The COVIDSafe App - 4 week update

Jim Mussared
jim.mussared@gmail.com
https://twitter.com/jim_mussared

Eleanor McMurtry
mcmurtrye@unimelb.edu.au
https://twitter.com/noneuclideangrl

with contributions from Vanessa Teague, Richard Nelson, and Geoffrey Huntley.

Written 24/05/2020.
Last updated: 27/05/2020

Status: Public

The COVIDSafe App - 4 week update
- Introduction
- Privacy background
- Summary of outstanding issues
- Summary of fixed issues
- Notes on specific issues
  - CVE-2020-12856
  - The Apple/Google Exposure Notification API
  - iPhone app running in the background
  - Code/issue sharing between OpenTrace variants
  - Using Amazon Web Services (AWS) for the server
  - Community engagement on critical security issues
  - The source code release
  - Privacy Policy and Privacy Impact Assessment
- Timeline
- Author notes

Introduction

The COVIDSafe app was launched for Android and iOS on 26/04/2020, and within hours several serious privacy and functionality issues were discovered by the tech community.

Four weeks later, this app continues to be a privacy risk for anyone who installs it and there is no ETA on when these issues will be resolved.

It is our recommendation that:

- The risks of using the COVIDSafe app should be explained to the public.
- People for whom tracking is a major concern do not install COVIDSafe.
- The move to the Apple/Google Exposure Notification API should be expedited.
- The privacy policy must be updated to more accurately reflect what the app actually does.
- Further investigation should be undertaken to understand how these issues were not detected during testing, why industry best-practices around reporting and managing security issues were not followed, and why the fixes took such a long time to acknowledge and implement.

Note an update to the app source code on GitHub was released on 26/05/2020; this document does not comment on this update as there has not been enough time to carefully analyse the changes.

Privacy background

Any contact tracing app needs to make a trade-off between effectiveness and user privacy. In order to record encounters between people, it must exchange data known as a “TempID” with nearby devices. However, it is very important that these TempIDs (or any other data transmitted) by the app changes on a regular interval as it will otherwise allow re-identification of the device over longer time periods: if you saw a TempID in Richmond an hour ago that you just saw again in Footscray, you know it was sent by the same phone.

Re-identification is a major issue because it allows a malicious actor to track the movements of a device, and therefore of its owner. This can happen in many ways:

- The same person can be detected at different locations and times; for example a person can be identified once (i.e. outside their house or place of work), and then detected at any number of locations subsequently.
- A person’s presence in a single location can be tracked over time (i.e. this person spent 37 minutes at this coffee shop today, and the same person was here yesterday for 24 minutes).
- The number of people in a given building can be detected (and whether it’s the same people as at an earlier time).

Some of these concerns have been dismissed due to some misinformation about what is already possible with Bluetooth and Wi-Fi. However:

- COVIDSafe forces users to enable Bluetooth if they didn’t already have it enabled.
- COVIDSafe enables a range of new ways to track a user that were not previously possible.
- Other apps using Bluetooth Low Energy (BLE) beacon-based tracking can make similar data available only to specific parties (e.g. the app developers and advertising

partners) as per their privacy policies. In contrast, the sort of tracking that COVIDSafe facilitates is available to anybody in Bluetooth range (~20-30 metres).

The tracking issues described in this document have all been relatively easy to exploit, and it only takes one person to package them up into a malicious app for others to use. **Most importantly though, these privacy issues are not inherent to the functionality of the app, and should have been caught during development and review.**

Even if the long-term tracking issues were fixed, it has also been our experience that most people are unaware of the fact that anybody in Bluetooth range is able to detect that the app is running on a given phone. In many cases, especially Android devices, it was not previously possible to detect the presence of a phone at all. This allows, for example, someone outside a building to confirm that somebody with the app is inside the building. While this doesn't allow for long-term tracking, this is not at all clear from the privacy policy or the public messaging around this app.

Summary of outstanding issues

There are seven main issues that have not been resolved:

- Persistent, long-term tracking of devices, even after the app is uninstalled (registered as CVE-2020-12856).
    - This was raised (by Alwen Tiu & Jim Mussared) on 05/05/2020.
    - This issue also allows other denial-of-service and privacy-related attacks (details not yet public).
    - This is a far more serious issue than any of the previous issues. It is not clear how the DTA plans to fix or mitigate it, nor has there been any communication of a planned fix date.
    - See more details below.
- TempID rotation is still broken on iPhone, allowing re-identification of devices and encounters not being recorded.
    - This was first described by Chris Culnane, Eleanor McMurtry, Robert Merkel and Vanessa Teague on 27/04/2020.
    - The root cause was discovered and reported (by Yaakov Smith, Hubert Seiwert, and Jim Mussared) with a suggested fix on 21/05/2020.
    - There are other issues relating to the way TempID expiry works that were raised (by Yaakov Smith) on 17/05/2020.
    - It's very important that expired TempIDs are not used, as this will lead to encounters that should be marked invalid by the server, reducing the effectiveness of this app at contact tracing
- The phone model name (e.g. "Samsung Galaxy G8") and device name (e.g. "Jim's Pixel 2") is available to any device in range, allowing for device re-identification and tracking.
    - This was raised (by Jim Mussared) on 27/05/2020. The fix is to update the privacy policy and to expedite the move to the Apple/Google Exposure Notification API.

- The source code for the server is not available, and none of the cryptography can be verified to be compliant with the privacy policy.
    - The privacy policy is effectively useless without a way to verify how the data is being managed. This is different to a regular Government use of private data where the data is hosted in government data centres. In COVIDSafe, the encrypted tokens are being stored on peoples phones and transmitted over radio.
    - There have been several instances of State Governments using insecure cryptography that were discovered by source code analysis. See e.g. "The New South Wales iVote System: Security Failures and Verification Flaws in a Live Online Election" (J. Halderman & V. Teague, 2015) and "How Not to Prove Your Election Outcome" (T. Haines, S. J. Lewis, O. Pereira & V. Teague, 2020).
    - See also "The missing server code, and why it matters" (Robert Merkel, Eleanor McMurtry, and Vanessa Teague).
- TempID rotation (when working correctly) is set to use a 2-hour expiry time. This is too long, and is far longer than Singapore's TraceTogether app which uses a 15-minute expiry time.
    - See "Tracing the challenges of COVIDSafe" (Chris Culnane, Eleanor McMurtry, Robert Merkel and Vanessa Teague).
- The distance measurement as implemented by COVIDSafe does not work, making the claimed "1.5 metres for 15 minutes" criterion used for contact tracing meaningless.
    - Furthermore, many users have been led to believe that the app only stores encounters that match these criteria. In reality, the app stores all the encounters it sees, and any filtering is done on the server after the app uploads its contacts.
    - See "Coronavirus Contact Tracing: Evaluating The Potential Of Using Bluetooth Received Signal Strength For Proximity Detection" (D. J. Leith, S. Farrell, 2020). More information at The Intercept, and the author's own experiments.
- There have been a number of different reports of this app interacting poorly with other Bluetooth-based apps.
    - Notably, this includes continuous glucose monitoring products, leading to missed alarms; see e.g. https://www.diabetes.co.uk/news/2020/apr/australian-covid-19-tracker-app-could-interfere-with-cgm-devices.html.
    - These reports started from the first day after launch (see Apple App Store reviews and Google App Store reviews) and seem to have gotten more prevalent from iPhone users since the background-mode behavior was fixed.
    - There have been tweets from official accounts claiming that the app attempts to work around these issues but no evidence of this has been found during analysis of the source code, nor is there any evidence of any fixes being made.

Summary of fixed issues

The following issues have been raised by the community and subsequently fixed:

- A long-term tracking issue related to incorrect cache invalidation.
  - This was raised (by Jim Mussared) with a suggested fix on 27/05/2020 and fixed on 14/05/2020.
- A separate long-term tracking issue related to device advertisements.
  - This was raised (by Jim Mussared) with a suggested fix on 27/05/2020 and fixed on 14/05/2020.
- A way for an attacker to crash the iPhone app remotely by sending a malformed advertising payload.
  - This was raised (by Richard Nelson and Jim Mussared) on 6/05/2020 and fixed on 14/05/2020.
- The iPhone app did not work while backgrounded in most situations.
  - A fix was identified by the community (see also here) (mainly Richard Nelson and Jim Mussared) and raised on 30/04/2020, and this fix was implemented on 14/05/2020.
  - This fix has not been publicly acknowledged by the DTA -- there has been no official communication that the iPhone now works in the background.
- A confusing piece of copy text led some users to believe that the app was telling them that they had COVID-19.
  - This was reported on 26/04/2020 (by Geoffrey Huntley) and fixed on 04/05/2020.

Notes on specific issues

CVE-2020-12856

The Common Vulnerabilities and Exposures (CVE) system provides a way of tracking security issues. Because of the severity, and because it affects multiple countries' apps, the persistent tracking issue in COVIDSafe has been assigned an ID of CVE-2020-12856.

The details of this issue are not public, however, a full write-up has been provided to the ASD & DTA, as well as the teams working on other OpenTrace-based apps (Singapore and Alberta). Additionally, it has been shared with Google and teams involved in other contact tracing apps based on a similar design that are also vulnerable to this issue.

In the absence of any engagement from these teams to discuss disclosure, public release, or commit to a fix date, the details are subject to a self-imposed 45 day embargo, starting on from the date it was first reported to the DTA, ending on 19/06/2020. Details will be posted at this GitHub repo.

Unlike the previous tracking issues, this issue allows an attacker to track a target device even after the app is uninstalled. It mostly affects Android, but a lesser (but still serious) variant is still possible on iPhone.

The Apple/Google Exposure Notification API

This initiative from Apple and Google was announced on 11/04/2020 (two weeks before the launch of COVIDSafe). It became available as a system update for iPhone and Android starting on 20/05/2020. COVIDSafe only started becoming functional for contact tracing in the same week, so (with the benefit of hindsight) there would have been very little disadvantage to waiting for this.

It should have been clear from the very first announcement that the Exposure Notification API would be incompatible with the OpenTrace-based approach that COVIDSafe was adopting (i.e. that it would be impossible, and outright disallowed by Apple/Google, to run the two protocols concurrently in the same app, meaning that a future transition would be very difficult). This alone should have been a strong reason to motivate switching (and deferring the launch of COVIDSafe), however the subsequent security and privacy issues have further reinforced that this sort of app should never have been attempted.

The Exposure Notification API's decentralised design means that there are limited touch points for centralised contact tracing, which has been raised by some as a concern. However:

- As a result of choosing a different path for COVIDSafe, there are now serious privacy and reliability issues and an unclear upgrade path.
- Even if the claims are correct that centralised contact tracing is more effective, this doesn't mean that the decentralised approach isn't also effective. Furthermore, Apple and Google may extend their API in the future based on real-word experience.
- There are already many ways for apps based on the Exposure Notification API to allow opt-in engagement with their exposure notification data while still preserving privacy by default.

By embracing the Exposure Notification API from the start, COVIDSafe could have launched on day one of the API's availability with a far more polished app, significantly less development cost and complexity, and been a true success story for the Australian Government.

References:

1. "How Google and Apple outflanked governments in the race to build coronavirus apps" -- Politico, 15/05/2020
2. "Australia's COVIDSafe Experiment, Phase III: Legislation for Trust in Contact Tracing" -- G. Greenleaf, K. Kemp, 15/05/2020
3. "Contact tracing apps are vital tools in the fight against coronavirus. But who decides how they work?" -- S. Lazar, M, Sheel, 12/05/2020
4. "Privacy Preserving Contact Tracing" -- Apple
5. "Exposure Notification APIs Addendum" -- Apple

iPhone app running in the background

At launch, there was extremely confusing messaging about whether the iPhone app worked in the background. We were able to confirm with the Singapore team that in the OpenTrace code that the COVIDSafe app is based on, the app is “not expected to work” in the background on iPhone. Analysis by Richard Nelson showed that the COVIDSafe app behaviour was not substantially different to the OpenTrace code, and later when the source code was released this was confirmed to be true. Richard’s experiments also showed that encounters were not being recorded while the app was in the background, unless another iPhone was nearby with the app in the foreground.

However, the Australian Government claimed that the COVIDSafe app at launch had received significant improvements to this behavior, which was clearly not the case. DTA CEO Randall Brugeaud further added to the confusion at the Select Senate Committee on COVID-19 on May 6th, mostly blaming Apple and older hardware for these issues without any justification or evidence.

The issue here is plain and simple -- the code was never expected to work, it couldn’t possibly have worked, and it was fixed by being completely rewritten in v1.2. In more detail: the initial version of the code scanned for nearby devices by starting a scanner every 185 seconds. After starting the scan, the scan would be stopped 180 second later. The timer used to start the next scan is not able to run when the app is in the background, resulting in the scanning functionality being disabled. The fix is to just not stop the scan. This analysis was provided to the DTA on 30/04/2020, and the v1.2 release that incorporated these suggestions was on 14/05/2020. There has been no public acknowledgement of these fixes.

More information is available from Richard Nelson at “The Unbroken iOS COVIDSafe application” and in the senate submission prepared by covid.watch.

However, issues still persist, notably compatibility with other Bluetooth-based apps. The only clear way forward here is to prioritise the move to the Apple/Google Notification API.

Code/issue sharing between OpenTrace variants

The COVIDSafe app is based on the OpenTrace code developed in Singapore. This code is also used in Alberta, Canada (for the ABTraceTogether app), and in Poland (for the ProteGo app).

Unfortunately, only the Android and iPhone app code was shared, which meant the Australian team had to re-implement the server logic independently. While Singapore released an example implementation of some OpenTrace server functionality, this does not form complete server code. Additionally, the OpenTrace server functionality is based on Google’s Firebase cloud services, whereas the Australian version is based on Amazon Web Services.

The code was shared with limited documentation, very few code comments, and no ability to track changes from upstream. Whatever mechanism the code was shared by, it appears to be equivalent to having just emailed a zip file of a snapshot of the code, and no further communication between the teams was possible. This has meant that any fixes made (either upstream or downstream) have not been able to be shared with the other countries. It has also made it difficult for the community to raise security issues, as each country needs to be notified independently. Additionally, a lack of understanding of how the parts of the code interoperate have led to privacy issues being introduced in the Australian app (e.g. the “device advertisement” tracking issue, and the TempID rotation on iPhone, both discovered by the community).

Using Amazon Web Services (AWS) for the server

This was almost certainly the right decision. A lot of attention has been paid to hypothetical concerns around the CLOUD Act, whereas real, practical concerns about the reliability and security of this data are easily answered by using AWS.

The process of onboarding Government services with AWS is well-understood and has had a lot of thought and attention paid to it over the past couple of years. AWS has received ASD certification via the IRAP.

Whilst it would be fantastic to have a locally owned and operated provider in this space, there is no other certified provider that comes close to the level of service and functionality offered by AWS.

Community engagement on critical security issues

No industry best-practices were implemented at launch, including but not limited to:

- A bug bounty (which would provide a clear path for reporting issues)
- A security contact address, separate from general enquiries (allowing for prioritisation and triaging).
- Source code available at launch (making it easier for analysis)
- Engagement on reported issues to coordinate disclosure.

Minor user interface updates were prioritised over privacy issues, with the justification of “sprint planning”. This is not how “agile software development” is supposed to work.

There has been no public acknowledgement of issues raised, nor has there been any communication around interim mitigations or workarounds.

This has been done better in other countries, e.g. UK’s NHSX recently described their engagement with the community in a blog post. See also the Twitter thread from Vanessa Teague. They also have a HackerOne bug bounty, specifically for their COVID-19 tracker app, as does Singapore.

In the case of the two long-term tracking issues that were fixed, neither recommended fixes were implemented. The recommendation was to remove the unnecessary features altogether to avoid any further risks, and also to add comments to prevent future regressions. Instead, workarounds were added to the existing fragile code.

It is worth noting how much better the response from the DTA was to the iPhone crash, compared to the privacy issues. As is to be expected, a crashing app is far easier for the public to understand than an invisible privacy leak. This difference was not just in terms of time-to-fix, but also in the engagement from the DTA.

Some improvements have been implemented; for example, issues raised via the newly-created support@covidsafe.gov.au address in the past week have at least received quick initial replies. However there continue to be no further engagement on these issues, nor any discussion around disclosure, or commitment to fixes.

The source code release

The source code for both apps was released on 08/05/2020, and has been updated within a couple of days of each subsequent release.

It is published with an unusually restrictive license limiting the rights of its users, there are no tests, the code contains very few comments, and it is impossible for a developer to build and run the application without first building their own test server (with no documentation on this process). At the very least a sample server should have been included.

Additionally, the repositories are read-only, there is no way for the community to provide fixes or improvements.

In addition to this, the Privacy Amendment (Public Health Contact Information) Act contains extremely ambiguous wording around what a researcher may legally do with this application. As a result, the Government has made it extremely unappealing to try and help them.

Privacy Policy and Privacy Impact Assessment

The COVIDSafe privacy policy (as at 24/05/2020) has received no updates since launch.

The authors of the Privacy Impact Assessment were not responsive in dealing with the privacy issues reported to them starting on the day after launch, nor did this PIA highlight any of the Bluetooth-related areas that needed to be investigated in this app.

It is clear that no Bluetooth payloads transmitted by the app were inspected as part of the assessment.

Timeline

This is a partial list of relevant events and actions taken by various parties. Please contact jim.mussared@gmail.com for more information, references, and confirmation before quoting any of these dates.

Day	Date	Notes
0	26/04/2020	COVIDSafe app launched
1	27/04/2020	First long-term tracking issues reported to privacy@health.gov.au, ASD, Maddocks (author of the PIA). First reports of the app interacting poorly with other Bluetooth devices (e.g. Continuous Glucose Monitors).
	28/04/2020	First four issues described in a single document that was distributed widely to the relevant teams (both through official and unofficial channels).
4	30/04/2020	First contact with Singapore OpenTrace team. TempID caching issue fixed same-day. The Singapore team confirms that iPhones in the background are “not expected to work”. ASD confirmed that they will “follow this up”. No further contact. The Cybersecurity CRC confirmed that they have forwarded this doc but are extremely dismissive of the findings. No further contact. Maddocks replied and promised to forward the doc. No further contact.
8	04/05/2020	First contact with DTA. v1.0.15 & v1.0.16 (Android) released containing only updates to graphics and animations and some minor text changes. The only issue fixed is the confusing wording raised by Geoff. risky.biz publishes a high-level summary of the known issues at this stage.
9	05/05/2020	v1.1 (iPhone) released. DTA confirms that they were first aware of the issues on 30/04/2020, but our contact still had not read the document. Full details of CVE-2020-12856 shared with the ASD/ACSC and DTA
10	06/05/2020	DTA CEO questioned by the Select Senate Committee on COVID-19. Topics include the iPhone background behavior and engagement with the tech community. Richard Nelson discovered the remote iPhone crash, reported to DTA.
12	8/05/2020	Source code of v1.0.16 (Android) and v1.1 (iPhone) released, confirming that there are no differences in the Bluetooth implementation to the upstream Singapore codebase.

13	9/05/2020	Same issues discovered in the ABTraceTogether app used by Alberta, Canada. Emailed, and Skype meeting arranged within 24 hours.
17	13/05/2020	DTA confirms that there will be a release tomorrow to fix the iPhone crash but it will fix none of the outstanding privacy issues.
18	14/05/2020	v1.0.17 (Android) and v1.2 (iPhone) released. Contrary to advice from the day before, fixes the first two privacy issues (along with the remote iPhone crash). DTA asked (via SMS to Jim Mussared) for availability to discuss fixes for CVE-2020-12856 in the next couple of days. Jim offered that they can call any time, but then they never followed through on arranging a time. No further contact received from the DTA, all follow-up emails ignored. (Edit: update after this doc was published, see below)
19	15/05/2020	Source code of v1.0.17 (Android) and v1.2 (iPhone) released.
20	16/05/2020	Source code of Alberta, Canada's ABTraceTogether released. None of the issues raised on 09/05/2020 have been fixed.
21	17/05/2020	v1.3 (iPhone) released.
22	18/05/2020	Source code of v1.3 (iPhone) released. iPhone crash fixed in Singapore OpenTrace.
23	19/05/2020	Full details of CVE-2020-12856 shared with the Singapore & Alberta teams (and other affected countries).
26	22/05/2020	iPhone TempID expiry issue raised with DTA (and Singapore & Alberta).
29	25/05/2020	This document was released publicly. **26 minutes later, update from the DTA with a planned release date for "the remaining Bluetooth issues".**

Author notes

Jim: I'm a hybrid hardware and software developer, with current professional experience with open-source development and designing/developing BLE-based products for George Robotics. Formerly worked in programming/electronics education at Grok Learning, and before that at Google Australia as a tech lead in the SRE team as well as some time working with the Android team.

Eleanor: I'm a research student at the University of Melbourne studying security and privacy, currently working on cryptographic voting with Vanessa Teague. I have been a software developer and tertiary educator for several years, and specialise in large-scale and processing-intensive programming. I also work with Blueprint for Free Speech, a not-for-profit organisation working to safeguard privacy and freedom of expression in an online era.