TEU00311

What is the Internet doing to me? (witidtm)

Stephen Farrell
stephen.farrell@cs.tcd.ie

https://github.com/sftcd/witidtm
https://down.dsg.cs.tcd.ie/witidtm

TRINITY COLLEGE DUBLIN
COLÁISTE NA TRÍONÓIDE, BAILE ÁTHA CLIATH
THE UNIVERSITY OF DUBLIN

Testing Apps for COVID-19 Tracking (TACT) Project

- Joint project since April 2020 with Dr. Doug Leith, also from TCD comp sci (Doug did most of the clever bits)
  - https://down.dsg.cs.tcd.ie/tact/
- This talk mostly based on one from July 2020
  - https://down.dsg.cs.tcd.ie/tact/tact-pearg-pressie.pdf
- There’s even a video of that
  - https://www.youtube.com/watch?v=Yva3py95H98&list=PLC86T-6ZTP5hG1r1T0vHX9uIm-CYjYcbA&index=14&t=344s
- More background added here

Background

- So far during the COVID-19 pandemic, the Irish Government and Health Services Executive (HSE) are locally perceived to have done a pretty good job managing the pandemic and already-stressed health care services
- A March 2020 paper (*) asserted that: “A mobile phone App can make contact tracing and notification instantaneous upon case confirmation.”
- Most people would like that to be true
- We were unsure if that was true or not, nor what privacy/security consequences might flow from population-scale uses of such Apps
- Trinity College Dublin (our employer) announced quick-turnaround funding for projects related to the pandemic (mostly aimed at medics)
- We applied for, and got funding for, Testing Apps for COVID-19 Tracing (TACT)

(*) Ferretti, Luca, et al. "Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing." Science 368.6491 (2020).

Caveats

- Many decisions, esp. earlier ones, had to be made quickly in the face of significant uncertainty
- Lots has changed since March 2020, and more change seems likely
- I believe everyone with whom we've interacted on this so far is trying to do good
- But: it's also important to discover and then improve/fix things that need fixing

Overview

- Non-Bluetooth Methods
- Bluetooth proximity concept
- The Google/Apple Exposure Notification (GAEN) system
- A replay attack
- Is BLE proximity detection effective?
- What traffic do GAEN Apps send? (on Android)
- Measuring deployments
- Conclusion

Non-Bluetooth Contact Tracing Methods

- “Traditional” phone/interview based methods
- GPS location based
  - Accuracy is an issue but no signal indoors is a killer
  - Severe privacy issues as Public Health Authority (PHA) gets to track all
- Location QR-code scanning
  - Scan QR code on wall, PHA tells everyone time/location of potential close contacts
  - Tried in New Zealand and also in UK (alongside Bluetooth)
- S. Korea: use CCTV & payment transaction details to determine who was a close contract to known cases
- Note that Chinese-style “apps displays green-tick or you’re not getting in” isn’t really contact tracing

Bluetooth Proximity

- Ferretti's paper wasn't the 1st to suggest using Bluetooth (BT) for contact tracing:
  - Yoneki, Eiko. "Fluphone study: Virtual disease spread using haggle." Proceedings of the 6th ACM Workshop on Challenged Networks. 2011.
- Basic idea: Radio signals weaken with distance from transmitter so might be usable to detect proximity
- BT is a short-range radio technology that's supported in almost all phones, in particular Bluetooth Low Energy (BLE) is supported in many modern phones
- BLE has a "beacon" feature where a phone can regularly transmit a beacon value, and all others in range can receive that value
- When a phone receives a BLE beacon, it also records the "Received Signal Strength Indicator" (RSSI) value, measured in decibels (dB)
  - -100 dB will represent "far", -50 would be "near" – decibel scale is logarithmic
- **NB:** BT is designed for data transfer (e.g. with headset) and **not** for distance estimation

BLE for Contact Tracing Concept

- Try get everyone in a region with a BLE capable phone to install the relevant app
- Phones regularly (say 4/second) transmit beacons that have privacy-friendly values
- Phones listen for and record time, beacon and RSSI values for 14 days – typically for 4 seconds every 4 minutes to conserve power (listening all the time consumes more power than briefly turning on transmitter to send a beacon)
- If someone tests positive for COVID-19, they trigger a public-health-authority (PHA) server to publish a set of “keys” (usually one “key” for each of the last 14 days) so that...
- Each “key” can be matched with stored beacon values if the two phones were nearby on that day
- Phones therefore also regularly download all the “keys” related infected folks
- If there’s a match between a “key” and a beacon value, and if the RSSI indicates the phones were “close enough for long enough” then the app alerts the user that they were a “close contact”
- Device owner should then go get tested, isolate etc.

Initial Issues

- How much trust are device owners putting in PHA?Mostly comes down to how to generate keys – centrally or on-device
  - Centrally generated keys are sent (in batches) to device and beacon values generated from those – Bluetrace scheme in Singapore & Australia does this – Allows PHA who to map from beacons to e.g. user so less good for privacy
    - We did an analysis of the open-source version of the Singapore app in April https://www.scss.tcd.ie/Doug.Leith/pubs/opentrace_privacy.pdf
  - Distributed key generation happens on the phone, keys are only uploaded after a positive test, so PHA can't abuse beacon values as in central case
  - This was the main controversy in the Mar/Apr timeframe
- Apps using BLE can exhaust battery quickly so mobile OSes (android & esp. apple) don't allow apps free use of BLE while the app is in the background
  - It becomes impractical to do any of this without co-operation from Google/Apple – UK, France and others tried but it fails as proximity detection fails too often when app in background

Google & Apple Co-operate

- In April Google and Apple announced that they would co-operate on a BLE based decentralised scheme
  - Their scheme is very like other decentralised schemes that were proposed by academics around that time (a good thing)
- Google/Apple Exposure Notification (GAEN) scheme
  - https://www.google.com/covid19/exposurenotifications/
  - https://covid19.apple.com/contacttracing
- Most PHAs (in EU anyway) seem to have adopted the GAEN approach
  - We’ve more or less focused TACT work on that since April
- Ireland, via the Health Services Executive (HSE) and Nearform (a Waterford based mobile app company), were initially developing their own (centralised?) scheme, but adopted the GAEN approach in May 2020

The GAEN System (1)

- GAEN App uses GAEN API implementation
- App handles interaction between handset and Public Health Authority servers
- API implementation handles key storage, BLE beacons and beacon/key (TEK) matching
- Handsets generate a Temporary Exposure Key (TEK) every day
- BLE beacons are sent @~4Hz and contain a Rolling Proximity Identifier (RPI) value that changes whenever BLE MAC address changes (about every 10 minutes)
  - $RPIK_i$:= HKDF(TEK_i,NULL,UTF8("EN-RPIK"),16)
  - $RPI_{i,j}$:= AES128($RPIK_i$,time-in-10-min-chunks)
  - Beacons also include encrypted TxPower value (unauthenticated encryption but not that bad here)
- Handsets listen for beacons for about 4s every 4 mins and record the time, the beacon value (RPI) and Received Signal Strength Indicator (RSSI) for 14 days
- If Alice tests positive, she causes her App to upload the set of TEKs she used for the last (up to) 14 days to her Public Health Authority server
- Bob’s handset downloads TEKs from his Public Health Authority server, perhaps every 2 hours, and checks if any TEK matches any stored RPI

The GAEN System (2)

- Stated goal of these Apps is to detect if two handsets were within 2m for more than 15 minutes, based essentially on matching RPIs to TEKs and the associated RSSI and TxPower values
  - Spoiler: I don’t believe that can be reliably achieved
- “Attenuation” is the measurement used, defined as TxPower-RSSI
  - TxPower might be -27dB, RSSI might be -80dB so attenuation then would be 53dB
  - Measurements also need to be amortised over matching beacons seen and maybe (not specified) with some outliers thrown away
- GAEN API implementation accepts thresholds from App code and returns attenuationDuration values for the “above”, “between” and “below” ranges
  - E.g. App might input “[55,62]” and, if there’s an RPI/TEK match, get back “[10,3,11]” meaning handsets were “closer” than “55dB attenuation” for 10 minutes, in between 55db and 62dB for 3 minutes and “further away than” 62dB for 11 minutes
- App then decides if that output implies a notification is needed
  - If notified, user is usually guided to isolate, go get tested, etc.

The GAEN System (3)

- Governance: Google and Apple are in control
- (IMO) Reasonable justifications for that:
  - There are ~200 countries, don’t want 200 schemes
  - Google/Apple mobile OS duopoly, and their knowledge of handset internals, means you couldn’t credibly tackle this problem without them
  - Don’t want: fake tracing Apps, Apps draining battery messing with BLE or Apps using worse crypto than GAEN spec
- Upshot however is that OS updates can directly affect notifications without any visibility to Public Health Authorities (e.g. calibration adjustments)
  - Spoiler: revisiting governance may be a good idea

The GAEN System (4)

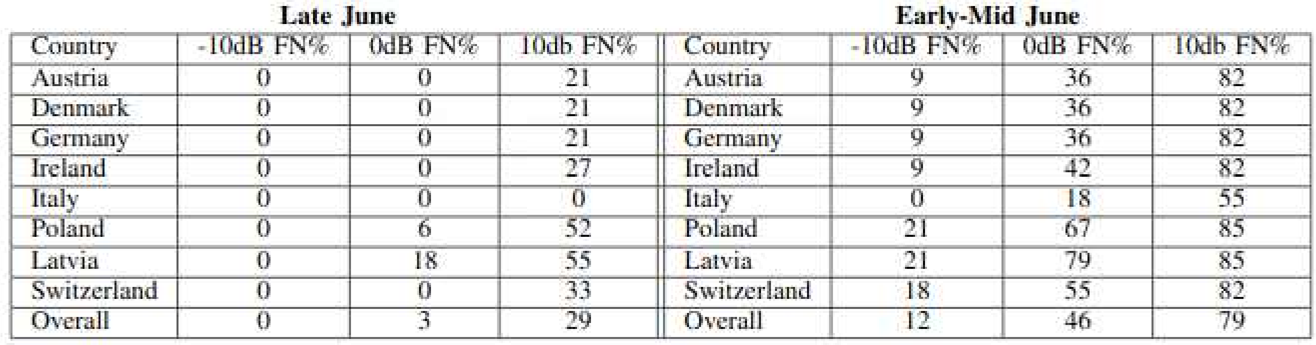
- Some interesting questions for these Apps might be:
  - How many people who would not have been found via manual contract tracing get notified?
  - How many people are notified sooner than would be the case with manual contact tracing?
  - How many notified people turn out to test positive for COVID-19, vs. the averages for testing at that time/in that locale?
  - What are the true/false positive/negative rates for notifications (where “true” == “within 2m for >15 mins” or whatever is the goal)
- So far, we’ve not seen the overall contact tracing systems (manual+App-based, together) being setup so as to be able to answer questions like those above. It’d be good if they were.
- How many downloads or the proportion of the population who’ve installed isn’t really a good metric
  - “You need 60% of the population” is not true – that refers to an unrealistic model in Ferretti et al where the only contact tracing is via such Apps

Replay Attack

- There’s an obvious replay attack: collect beacons (likely from someone who’s positive) and re-transmit/spread those to others elsewhere
  - E.g. use raspberry Pi computers at two collector and spreader locations, connected via the Internet – collector listens for BLE beacons, then sends to spreader which can re-broadcast those values in distant location – other phones near the spreader may be treated as close contacts if someone near the collector later tests positive
  - Locate the collector near a COVID-19 testing station
  - Locate the spreader near potential victims e.g. in hospital Emergency Dept
  - https://down.dsg.cs.tcd.ie/tact/replay.pdf
- We calculate an (under)estimate for the amplification factor for such an attack
  - With conservative early-May Irish figures, “collector” at COVID testing station and “spreader” at hospital emergency department, each true-positive case could produce 4 or more false positive notifications
- Attack is obvious, hasn’t been mitigated and hasn’t (yet) happened (AFAIK)
- A variation of this attack also affects centralised systems e.g. Singapore’s
- As beacons are unauthenticated, attacks like this are very very hard to mitigate

Does BLE Proximity Work?

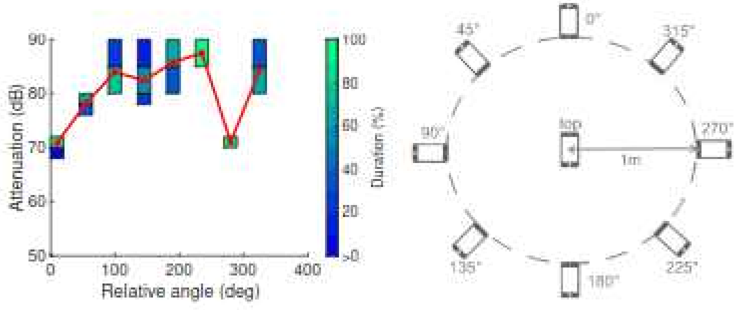
- Pairwise tests with different handset types at 1m for 30+ minutes still produce false negatives if you assume some “noise” due to orientation that affects attenuation
  - https://down.dsg.cs.tcd.ie/tact/posorient.pdf
- On June 13th Google shipped an update that added new calibration adjustments to attenuation calculations. We had tested before that, so we re-did the same tests.
- Percentage false negative seen for various Country configurations:



Late June				Early-Mid June			
Country	-10dB FN%	0dB FN%	10db FN%	Country	-10dB FN%	0dB FN%	10db FN%
Austria	0	0	21	Austria	9	36	82
Denmark	0	0	21	Denmark	9	36	82
Germany	0	0	21	Germany	9	36	82
Ireland	0	0	27	Ireland	9	42	82
Italy	0	0	0	Italy	0	18	55
Poland	0	6	52	Poland	21	67	85
Latvia	0	18	55	Latvia	21	79	85
Switzerland	0	0	33	Switzerland	18	55	82
Overall	0	3	29	Overall	12	46	79

“Noise”

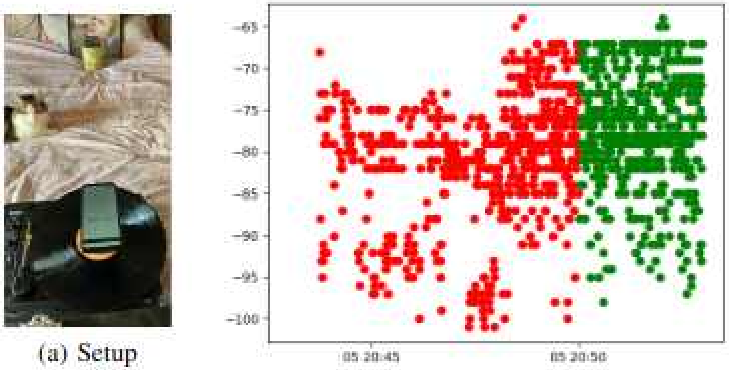
- Handsets package antennae in different ways and so orientation can change attenuation by itself (as can other things)
- We’re not sure how to model this but it seems to be able to affect RSSI (and hence attenuation) by 10-20dB



(a)

(b)

Fig. 4: (a) Measured attenuation duration vs the angle between two handsets placed 1m apart. (b) Schematic showing orientation to which each angle corresponds.



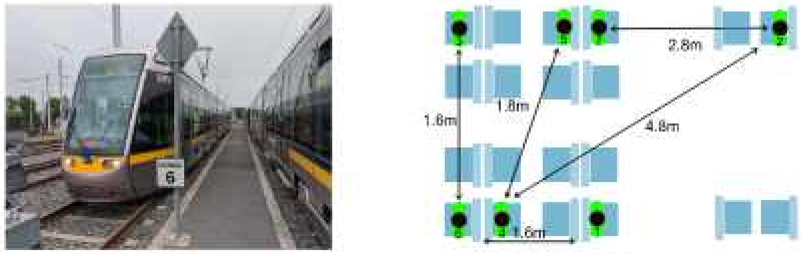
(a) Setup

(b) RSSI

Cat video! https://down.dsg.cs.tcd.ie/tact/changes.mov

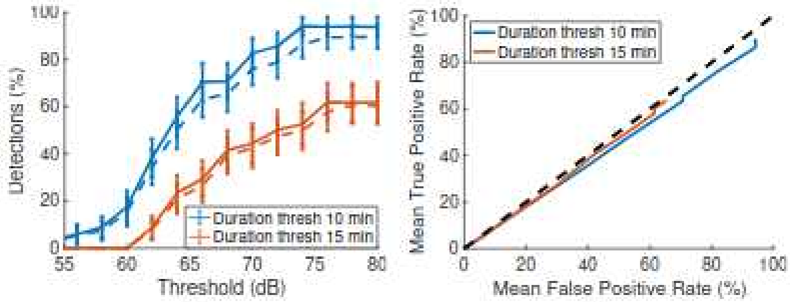
Real-World Scenario Testing

- We did tests of real-world scenarios to see how those affect RSSI, attenuation etc.
  - Walking, cycling, sitting around a table, on a park bench, between cars in parallel
- Two are noteworthy:
  - On a commuter bus https://www.scss.tcd.ie/Doug.Leith/pubs/bus.pdf
  - On a tram https://www.scss.tcd.ie/Doug.Leith/pubs/luas.pdf
- Those seem like scenarios where these Apps, if they work, could help with contacts that would otherwise be missed but the metallic surrounds affected BLE distance estimation badly



(a) (b)

Fig. 1: (a) Tram on which measurements were collected. (b) Relative positions of participants during tests.



(a) (b)

Fig. 8: Exposure notification true and false positive rates when a simple threshold strategy is applied to the GAEN tram dataset. (a) True and false positive rates vs attenuation level and duration thresholds, solid lines indicate true positive rates and dashed lines the corresponding false negative rates. (b) ROC plot corresponding to mean rates in (a), dashed line indicates 45° line.

What traffic is sent? (on Android)

- The Android implementation of the GAEN API is a part of Google Play Services. If you disable Google Play Services these Apps won’t work.
- We mitm’d and unpinned a set of GAEN Apps and captured traffic traces of App traffic and traffic from Google Play Services
  - https://www.scss.tcd.ie/Doug.Leith/pubs/contact_tracing_app_traffic.pdf
- Overall the Apps seem well-behaved but Google Play Services shares long term identifiers with Google and also frequently connects in a way that would allow IP-based location tracking

Example: Irish CovidTracker App

- For all apps, we only tested onboarding and TEK downloads – we didn’t touch anything that the client might e.g. do after a positive test as that could interfere with a running service
- Irish App has some issues:
  - An unnecessary “supercookie” – A JWT authToken used as a bearer token to download TEKs (no other services had such an individual cookie) – the HSE’s Data Protection Impact Assessment (DPIA) states that they don’t log that kind of information but it’d be better to not have it in the protocol
  - The App allows users to opt-in to sending “metrics” that mix devops (whether App opened that day) and medical information (how many times notified) – we recommend separating those into different security contexts
  - There’s some path-not-taken code that makes a call to Google Firebase
- Most apps were open-source, had DPIAs, did certificate pinning and were pretty clean in terms of data-sent, Irish App would be “mid-table” if this were a league

Google Play Services

- We turned off everything we could, including Google Play Services “Usage & Diagnostics” and tried to get to a minimal configuration where a GAEN App can run (reminder: that requires Google Play Services to be running)
- Roughly every 6 hours, Google Play Services connects to a “/checkin” API and sends back information including the handset IMEI, phone number, SIM serial number, WiFi MAC address and the email address associated with the handset
- Roughly every 20 minutes, Google Play Services connects to a “/p/log/batch” API including a cookie that is present in the “/checkin” HTTP request, thereby linking many long-term persistent hard-to-change identifiers with the IP address that can be geo-located
- The above messages contain additional telemetry – how much depends on settings and what other Apps are running (but is opaque); there was some variation in the frequency of connecting to the “/p/log/batch” endpoint, depending on settings.
- There is no published DPIA for the Android GAEN API implementation nor of Google Play Services (that we know about).
- Google Play Services is closed-source.
- That all seems hugely invasive to us and is required if you want to use a GAEN App on Android.

Measuring Deployments

- The set of TEKs that Bob’s phone downloads are public, so we can count those
- We’re currently measuring those for 33 different regions around the world – hourly download of TEKs to our measurement server
  - Started in June, added regions as they deployed or when we got around to them
  - https://down.dsg.cs.tcd.ie/tact/tek-counts/ shows counts, updated a couple of times a day
- Recall that you upload 14 TEKs, and the health authority might publish some or all of those, so if we see 1 new TEK for Ireland today, there’ll (maybe) be 1 new TEK added to the count for each of the last 14 days – we do have hourly snapshots so can estimate how many TEKs are being uploaded each day
- Note: these measurements can’t tell us if these Apps “work” but might tell us they don’t work (in some places)

Measurement Campaign

- For each deployment (of 33), we needed for figure out the TEK download URLs used
  - Doug Leith did all the smart stuff here!
- Sometimes URL is clear from open-source code and documentation (maybe with some guessing)
- Mostly though, we had to install App on rooted handset and run a man-in-the-middle attack to decrypt the HTTPS traffic
- In a bunch of cases we also had to “unpin” the CA certs with which the App was shipped
- There are lots of server-side oddities:
  - Ireland and other Nearform apps require an unnecessary and privacy unfriendly refreshToken to “authorise” the TEK download (it doesn’t need to be authorised as everyone does it!)
  - Canadian server uses a hard-coded secret key to obfuscate URL
  - Many servers include timestamps in URLs
  - Many servers require download of an index or manifest before getting TEKs
- We then script up the hourly downloads, lots of scripts at:
  - https://github.com/sftcd/tek_transparency

Measurement Difficulties

- The GAEN design did not include anything that would allow anyone to measure it’s efficacy, and health authorities don’t publish much detail
  - Oops! Would we do that with a drug? Is a population-scale deployment of an app that different?
- That meant we needed the measurement tooling from the previous slide
- Server implementations have also done other odd things, mainly adding fake TEKs
  - We can often work around those fakes in our estimates but it’s another complexity esp as they turn that kind of thing on and off over time
- In early July Ireland and Northern Ireland started to share TEKs (which makes sense but also makes estimating harder)
  - In late October, additional EU countries started sharing and a common server in the US now supports (I think) 12 states
- Ireland, Germany and Switzerland do now emit some (different) statistics in different ways so we started screen-scaping those

Shortfalls (1)

- If we know:
  - Number of cases per day and population in the region
  - Percent of population running app
  - Number of TEKs uploaded per day
- Then we can calculate whether or not there’s a shortfall in the number of TEK uploads we should expect
- We express that as a percentage, ideally, close to zero
- We did that in October
  - https://down.dsg.cs.tcd.ie/tact/survey10.pdf

$$shortfall = \frac{\frac{cases}{population} - \frac{uploads}{active\_users}}{\frac{cases}{population}}$$

- Result: not great, esp Italy and Poland (both with shortfalls of >90%)

Shortfalls (2)

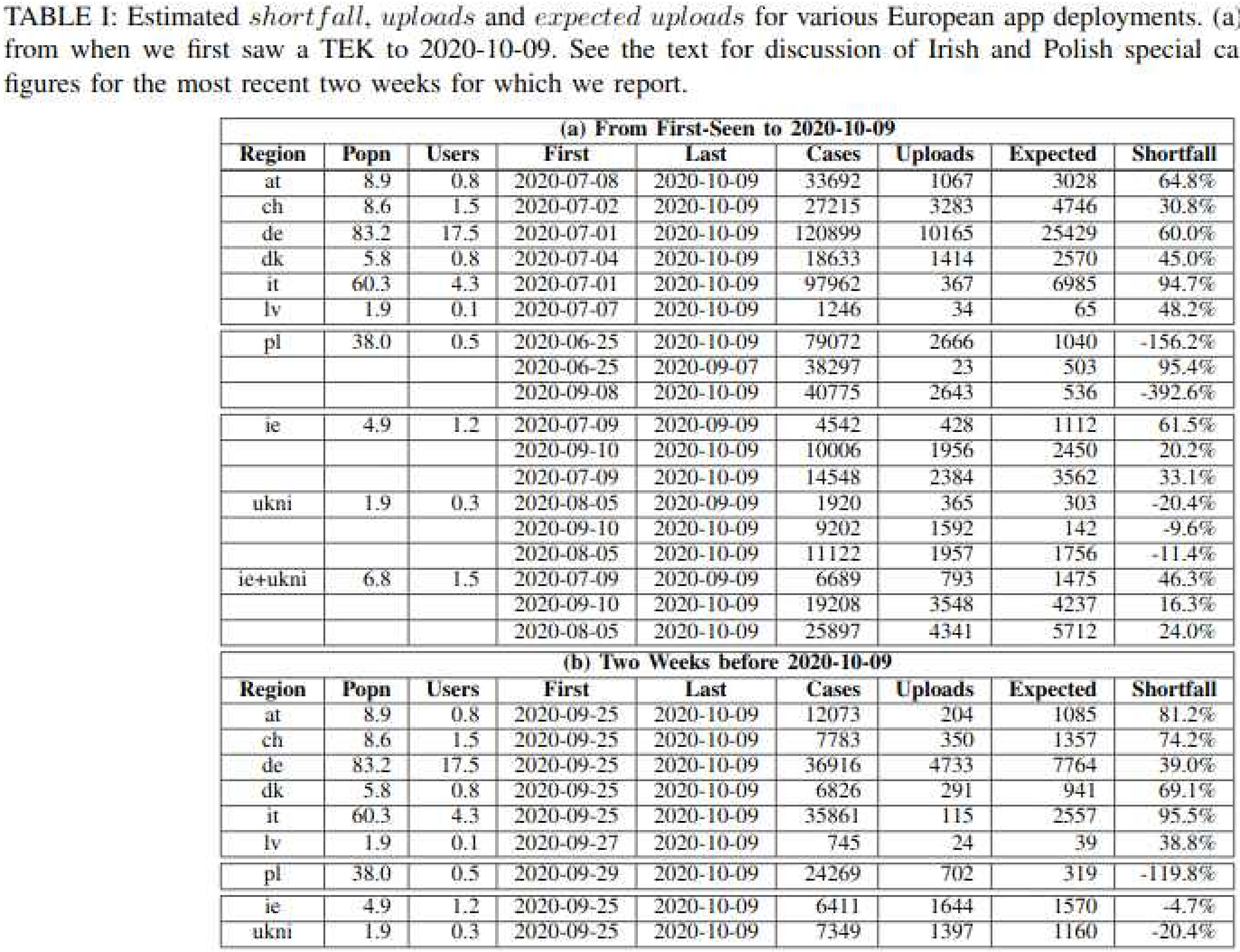


TABLE I: Estimated *shortfall*, *uploads* and *expected uploads* for various European app deployments. (a) shows the figures from when we first saw a TEK to 2020-10-09. See the text for discussion of Irish and Polish special cases. (b) shows the figures for the most recent two weeks for which we report.

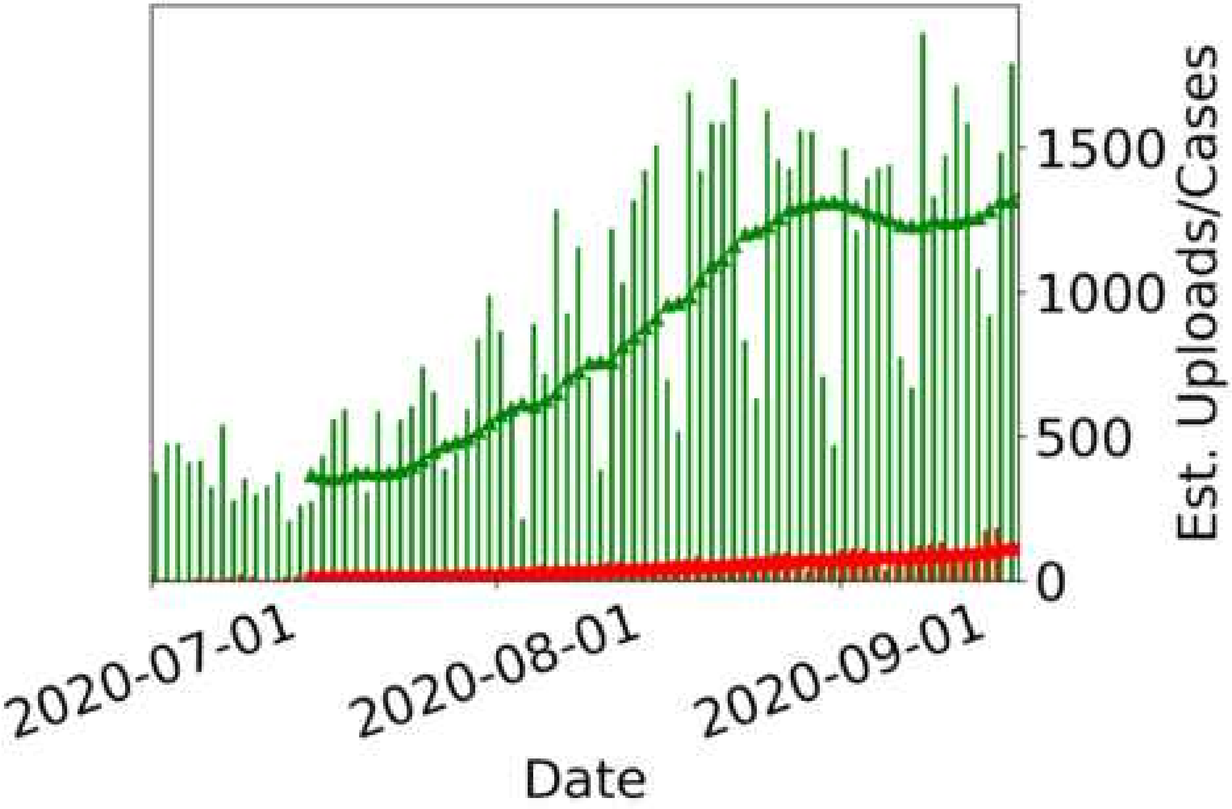
(a) From First-Seen to 2020-10-09

Region	Popn	Users	First	Last	Cases	Uploads	Expected	Shortfall
at	8.9	0.8	2020-07-08	2020-10-09	33692	1067	3028	64.8%
ch	8.6	1.5	2020-07-02	2020-10-09	27215	3283	4746	30.8%
de	83.2	17.5	2020-07-01	2020-10-09	120899	10165	25429	60.0%
dk	5.8	0.8	2020-07-04	2020-10-09	18633	1414	2570	45.0%
it	60.3	4.3	2020-07-01	2020-10-09	97962	367	6985	94.7%
lv	1.9	0.1	2020-07-07	2020-10-09	1246	34	65	48.2%
pl	38.0	0.5	2020-06-25	2020-10-09	79072	2666	1040	-156.2%
			2020-06-25	2020-09-07	38297	23	503	95.4%
			2020-09-08	2020-10-09	40775	2643	536	-392.6%
ie	4.9	1.2	2020-07-09	2020-09-09	4542	428	1112	61.5%
			2020-09-10	2020-10-09	10006	1956	2450	20.2%
			2020-07-09	2020-10-09	14548	2384	3562	33.1%
ukni	1.9	0.3	2020-08-05	2020-09-09	1920	365	303	-20.4%
			2020-09-10	2020-10-09	9202	1592	142	-9.6%
			2020-08-05	2020-10-09	11122	1957	1756	-11.4%
ie+ukni	6.8	1.5	2020-07-09	2020-09-09	6689	793	1475	46.3%
			2020-09-10	2020-10-09	19208	3548	4237	16.3%
			2020-08-05	2020-10-09	25897	4341	5712	24.0%

(b) Two Weeks before 2020-10-09

Region	Popn	Users	First	Last	Cases	Uploads	Expected	Shortfall
at	8.9	0.8	2020-09-25	2020-10-09	12073	204	1085	81.2%
ch	8.6	1.5	2020-09-25	2020-10-09	7783	350	1357	74.2%
de	83.2	17.5	2020-09-25	2020-10-09	36916	4733	7764	39.0%
dk	5.8	0.8	2020-09-25	2020-10-09	6826	291	941	69.1%
it	60.3	4.3	2020-09-25	2020-10-09	35861	115	2557	95.5%
lv	1.9	0.1	2020-09-27	2020-10-09	745	24	39	38.8%
pl	38.0	0.5	2020-09-29	2020-10-09	24269	702	319	-119.8%
ie	4.9	1.2	2020-09-25	2020-10-09	6411	1644	1570	-4.7%
ukni	1.9	0.3	2020-09-25	2020-10-09	7349	1397	1160	-20.4%

Shortfalls (3)



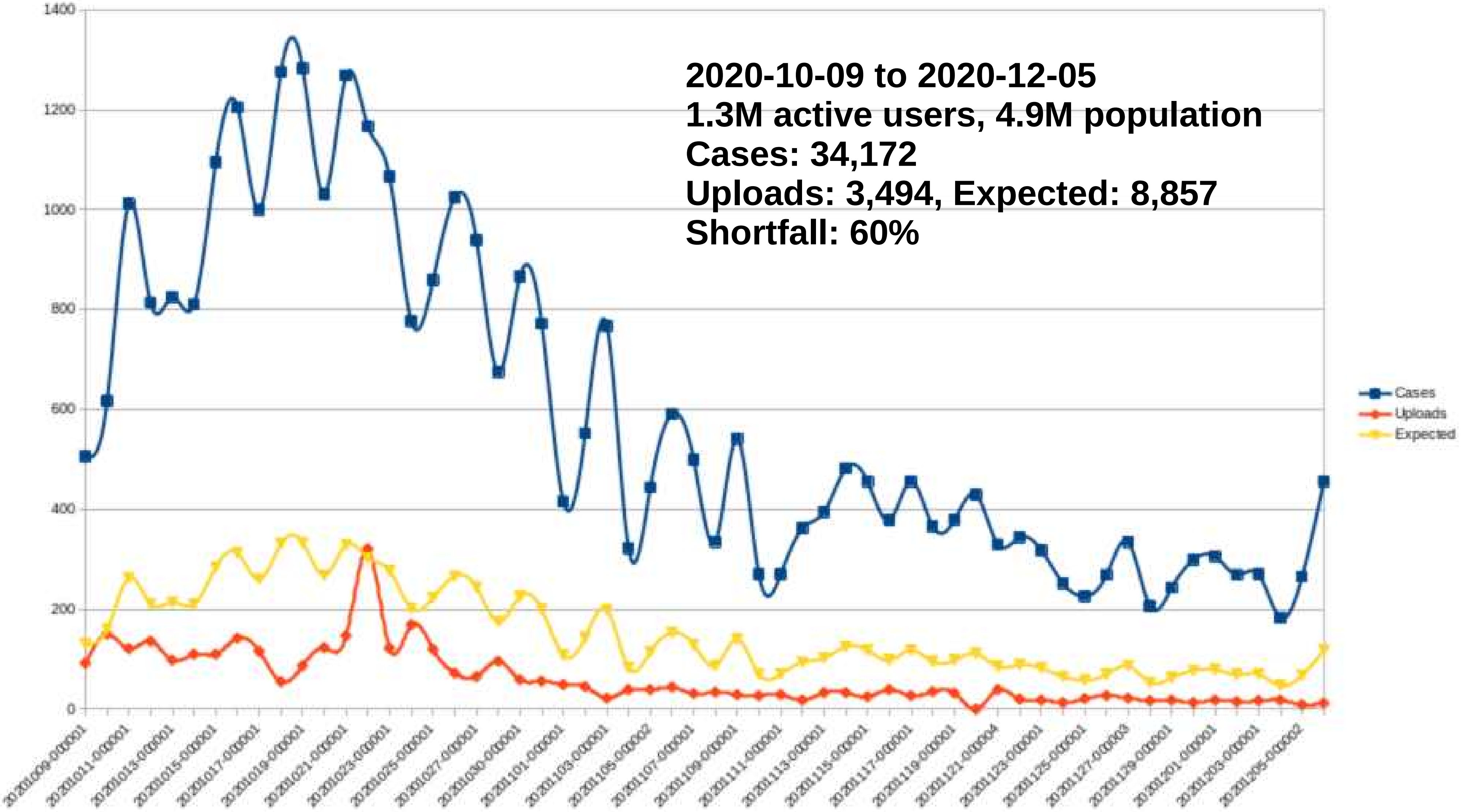
German Cases vs. Uploads with ~21% using their app

Plan is to look again at the data around year-end

Actual Irish Shortfall

- HSE App now includes some in-App statistics
  - Not sure when introduced
- Since early Oct we’ve been downloading those hourly as part of our measurement campaign
- These numbers show a similar shortfall but are no longer estimates – they’re the HSE’s own figures

Actual Irish Shortfall



2020-10-09 to 2020-12-05
1.3M active users, 4.9M population
Cases: 34,172
Uploads: 3,494, Expected: 8,857
Shortfall: 60%

Reasons for shortfalls?

- Really unclear as of now, could be some mixture of:
  - Uploads need anti-spam protection via a “code” that only allows those really testing positive to upload, and maybe that’s too hard for a person to handle when they’ve just been told their test is positive
  - People actually just don’t trust these systems despite the design and all the advertising
  - Demographic factors: maybe active users are more likely young and asymptomatic, or maybe active users are more likely older, more careful and less likely to test positive
  - Geography: maybe some regions have lots of cases but other regions have lots of active users
  - Uncounted uninstalls: maybe people are turning this stuff off for some reason
- In any case with substantial shortfalls such as in Italy or Poland you would really have to wonder if there’s any utility in those apps (so far)
  - Even Ireland and Switzerland seem to have maybe ~33% shortfalls which (IIUC) was not expected back in March

Conclusions

- These Apps are being deployed, governments are encouraging adoption for entire populations
- Replay attacks (and perhaps others) exist and have yet to be mitigated
- BLE-based proximity detection may not work as well as claimed
  - We do expect that to improve over time, but are unsure if it will ever be reliable
  - The “<2m && >15 mins” criterion may be asking the wrong question – we find it hard to see that that can be reliably determined, we don’t know if some other (epidemiologically) useful criterion could be reliably determined
- The Android implementation of the GAEN API comes with serious privacy problems
  - If you don’t care about Google tracking you, then you have no problem
  - If you don’t want to install a GAEN App, then you have no new problem
  - If you care about Google tracking you and want to run a GAEN App, you have a new problem
- We have no information about the internals of the Apple implementation
- Measurements show shortfalls that may indicate that these overall systems don’t seem to work as well as envisaged
- The GAEN system still requires more public documentation, perhaps needs a new “quiet mode” to justify population-wide acceptability and it may be wise to revisit the governance setup for the overall GAEN system
- We also need to start planning to get these systems turned off

Resources

- All details are available at: https://down.dsg.cs.tcd.ie/tact/
- Seven of the 10 documents there are tech-reports, not (yet) peer-reviewed, 3 have been peer-reviewed
- We have published data sets for the bus, tram and pairwise data sets and publish the daily TEK counts
- We have published code for the TEK survey and a modified version of the Google exemplar GAEN App that was used in tests
- We have an (unpublished, it’d help with the replay attack) App that supports other tests by doing the transmitting GAEN functions without using the GAEN API but that interops with the Android and Apple implementations – happy to make available to other researchers

Thanks

Offline questions welcome too
stephen.farrell@cs.tcd.ie

These slides:
https://down.dsg.cs.tcd.ie/tact/