solution to curb disruptive apps at runtime

Talk outline

- Background
- **Understand disruptive app behavior**
- **DefDroid: curb disruptive app behavior at runtime**

Data collection

user forums



open-source project issue trackers

app issue complaints

9,000+ initial cases

~~crash, ANR, feature bug~~

287 DAB cases
182 unique apps

Example of disruptive app behavior

App	Version	Disruptive behavior
ownCloud	1.5.4	Sync with server too often and request a lot of info, battery drain, data hog
WHERE	3.2.1	Continuously use GPS on standby, draining battery
F-Droid	0.72	Create “infinite” copies of XML files due to a bug, eating up storage
DAVDroid	0.6	Sync lots of high resolution contact pictures, data hog
Facebook	40.0	CPU spin in network handling, leaking audio session, draining battery
Chrome	28.0	Overuse Gyroscope sensor in background, draining battery

Example of disruptive app behavior

App	Version	Disruptive behavior
ownCloud	1.5.4	Sync with server too often and request a lot of info, battery drain, data hog
WHERE	3.2.1	Continuously use GPS on standby, draining battery
F-Droid	0.72	Create “infinite” copies of XML files due to a bug, eating up storage
DAVDroid	0.6	Sync lots of high resolution contact pictures, data hog
Facebook	40.0	CPU spin in network handling, leaking audio session, draining battery
Chrome	28.0	Overuse Gyroscope sensor in background, draining battery

Even expert developers can make mistakes

Users often still need to use the apps

This issue makes this program less acceptable to me than Sipdroid. Otherwise, I am happy with the UI and satisfied with stability

-- user comment on a severe battery drain issue in CSipSimple

4.1 out of 5 rating for apps with disruptive behavior in our study

Root causes of DAB are diverse

Refresh widget every 2.5 seconds.
Severe battery drain!

```
<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider
  android:minHeight="146dp"
  android:minWidth="146dp"
  android:updatePeriodMillis="2500"/>
```

Root causes of DAB are diverse

**Refresh widget every 2.5 seconds.
Severe battery drain!**

```
<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider
  android:minHeight="146dp"
  android:minWidth="146dp"
  android:updatePeriodMillis="2500"/>
```

When users are in environment with poor GPS-signal, it will keep searching for GPS signal!

```
private requestLocationUpdate(LocationManager lm) {
  lm.requestSingleUpdate(provider, mLocationListener);
}
private LocationListener mLocationListener =
  new LocationListener() {
  public void onLocationChanged(Location location) {
    renderData(getWeatherForLocation(location));
    scheduleRefresh(); // refresh in 60 minutes
    disableOneTimeLocationListener();
  }};
```

BetterWeather

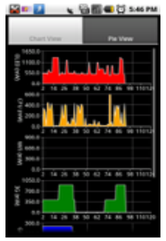
Other findings

- Common patterns
- Triggering conditions
- Fix strategy and time

How to deal with disruptive app behavior?

User

- difficult to diagnose
- difficult to fix



Ordinary

Developer

- difficult to ship always-friendly code

Inexperienced

OS

?

Talk outline

- Background
- Understand disruptive app behavior
- **DefDroid: curb disruptive app behavior at runtime**

DefDroid

Goal

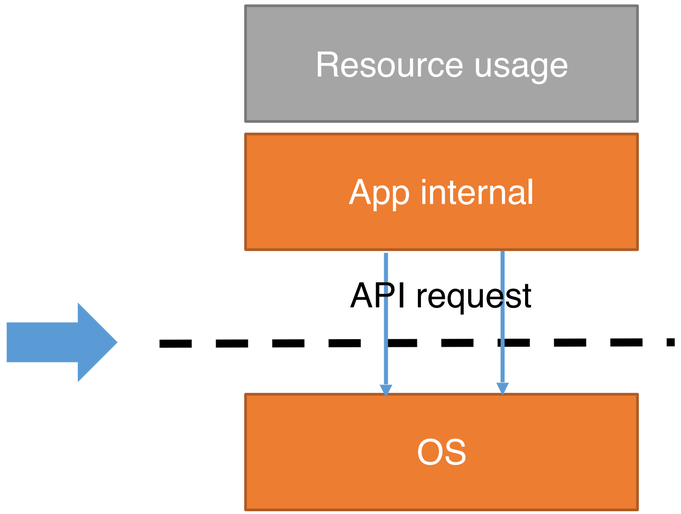
- making OS defensive to react to common disruptive app behavior
- protect end users from bad experiences

Main challenges

- handle a diverse set of disruptive behavior
- preserve app main functionality
- control overhead

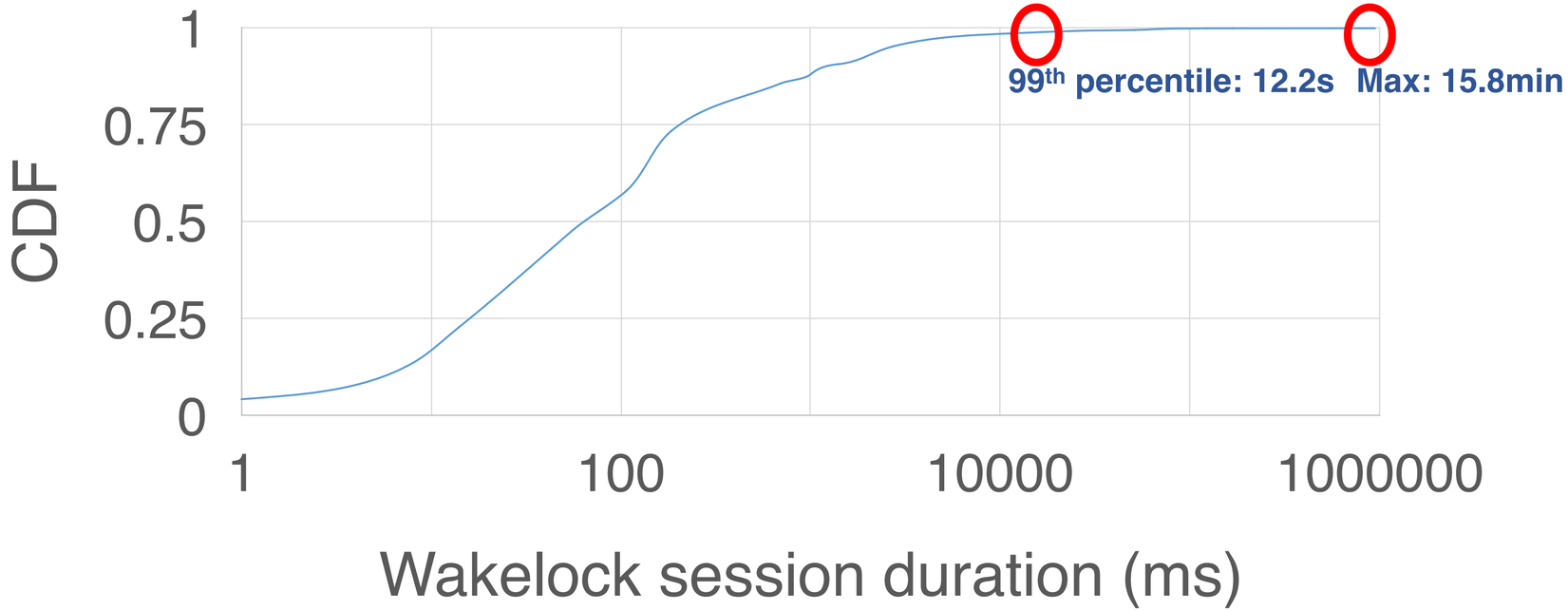
Key idea of DefDroid

» Monitor important app interactions with OS



Key idea of DefDroid

- Monitor important app interactions with OS
- Define disruptive behavior using profiling + user complaints



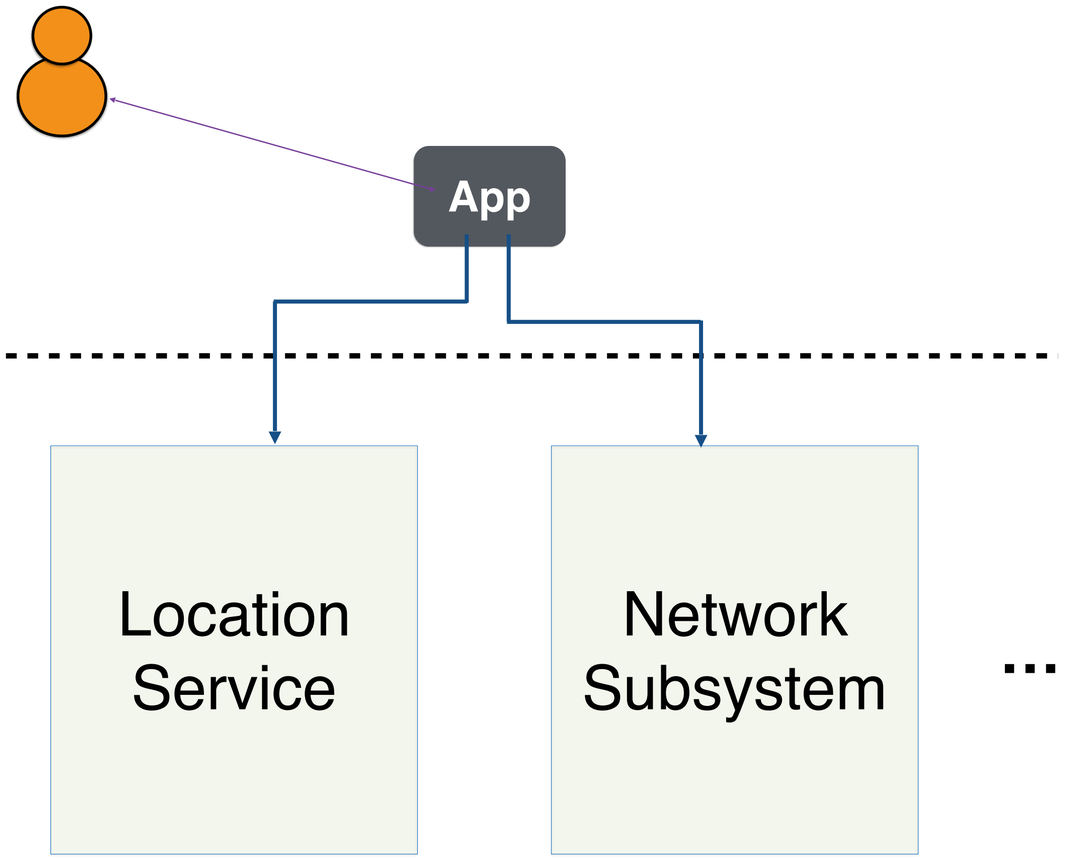
Key idea of DefDroid

- Monitor important app interactions with OS
- Define disruptive behavior using profiling + user complaints
- Apply **defensive actions** at request granularity

Key idea of DefDroid

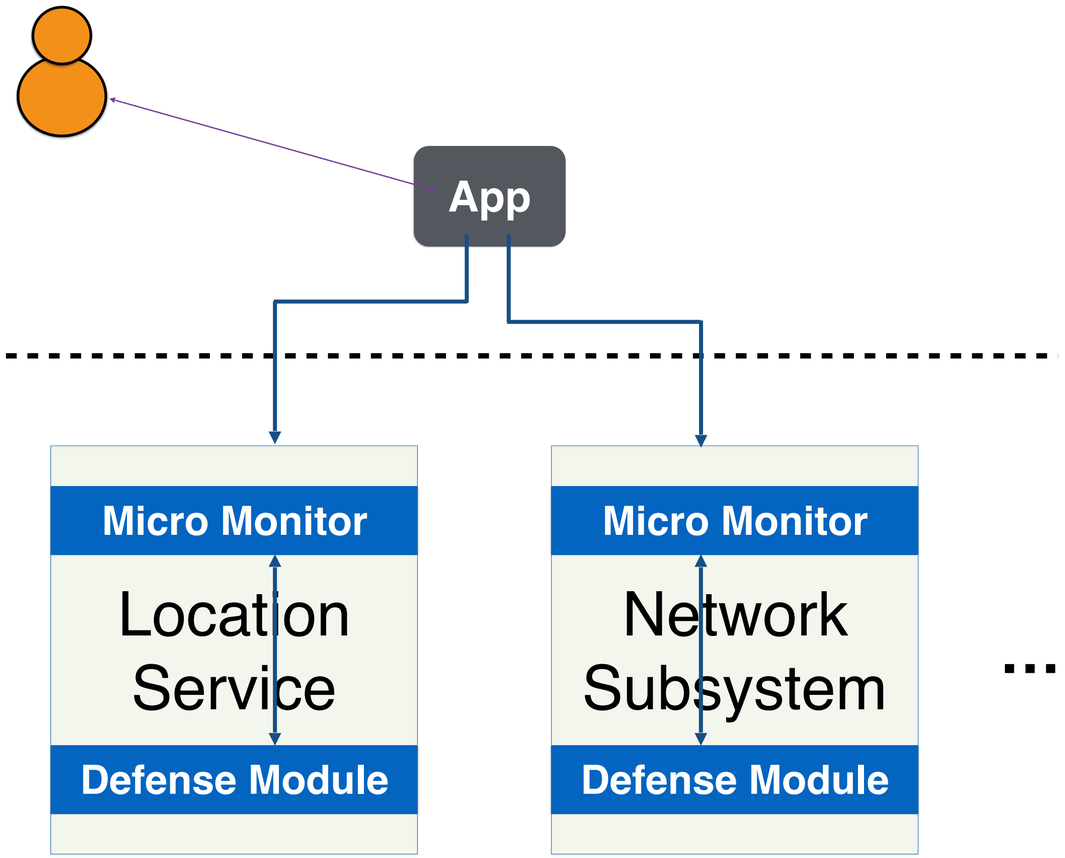
- Monitor important app interactions with OS
- Define disruptive behavior using profiling + user complaints
- Apply **defensive actions** at request granularity
  - enforce exponential back-off
  - release long-held resource
  - reduce aggressive update frequency
  - decrease scheduling priority

Extensible defense modules

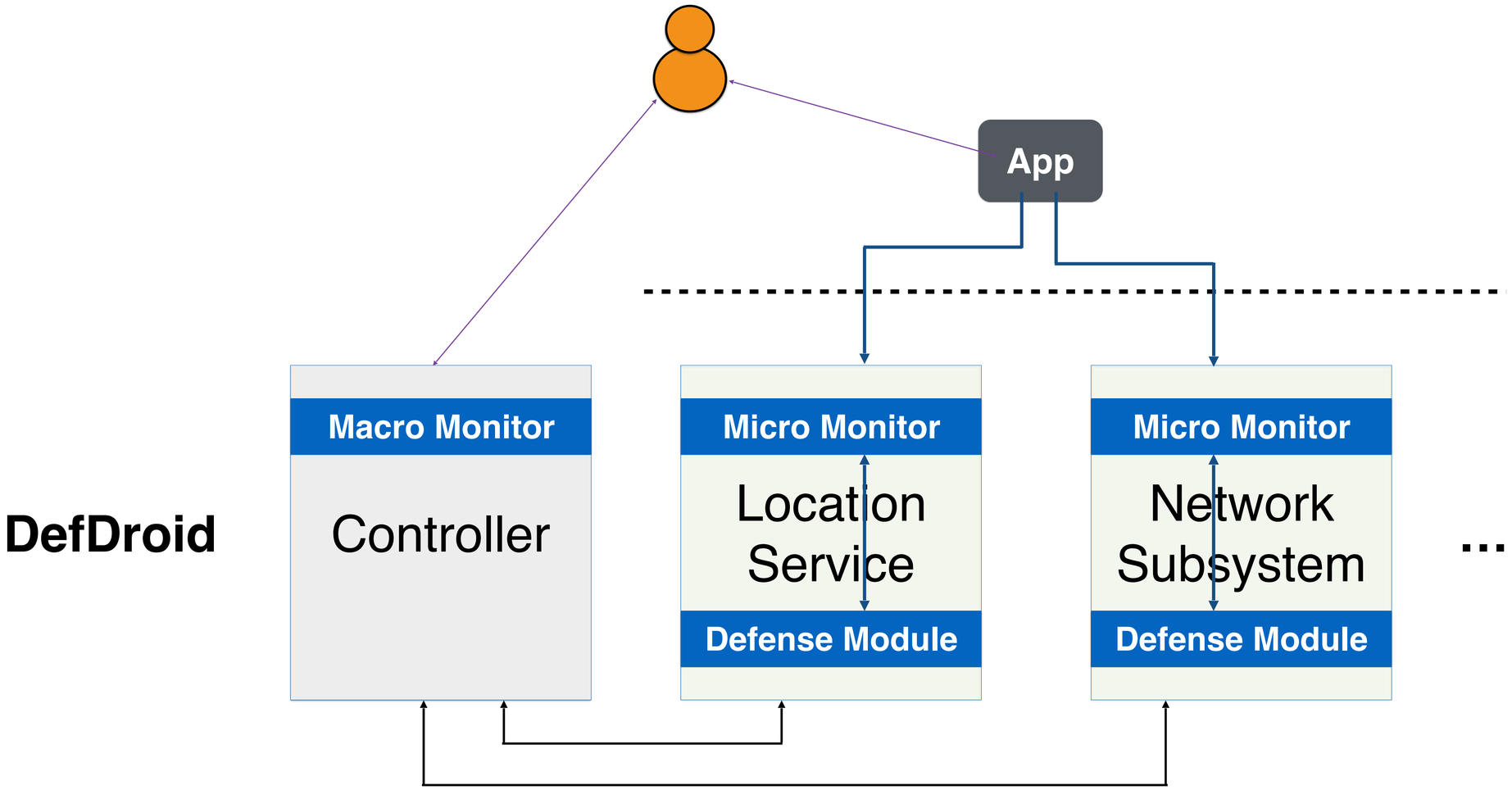


Extensible defense modules

DefDroid



Extensible defense modules



Avoid being over-defensive

Immune

Foreground app user currently interacting with

Avoid being over-defensive

Immune

Foreground app user currently interacting with

Conservative

Criteria of disruptive behavior

Avoid being over-defensive

Immune

Foreground app user currently interacting with

Conservative

Criteria of disruptive behavior

Temporary

Length of DefDroid actions

Avoid being over-defensive

Immune

Foreground app user currently interacting with

Conservative

Criteria of disruptive behavior

Temporary

Length of DefDroid actions

Informative

User interfaces

DefDroid implementation

Built on Android 4.4

- ported to 5.1
- support both AOSP and CM releases

Implemented 11 defense modules

- location, wakelock, sensor, alarm, network, etc.

Evaluation

- How effective is DefDroid in handling real-world DAB?
- What is the impact to app usability?
- How DefDroid performs in the field?
- What is the overhead of DefDroid?

Experimental setup

Devices

- Motorola G, Google Nexus 4

Dataset

- 96 real-world DAB cases reproduced from our study
- 32 new real-world DAB cases we later collected

Overall result

Studied DAB cases

Total	Reacted
96	94

New DAB cases

Total	Reacted
32	31

Reacted means DefDroid took some actions that lead to positive effect (e.g., reduction in power consumption)

Resource consumption reduction

30 Minute Experiment Session

Resource	Reduction		
	Min	**Median**	**Max**
Energy	130 J, 6%	340 J, 21%	866 J, 87%
Cellular data	17 MB, 15%	54 MB, 34%	183 MB, 86%
Storage	30 MB, 25%	48 MB, 37%	145 MB, 90%

Large-scale deployment to PhoneLab

http://phone-lab.org

- **185 users** using Google Nexus 5 running DefDroid
- **43 days** from 2015/09/21 to 2015/11/3
- Took actions on **81** apps in **105** of participants
- Reproduced **57** cases
- No user complaints about DefDroid breaking app usability

Disruptive behavior found in the field

App	Disruptive behavior
cClock	Constant GPS search
The Economist	Keep display on after use
TWC WiFi Finder	Keep (dis)connecting
NY Times	Excessive sensor requests
Sina News	Long-held GPS
Xiami	Excessive wakelock requests
Kik	Frequent wake-up phone
Zillow	Excessive toasts (notifications)

Disruptive behavior found in the field

App	Disruptive behavior
cClock	Constant GPS search
Zillow	Excessive toasts (notifications)

"Biggest issue is the battery drain. This app keeps checking for (hardly) available TWC hotspots, even though I'm already connected to a strong (home/work) wifi. The battery drained from 100% to 15% in 1 to 2 hours."

Overhead

	CPU load	Memory	Power
Android	37.5%	1721.9 MB	1688.4 mW
DefDroid	39.2%	1749.8 MB	1719.8 mW
Overhead	**1.7 ± 0.3%**	**27.9 ± 0.9 MB**	**31.4 ± 7.4 mW**

Conclusion

» Problem Disruptive app behavior widely exists in the field, frustrating users

» Study Characteristics of 287 real-world issues of disruptive apps

» Solution DefDroid: system-level approach to curb disruptive apps at runtime

» Evaluation real-world cases, deployed to 185 users, found new issues

http://defdroid.org

Limitations

- Different DefDroid modules are implemented separately
- May overrule user configurations for an app
- Policies of defense mechanisms are static
- New unknown DAB patterns

Mobile apps often less mature

Top 1,000 mobile apps

Platform	Avg. age	% of apps by individual developers
App Store	2.5 years	> 12 %
Google Play	1.5 years	> 5%

Popular desktop applications

Application	History
MySQL	19 years
Firefox	12 years
Chrome	7 years
…	

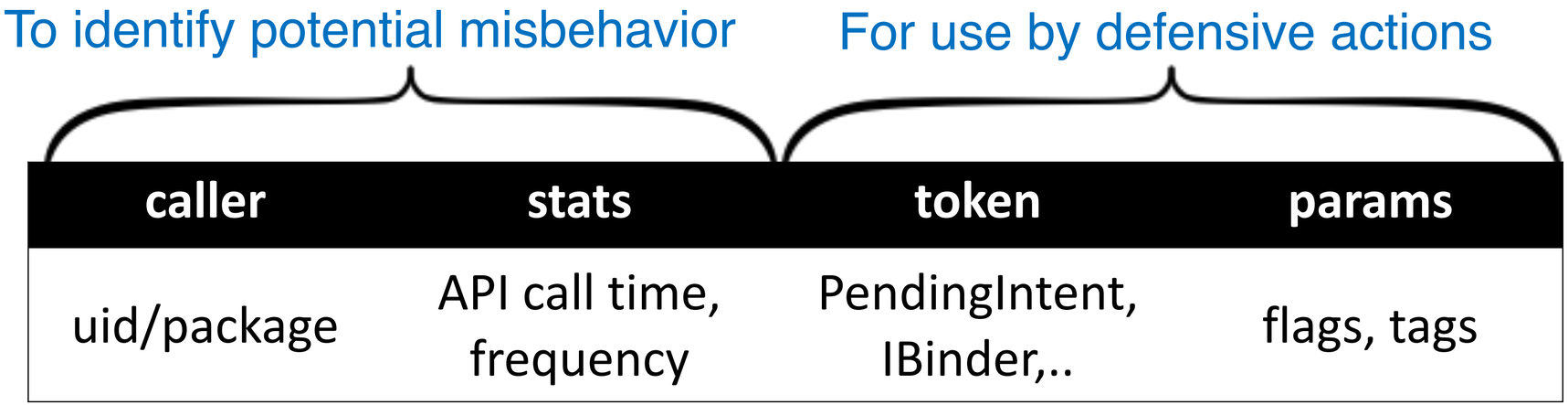
Monitors

System-level conditions

- battery, storage usage, network condition, etc.

App-level activities

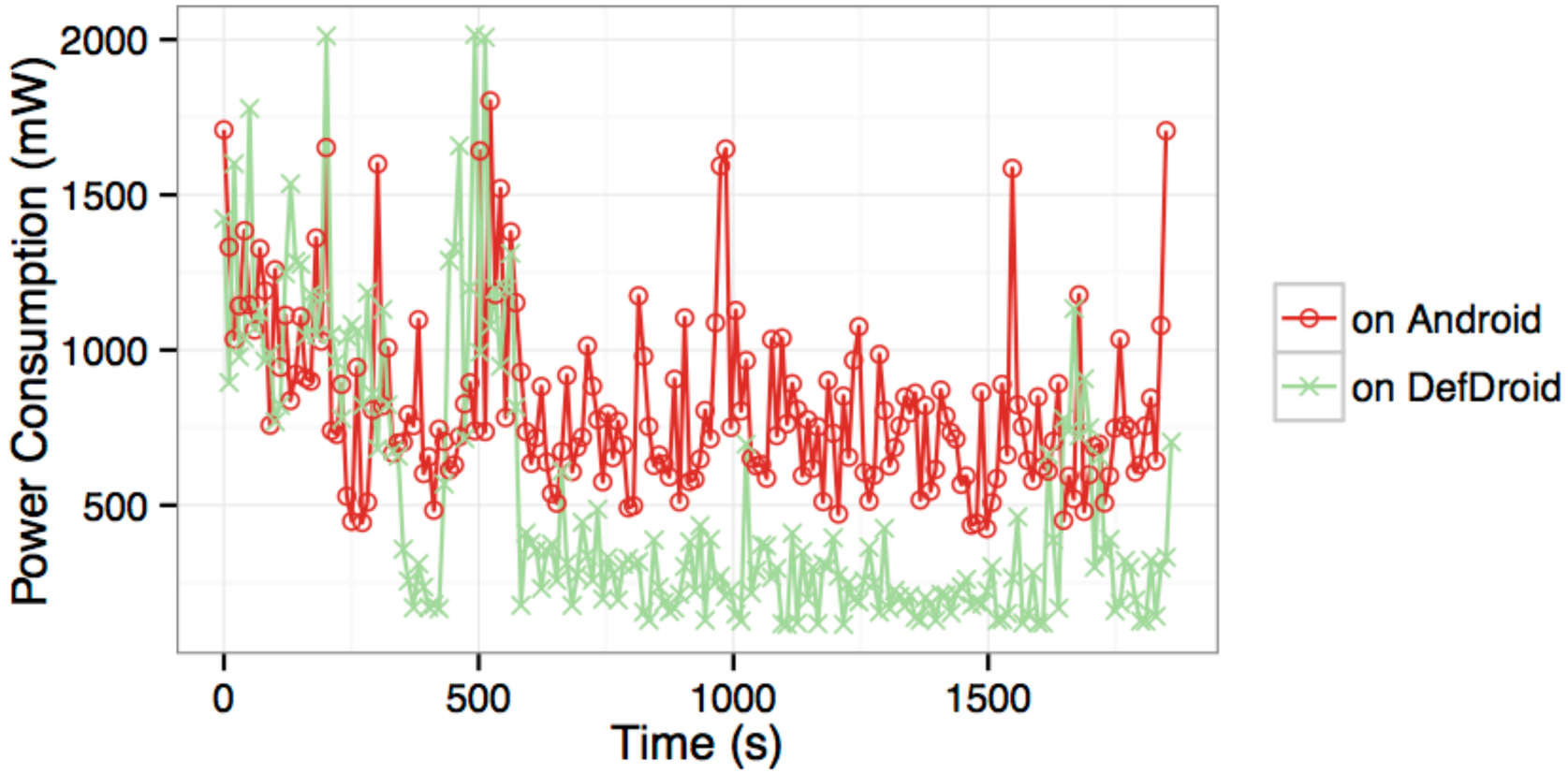
- important API calls, stats, tokens, etc.



Defensive actions

Action type	Example
Release	release long-held wakelock, GPS
Slowdown	enlarge alarm interval, reduce sensor frequency
Delay	delay excessive cellular network requests
Approximate	use coarse-grained location
Downgrade	switch to passive GPS provider
Deprioritize	decrease scheduling priority
Block	reject frequent alarm requests for a while
Warn	warn about large data consumption
Kill	kill naughty app processes or services

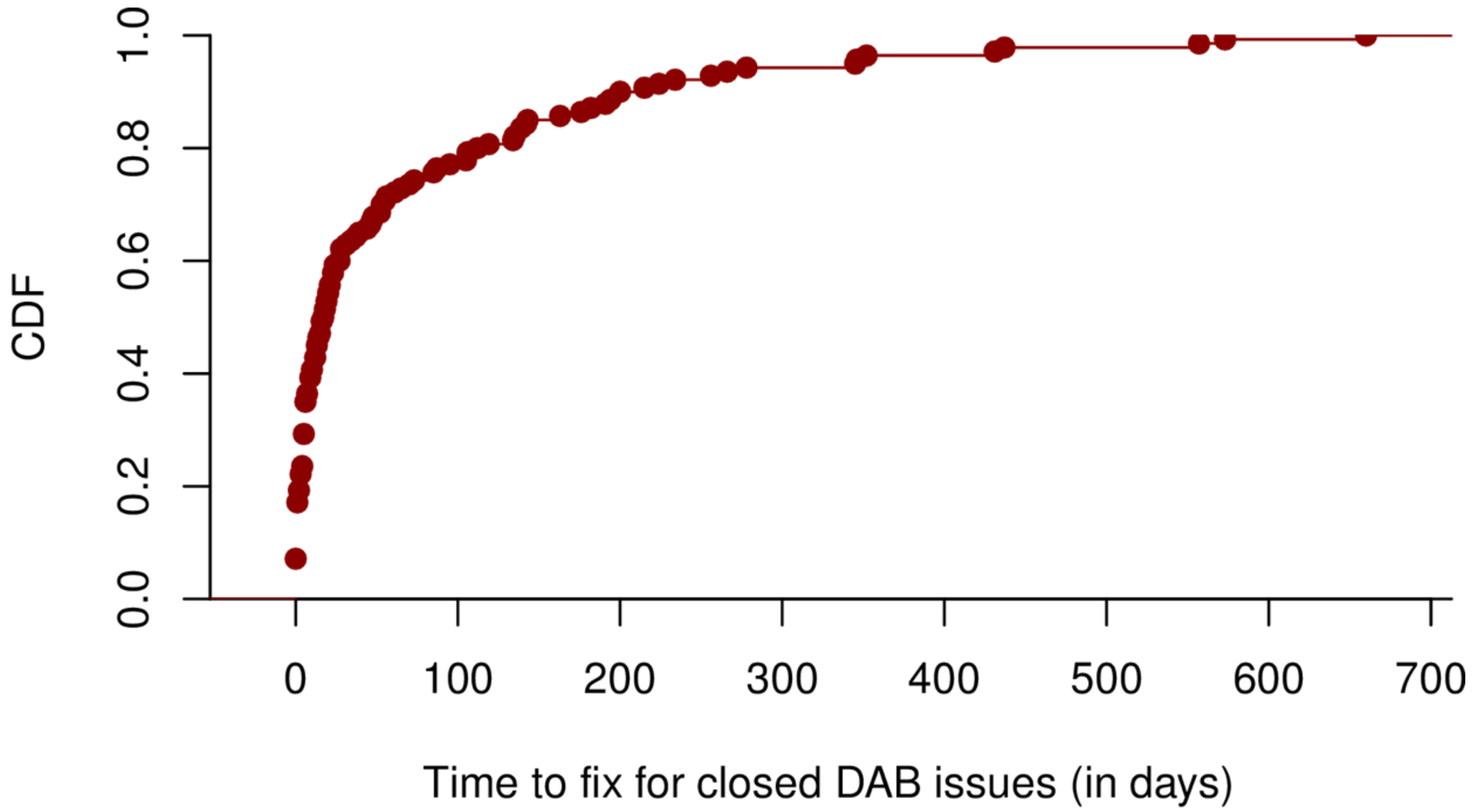
Example run of a disruptive app



Common patterns of disruptive app behavior

Pattern	Cases
Wakelock leak or overuse	25
GPS leak or overuse	19
Sensor leak or overuse	12
Other resource leak or overuse	15
Frequent wake-up alarms	12
Frequent broadcasts, receivers	6
Aggressive sensor/GPS updates	15
Frequent connection, aggressive retry	44
Excessive cellular data transfer	42
Excessive storage use	33
High CPU usage	31
Excessive or stuck notifications	33

Time to fix for closed DAB issues



Impact to app execution

Impact	cases
No impact	40
Deferred execution	49
Failed requests	31
Checked exceptions	5
Termination	0