UNIVERSITY OF CAMBRIDGE

Measuring security and cybercrime

Daniel R. Thomas

Cambridge Cybercrime Centre, Department of Computer Science and Technology, University of Cambridge, UK

SecHuman 2018

GPG: 5017 A1EC 0B29 08E3 CF64 7CCD 5514 35D5 D749 33D9

Firstname.Surname@cl.cam.ac.uk

Format

1. Group warm up (5 minutes)
2. Short lecture (35 minutes).
3. Experimental design and review (50 minutes)
   - 3.1 Designing an experiment to measure security or cybercrime (30 minutes)
   - 3.2 Plenary feedback (20 minutes)

What is security and how to we measure it?

- Discuss in groups for 2 minutes
- Then we will listen to some of the ideas

Measuring security and cybercrime is important

- Is security getting better or worse?
- Did this intervention work?
- Is there a difference in security between these products?

Two examples of security measurement research

- Measuring security of Android
- Measuring DDoS attacks (cybercrime)

Drawing out the principles, insights, and mistakes as we go along.

Security metrics for the Android ecosystem[1]

https://androidvulnerabilities.org/

Daniel R. Thomas

Andrew Rice

Alastair R. Beresford

Daniel Wagner

[1] Daniel R. Thomas, Alastair R. Beresford, and Andrew Rice. 2015. Security metrics for the Android ecosystem. In *ACM CCS workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)*. ACM, Denver, Colorado, USA, (Oct. 2015), 87–98. ISBN: 978-1-4503-3819-6.

Smartphones contain many apps written by a spectrum of developers

How “secure” is a smartphone?

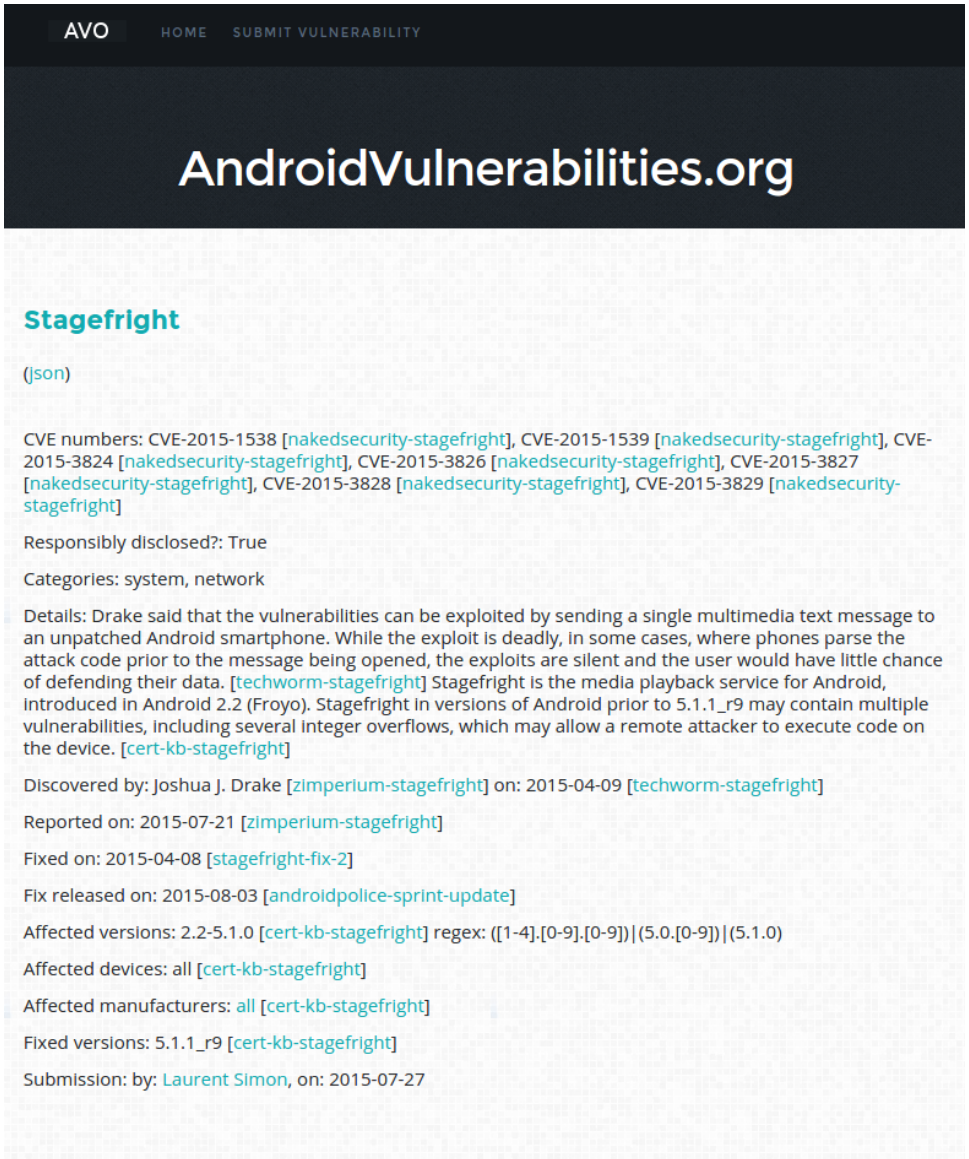
Root/kernel exploits are harmful

- Root exploits break permission model
- Cannot recover to a safe state
- In 2012 37% Android malware used root exploits
- We're interested in critical vulnerabilities, exploitable by code running on the device

Hypothesis: devices vulnerable because they are not updated

- Anecdotal evidence was that updates rarely happen
- Android phones, sold on 1-2 year contracts

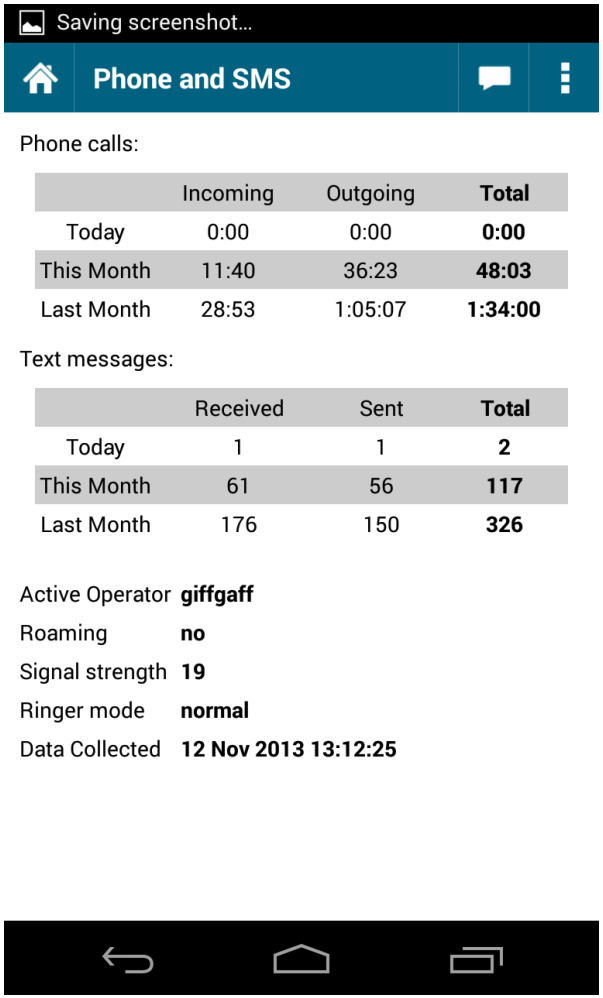
No central database of Android vulnerabilities: so we built one



Device Analyzer gathers statistics on mobile phone usage

- Deployed May '11
- 30 000 contributors
- 4 000 phone years
- 180 billion records
- 10TB of data
- 1089 7-day active contributors

(2015 numbers)



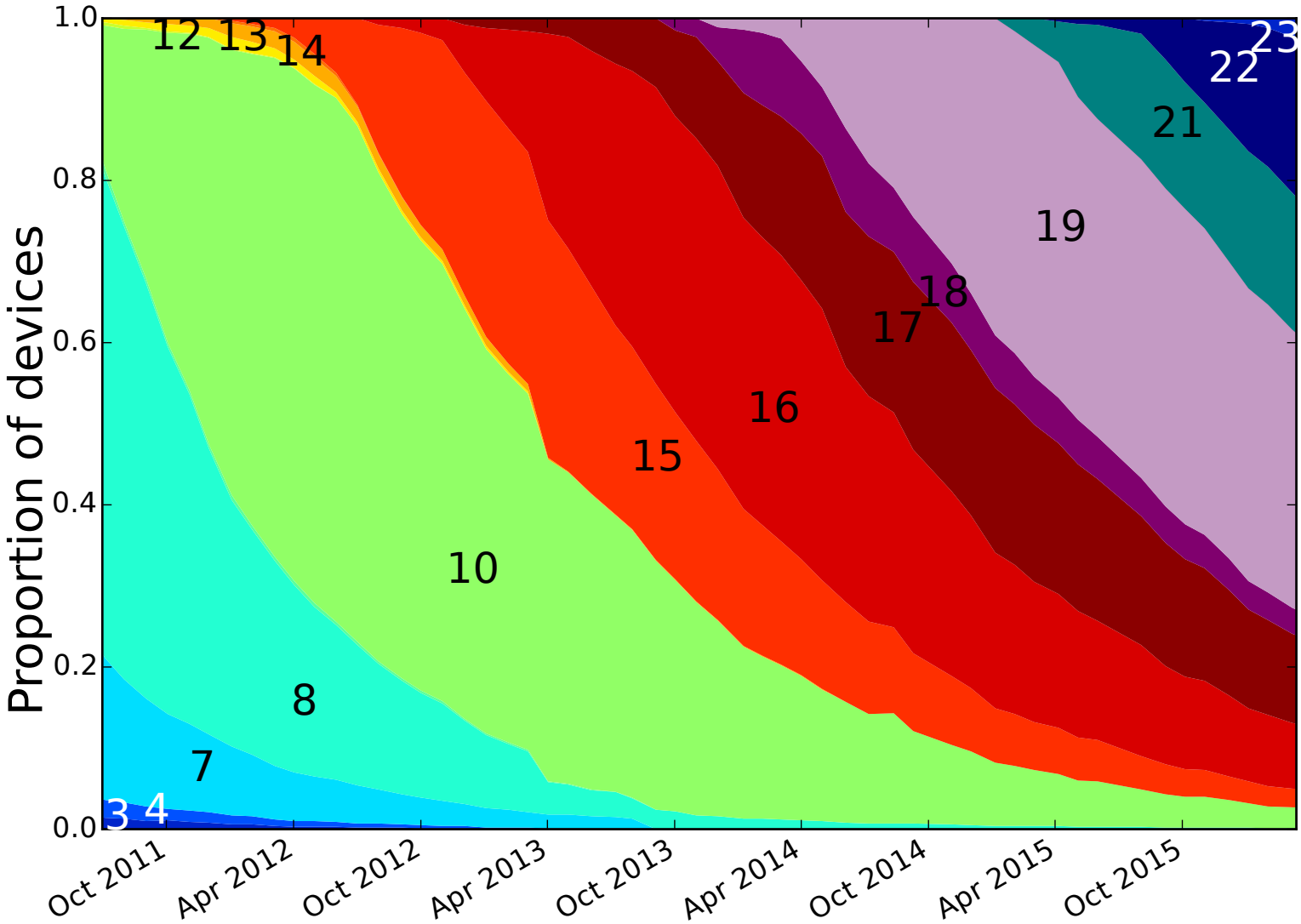
Device Analyzer gathers wide variety of data

Including: system statistics

- OS version and build number
- Manufacturer and device model
- Network operators

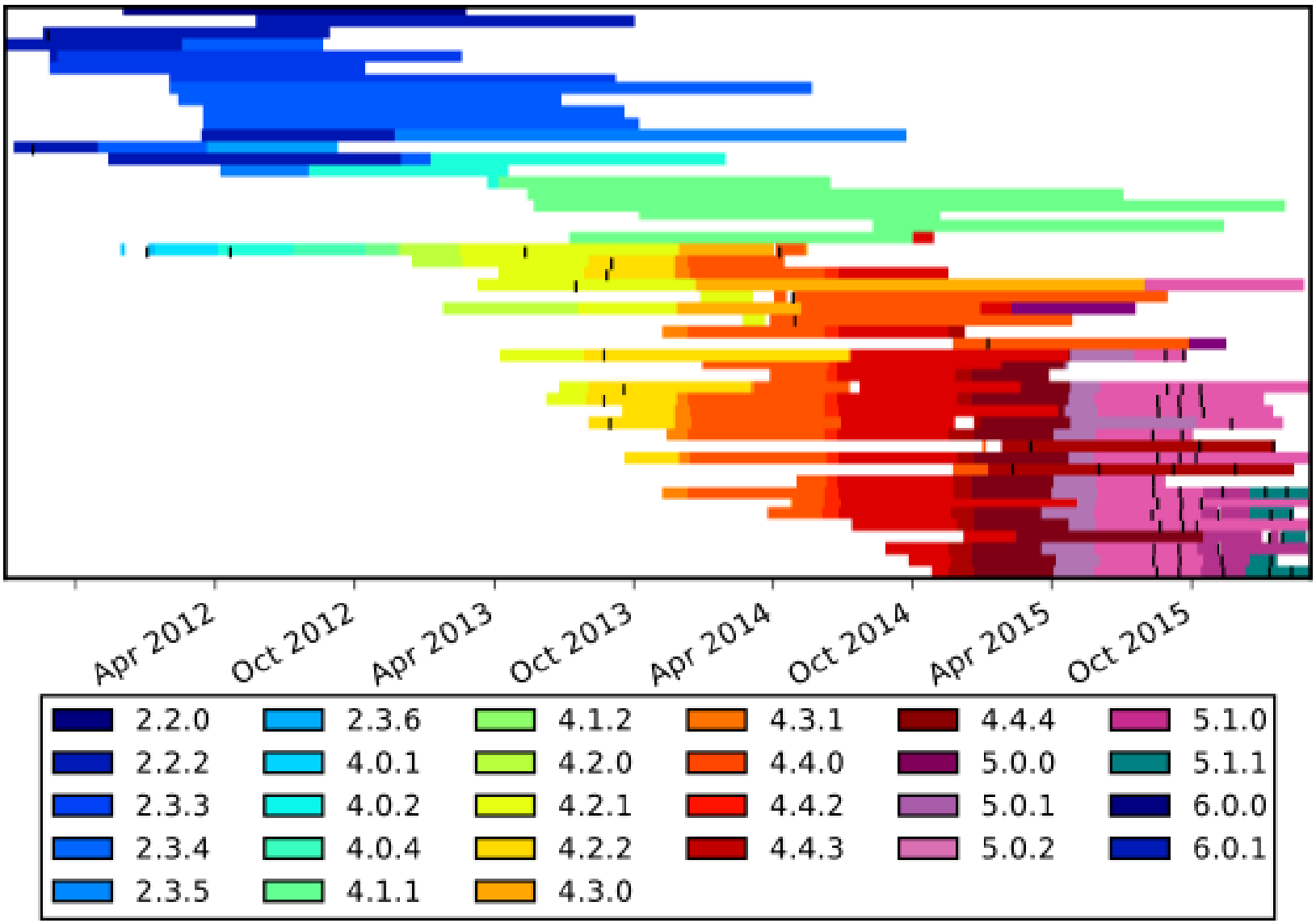
Is the *ecosystem* getting updated?

Google data: device API levels



Are *devices* getting updated?

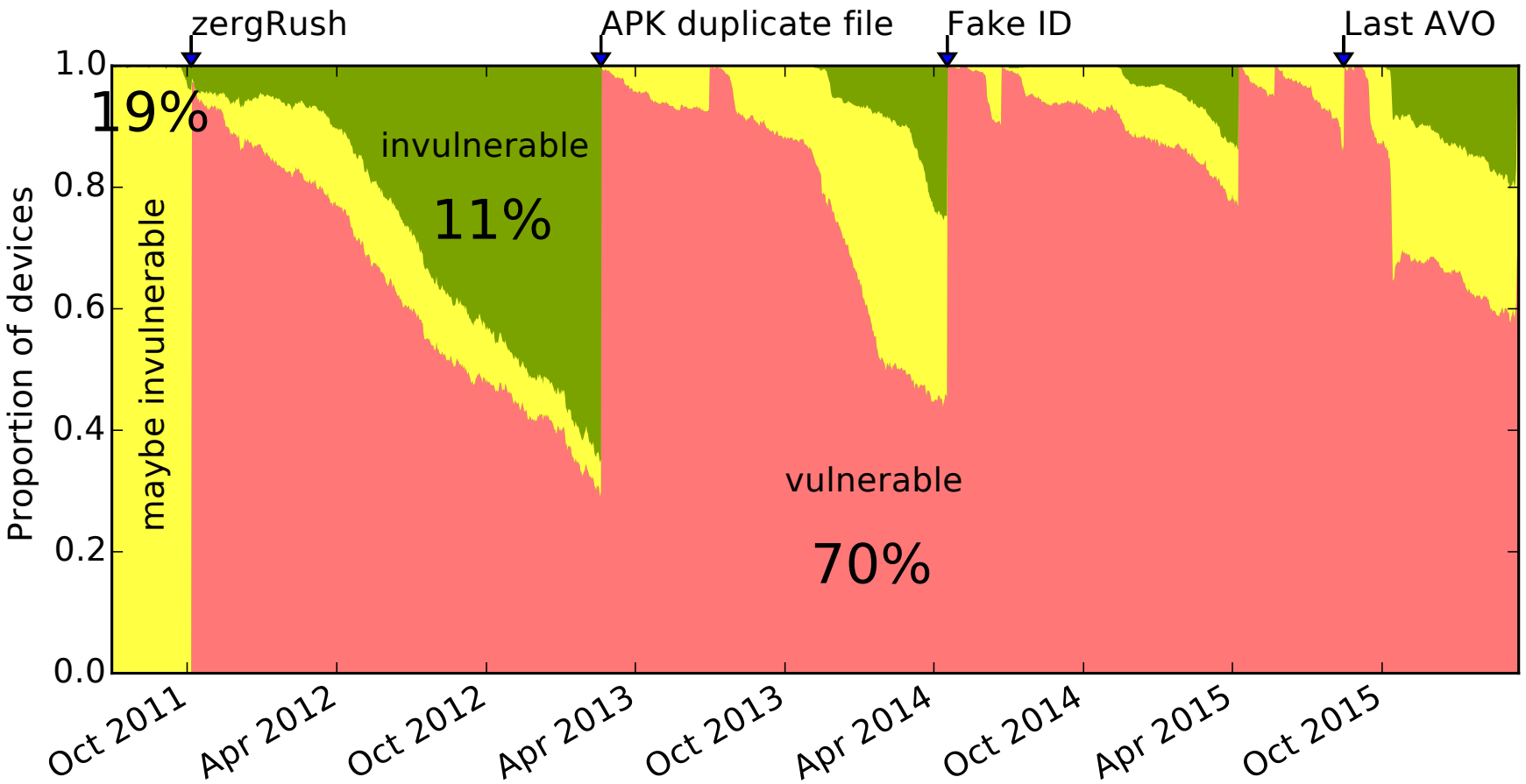
LG devices by OS version



Connecting the two data sets: assume OS version $\rightarrow$ vulnerability

- We have an OS version from Device Analyzer
- We have vulnerability data with OS versions
- Match on OS and Build Number and assign:
  - Vulnerable
  - Maybe invulnerable
  - Invulnerable (not known vulnerable)

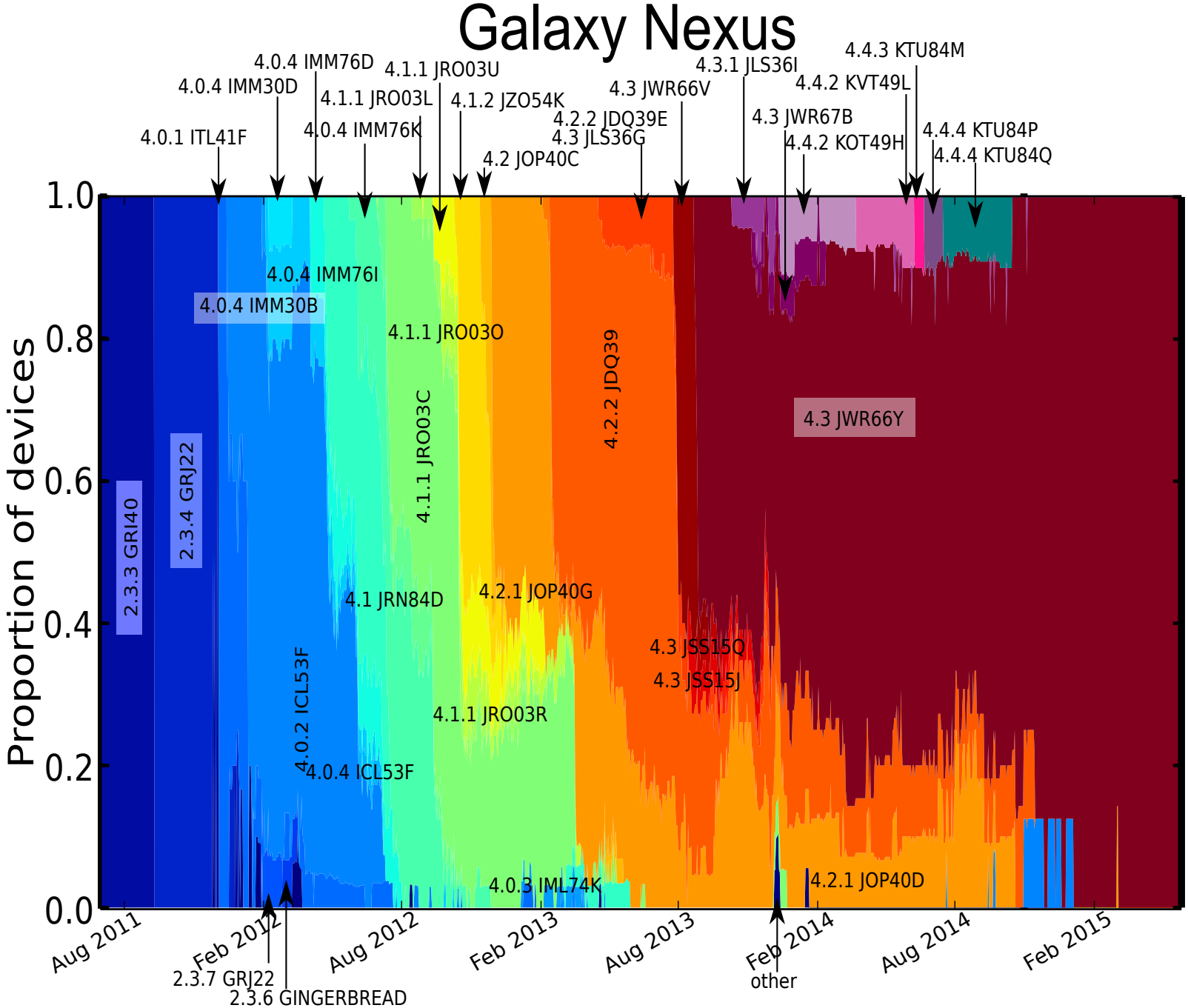
Vulnerability varies over time



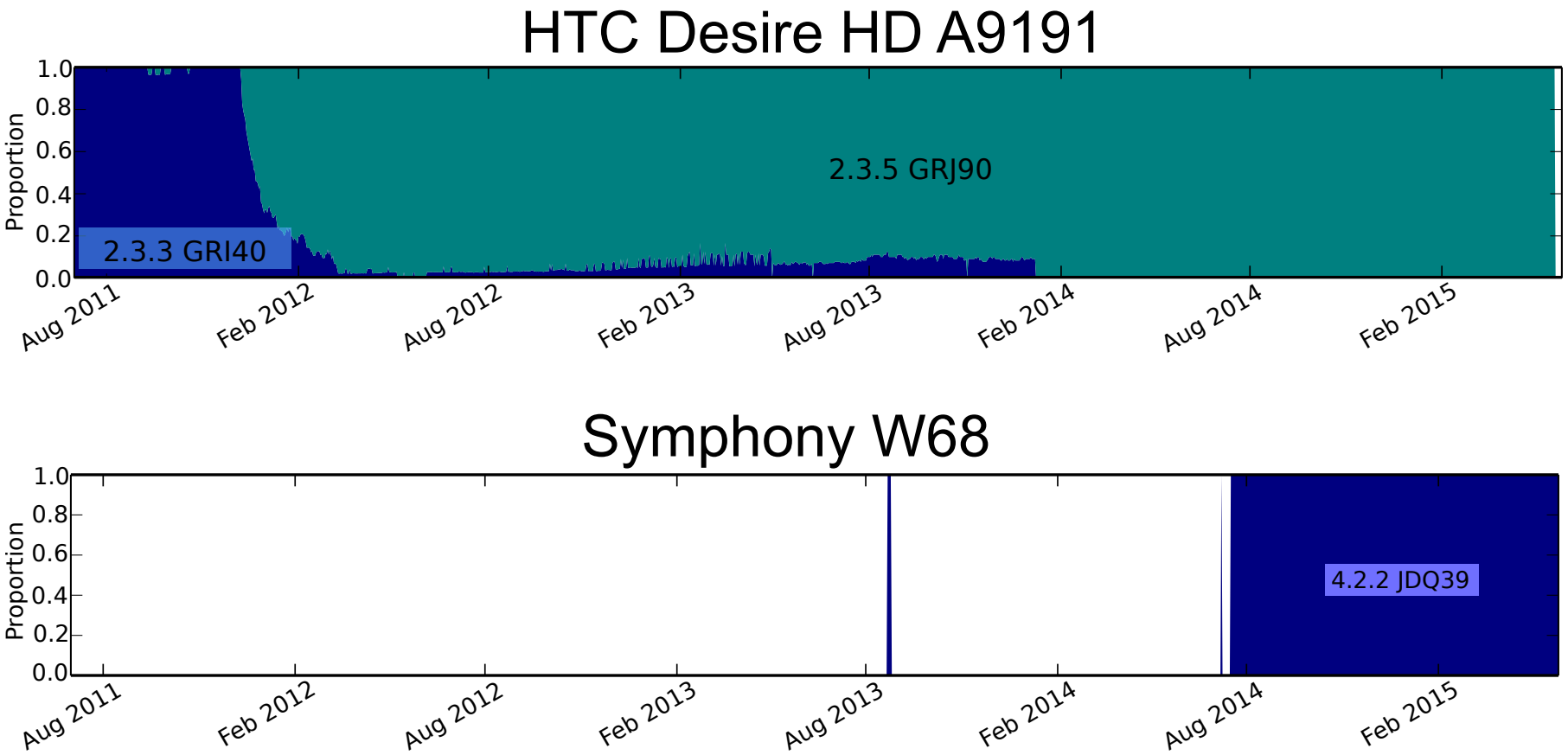
The FUM metric measures the security of Android devices

$$\mathrm{FUM} = 4f + 3u + 3\frac{2}{1 + e^{m}}$$

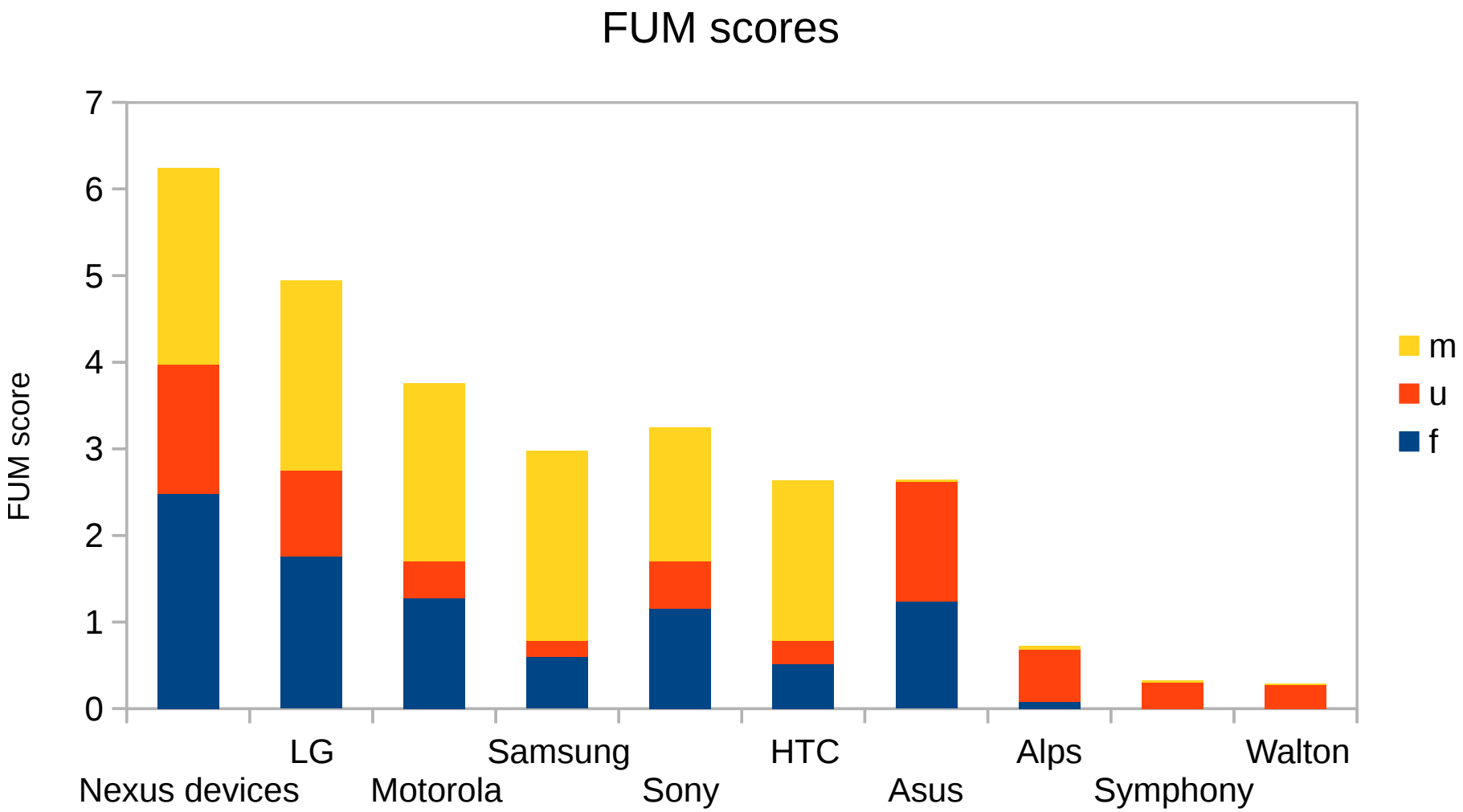
*f*ree from (known) vulnerabilities
*u*pdated to the latest version
*m*ean unfixed vulnerabilities


Galaxy Nexus
4.0.1 ITL41F
4.0.4 IMM30D
4.0.4 IMM76D
4.0.4 IMM76K
4.1.1 JRO03L
4.1.1 JRO03U
4.1.2 JZO54K
4.2 JOP40C
4.2.2 JDQ39E
4.3 JLS36G
4.3 JWR66V
4.3.1 JLS36I
4.3 JWR67B
4.4.2 KOT49H
4.4.2 KVT49L
4.4.3 KTU84M
4.4.4 KTU84P
4.4.4 KTU84Q
Proportion of devices
1.0
0.8
0.6
0.4
0.2
0.0
4.0.4 IMM76I
4.0.4 IMM30B
4.1.1 JRO03O
4.1.1 JRO03C
4.2.2 JDQ39
4.3 JWR66Y
2.3.4 GRJ22
2.3.3 GRI40
4.1 JRN84D
4.2.1 JOP40G
4.3 JSS15Q
4.3 JSS15J
4.0.2 ICL53F
4.1.1 JRO03R
4.0.4 ICL53F
4.0.3 IML74K
4.2.1 JOP40D
Aug 2011
Feb 2012
Aug 2012
Feb 2013
Aug 2013
Feb 2014
Aug 2014
Feb 2015
2.3.7 GRJ22
2.3.6 GINGERBREAD
other

Lack of security updates



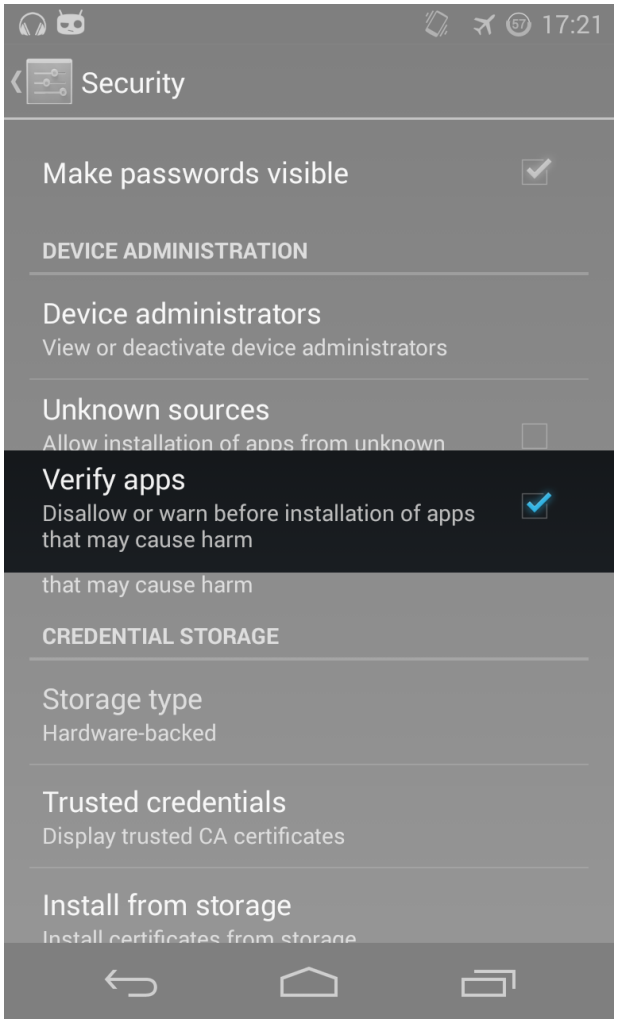
Comparing manufacturers



Why is fixing vulnerabilities hard: software ecosystem is complex

- Division of labour
  - Open source software
  - Core OS production
  - Driver writer
  - Device manufacturer
  - Retailer
  - Customer
- Apple and Google have different models
  - Hypothesis: Apple's model is more secure

Google to the rescue



- Play Store
- Verify apps
- Android Security Patch Level
- Later: Android Enterprise Recommended

What happened next?

- Plenty press coverage
- Contacts with Google, manufacturers, UK Home Office
- FTC cites work.
- Google uses graphs to pressure manufacturers to improve update provision
- We move on: no further collection of vulnerability data, no updated scores.

1000 days of UDP amplification DDoS attacks[2]

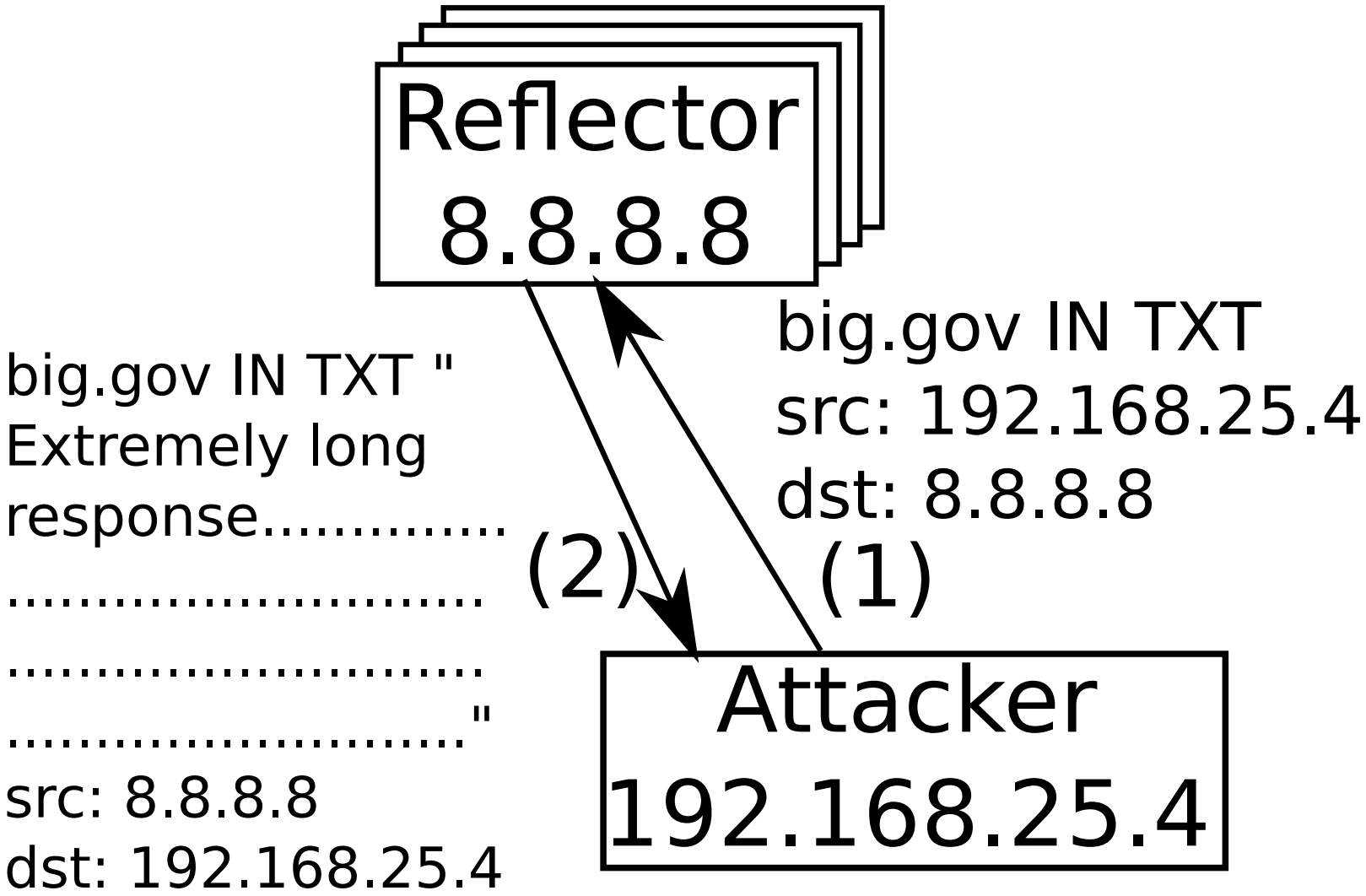
Daniel R. Thomas

Richard Clayton

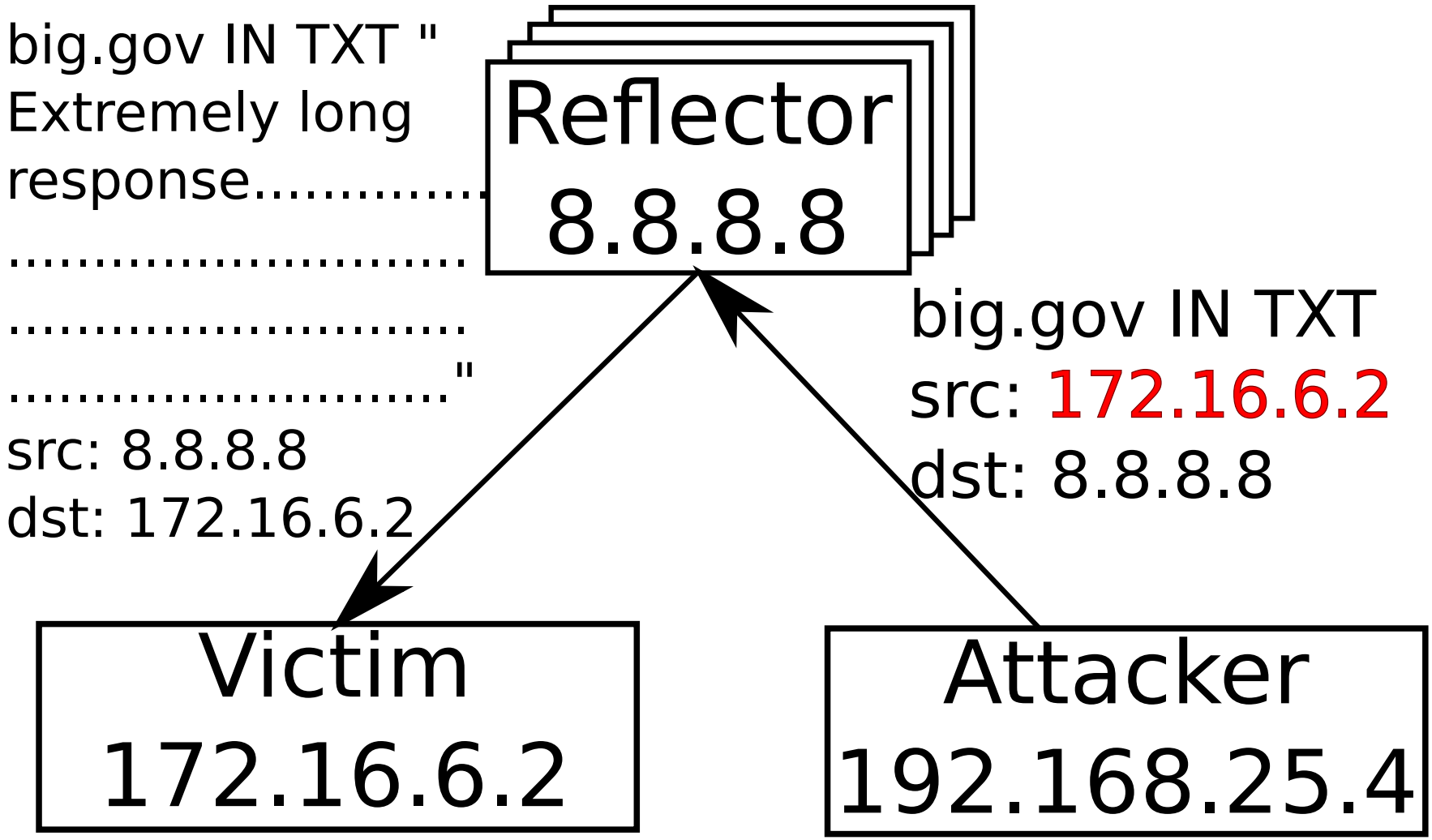
Alastair R. Beresford

[2]Daniel R. Thomas, Richard Clayton, and Alastair R. Beresford. 2017. 1000 days of UDP amplification DDoS attacks. In *APWG Symposium on Electronic Crime Research (eCrime)*. IEEE, (Apr. 2017).

UDP scanning



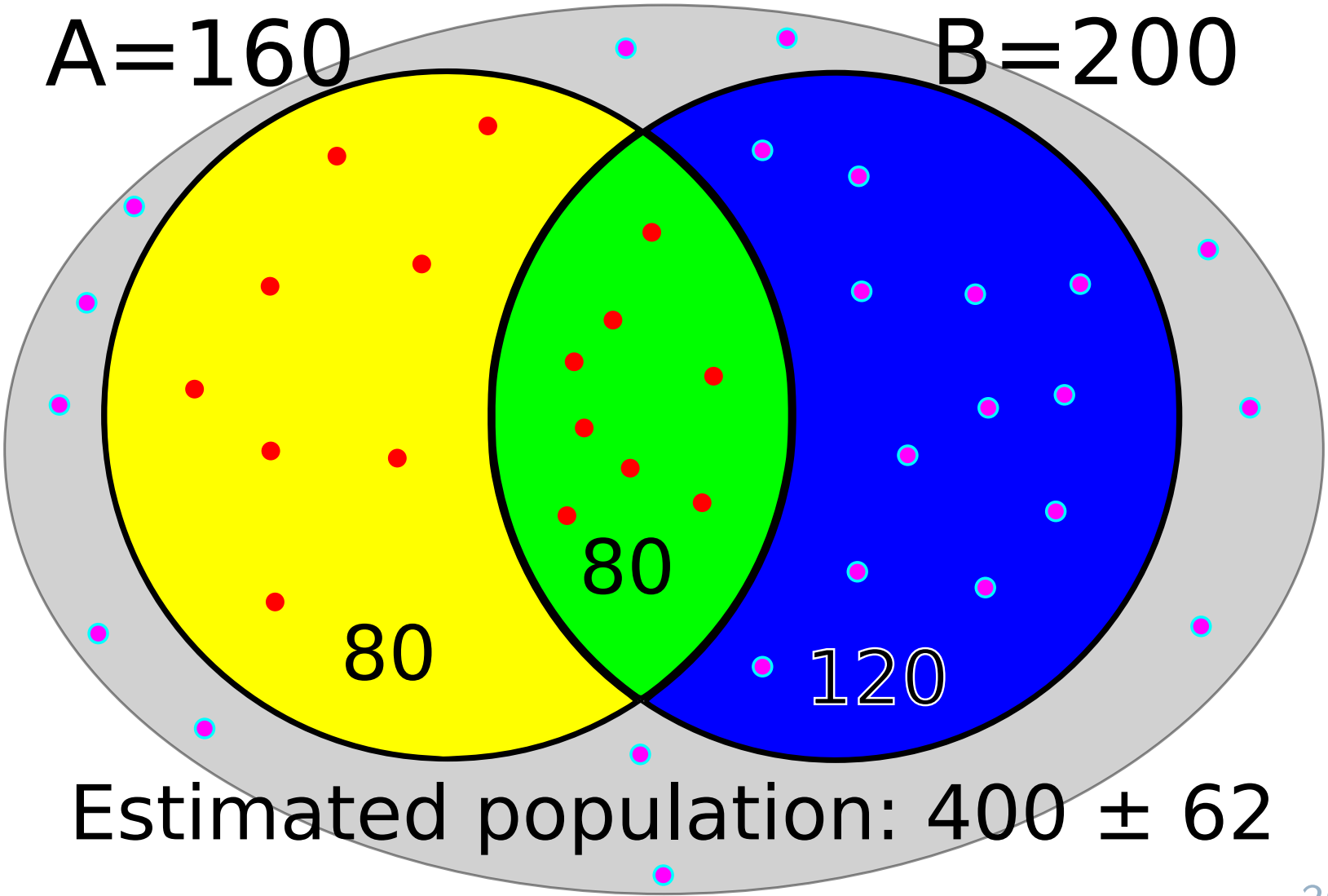
UDP reflection DDoS attacks



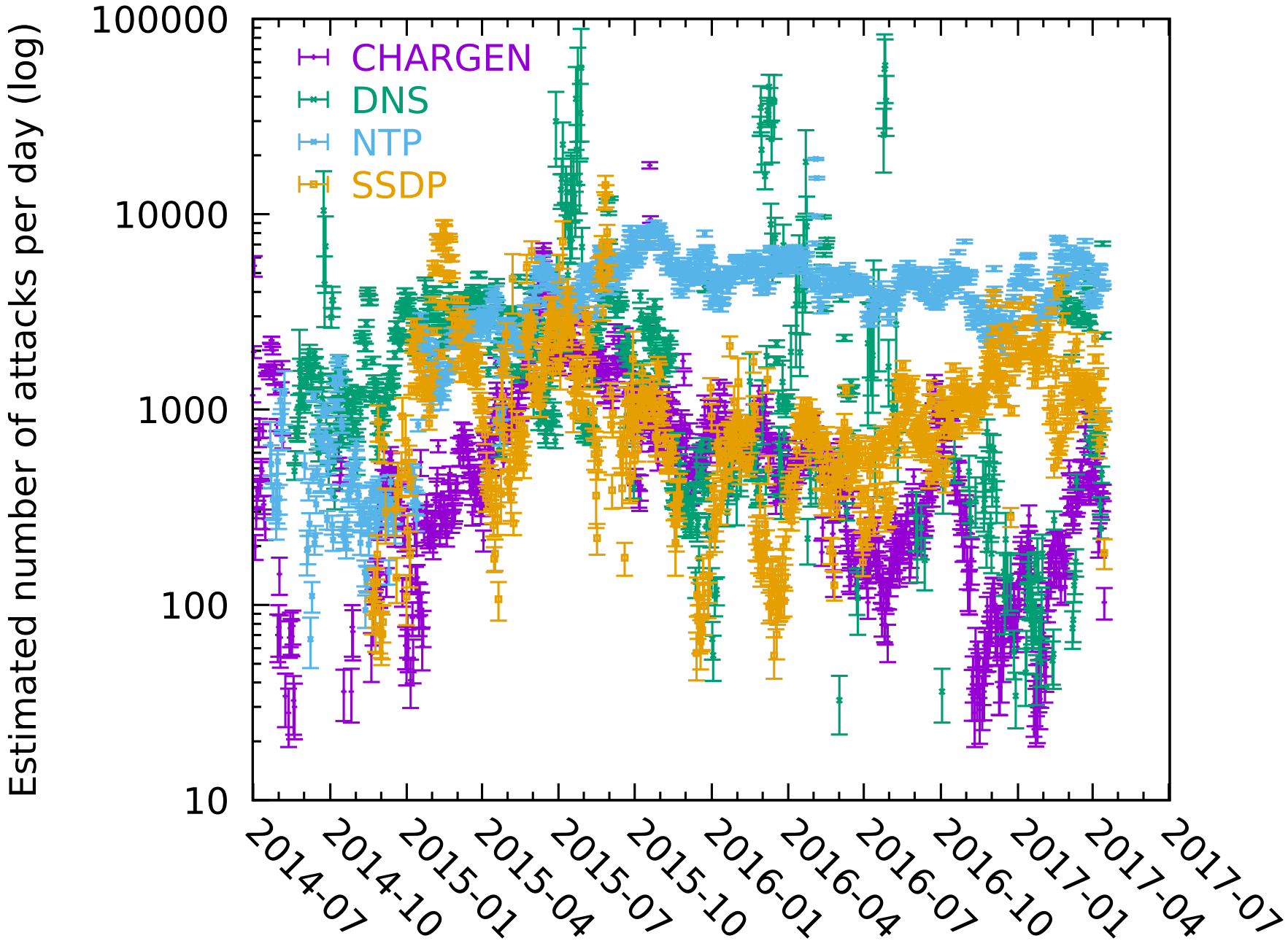
We run lots of UDP honeypots

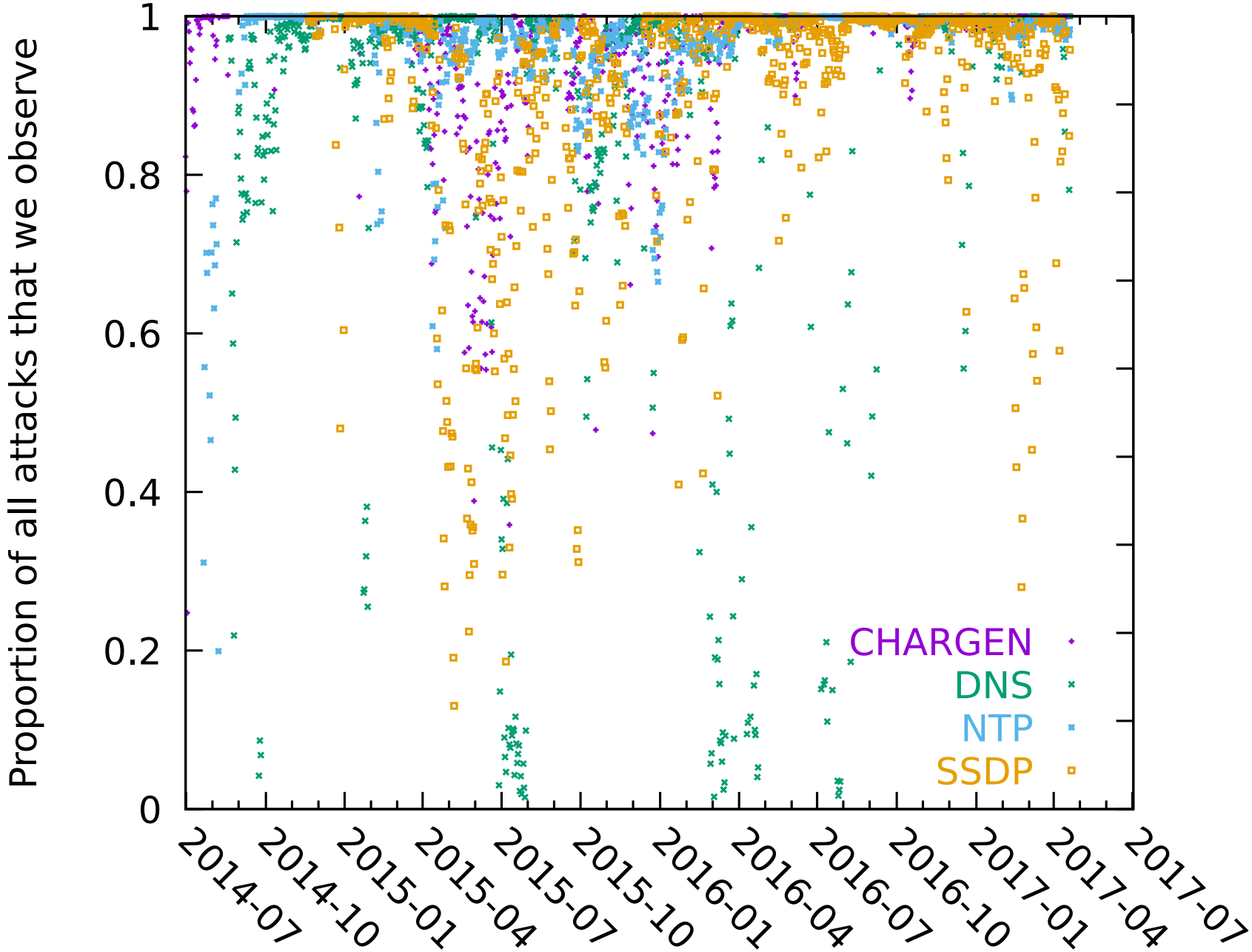
- Median 65 nodes since 2014
- Hopscotch emulates abused protocols
  QOTD, CHARGEN, DNS, NTP, SSDP, SQLMon, Portmap, mDNS, LDAP
- Sniffer records all resulting UDP traffic
- (try to) Only reply to black hat scanners

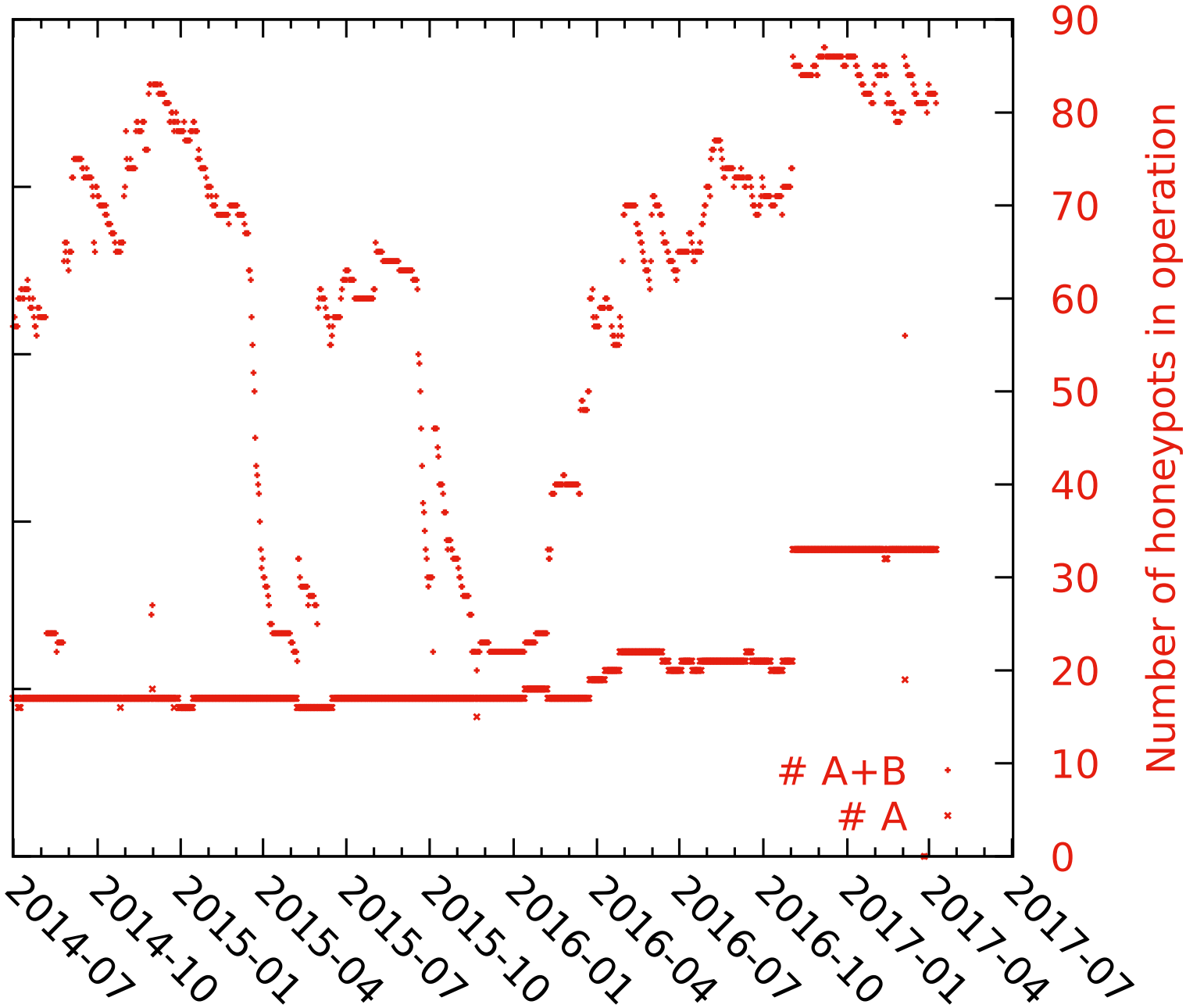
Total attacks estimated using capture-recapture

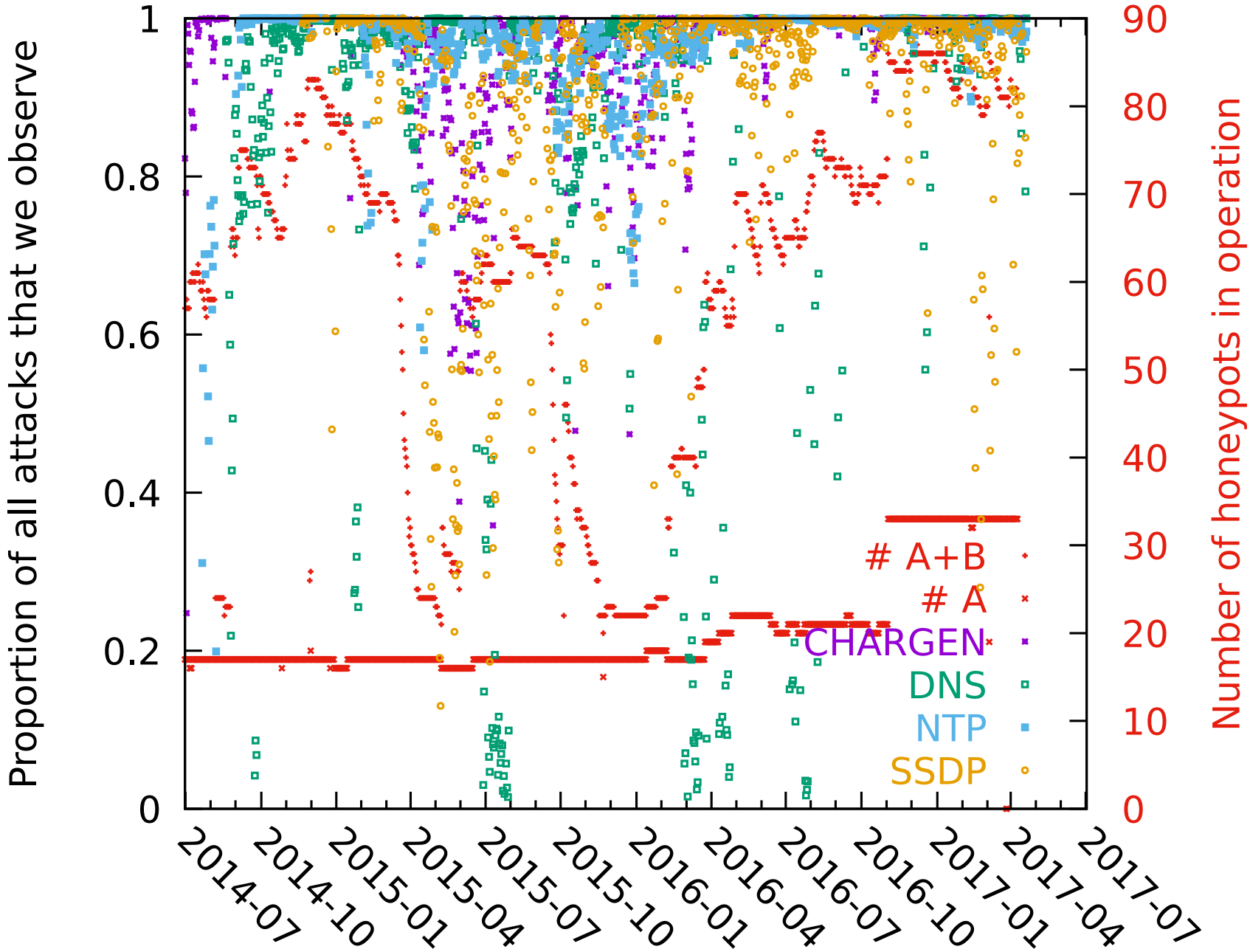


Estimated population: 400 ± 62


Estimated number of attacks per day (log)
100000
10000
1000
100
10
CHARGEN
DNS
NTP
SSDP
2014-07
2014-10
2015-01
2015-04
2015-07
2015-10
2016-01
2016-04
2016-07
2016-10
2017-01
2017-04
2017-07


Proportion of all attacks that we observe
1
0.8
0.6
0.4
0.2
0
2014-07
2014-10
2015-01
2015-04
2015-07
2015-10
2016-01
2016-04
2016-07
2016-10
2017-01
2017-04
2017-07
CHARGEN
DNS
NTP
SSDP


90
80
70
60
50
40
30
20
10
0
Number of honeypots in operation
A+B
A
2014-07
2014-10
2015-01
2015-04
2015-07
2015-10
2016-01
2016-04
2016-07
2016-10
2017-01
2017-04
2017-07


Proportion of all attacks that we observe
Number of honeypots in operation
1
0.8
0.6
0.4
0.2
0
90
80
70
60
50
40
30
20
10
0
2014-07
2014-10
2015-01
2015-04
2015-07
2015-10
2016-01
2016-04
2016-07
2016-10
2017-01
2017-04
2017-07
A+B
A
CHARGEN
DNS
NTP
SSDP

This was ethical

- We reduce harm by absorbing attack traffic
- We don't reply to white hat scanners (no timewasting)
- We used leaked data for validation, this was necessary and did not increase harm.
- Further discussion of the ethics of using leaked data for research tomorrow.

This is a solvable problem

- BCP38/SAVE
- Follow the money
- Enforce the law
- Warn customers it is illegal

Experimental design [30 minutes]

How would you measure the relative security of different:

BO	Banks
BOT	CPU vendors
DO	Residential ISPs
DU	Operating systems
E	Cycle lock manufacturers
GE	IoT manufacturers
HER	Offices
MH	Elections
OB	Online payment providers
RE	Smartphones

What data would you need to collect?

How would you collect it?

Would it be possible to cheat your measurement without actually improving security?

Plenary discussion [20 minutes]

Feedback from each group on their experimental design.

Thank you! Questions?

Daniel R. Thomas
Daniel.Thomas@cl.cam.ac.uk
@DanielRThomas24
https://www.cl.cam.ac.uk/~drt24/
5017 A1EC 0B29 08E3 CF64 7CCD 5514 35D5 D749 33D9

Daniel Thomas is supported by the EPSRC [grant number EP/M020320/1].

References I

[1] Daniel R. Thomas, Alastair R. Beresford, and Andrew Rice. 2015. Security metrics for the Android ecosystem. In *ACM CCS workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)*. ACM, Denver, Colorado, USA, (Oct. 2015), 87–98. ISBN: 978-1-4503-3819-6.

[2] Daniel R. Thomas, Richard Clayton, and Alastair R. Beresford. 2017. 1000 days of UDP amplification DDoS attacks. In *APWG Symposium on Electronic Crime Research (eCrime)*. IEEE, (Apr. 2017).