Technische und Rechtliche Analyse der Stopp Corona App des Österreichischen Roten Kreuzes

Ulrich Bayer[3], Andreas Bernauer[3], Marco Blocher[2], Benedikt Gollatz[1], Aljosha Judmayer[3], Michael Koppmann[3], Christian Kudera[3], Thomas Lohninger[1], Georg Merzdovnik[3], Armin Ronacher, Max Schrems[2]

epicenter.works - Plattform Grundrechtspolitik[1]

Widerhofergasse 8/2/4
1090 Wien
Österreich

NOYB – European Center for Digital Rights[2]

Goldschlagstr. 172/4/2
1140 Wien
Österreich

SBA Research gGmbH[3]

Floragasse 7/5. Stock
1040 Wien
Österreich

Inhaltsverzeichnis

1. Einleitung

1.1. Scope und Disclaimer

Mit der *Stopp Corona App*[1] stellt das **Österreichische Rote Kreuz** („ÖRK“)[2] eine Contact-Tracing-App zur Eindämmung von Neuinfektionen mit SARS-CoV-2 („Coronavirus“) in Österreich zur Verfügung. Der zu diesem Zeitpunkt nicht der Allgemeinheit zugänglich gemachte Quellcode der Version 1.1 wurde epicenter.works, noyb, SBA Research und Armin Ronacher (unabhängig) unter Einhaltung einer Verschwiegenheitserklärung (NDA)[3] für eine Analyse zur Verfügung gestellt. Die Konsequenz des NDA ist, dass der vorliegende Bericht 48 Stunden vor Veröffentlichung an Accenture GmbH und das ÖRK geschickt werden musste. Unsere drei Organisationen und Armin Ronacher haben für diese Überprüfung keine finanzielle Vergütung oder sonstigen Mehrwert erhalten. Wir sehen es als unsere Aufgabe, die Bevölkerung möglichst neutral über diese App aufzuklären, wobei wir uns dabei auf die technische und rechtliche Ebene fokussieren, mit Schwerpunkt auf Fragen des Datenschutzes und der IT-Security .

Folgende Dokumente/Dateien lagen den an diesem Projekt beteiligten Organisationen und Personen vor:

- der Quellcode der Android und iOS Apps und der Server-Anwendung vom 07.04.2020[4]
- Datenschutzinformation *Stopp Corona App*[5]
- Bericht zur Datenschutz-Folgenabschätzung
- „allgemeine“ Einwilligungserklärung
- Einwilligungserklärung für den „Symptom-Checker“
- Allgemeine Nutzungsbedingungen, wie im Quellcode hinterlegt
- FAQs[6]

Die Beurteilung der epidemiologischen oder medizinischen Sinnhaftigkeit der *Stopp Corona App* ist nicht Gegenstand dieser Analyse. Ebensowenig wird die Compliance mit Konsumentenschutzrecht, dem E-Commerce-Gesetz oder anderen Rechtsgebieten geprüft, die die Beziehung zwischen dem ÖRK und den User*innen betreffen.

[1] https://participate.roteskreuz.at/stopp-corona/ (abgerufen am 18.04.2020).

[2] Das Österreichische Rote Kreuz ist die nationale Rotkreuz-Gesellschaft (gemäß der Genfer Konventionen) in Österreich und als Verein gemäß Vereinsgesetz 2000 konstituiert (ZVR 432857691).

[3] Volltext des NDA: https://epicenter.works/document/2465

[4] SHA256(StoppCorona1.1-QA_215-Android.zip)=
568f992d856ac3bd1b26ed5f09f8edce1316af5127a82775953afe43a93beb4a
SHA256(StoppCorona1.1-QA_580-iOS.zip)=
938b7833ba7a275277c02f81f412c0a4a9ccd7c9ce51a6fbf8bae4211ba9ce6c
SHA256(2020-04-09-CovidAppSources.zip)=
0e6c763178819b0c62cddc344497209ac5c94d3457c47315f0a889bd0a46c0da

[5] https://www.roteskreuz.at/site/faq-app-stopp-corona/datenschutzinformation-zur-stopp-corona-app/ (abgerufen am 18.04.2020).

[6] https://www.roteskreuz.at/site/faq-app-stopp-corona/ (abgerufen am 18.04.2020).

1.2. Executive Summary

Die Debatte um Contact-Tracing-Apps wird erst seit einigen Wochen geführt und ist daher noch sehr jung. Das Österreichische Rote Kreuz hat vergleichsweise früh mit der Entwicklung einer App begonnen, während viele andere Ländern gerade erst Konzepte diskutieren. Nach Überprüfung des Quellcodes haben wir den Eindruck, dass viele der Anforderungen an die App erst nach Entwicklungsstart hinzu kamen (z.B. automatischer Handshake). Zwar wurde immer ein datenschutzfreundlicher Ansatz verfolgt, jedoch führten die zusätzlichen Anforderungen und technische Beschränkungen auf den Smartphone-Betriebssystemen von Google und Apple zu einer Architektur, die einige Probleme mit sich bringt.

Wir konnten mit unserer Überprüfung des Quellcodes einige ernstzunehmende Datenschutzprobleme identifizieren, die zum Teil bereits mit einem Hotfix repariert wurden. Aus rechtlicher Perspektive äußern wir einige Verbesserungsvorschläge, trotzdem ist das Konzept der App aus unserer Sicht datenschutzkonform. Die technische Sicherheitsüberprüfung hat keine kritischen Sicherheitslücken ergeben, jedoch scheinen einige Verbesserungsvorschläge angebracht.

Einige Datenschutzprobleme der App können in der aktuellen Architektur kaum gelöst werden. In der internationalen Debatte haben sich mittlerweile sehr vielversprechende Ansätze herausgebildet. Datenschutzfreundliche Protokolle wie DP-3T werden von der EU-Kommission[7] und internationalen Wissenschaftler*innen[8] unterstützt. Diese lösen auch das Problem der fehlenden Unterstützung von automatisierten Handshakes auf Apple Smartphone Geräten.

Dieser Bericht verfolgt die zwei Ziele: (1) über die Funktionsweise der App aufzuklären, und (2) konkrete Lösungsvorschläge für deren Verbesserung aufzuzeigen. In diesem Kontext bedanken wir uns beim Roten Kreuz und Accenture für einen professionellen und lösungsorientierten Austausch. Obwohl der Bericht einige kritische Punkte aufgeworfen hat, wurden diese Probleme eingestanden und auf die konkreten schnell Lösungsvorschläge eingegangen.

1.3. Übersicht der Empfehlungen

Von den 26 Empfehlungen dieses Berichts wurden 16 durch einen Hotfix umgesetzt, der bei Veröffentlichung des Berichts bereits zum Download zur Verfügung steht. Bei 3 Empfehlungen wurde eine Nachbesserung im nächsten Release mit Ende der 18. Kalenderwoche angekündigt. Weitere 4 Empfehlungen werden erst mit der Umstellung auf die neue Architektur - vermutlich in 4 Wochen - gelöst werden. Bei den verbleibenden drei juristischen Empfehlungen sieht das ÖRK keinen Handlungsbedarf.

[7] https://ec.europa.eu/info/sites/info/files/5_en_act_part1_v3.pdf (Seite 10, abgerufen 21.04.2020)
[8] Öffentlicher Brief von knapp 300 Kryptograph*innen in Unterstützung eines dezentralen Ansatzes für Contact-Tracing. https://drive.google.com/file/d/1OQg2dxPu-x-RZzETlpV3lFa259Nrpk1J/view

Kapitel	Empfehlung
2.2.2.	Die vom Co-Epi-Projekt vorgestellten Sicherheitseigenschaften für Contact-Tracing-Apps stellen aus der Sicht des Datenschutzes wichtige Empfehlungen dar. Wir empfehlen, dass die gewählte Architektur diese empfohlenen Sicherheitseigenschaften berücksichtigt.
	Rückmeldung ÖRK
	Die erstrebenswerten Sicherheitseigenschaften von Contact-Tracing-Apps sind ein Grundpfeiler in der Gestaltung solcher Apps. Wir finden diese Eigenschaften wichtig und versuchen in der Umsetzung uns bestmöglich daran zu orientieren.
2.2.3.	**Empfehlung**
	Zum Entwicklungsstart der ÖRK-App standen keine architekturellen Ansätze zur Verfügung, wodurch das ÖRK und Accenture gezwungen waren, eine eigene Architektur zu entwerfen und zu implementieren. Mittlerweile werden in der Fachwelt unterschiedliche Ansätze (z.B. DP-3T, Co-Epi/CovidWatch) diskutiert. Wir empfehlen mittel- bis langfristig den Umstieg auf eine dezentrale Architektur, welche von internationalen Experten aus unterschiedlichen wissenschaftlichen Disziplinen empfohlen wird.
	Rückmeldung ÖRK
	Lösung mit Umsetzung von DP-3T Wir bevorzugen weiterhin klar einen dezentralen Architekturansatz. Daher sind im engen Austausch mit DP-3T[1)] und Google&Apple[2)] damit deren Ansätze unsere Anforderungen erfüllen, und haben dazu sehr positive Rückmeldungen erhalten. Sobald der Ansatz eine praxistaugliche Reife und Verfügbarkeit erreicht, werden wir die Architektur auf DP-3T umstellen und auf Interoperabilität mit anderen Ländern achten. (1) DP-3T: regelmäßige Abstimmungsmeetings mit Prof. Capkun (ETH Zürich) sowie Prof. Bugnion (EPFL) um unsere Anforderungen verschiedene Warntypen, manuelle Handshakes, Tokens etc. in die DP-3T Umsetzung aufzunehmen. (2) Google und Apple GF in Österreich, der globalen Partnerschaft des Umsetzungspartners Accenture mit Google&Apple bezüglich early adopter Zugänge der angekündigten Lösung; dem technischen 3rd Level Support von Apple für mobile Apps und Lösung der iOS Background Limitierungen sowie

	Klärung eines teilweisen Einsatzes des Google&Apple Ansatzes für das Tracing von Geräten ohne die Verwendung der Messaging Funktionalität.
2.3.1.	**Empfehlung**
	Die Statistikmeldungen an den Server müssen datenschutzfreundlich umgestaltet oder ganz entfernt werden, ansonsten sind theoretisch auf seiten des Servers vom RK Rückschlüsse auf Kontakte oder Infektionsketten möglich.
	Rückmeldung ÖRK
	Lösung umgesetzt in Release 22.4.2020 Die Empfehlung wurde aufgegriffen und die Statistikmeldung entfernt. Ein erneuter Einbau wird nur mit den im Kapitel 2.3.1. beschriebenen Vorgaben erfolgen.
2.3.2.	**Empfehlung**
	Das Tracken einzelner Smartphones durch dritte, mittels lokalem Aufzeichnen der öffentlichen Schlüssel in Handshakes, sowie die damit verbundene Möglichkeit zum Erstellen von Bewegungsprofilen von User*innen der App muss technisch, beispielsweise durch wechselnde Schlüsselpaare, ausgeschlossen werden.
	Rückmeldung ÖRK
	Lösung geplant für Release 30.4.2020 Die Anregung wurde aufgegriffen und befindet sich in Umsetzung. Hinweis: Der Austausch der public keys wird aktuell durch verschiedene Mechanismen abgesichert und wird nicht direkt z.B. per Bluetooth Broadcast zu Verfügung gestellt.
2.3.3.	**Empfehlung**
	Wir empfehlen, die Nutzung von p2pkit für den automatischen Handshake und die Nutzung von Google-Nearby-Messages für den manuellen Handshake zu überdenken.
	Rückmeldung ÖRK
	Lösung mit Umsetzung von DP-3T Wie bei 2.2.3 beschrieben lösen wir mit der Umsetzung von DP-3T und dem Google&Apple Mechanismus zum direkten Austausch der

	Handshake-Informationen zwischen zwei Geräten die bisher dafür genutzten Mechanismen ab.
2.3.4.	**Empfehlung**
	Wir empfehlen, dass keine Kommunikation mit dem Server des ÖRK oder einem Drittdienst stattfindet, bevor die User*innen der Datenverarbeitung zugestimmt haben.
	Rückmeldung ÖRK
	Lösung umgesetzt in Release 22.4.2020 Die Empfehlung wurde aufgegriffen und im aktuellen Release bereits umgesetzt.
2.3.5.	**Empfehlung**
	Es muss soweit als möglich ausgeschlossen werden, dass Infektionsnachrichten an bekannte öffentliche Schlüssel von Dritten zuordenbar sind.
	Rückmeldung ÖRK
	Lösung geplant für Release 30.4.2020 und DP-3T Umstellung Der Empfehlung wird gefolgt. Es ist krimineller Energie kombiniert mit technischer Expertise erforderlich, trotz bestehender Sicherheitsvorkehrungen die Informationen analysieren zu können. Um den Angriffsvektor für theoretisch bestehende statistische Rückschlüsse weiter zu minimieren, werden bis zum Release am 30.4.2020 rotierende Schlüssel eingesetzt.
2.4.4.1.	**Empfehlung**
	Beim Einsatz von kryptografischen Algorithmen empfehlen wir die Beachtung von Best-Practice-Empfehlungen bezüglich Mindestschlüssellängen und Padding-Verfahren.
	Rückmeldung ÖRK
	Lösung mit Umsetzung von DP-3T Die Empfehlung wird gerne aufgegriffen und eine Umsetzung mit auch für iOS verfügbaren, besser geeigneten Padding Schemes wird gerade analysiert. Mit Umstellung auf DP-3T ist das Thema jedenfalls adressiert.
	Empfehlung

2.4.5.1.	In Bezug auf die Abhörsicherheit ist es eine sinnvolle Maßnahme, für eine weitere Erhöhung der Sicherheit das Vertrauen im Zusammenhang mit der Zertifikatsausstellung auf nur eine CA zu beschränken („Certificate Pinning“). Wir empfehlen diese Maßnahme in die nächste Version einzubauen.
	Rückmeldung ÖRK
	Lösung umgesetzt in Release 22.4.2020 Die Empfehlung wurde aufgegriffen und im aktuellen Release bereits umgesetzt.

Rechtliche Analyse

In einer kurzen juristischen Analyse nach der DSGVO (mit Fokus auf Artikel 5, 6 und 13 DSGVO) sind folgende Probleme und offene Punkte identifiziert worden.

Kapitel	Empfehlung
3.1.2	Eine klare Benennung aller Sub-(Sub-) Auftragsverarbeiter in der Datenschutzinformation ist nachzuholen.
	Rückmeldung ÖRK
	Anmerkung wird aufgenommen.
3.1.2	**Empfehlung**
	Angleichung der Zwecke der Datenverarbeitung zwischen AVV und Datenschutzrichtlinie.
	Rückmeldung ÖRK
	Anmerkung wird aufgenommen.
3.1.2	**Empfehlung**
	Klare Trennung der beiden Google-Dienste (Nearby und Firebase) und der jeweils Verantwortlichen in der Datenschutzinformation ist sicherzustellen.
	Rückmeldung ÖRK
	Anmerkung wird aufgenommen.
3.1.2	**Empfehlung**
	Klarstellung in der Datenschutzinformation zur Verwendung von Apple Push Notification Service ist erforderlich.
	Rückmeldung ÖRK
	Anmerkung wird aufgenommen.
3.1.2	**Empfehlung**
	Die Verwendung alternativer Auftragsverarbeiter, die nicht unter US-Gesetze fallen, wird empfohlen.
	Rückmeldung ÖRK
	Anmerkung wird für Telefonnummern (TAN) in der aktuellen Release umgesetzt. Für pseudonymisierte Daten wird an einer Umsetzung gearbeitet.

3.1.4	**Empfehlung**
	Es wäre wünschenswert, jene Daten, die konkret technisch an Gesundheits- und Bezirksverwaltungsbehörden beauskunftet werden können, sowie die im österreichischen Recht bekannten Fälle der Beauskunftung klar zu benennen.
	Rückmeldung ÖRK
	Bisher gab es keine Beauskunftungsanfrage oder Anstrebungen hierzu. Wurde in der Datenschutzinformation klargestellt.
3.2.1	**Empfehlung**
	Es ist klarzustellen, zu welchem Zweck IP-Adressen verarbeitet werden und auf welche Rechtsgrundlage jene Datenverarbeitungen gestützt werden, die bereits vor Erteilung der Einwilligung erfolgen.
	Rückmeldung ÖRK
	IP-Adressen werden nicht gespeichert. Die Aufrufe vor Einwilligung wurden korrigiert.
3.2.1	**Empfehlung**
	Die Speicherdauer von IP-Adressen ist unklar. Auch ist die Speicherdauer der „digitalen Handshakes" in der Datenschutzinformation auszuweisen.
	Rückmeldung ÖRK
	IP-Adressen werde nicht gespeichert, Information werden ergänzt wenn nicht bereits klargestellt.
3.2.3	**Empfehlung**
	Die Notwendigkeit einer zweiten, gesonderten Einwilligung für den Symptom-Checker ist zu überdenken.
	Rückmeldung ÖRK
	Dies wurde nach sorgfältiger Beratung in Hinblick auf Artikel 7(2) DSGVO für notwendig erachtet.
3.2.3	**Empfehlung**
	Der Benachrichtigungszeitraum (54 Stunden) sollte in allen Dokumenten einheitlich aufscheinen.
	Rückmeldung ÖRK
	Der Benachrichtigungszeitraum ist im Sinne des Containment 2.0 je nach Stand der Wissenschaft konfigurierbar.. Die Datenschutzinformation wurde angepasst.

3.2.4	**Empfehlung**
	Es bestehen massive Bedenken, ob die Statistik-Funktion dem Gebote der Datenminimierung entspricht.
	Rückmeldung ÖRK
	Die Anregung wurde aufgegriffen und im aktuellen Release bereits umgesetzt.
3.4.1	**Empfehlung**
	Modalitäten und Auswirkungen eines Widerrufs der erteilten Einwilligungen sollten verständlich dargestellt werden. Einzelne Handshakes sollen vom Gerät gelöscht werden können.
	Rückmeldung ÖRK
	Anregung wird gerne aufgegriffen und zur Priorisierung in den Backlog möglicher Erweiterungen gestellt.
3.4.2.	**Empfehlung**
	Die App muss die Handshakes umgehend nach der Löschfrist auch tatsächlich löschen.
	Rückmeldung ÖRK
	Die Anregung wurde aufgegriffen und befindet sich in Umsetzung.
3.4.2.	**Empfehlung**
	Die Speicherdauer von IP-Adressen muss angegeben werden.
	Rückmeldung ÖRK
	IP-Adressen werde nicht gespeichert, Information muss ergänzt werden wenn nicht bereits klargestellt.
3.5.5	**Empfehlung**
	Es scheinen weitere „angemessen Maßnahmen“ (im Sinne des Artikel 5(1)(c) DSGVO) zu bestehen, um falsche Informationen soweit wie möglich zu vermeiden.
	Rückmeldung ÖRK
	Aus fachlicher Sicht (siehe epidemologische Erläuterungen) werden grundsätzlich falsch-positive Meldungen akzeptiert oder antizipiert. Die ist ident zur analogen Welt, wo ebenfalls viele falsch-positive Fälle getestet werden um dann ein negatives Testergebnis zu erhalten.

3.5.6	**Empfehlung**
	Es ist zu empfehlen, die Übermittlungszeiten in der Datenschutzrichtlinie oder in den FAQs anzuführen.
	Rückmeldung ÖRK
	Aktuell wird die Meldung mit einem maximalen Verzug von einer Stunde übermittelt. Verbesserungen können gerne aufgenommen werden. Hinweis zur Einordnung: Der aktuell behördliche Informationsfluss „Symptom > Testung > Ergebnis > Ausforschung der Sozialkontakte > Benachrichtigung“ benötigt typischerweise Tage – gegenüber 1 Stunde in der App.
3.6	**Empfehlung**
	Zumindest die eigene UUID ist dem User ersichtlich zu machen oder eine alternative Möglichkeit der eindeutigen Identifizierung zu schaffen, um die Ausübung von Betroffenenrechten zu ermöglichen.
	Rückmeldung ÖRK
	Dieser Verbesserungswunsch kann gerne aufgenommen und gemeinsam priorisiert werden. Faktisch wird die UUID nach dem Ausbau der Statistikmeldung nicht weiter verwendet.

1.4. Allgemeine Funktionsweise

Die *Stopp Corona App* des ÖRK dient dazu, Kontakte zwischen Mobiltelefonen, auf denen die Applikation installiert ist, aufzuzeichnen und im Falle einer später festgestellten Infektion die aufgezeichneten Kontakte retrospektiv zu warnen. Dazu wird einmalig auf jedem Gerät eine zufällige ID sowie ein Schlüsselpaar, bestehend aus einem öffentlichen und einem privaten Schlüssel, erzeugt.

Die *Stopp Corona App* verfolgt dabei ein Modell, bei dem teilnehmende Mobiltelefone Daten miteinander über eine zentrale Infrastruktur austauschen und lokal im Mobiltelefon speichern. Um Kontakte zwischen teilnehmenden Geräten festzustellen, versucht jedes dieser Geräte sogenannte “Handshakes” mit anderen Geräten in seiner Umgebung durchzuführen. Die Anwender*innen können entscheiden, ob die Handshakes automatisiert oder manuell durchgeführt werden sollen. Ziel eines Handshakes ist es, den öffentlichen Schlüssel (Public Key) an das jeweils andere Gerät zu übertragen. Durch Begrenzungen der übertragbaren Datenmenge mittels der eingesetzten Bluetooth-Low-Energy-Technologie (BLE) wird dazu allerdings neben BLE auch ein zentraler Cloud-Dienst herangezogen (p2pkit für automatische Handshakes und Google Nearby für manuelle Handshakes).

Etwas vereinfachend lässt sich die bestehende Architektur in zwei Teile trennen: zum einen die Infrastruktur, die notwendig ist, um im Falle eines Kontakts Handshakes (bzw. öffentliche

Schlüssel) auszutauschen, und zum anderen die Infrastruktur, die notwendig ist, um Infektionsmeldungen an die betroffenen Kontakte zu senden.

Registriert die Applikation nun einen Kontakt, d.h. ein anderes Mobiltelefon mit installierter Applikation in unmittelbarer Nähe, soll mittels des Handshake der jeweilige öffentliche Schlüssel ausgetauscht werden. Um Handshakes durchzuführen, wird auf zwei unterschiedliche Methoden zurückgegriffen. Erstens kann ein manueller Handshake durchgeführt werden, im Rahmen dessen der öffentliche (statische) Schlüssel mittels der Nearby-Messages-API durch die Google-Nearby-Cloud-Infrastruktur ausgetauscht wird. Zu diesem Zweck wird der öffentliche Schlüssel auf Google-Servern hinterlegt, und auf Basis von Ultraschall oder Bluetooth wird ein Token ausgetauscht, mit dem man diesen öffentlichen Schlüssel nachschlagen und herunterladen kann. Zweitens können automatische Handshakes durchgeführt werden. In diesem Fall wird der öffentliche Schlüssel durch die p2pkit-Infrastruktur des Dienstleisters Uepaa ausgetauscht. Funktionell ist dies analog zur Google-Variante allerdings ohne Ultraschall.

Zusätzlich zur Übertragung der öffentlichen Schlüssel wird aktuell sowohl von der iOS- als auch der Android-Applikation jeweils eine Nachricht an den Server des ÖRK geschickt, dass ein solcher Handshake stattgefunden hat. Diese Nachricht beinhaltet die initial erzeugte (statische) ID des jeweiligen Clients sowie einen auf die aktuelle Stunde gerundeten Zeitstempel.

Stellt nun einer der am Handshake Beteiligten fest, infiziert worden zu sein (Red Warning), oder möchte diesbezüglich einen Verdacht äußern (Yellow Warning), dann kann über die Applikation ein TAN (SMS) angefragt werden. Nach Eingabe des TAN kann die Applikation dazu veranlasst werden, an alle Kontakte der letzten Zeit eine verschlüsselte Nachricht zu schicken. Der Verdacht (Yellow Warning) wird über einen Selbsttest eruiert. Wenn der/die User*in angibt, dass er/sie an wiederkehrendem trockenem Husten und/oder einer Körpertemperatur über 38°C leidet, wird er/sie zur Absetzung einer Meldung über den Verdacht einer Erkrankung an COVID-19 aufgefordert. Sollte sich dieser Verdacht bei einer ärztliche Begutachtung als unbegründet herausstellen,kann durch den/die User*in eine Entwarnung an die jeweiligen Kontakte versendet werden.

Um die Infektionsnachricht, Verdachtsnachricht oder Entwarnungsnachricht zuzustellen, wird sie verschlüsselt an den Server des ÖRK geschickt. Dieser kann die Nachricht selbst nicht lesen und kennt auch die involvierten öffentlichen Schlüssel nicht. Der Server fungiert lediglich als Anlaufstelle für alle Applikationen, um dort neue verschlüsselte Nachrichten zu hinterlegen bzw. in periodischen Abständen neue verschlüsselte Nachrichten abzuholen. Um die Anzahl der herunterzuladenden Nachrichten zu reduzieren, wird zusätzlich auf Basis eines Prefix (erstes Byte des SHA256 Hashes des öffentlichen Schlüssels) des eigenen öffentlichen Schlüssels die Gesamtanzahl auf 1/256 aller verschlüsselten Nachrichten reduziert und heruntergeladen. Dazu muss dieser Prefix ebenfalls bei der Benachrichtigung, zusammen mit der verschlüsselten Nachricht, an den Server übertragen werden.

Um nun herauszufinden, ob eine Infektionsmeldung für ein teilnehmendes Gerät existiert, versucht jedes Gerät die heruntergeladenen Daten mit dem eigenen privaten Schlüssel zu

entschlüsseln und ignoriert die Nachrichten, die es nicht entschlüsseln kann, weil sie für ein anderes Gerät verschlüsselt wurden.

Sollte ein Entschlüsselungsversuch erfolgreich sein, benachrichtigt die Applikation den/die User*in, dass er/sie potenziell betroffen ist. In der Nachricht selbst befindet sich nur die Information, dass es sich um einen Verdacht (Yellow Warning) oder eine potenzielle Infektion (Red Warning) handelt, zusammen mit einem auf die Stunde gerundeten Zeitpunkt der Begegnung, die den Anlass für die Benachrichtigung darstellt. Die Applikation schickt eine Benachrichtigung an den Server des ÖRK, dass eine Warnung auf diesem Gerät eingegangen ist. Diese beinhaltet die initial erzeugte (statische) ID des Geräts sowie einen auf die Stunde gerundeten Zeitstempel.

2. Technische Analyse

2.1. Einleitung und methodischer Ansatz

Die Entwicklung sicherer Software ist eine Grundvoraussetzung für eine Applikation, die personenbezogene Daten verarbeitet und böswilligen Angriffen standhalten muss. Aus diesem Grund haben sich eine Vielzahl von Best Practice Empfehlungen und Konzepten etabliert, welche die sichere Entwicklung forcieren. Ein in der Fachliteratur oft herangezogener Standard ist der Security Development Lifecycle (SDL)[9] und die zugehörige Process Guidance[10] (Letztversion 5.2 aus 2012):

- Secure by Design: Stellt sicher, dass bereits in der Planungsphase einer Software auf sicherheitsrelevante Aspekte Rücksicht genommen wird. Hierzu bedarf es einer sicheren Architektur, welche sich aus der Erhebung von Angriffsszenarien und durch eine Bedrohungsanalyse ableitet.
- Secure by Default: Berücksichtigt, dass in einer Software trotz sorgfältiger Planung Schwachstellen enthalten sein können. Daher müssen die Komponenten einer Software immer mit den niedrigst möglichen Berechtigungen betrieben werden. Des Weiteren darf ein Sicherheitskonzept nicht nur auf einer Maßnahme beruhen, sondern muss über vielschichtige und tiefgreifende Maßnahmen verfügen.
- Privacy by Design: Die Architektur und das Design einer Applikation müssen die minimale Verarbeitung von Daten vorsehen und Datenschutzrisiken im Vorhinein durch das Design ausgeschlossen werden. Vor der Verarbeitung muss die Zustimmung des Anwenders eingeholt werden. Die Erhebung selbst muss transparent und für den Anwender klar ersichtlich sein. Die erhobenen Daten müssen sowohl bei der Übertragung als auch bei der Speicherung bestmöglich gegen den Zugriff von Dritten geschützt werden.
- Privacy by Default: Fordert den Datenschutz durch datenschutzfreundliche Voreinstellungen. Das heißt, dass eine Applikation bei der Inbetriebnahme mit den datenschutzfreundlichsten Einstellungen betrieben werden soll. Mithilfe dieses Konzepts sollen vor allem Anwender geschützt werden die weniger technikaffin sind.

Die klassische Methode um die Einhaltung der vorgestellten Konzepte im Anschluss an die Entwicklung sicherzustellen ist ein unabhängiges Software Security Audit. Hierbei wird der Source Code nach Designfehlern, Security Schwachstellen und Missachtungen von Best Practice Empfehlungen durchsucht. Ein Software Security Audit ist ein aufwendiges Verfahren, welches für die von Accenture zur Verfügung gestellten Komponenten (Android Source Code, iOS Source Code, Push Service Backend, SMS Notification Backend und RCA CoronaApp Backend) mindestens 20 Projekttage erfordern würde. Für die durchgeführte Analyse stand nur ein Bruchteil der erforderlichen Zeit zur Verfügung. Weiters wäre eine abgetrennte Testinfrastruktur notwendig, um bösartige Testfälle ohne Beeinträchtigung des Produktionsbetriebs durchzuführen zu können. Aus diesen Gründen handelt es sich bei den nachfolgenden Ergebnissen ausschließlich um eine

[9] https://www.microsoft.com/en-us/securityengineering/sdl.
[10] https://www.microsoft.com/en-us/download/details.aspx?id=29884.

Ersteinschätzung, welche im Rahmen eines Quick-Checks erstellt wurde. Es ist vorstellbar, dass bei einer Offenlegung des Source Codes (Open Source Freigabe) weitere Probleme identifiziert werden, welche in der nachfolgenden Analyse nicht adressiert werden.

2.2. Architektur

Die grundsätzliche Architektur des Systems wird bereits Abschnitt “Allgemeine Funktionsweise” erklärt. Dieser Abschnitt beschäftigt sich mit tiefergehenden Konsequenzen der Architektur.

2.2.1. Handshakes

Geräte tauschen untereinander (mit einer Indirektion durch die p2pkit-Cloud) Public Keys miteinander aus während des automatischen Handshakes. Die Idee bzw Basis der Architektur kann daher durchaus als dezentral gesehen werden wenn dieser Austausch direkt von Gerät zu Gerät passieren könnte.

Es gilt, dass der Public Key sich nicht im Laufe der Zeit ändert und dieser damit als eindeutiges Identifikationsmerkmal gesehen werden kann. Dies ist aus der Sicht des Datenschutzes nicht wünschenswert. Ein passiver Angreifer kann mittels dieses Merkmals Personen eindeutig wiedererkennen, wenn sie sich mehrmals an einem Standort aufhalten. Die Anwendung hebelt damit auch Sicherheitsfeatures des Bluetooth-Stacks in Mobiltelefonen aus: normalerweise rotiert der Bluetooth-Stack automatisch die MAC-Adresse (das Identifikationsmerkmal eines Bluetooth-Transceivers) um eine solche Wiedererkennung zu verhindern.

Dadurch, dass Meldungen auch für einen bestimmten Empfänger verschlüsselt werden entsteht hier auch ein gewisses Skalierungsproblem. Der heruntergeladene Teil der Datenbank stellt etwa 1/256 der gesamten Datenbank dar und bestimmt sich nach dem ersten Byte der SHA256-Summe des eigenen Public Keys, der bei Handshakes ausgetauscht wird. Dies bedeutet in Folge auch, dass, wenn die Anwendung so angepasst werden würde, dass beim Handshake ausgetauschte Public Keys rotiert werden, zunehmend größere Teile der Datenbank heruntergeladen werden müssten. Es besteht damit die Gefahr, dass das System zum Höhepunkt der Pandemie überlastet ist und Skalierbarkeitsprobleme aufweist.

Grundsätzlich ist anzumerken, dass die Systemarchitektur nicht per se unsicher ist. Jedoch ist sie aus Sicht der Skalierbarkeit und der Abhängigkeit von zentralen Servern nicht erstrebenswert. Durch die Limitierungen von BLE ist es fraglich, ob die bestehende Architektur kompatibel umsetzbar ist, ohne auf einen Cloud-Service zurückgreifen zu müssen. Um Handshakes zwischen Geräten auszutauschen, müsste eine Verbindung zwischen den Geräten aufgebaut werden, was unter iOS nicht alleine auf Basis von BLE umsetzbar ist. Das Zurückgreifen auf Cloud-Dienste Dritter für die Durchführung von Handshakes bricht aber mit Privacy-by-Design-Prinzipien (siehe Abschnitt "Privacy").

2.2.2. Sicherheitseigenschaften

Das Co-Epi-Projekt[11] hat eine Liste erstrebenswerter Sicherheitseigenschaften von Contact-Tracing-Apps definiert, welche nachfolgend vorgestellt werden:

- **Server Privacy:** Unter Server Privacy versteht man, dass die zentrale Server-Infrastruktur keine Rückschlüsse auf die User ziehen kann (z.B. dessen Kontakte oder Positionsdaten). Diese Eigenschaft ist wünschenswert, da es die Folgen eines Angriffs auf die Zentrale Infrastruktur reduziert. Je weniger Informationen dem Server bekannt sind, desto weniger kann bei der Kompromittierung durch einen Angreifer entwendet werden. Naturgemäß wird die Server Privacy für User die ihre Infektionen melden herabgesetzt, da hier mehr Informationen bekannt gegeben werden müssen.
- **Source Integrity:** Source Integrity bezeichnet die Unmöglichkeit, Infektionsnachrichten an User zu senden, mit denen der User nicht in Kontakt gestanden ist. Des Weiteren darf keine Möglichkeit bestehen, dass ein User die Identität eines anderen Users vortäuscht.
- **Broadcast Integrity:** Broadcast Integrity bezeichnet die Unmöglichkeit, dass ein User während des Handshakes Informationen über eine andere Identität als sich selbst versendet. Des Weiteren darf ein User während des Handshakes keine falsche Identität annehmen dürfen.
- **No Passive Tracking**: No Passive Tracking bezeichnet den Umstand, dass passives Mithören von Bluetooth-Übertragungen keine Standortinformationen über User offenbart, solange diese keine Infektionsmeldungen erstellt haben.
- **Receiver Privacy:** Unter Receiver Privacy wird verstanden, dass der Empfänger einer Infektionsmeldung keine Informationen über sich selbst an andere preisgibt.
- **Reporter Privacy:** Reporter Privacy bezeichnet den Umstand, dass nur das Minimum an Information von infizierten Personen geteilt wird. Konkret sollen keine Daten an Personen weitergegeben werden, mit denen man nicht in Kontakt gestanden ist. Personen, mit denen man Kontakt hatte, soll maximal der ungefähre Zeitpunkt der Begegnung offenbart werden.

Im Folgenden betrachten wir die vorgestellten Sicherheitseigenschaften für die Architektur der *Stopp Corona App*.

2.2.2.1. Server Privacy

Im Falle der ÖRK App ist Server Privacy für den Austausch der Infektionsnachrichten *aus der Sicht des Protokolls* gewahrt. Aufgrund der Nutzung der Infrastruktur von p2pkit und Google Nearby, sowie der implementierten Statistikfunktionen in der App, wird die Server Privacy *insgesamt aber verletzt*. Ersteres ist problematisch, da bei jedem Handshake auf Grund von den technischen Limitierungen von BLE der Public Key entweder zur Infrastruktur von p2pkit oder Google Nearby übertragen werden muss. Da beide Dienste über keine End-to-End Encryption (E2EE) verfügen, sind die Public Keys auf den Servern im Klartext abgelegt. Dies macht p2pkit und Google Nearby Cloud zu einem interessantes Ziel für

[11] https://github.com/TCNCoalition/TCN (abgerufen am 19.04.2020).

Angreifer. Mit der Kombination von Zugriffslogs und den Public Keys lässt sich der komplette Social Graph (“Wer hatte wann mit wem Kontakt”) rekonstruieren. Näheres zur datenschutzrelevanten Verwendung von diesen Datendiensten findet sich auch im Abschnitt “2.3.3. Nutzung von Drittdiensten”.

Zusätzlich dazu besitzt das System momentan Statistik-Meldungen die in der aktuellen Implementierung ebenfalls Server Privacy verletzen (siehe Abschnitt “2.3.1. Statistikmeldungen”)

2.2.2.2. Source Integrity

Da ein Public Key, der den Empfänger einer Nachricht bestimmt, auch von Dritten übermittelt worden sein kann (siehe Broadcast Integrity) und Infektionsnachrichten nicht signiert sind, kann es auch zur Übermittlung von Infektionsnachrichten an Personen kommen, mit denen ein Infizierter nicht im Kontakt gestanden ist. Source Integrity ist damit nicht gegeben.

2.2.2.3. Broadcast Integrity

Da bei Handshakes nur der Public Key einer Person ausgetauscht wird, kann ein bösartiger User seinen Schlüssel gegen den Schlüssel eines anderen Users austauschen und somit eine falsche Identität vortäuschen. Broadcast Integrity ist damit nicht gegeben. Konkret bedeutet dies, dass ein User Public Keys von Dritten sammeln und damit falsche Kontakte vortäuschen kann in dem er die Public Keys von diesen Dritten anstatt des eigenen verteilt.

2.2.2.4. No Passive Tracking

Zwar werden auf dem BLE-Layer gemäß der BLE-Spezifikation selbst rotierende Identifier ausgetauscht, allerdings wird beim automatischen Handshake der gegenwärtig nicht rotierende Public Key des Users in der p2pkit-Cloud abgelegt. diese lassen sich aber in der p2pkit-Cloud gegen statische Keys "eintauschen". Theoretisch ließe sich dieses Verhalten in der p2pkit-Cloud erkennen und durch Rate Limiting begrenzen, allerdings ist grundsätzlich davon ausgehen, dass sich das System für passives Tracking ausnutzen lässt.

Weiteres zu den Problemen die sich durch Passive Tracking ergeben siehe Abschnitt “2.3.2. Bewegungsprofile durch Tracking im physischen Raum”.

2.2.2.5. Receiver Privacy

Dies wird in der Architektur bei hinreichend großer Userzahl gewahrt, in der konkreten Implementierung wird damit aber gebrochen (siehe Abschnitt "2.3.5. Verfolgbarkeit von Benachrichtigungen innerhalb eines kurzen Zeitraums an bekannte öffentliche Schlüssel"). Bei Datenbankzugriffen durch die User gibt dieser das erste Byte der SHA256-Summe seines Public Keys bekannt. Der User lässt sich so in eine von 256 Klassen von Usern einordnen. Solange die Anzahl der User im System genügend groß ist, sollten sich User so nicht identifizieren lassen. Bei einer kleinen Anzahl von Usern wäre diese Informationspreisgabe jedoch problematisch.

2.2.2.6. Reporter Privacy

Architekturell gelten hier dieselben Erwägungen wie zu Receiver Privacy. In der konkreten Implementierung wird mit der Reporter Privacy jedoch gebrochen (siehe Abschnitt "2.3.5. Verfolgbarkeit von Benachrichtigungen innerhalb eines kurzen Zeitraums an bekannte öffentliche Schlüssel"). Im Übrigen muss sich ein infizierter User grundsätzlich auch mit einer TAN authentifizieren und TAN und Telefonnummer werden bei der Infektionsmeldung an das ÖRK übertragen.

➔ **Die vom Co-Epi-Projekt vorgestellten Sicherheitseigenschaften für Contact-Tracing-Apps stellen aus der Sicht des Datenschutzes wichtige Empfehlungen dar. Wir empfehlen, dass die gewählte Architektur diese empfohlenen Sicherheitseigenschaften berücksichtigt.**

2.2.3. (Distributed) Contact Tracing-Architekturen

In den letzten Wochen wurden eine Vielzahl von zentralisierten und dezentralisierten Architekturen zum Contact Tracing diskutiert, welche durch unterschiedlichste konzeptionelle und technische Eigenschaften charakterisiert sind. Im akademischen Umfeld findet momentan ein rege Debatte statt, welche Architekturen und konzeptionelle Ansätze sich am besten für Contact Tracing-Applikationen eignen. Generell ist das Thema eine sehr junge Wissenschaft, wo es noch an Erfahrungswerten und Evaluierungen mangelt. Aus diesem Grund gibt es momentan auch keine Best-Practice-Empfehlungen für die Entwicklung von Contact Tracing-Architekturen.

Jedoch kristallisiert sich langsam heraus, dass die Kompatibilität mit dem BLE-Stack eine wichtige Anforderung ist. Zur Erfüllung der Kompatibilität darf die Nachricht nicht länger als 20 Byte sein, da dies im BLE Stack nicht vorgesehen ist. Sowohl das Protokoll DP-3T[12] sowie das Google/Apple Privacy Preserving Contact Tracing-Protokoll[13] unterstützen diese Anforderung und tauschen grundsätzlich nur zufällige Identifier aus, die zusammen mit der MAC-Adresse des Bluetooth Stacks rotieren. Damit ergeben sich gegenüber dem, was ein Mobiltelefon schon von sich aus übermittelt, keine weiteren Angriffsvektoren. Erst zum Zeitpunkt einer Infektionsmeldung werden Secret Keys über eine zentrale Infrastruktur preisgegeben, aus denen sich die Identifier rückrechnen lassen, welche von anderen Geräten heruntergeladen werden können.

Dieses System unterscheidet sich von dem der *Stopp Corona App*. Auch die *Stopp Corona App* lädt Infektionsmeldungen von einer zentralen Infrastruktur herunter; diese sind allerdings speziell für den Empfänger verschlüsselt. Bei DP-3T oder dem Apple/Google-System würde das Gerät stattdessen Daten herunterladen, mit denen alle — pseudonymisierten — möglichen Infektionen berechenbar sind. Diese Ansätze standen zum Entwicklungsstart der ÖRK App noch nicht zur Verfügung, stellen heute aber eine praxistaugliche Alternative dar.

[12] https://github.com/DP-3T/documents (abgerufen am 19.04.2020).
[13] https://www.apple.com/covid19/contacttracing/ (abgerufen am 19.04.2020).

Nachfolgend wird die *Stopp Corona App*-Architektur im Detail mit anderen Architekturen verglichen.

2.2.3.1. BLE Layer Data Exchange

Auf der Bluetooth-Ebene tauscht die *Stopp Corona App* einen Identifier aus, der genutzt wird, um Daten durch Google Nearby oder p2pkit über einen Cloud-Service auszutauschen. Diese Datenübertragung besteht aktuell aus dem Austausch eines Public Keys sowie, im Falle des manuellen Handshakes einer vierstelligen Zufallszahl, die dem User angezeigt wird. Im Vergleich dazu tauschen dezentrale Systeme wie DP-3T, Co-Epi/CovidWatch oder Apple/Google ein zufälliges Token ohne Verwendung eines Cloud-Services aus.

2.2.3.2. Infection Exchange

Infektionsnachrichten werden sowohl bei der *Stopp Corona App* als auch auch bei dezentralen Systemen durch ein zentrales Backend ausgetauscht. Insofern ist der Terminus “dezentrales System” nur begrenzt zutreffend. Mit der *Stopp Corona App* hinterlegen Geräte Infektionsnachrichten, die für ein bestimmtes anderes Endgerät verschlüsselt sind. Bei DP-3T und anderen werden in der Regel Secrets oder Seeds von Infizierten ausgetauscht, aus denen die zufälligen Tokens, die im Kontaktfall möglicherweise ausgetauscht wurden, berechenbar sind. In beiden Fällen werden Kontakte erst am Endgerät des Users erkannt und können vom zentralen System nicht erkannt werden.

2.2.3.3. Source Integrity

Der Ansatz der *Stopp Corona App* würde es theoretisch erlauben, dass Infektionsnachrichten nur von Personen zu Personen erfolgen, die miteinander im Kontakt gestanden sind. Sofern im Kontaktfall sichergestellt wird, dass die Public Keys jeweils beider Geräte ausgetauscht werden, könnten Infektionsnachrichten signiert und diese Signatur von den Empfängern geprüft werden. Das Signieren von Infektionsnachrichten erfolgt in der vorliegenden Version der App nicht. Ein Problem, welches in der gegenwärtigen Architektur durch den Einsatz von Signaturen entstehen würde, ist, dass Anwender beim Empfang einer Infektionsnachricht die Identität des Absenders feststellen könnten was dem Ansatz der Reporter Privacy widerspricht.

In einer dezentralen Architektur werden grundsätzlich nur temporäre Tokens ausgetauscht. Einem User ist es dadurch in einem solchen System möglich, über einen gewissen Zeitraum die Tokens anderer Usern weiterzuverbreiten. Das Apple/Google-Protokoll schränkt dies ein, indem Tokens nur für etwa 30 Minuten gültig sind. Ist aus den Infektionsmeldungen ableitbar, für welchen Zeitraum getauschte Tokens jeweils gültig waren, können Empfänger Tokens, die nicht in dem passenden 30-Minuten-Fenster empfangen wurden, ignorieren.

2.2.3.4. Infektionsmeldungen

Die *Stopp Corona App* erfordert momentan, dass bei einer Infektionsmeldung die infizierte Person dem ÖRK ihre Telefonnummer preisgeben muss. Dies ist auch dem Umstand geschuldet, dass man das missbräuchliche Absetzen einer Infektionsmeldung verhindern will

und ggf. mit der Person Kontakt aufnehmen können will. Je nachdem, welche Interaktionen mit der Person notwendig sind, lässt sich dies jedoch umgehen. Es wäre möglich, die Infektionsmeldung selbst von der Authentifizierung für eine solche zu trennen und so einen Rückschluss von Telefonnummer zu konkreten Infektionsmeldungen am Backend nicht zu ermöglichen.

DP-3T und andere Protokolle gehen davon aus, dass eine Infektionsmeldung nur durch einen positiven Covid-19-Test erfolgen kann. Als vorgeschlagenes Protokoll werden hier inaktive Authentifizierungscodes übertragen, die erst mit einem positiven Testresultat aktiviert werden. Da die Authentifizierungscodes z.B. von Ärzten ins System eingetragen werden, kann so eine Unschärfe erzeugt werden, die keinen Rückschluss auf die konkrete Person erlauben. Die derzeitige Struktur von Meldungen auf Basis von Symptomen (yellow) und Testresultaten (red/green) bietet einen Mehrwert und sollte nach Möglichkeit beibehalten werden.

➔ **Zum Entwicklungsstart der ÖRK App standen keine architekturellen Ansätze zur Verfügung, wodurch das ÖRK und Accenture gezwungen waren eine eigene Architektur zu implementieren. Mittlerweile werden in der Fachwelt unterschiedliche Ansätze (z.B. Co-Epi/CovidWatch, DP-3T, NOVID20, Pepp-Pt) diskutiert. Wir empfehlen den Umstieg auf eine Architektur, welche von internationalen Experten aus unterschiedlichen wissenschaftlichen Disziplinen empfohlen wird.**

2.3. Datenschutz

Die Entwicklung einer App, die sensible Daten wie Kontakte und den Gesundheitszustand der User verarbeitet, ist daran zu messen, ob dabei Privacy-by-Design-Prinzipien beachtet wurden, die die Beachtung von Datenschutzaspekten während der Entwicklung sicherstellen sollen. Zentral ist das Prinzip "preventive not remedial", dass also Datenschutzrisiken im Vorhinein ausgeschlossen werden und nicht nachträglich durch weitere Maßnahmen gelindert werden sollen. Im vorliegenden Kontext ist insbesondere das Risiko auszuschließen, dass erfolgte Kontakte und mögliche Infektionsketten durch das ÖRK oder Dritte rekonstruiert werden können oder mit gegenüber vorher bestehendem Wissen signifikante Änderungen in der Bewertung der Wahrscheinlichkeiten solcher Vorkommnisse vorgenommen werden können.

Grundsätzlich ist festzustellen, dass bei der Entwicklung der *Stopp Corona App* dieser Ansatz in gewisser Weise verfolgt worden ist. Das Design ist zunächst so gewählt, dass es möglich ist, dass das Backend des ÖRK nicht in Erfahrung bringt, wer mit wem in Kontakt gestanden hat und welche Infektionsketten möglicherweise entstanden sein könnten, während dennoch im Infektionsfall entsprechende Infektionsnachrichten weitergeleitet werden können. Leider wird diese Garantie in der vorliegenden Version der App verletzt, da insbesondere im Betrieb der App anfallende Kommunikationsmetadaten wie IP-Adressen und Zeitpunkte der Übermittlung nicht beachtet wurden, die Rückschlüsse auf Personen, Art und Inhalt der Kommunikation erlauben. Damit wird mit Privacy-by-Design-Prinzipien gebrochen.

2.3.1. Statistikmeldungen

Das Backend bietet einen API-Endpoint */Rest/v3/track-events*, über den Installationen der App zeitnah einen erfolgten Handshake oder den Erhalt einer erfolgreich entschlüsselten Infektions- oder Entwarnungsmeldung einmelden. Dabei wird die eindeutige Gerätenummer und der auf eine Stunde genaue Zeitpunkte erfolgter Handshakes, sowie empfangener und erfolgreich entschlüsselter Infektions- sowie Entwarnungsmeldungen an den Server übertragen. In Kombination mit den entstehenden Metadaten der Kommunikation (IP-Adressen der meldenden Geräte, Zeitpunkte) können damit Rückschlüsse gezogen werden, zwischen welchen Personen Handshakes stattgefunden haben und welche Infektionsketten möglich sind.

Wir raten dringend, soweit die Nutzung derartiger Statistikmeldungen überhaupt notwendig ist, diese Meldungen

- nicht zeitnah zum erfolgten Event, sondern von jedem Gerät etwa täglich zu einem bestimmten Zeitpunkt,
- ohne die Übermittlung eindeutiger Kennungen,
- nicht in Form von Timestamps, sondern in Form von summary statistics (also Anzahl der erfolgten Meldungen), und
- nicht in Form echter Werte, sondern mit einem Local-Differential-Privacy-Mechanismus verrauschte Daten

zu übertragen.

➔ **Die Statistikmeldungen an den Server müssen datenschutzfreundlich umgestaltet oder entfernt werden, ansonsten sind Rückschlüsse über Kontakte oder Infektionsketten möglich.**

2.3.2. Bewegungsprofile durch Tracking im physischen Raum

Wie im Kapitel zu “No Passive Tracking” ausgeführt, ist es möglich eine Installation der App im physischen Raum zu verfolgen und damit Bewegungsprofile zu erstellen. Dazu muss die App im Modus für automatische Handshakes sein, was auf Android-Geräte die Standardeinstellung ist. Auf iOS-Geräten funktioniert dieses Tracking aufgrund von Limitierungen des Betriebssystems derzeit noch nicht. Aufgrund des automatischen Austausch des public keys durch die App und der der statischen Natur des public keys kann ein einzelnes Gerät über einen beliebig langen Zeitraum wiedererkannt werden. Über Sensoren auf öffentlichen Plätzen oder in Verkehrsmitteln wäre damit ein Tracking und das Erstellen von Bewegungsprofilen einzelner Personen möglich.

Der Bluetooth Stack verhindert dies von Haus aus seit etwa 2014 in Smartphones, indem es die MAC-Adresse randomisiert; das aktuelle Protokoll hebelt allerdings dieses Sicherheitsfeature aus, indem immer derselbe Schlüssel geteilt wird. Zudem liegen damit, wie bereits bei “Server Privacy” angemerkt, alle Public Keys auf der p2pkit Cloud.

Um das Problem hier etwas besser zu visualisieren, kann man sich den Fall vorstellen, in dem ein Supermarkt einen passiven Bluetooth-Sniffer einsetzt, um Personen wiederzuerkennen, die häufig einkaufen.

➔ **Das Tracken einzelner Geräte im physischen Raum und dadurch das Erstellen von Bewegungsprofilen von Usern der App muss technisch ausgeschlossen werden.**

2.3.3. Nutzung von Drittdiensten

2.3.3.1. p2pkit

Automatisierte Handshakes werden mittels des Drittdienstes p2pkit über das Internet abgewickelt. Dabei entstehen sowohl Metadaten der Kommunikation zwischen den Geräten und Statistikmeldungen der beteiligten Geräte gegenüber p2pkit. Es werden p2pkit dabei Informationen zum Zeitpunkt des Handshakes, die Tatsache, dass es sich um eine Nutzung der *Stopp Corona App* handelt, eine p2pkit-eigene pseudonyme Userkennung, Betriebssystem, Betriebssystemversion und Modell des einmeldenden Geräts übertragen. Durch diese Daten, in Kombination mit weiteren anfallenden Metadaten der Kommunikation können Rückschlüsse auf erfolgte Kontakte gezogen werden.

2.3.3.2. Google Nearby Messages

Zum Zwecke des manuellen Handshakes wird Google Nearby Messages verwendet. Dieser Dienst wird über die Google-Infrastruktur vermittelt. Anfallende Metadaten wie IP-Adressen erlauben wieder Rückschlüsse auf erfolgte manuell registrierte Kontakte.

➔ **Wir empfehlen die Nutzung von p2pkit für den automatischen Handshake und die Nutzung von Google Nearby Messages für den manuellen Handshake zu überdenken.**

2.3.4. Datenverarbeitung vor der Einwilligung

Beim ersten Start der Applikation wird dem User eine kurze Erklärung der Funktionsweise der App angezeigt. Anschließend wird er aufgefordert der Datenverarbeitung zuzustimmen um die App nutzen zu können. Im Hintergrund werden allerdings bereits beim Initialisieren der App mehrere Requests abgesetzt:

- Download eines JSON-Objekts von Microsoft Azure, welches die Konfiguration für die Applikation beinhaltet
- Download der Infektionsnachrichten von Microsoft Azure
- Subscription bei einem Topic von Google Firebase, welches als Push Notification genutzt wird

Zumindest der Download der Infektionsnachrichten und die Subscription bei Google Firebase sind aus technischer Sicht nicht vor der Zustimmung der Datenverarbeitung notwendig. Aber auch der Download der Konfiguration kann technisch so gelöst werden, dass der User erst der Datenverarbeitung zustimmen muss.

➔ **Wir empfehlen, dass keine Kommunikation mit dem Server des ÖRK oder einem Drittdienst stattfindet, bevor der Datenverarbeitung durch den User zugestimmt wurde.**

2.3.5. Verfolgbarkeit von Benachrichtigungen innerhalb eines kurzen Zeitraums an bekannte öffentliche Schlüssel

Jeder verschlüsselten Infektionsnachricht, die auf den Servern des ÖRK zum Download veröffentlicht wird, ist auch der Präfix (erstes Byte des SHA256-Hashes des Fingerprints eines öffentlichen Schlüssels) beigefügt. Betrachtet man nun ein kleines Zeitintervall, dann ist es möglich, durch Herunterladen aller in diesem Intervall neu hinzugekommenen Nachrichten vom Server einzelne bekannte öffentliche Schlüssel als potenzielle Empfänger auszuschließen, oder unter bestimmten Umständen Rückschlüsse auf mögliche Sender und Empfänger von Infektionsnachrichtenziehen (wenn die öffentlichen Schlüssel aller Empfänger bekannt sind).

Beispiel 1: Person A hat einen manuellen Handshake mit Person B durchgeführt und den öffentlichen Schlüssel PK_B von Person B erhalten. Somit kann Person A den Schlüssel PK_B der Person B zuordnen. Wenn Person A in der Zukunft eine Infektionsnachricht erhält, dann kann sie alle zu diesem Zeitpunkt neu veröffentlichten Nachrichten auf dem Server überprüfen, und feststellen, ob zum Präfix des Schlüssels PK_B ebenfalls eine Infektionsnachricht hinterlegt wurde. War dieser Präfix nicht unter den neu hinzugekommen Nachrichten, kann Person A ausschließen, dass Person B soeben auch eine Warnung bekommen hat.

Beispiel 2: Die Tatsache, dass im Präfix von Person A in einem bestimmten Zeitraum eine Infektionsnachricht abgelegt wurde, erhöht die Wahrscheinlichkeit, dass Person A mit einer infizierten Person Kontakt hatte. Hat nun etwa eine Gruppe von Personen A, B, C, D, … in einem kurzen Zeitraum untereinander Handshakes durchgeführt, ist jedes Mitglied dieser Gruppe jeweils im Besitz der öffentlichen Schlüssel der anderen Personen PK_A, PK_B, PK_C, PK_D, … in der Gruppe. Werden nun zu den Präfixen von PK_A, PK_B, PK_C, PK_D, … in einem bestimmten kurzen Zeitraum Infektionsnachrichten angelegt, so ist es nun wahrscheinlicher, dass eine Infektionsmeldung einer Person innerhalb dieser Gruppe stattgefunden hat.

Beispiel 3: Da eine meldende Person keine Infektionsnachricht an sich selbst erhält, wäre es im Szenario von Beispiel 2 sogar möglich zu bestimmen, welche Person die Infektionsmeldung durchgeführt hat wenn ihrem öffentlichen Schlüssel ein eigenes Präfix zugeordnet ist, also in Analogie zu Beispiel 1 der Empfang einer Infektionsnachricht ausschließbar ist.

Das Erkennen zeitlicher Korrelationen dieser Art kann durch wechselnde öffentliche Schlüssel und das Vermischen von neuen Nachrichten am Server (cf. sogenannte Mix-Netze [14]) erschwert werden. Ein solches Konzept kommt derzeit allerdings nicht zum Einsatz.

[14] https://dl.acm.org/doi/pdf/10.1145/358549.358563.

➔ **Es muss soweit als möglich ausgeschlossen werden, dass Infektionsnachrichten an bekannte öffentliche Schlüssel von Dritten zuordenbar sind.**

2.4. Security

2.4.1. Android App

2.4.1.1. Betriebssystemberechtigungen

Damit eine App bestimmte sensible Betriebssystemberechtigungen nutzen kann, müssen diese erst von dem Anwender freigegeben werden. Allerdings darf eine App nicht nach Erlaubnis für alle möglichen sensiblen Betriebssystemberechtigungen fragen, sondern nur für diejenigen, die vom Entwicklungsteam definiert wurden. Für die allgemeine Funktionsweise der App ist der Zugriff auf das Internet erforderlich. Des Weiteren verfügt die App über die Berechtigung die Batterieoptimierung zu deaktivieren, damit die App auch im Hintergrund und Schlafmodus ausgeführt werden kann. Diese Funktion wird für den automatischen Handshake benötigt.

Die Programmbibliothek des p2pkit erfordert folgende Berechtigungen:

- android.permission.ACCESS_WIFI_STATE
- android.permission.CHANGE_WIFI_STATE
- android.permission.CHANGE_NETWORK_STATE
- android.permission.ACCESS_NETWORK_STATE
- android.permission.BLUETOOTH
- android.permission.BLUETOOTH_ADMIN
- android.permission.ACCESS_COARSE_LOCATION

Die App hat somit die Möglichkeit den Zustand des Netzwerks und der W-LAN Schnittstelle auszulesen und zu verändern. Zusätzlich ist der Zugriff auf die Bluetooth Schnittstelle möglich und es kann eine Bluetooth Verbindung zu anderen Geräten hergestellt werden. Die letzte Berechtigung ermöglicht den Zugriff auf Standortdaten, wobei hierfür nur Informationen der W-LAN Schnittstelle und des mobilen Netzwerkes ("Handynetz") verarbeitet werden. Die Genauigkeit des Standorts entspricht daher ungefähr der eines Häuserblocks. Diese Berechtigung ermöglicht keinen Zugriff auf das GPS.

Die App verwendet zusätzlich Google Nearby Messages, welches mittels Bluetooth, Bluetooth Low Energy, W-LAN Informationen und dem Mikrofon (Signale im Ultraschallbereich) kurze Nachrichten austauscht und anschließend einen längeren Nachrichtenaustausch über das Internet ermöglicht. Da Google Nearby Messages auf Android ein Systemdienst ist, müssen die Berechtigungen nicht durch die Entwickler definiert werden. Der Anwender erhält bei der ersten Verwendung von Google Nearby Message eine Abfrage ob er mit dem Zugriff auf Standortdaten und das Mikrofon einverstanden ist.

Insgesamt konnte in dem analysierten Source Code im Bezug auf die angefragten Berechtigungen kein Fehlverhalten der App festgestellt werden. Die geforderten

Berechtigungen werden zielgerichtet eingesetzt. Es wurde kein Hinweis gefunden, dass Standortdaten oder Tonaufnahmen aufgezeichnet oder aus dem Gerät ausgeleitet werden.

2.4.1.2. Ablage des Schlüsselpaars auf Android

Das RSA-Schlüsselpaar wird über den Android Keystore Provider in einem geschützten Bereich des Mobilgeräts gespeichert. Sämtliche kryptografische Operationen werden durch spezielle Systemschnittstellen aufgerufen, sodass nicht einmal die App selbst Zugriff auf das rohe Schlüsselmaterial besitzt. Der Keystore Provider sorgt außerdem dafür, dass nur die App selbst berechtigt ist, Operationen mit diesem Schlüsselpaar durchzuführen; anderen Apps ist der Zugriff untersagt. Die öffentlichen Schlüssel der bei den Handshakes gesammelten Geräte werden in einer SQLite-Datenbank lokal im Datenverzeichnis der App abgelegt.

2.4.1.3. Anmerkung zur vierstelligen Zahl des manuellen Handshakes

Bei einem manuellen Handshake wird den Usern die eigene vierstellige Zahl und alle vierstelligen Zahlen der User in der Umgebung angezeigt. Aufgrund der verhältnismäßig geringen Anzahl an Möglichkeiten für die vierstellige Zahl besteht eine geringe Wahrscheinlichkeit, dass für zwei User die gleiche Nummer generiert wird.
In diesem Fall könnten diese zwei User bei der Auswahl für den Handshake von den anderen Teilnehmern nicht mehr auseinandergehalten werden. Dies könnte durch einen Neustart der App gelöst werden.
Bei einer vierstelligen Zahl ist die Wahrscheinlichkeit, dass dieses Ereignis bei einem bestimmten manuellen Handshake auftritt nur 1:10.000. Die Wahrscheinlichkeit, dass dies niemals passiert ist jedoch nach 7000 Handshakes nur mehr ca. 50%. In anbetracht der Userzahlen wäre es deshalb sinnvoll die Anzahl der Stellen zu erhöhen und/oder auf Alphanumerische Werte umzustellen.

2.4.1.4. Anmerkung zur potenziellen Exposition von Betriebssystemschwachstellen

Da die Applikation auf BLE setzt und dazu Bluetooth aktivieren und Nachrichten Broadcasten muss, kann es auf älteren ungepatchten Android Versionen dazu kommen, dass diese über Bluefrag (CVE-2020-0022)[15] attackiert werden können.
Dies ist eigentlich ein Problem des darunterliegenden Betriebssystems sowie der Versorgung mit Patches vom Hersteller und nicht der Applikation selbst. Die Applikation könnte jedoch das Security Patch Level des Devices prüfen[16] und ggf. den Dienst verweigern sofern kein Patch installiert ist. In diesem fall müsste das Patch level min. 2020-02-05 oder später sein[17].

2.4.2. iOS App

Dynamische Tests der Stopp Corona App wurden auf einem iPhone 7 mit iOS 13.3 durchgeführt. Ein tiefgreifenderer Einblick konnte über die Anwendung eines Jailbreaks mittels checkra1n 0.10.1 beta erreicht werden.

[15] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-0022.
[16] https://developer.android.com/reference/android/os/Build.VERSION#SECURITY_PATCH.
[17] https://source.android.com/security/bulletin/2020-02-01.

2.4.2.1. Betriebssystemberechtigungen

Damit eine App bestimmte sensible und datenschutzrelevante Betriebssystemfunktionen nutzen kann, müssen diese erst von dem User freigegeben werden. Dem User wird dabei ein Popup eingeblendet, in dem er auswählen kann, ob eine App diese Funktion verwenden darf oder nicht. Das Popup wird dann eingeblendet, wenn die App zum ersten Mal nach der Installation auf diese bestimmte Betriebssystem Funktionalitäten zugreifen möchte, die Entscheidung kann aber später in den Betriebssystemeinstellungen wieder abgeändert werden. Ein User kann also einer App den Zugriff auf eine bestimmte Funktion verwehren, in vielen Fällen wird die App dann aber in der Praxis nicht mehr richtig verwendet werden können, weil sich zentralen Funktionalitäten auf die Interaktion mit der Umwelt stützen (Standortbestimmung, Tonaufnahme über das Mikrofon, Interaktion über Bluetooth, etc). Im konkreten Fall bei der *Stopp Corona App* würde der Handshake dann nicht mehr reibungslos funktionieren.

Weiters darf eine bestimmte App auch nicht nach Erlaubnis für alle möglichen sensiblen Betriebssystem Funktionalitäten fragen, sondern nur für diejenige, die vom Entwicklungsteam definiert werden. Diese Berechtigungen können anhand der Einträge in der Datei Info.plist im Programmverzeichnis der installierten App eingesehen werden, wobei zusätzlich eine textuelle Begründung des Entwicklungsteams angegeben ist. Für die *Stopp Corona App* sind die folgenden Berechtigungen hinterlegt, nach der die App bei Bedarf um Erlaubnis fragen kann:

- NSBluetoothAlwaysUsageDescription = "The permissions are needed to improve the accuracy of the digital handshake.";
- NSBluetoothPeripheralUsageDescription = "The permissions are needed to improve the accuracy of the digital handshake.";
- NSLocationAlwaysUsageDescription = "The permissions are needed to improve the accuracy of the digital handshake.";
- NSLocationWhenInUseUsageDescription = "The permissions are needed to improve the accuracy of the digital handshake.";
- NSMicrophoneUsageDescription = "The permissions are needed to improve the accuracy of the digital handshake.";

Die ersten beiden Einträge betreffen den Gebrauch von Bluetooth, wobei die zwei Einträge aufgrund einer Rückwärtskompatibilität für ältere iOS-Versionen vorhanden sind. Laut Dokumentation weist NSBluetoothPeripheralUsageDescription alle iOS Versionen von 6 bis exklusive 13 auf die Verwendung von Bluetooth hin, ab iOS Version 13 wird der Eintrag NSBluetoothAlwaysUsageDescription benötigt.

Der Eintrag NSLocationWhenInUseUsageDescription erlaubt die Anfrage nach der Standortbestimmung bei Benutzung der App, NSLocationAlwaysUsageDescription nach der Standortbestimmung während sich die App im Hintergrund befindet. Letzterer Eintrag ist aller Wahrscheinlichkeit deswegen notwendig, damit der in der neueren Version der App eingeführte automatisierte Handshake funktioniert.

NSMicrophoneUsageDescription erlaubt den Zugriff auf das Mikrofon, damit der manuelle Handshake, der mittels Google Nearby auch auf dem Austausch von Tönen im Ultraschallbereich basiert, möglich ist.

Insgesamt konnte in dem analysierten Source Code im Bezug auf die angefragten Berechtigungen kein Fehlverhalten der App festgestellt werden. Die geforderten Berechtigungen werden zielgerichtet eingesetzt. Es wurde kein Hinweis gefunden, dass Standortdaten oder Tonaufnahmen aufgezeichnet oder in das Internet versendet werden.

2.4.2.2. Ablage des Schlüsselpaars auf iOS

Das RSA-Schlüsselpaar wird in der Keychain, dem im Apple-Ökosystem eingebetteten Passwort Tresor, abgelegt. Der Zugriff, sowohl auf den öffentlichen, als auch auf den privaten Schlüssel in der Keychain, ist dabei aufgrund des Zugriffsattributs kSecAttrAccessibleWhenUnlocked nur im entsperrten Zustand des Geräts möglich.

Die öffentlichen Schlüssel der bei den Handshakes gesammelten Geräte werden in einer SQLite-Datenbank lokal im Datenverzeichnis der App abgelegt. Das gleiche gilt für den eigenen öffentliche Schlüssel; auch dieser wird in der selben SQLite-Datenbank abgelegt, dies passiert allerdings erst beim ersten Absenden einer Erkrankungsmeldung.

2.4.2.3. Anmerkung zur vierstelligen Zahl des manuellen Handshakes

Bei einem manuellen Handshake wird den Usern die eigene vierstellige Zahl und alle vierstelligen Zahlen der User in der Umgebung angezeigt. Aufgrund der verhältnismäßig geringen Anzahl an Möglichkeiten für die vierstellige Zahl besteht eine geringe Wahrscheinlichkeit, dass für zwei User die gleiche Nummer generiert wird.
In diesem Fall könnten diese zwei User bei der Auswahl für den Handshake von den anderen Teilnehmern nicht mehr auseinandergehalten werden. Dies könnte durch einen Neustart der App gelöst werden.
Bei einer vierstelligen Zahl ist die Wahrscheinlichkeit, dass dieses Ereignis bei einem bestimmten manuellen Handshake auftritt nur 1:10.000. Die Wahrscheinlichkeit, dass dies niemals passiert ist jedoch nach 7000 Handshakes nur mehr ca. 50%. In anbetracht der Userzahlen wäre es deshalb sinnvoll die Anzahl der Stellen zu erhöhen und/oder auf Alphanumerische Werte umzustellen.

2.4.3. Backend

Das Backend teilt sich in die drei Komponenten Push Service Backend, SMS Notification Backend und RCA CoronaApp Backend auf:

- Push Service Backend: Abwicklung der Push Notifications mit dem Google Firebase Service
- SMS Notification Backend: Versendung von TANs mittels SMS
- RCA CoronaApp: REST Endpoints zur Verarbeitung der Anfragen der mobilen Applikationen

Ein kurzer Code-Review des Quellcodes zeigt, dass grundsätzlich im Hinblick auf Security nach modernen Best Practice Empfehlungen entwickelt wurde. Im Review haben wir

unseren Fokus primär auf risikoreiche und leicht auszunutzende Standardschwachstellen gelegt. Es folgt eine kurze Zusammenfassung an Hand der risikoreichsten Schwachstellenklassen laut OWASP (OWASP Top 10):

1. Injection-Schwachstellen: Aufgrund der konsequenten Verwendung von prepared Statements und ORM-Bibliotheken wurden keine SQL-Injections entdeckt. OS Command Injections sind nicht möglich, da das Backend keine OS Kommandos mit eingegebenen Daten aufruft.
2. Fehler in der Authentifizierung: Die App unterstützt keine User oder Passwörter.
3. Verlust der Vertraulichkeit von sensiblen Daten: Serverseitig werden für Geräte, die eine TAN anfordern, Mobiltelefonnummern gespeichert. Weiters werden serverseitig Infektionsmeldungen gespeichert. Mit der Infektionsmeldung schickt die mobile App auf Wunsch des Users einen Verdacht oder einen bestätigten Fall von COVID-19 zum Server. Sie enthält neben der Art der Meldung (gelb - Verdacht, rot - bestätigter Fall, grün - Gesundmeldung nach Verdacht), die Mobiltelefonnummer und eine verschlüsselte Meldung für jeden abgespeicherten Kontakt (digitalen Handshake). Dieses Verfahren wurde in Abschnitt 2.2. Architektur detaillierter betrachtet.
4. XML External Entities: Es konnten keine Stellen gefunden werden, wo die Backend-Applikation XML-Daten von den mobilen Apps entgegennimmt. Daher ist dieser Angriffsweg nicht möglich.
5. Fehler in der Zugriffskontrolle: Die mobilen Apps haben vollen Zugriff auf das REST-API. Hier findet keine weitere Zugriffskontrolle statt. Es wird verlangt, dass REST-Aufrufe den HTTP-Header AuthorizationKey mit einem bestimmten Wert gesetzt haben. Dieser Wert ist bei den mobilen Apps hinterlegt und damit quasi öffentlich. Der Header erfüllt seinen Zweck unbeabsichtigte Aufrufe zu erschweren. Mehr Bedeutung darf diesem nicht beigemessen werden.
6. Sicherheitsrelevante Fehlkonfigurationen: Es bestand keine Einsicht in Konfigurationsdaten. Dieser Punkt kann daher nicht beurteilt werden.
7. Cross-Site-Scripting (XSS): Das Backend validiert Daten an der REST-Schnittstelle so wie in Security Best Practices empfohlen. Ausgehende Daten entsprechen dem REST-Standard.
8. Unsichere Deserialisierung: Es werden im Backend keine serialisierten Java-Strings verarbeitet. Daher besteht keine diesbezügliche Gefahr.
9. Nutzung von Komponenten mit bekannten Schwachstellen: An Hand der Maven-Builddateien (pom.xml) konnten wir nachvollziehen, welche Drittbibliotheken in welchen Versionen verwendet werden. Eine Prüfung dieser Abhängigkeiten hat keine ausnutzbaren Schwachstellen in den eingesetzten Drittbibliotheken gezeigt.
10. Unzureichendes Logging und Monitoring: Diese Eigenschaft kann im Zuge eines Code-Reviews nicht beurteilt werden.

Über die angeführten Erkenntnisse hinausgehend, wurden einzelne sicherheitsrelevante Probleme identifiziert. Diese wurden an Accenture gemeldet und können aufgrund der unterzeichneten NDA erst nach einem 15-tägigen Responsible Disclosure Prozess offengelegt werden. Die festgestellten Probleme haben keine Auswirkungen auf die Sicherheit der Anwender.

2.4.4. Kryptografie

2.4.4.1. Abweichung von Best-Practice-Empfehlungen

Die Verschlüsselung von Infection Messages erfüllt im Anwendungsfall der Applikation hauptsächlich den Zweck zu beweisen, dass der öffentliche Schlüssel dem Sender der Nachricht bekannt war. Der Inhalt der Nachricht beinhaltet bis auf den gerundeten Zeitstempel der Begegnung und dem Typ der Warnung (Rot/Gelb/Grün) sowie eine zufälligen UUID für diese Nachricht keine relevanten Daten.
In der Applikation kommt dazu RSA (RSA/None/PKCS1Padding) mit einer Schlüssellänge von 1024 bit zum Einsatz. Um dem Stand der Technik zu entsprechen, müsste dieser Algorithmus mindestens mit einer Schlüssellänge von 2048 bit verwendet werden[18] (dies würde die Nachrichten größer sowie die Entschlüsselung ca. um den Faktor 7 langsamer machen). Ebenfalls wird das hauptsächlich noch aus Rückwärts-Kompatibilitätsgründen vorhandene Padding Scheme PKCS#1v1.5 verwendet (definiert in RFC 8017[19]). Neue Applikationen sollten, sofern RSA verwendet werden muss, jedoch zumindest auf ein ebenfalls in RFC 8017 definiertes OAEP Padding Scheme setzen, um die Wahrscheinlichkeit verschiedener Angriffe auf das Verschlüsselungssystem zu reduzieren (cf. [20]).
Ebenfalls hat es sich als best practice etabliert, sofern RSA zur Verschlüsselung von Daten benutzt werden soll, auf hybride Verschlüsselungs-Systeme zu setzen, bei denen RSA nur zur Verschlüsselung eines symmetrischen Schlüssels verwendet wird, mit welchem dann die eigentlichen Daten ver- bzw. entschlüsselt werden. Im Fall der *Stopp Corona App* wird RSA direkt verwendet, um die Nachrichten bestehend aus (Padding aus 37 “\0” Bytes, Warnungs Typ (Red,Yellow,or Green), Zeitstempel gerundet auf Stunde, zufällige UUID für diese Nachricht) zu Verschlüsseln. Abgesehen davon werden keine Daten von der Applikation mit diesem Verfahren verschlüsselt.

Können, beispielsweise aus Kompatibilitätsgründen, keine anderen Verfahren (e.g., ECC) eingesetzt bzw. das Protokoll verändert werden, wird empfohlen zumindest 2048 bit RSA mit OAEP padding zu verwenden welches ebenfalls in der eingesetzten javax.crypto.Cipher[21] Bibliothek in der mindest API version der App (23) vorhanden sind.

2.4.4.2. Keine Verifikation von Warnungen

Um eine Warnung an einen Teilnehmer zu schicken, reicht es aus, dessen öffentlichen Schlüssel zu kennen. Wenn eine vom Server abgeholte verschlüsselte Nachricht von der Applikation erfolgreich entschlüsselt werden konnte, gibt es keine Plausibilitätsprüfung mehr, ob diese Warnung wirklich auf einem vergangenen (auch lokal aufgezeichneten) Kontakt basiert, oder ob der darin enthaltene Zeitpunkt wirklich plausibel ist. Der für die gültige Verschlüsselung notwendige öffentliche Schlüssel stellt somit den einzigen Beweis einer vorherigen Begegnung da. Werden diese Schlüssel aber öffentlich gemacht oder wie im Fall von p2pkit oder Google Nearby Messages über eine zentrale Stelle verschickt, so können

[18] https://www.keylength.com/en/compare/
[19] https://tools.ietf.org/html/rfc8017#section-7.2
[20] https://www.iacr.org/archive/eurocrypt2000/1807/18070374-new.pdf
[21] https://developer.android.com/reference/javax/crypto/Cipher

diese Schlüssel missbraucht werden, um falsche Warnungen an deren jeweiligen Besitzer zu verschicken, obwohl kein tatsächlicher physischer Kontakt stattgefunden hat. Um dies zu bewerkstelligen, braucht ein Angreifer lediglich einen gültigen TAN. Mit einem TAN kann er maximal 500 solcher Nachrichten zu je 512 Byte an den Server schicken, welcher diese dann weiter an die jeweiligen Clients ausliefern würde. (Siehe auch die Bemerkungen zu Source Integrity und Broadcast Integrity im Abschnitt 2.2.2. Sicherheitseigenschaften)

Um das Verfolgen eines einzelnen statischen Schlüssels zu erschweren und um die Wahrscheinlichkeit solcher Falschmeldungen zu reduzieren, sollte das verwendete Schlüsselpaar ständig gewechselt werden, sowie der in der entschlüsselte Timestamp mit dem jeweiligen für dieses Zeitfenster verwendete Schlüsselpaar abgeglichen werden.

2.4.5. Kommunikation zwischen Client und Server

2.4.5.1. Certificate Pinning

Die *Stopp Corona App* prüft bei einem Verbindungsaufbau zum Server das TLS-Zertifikat der Webservice-Schnittstellen gegen eine Liste an öffentlichen Zertifikaten, die im Betriebssystem der Smartphones hinterlegt sind und von Certificate Authorities (CAs) ausgestellt wurden. In Bezug auf die Abhörsicherheit ist es eine sinnvolle Maßnahme, für eine weitere Erhöhung der Sicherheit, das Vertrauen im Zusammenhang mit der Zertifikatsausstellung auf nur eine (optional selbst verwaltete) CA zu beschränken („Certificate Pinning“).

Wenn ein Angreifer eine einzelne CA kompromittiert, kann dieser Man-in-the-Middle-Angriffe auf alle TLS-Verbindungen durchführen, die auf die Vertrauenswürdigkeit der gesamten Liste der CAs bauen. Das liegt daran, dass ein TLS-Serverzertifikat nicht fest an eine bestimmte CA gebunden ist, sondern potenziell jede dieser etwa 150 CAs TLS-Zertifikate für jede Domäne ausstellen darf.

Certificate Pinning kann diese Gefahr signifikant verringern, da in diesem Fall der Client (im vorliegenden Fall die *Stopp Corona App*) nicht mehr einer ganzen Liste vertraut, sondern lediglich einer oder zwei CAs oder gar einem spezifischen Zertifikat.

➔ In Bezug auf die Abhörsicherheit ist es eine sinnvolle Maßnahme, für eine weitere Erhöhung der Sicherheit, das Vertrauen im Zusammenhang mit der Zertifikatsausstellung auf nur eine CA zu beschränken („Certificate Pinning“). Wir empfehlen diese Maßnahme in die nächste Version einzubauen.

2.4.5.2. Serverseitige TLS-Konfiguration

Die App kommuniziert sowohl bei der Android als auch der iOS Version über HTTPS mit dem Backend. Dabei gilt die folgende TLS-Konfiguration.

Der Verbindungsaufbau ist über die folgenden Protokolle möglich:

- TLS 1.2

Die folgenden Protokolle sind *nicht* erlaubt:

- SSL 2
- SSL 3
- TLS 1.0
- TLS 1.1
- TLS 1.3

Die folgenden Cipher-Suites sind für TLS 1.2 erlaubt:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030) ECDH secp384r1 (entspricht 7680 bits RSA)
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f) ECDH x25519 (entspricht 3072 bits RSA)
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0x9f) DH 2048 bits
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (0x9e) DH 2048 bits

Die gegenwärtige TLS-Konfiguration wird durch die Testsuite von SSL Labs[22] mit der Gesamtnote A bewertet, wobei die Bestnote A+ wäre. Dabei erreichen das genutzte Zertifikat und der Protokollsupport die volle Punkteanzahl wohingegen der Schlüsselaustausch und die Cipher-Stärke 9/10 Punkten erreichen. Insgesamt entspricht die Konfiguration den Best Practice Empfehlungen und ist zufriedenstellend konfiguriert.

[22] https://www.ssllabs.com

3. Rechtliche Analyse

3.1. Involvierte Akteure - Datenschutzrechtliche Rollenverteilung

3.1.1. ÖRK als Verantwortlicher

Im Zusammenhang mit dem Download und der Installation der *Stopp Corona App*, bei den einzelnen Erfassungsvorgängen („digitaler Handshake“), bei der Meldung (des Verdachts) einer Covid-19-Erkrankung und einer Entwarnung bezüglich einer gemeldeten Erkrankung kommt es zur Verarbeitung personenbezogener Daten der User.[23] Datenschutz- rechtlicher Verantwortlicher (Artikel 4(7) DSGVO) für diese Verarbeitungen ist das ÖRK was in deren Datenschutzinformation (Artikel 13/14 DSGVO) zum Ausdruck gebracht wird.

3.1.2. Auftragsverarbeiter und technische Dienstleister

Das ÖRK stellt die *Stopp Corona App* nicht im Alleingang zur Verfügung, sondern bedient sich diverser technischer Dienstleister, die als datenschutzrechtliche (Sub-)Auftrags-verarbeiter (Artikel 4(7) DSGVO) bzw. zu qualifizieren sind. Konkret handelt es sich dabei um:

Accenture GmbH („Accenture“)

An Accenture wurden Entwicklung, Betrieb (Hosting/Backend) und Wartung der Software ausgelagert. Accenture ist damit Auftragsverarbeiter (Artikel 4(8) DSGVO) des ÖRK. Einige weiteren Dienstleister sind wiederum Sub-Auftragsverarbeiter von Accenture - andere sind direkt Auftragsdatenverarbeiter des ÖRK.

Microsoft-Konzern („Microsoft“)

Accenture verwendet den Cloudservice Azure der Microsoft Corporation („Microsoft”) als Sub-Auftragsverarbeiter. Microsoft untersteht dem EU-US-Privacy Shield, sodass eine Datenübermittlung gemäß Artikel 45(3) DSGVO grundsätzlich legal ist. Punkt 5.3.1 der Datenschutzinformation deutet an, dass es neben Microsoft noch weitere Sub-Auftragsverarbeiter gibt, lässt jedoch offen, um welche es sich dabei handelt, da in weiterer Folge nur Microsoft genannt wird (*„Accenture besorgt unter Heranziehung weiterer, einzeln genehmigter Dienstleister das Hosting und den technischen Betrieb der App und des Servers* [...]“).

Artikel 13(1) DSGVO erlaubt die Offenlegung von lediglich „Kategorien von Empfängern”. Wenn diese bekannt sind, ist jedoch von einer Pflicht zur Benennung der einzelnen

[23] Hinzu kommt die Verarbeitung für statistische Zwecke auf die in diesem Report nur oberflächlich eingegangen wird, da das das Ergebnis einer Aggregation zu Statistiken in der Regel keine Personenbezüge mehr aufweist.

Empfänger (also auch aller Sub-Auftragsverarbeiter) auszugehen.Auf Nachfrage legte Accenture eine Liste von Microsofts' Sub-Sub-Auftragsverarbeitern vor.

➔ **Eine klare Benennung aller Sub-(Sub-)Auftragsverarbeiter in der Datenschutzinformation ist nachzuholen.**

Uepaa AG („Uepaa")

Der automatische „digitale Handshake" wird durch die Software p2pkit von Uepaa umgesetzt, das als Sub-Auftragsverarbeiter von Accenture agiert.[24] Uepaa zieht Amazon Web Services Inc. („AWS") als Sub-Sub-Auftragsverarbeiter heran. Eine Datenübermittlung an Uepaa ist grundsätzlich zulässig, da für die Schweiz ein Angemessenheitsbeschluss gemäß Artikel 45(3) DSGVO besteht. Auch eine Datenübermittlung an Amazon unterliegt einem Angemessenheitsbeschluss, da Amazon dem EU-US-Privacy Shield untersteht.

Problematisch erscheint hingegen die vom ÖRK verlinkte Datenschutzrichtline der Uepaa.[25] Unklar ist, ob diese Datenschutzrichtlinie überhaupt anwendbar ist, wenn Uepaa nur als Auftragsverarbeiter agiert. Die Richtlinie beinhaltet jedoch auch Informationen zu Daten der User („End User Data"). Sie wurde gut zwei Jahre vor Einführung der DSGVO geschrieben und enthält weder alle nach Artikel 13 DSGVO erforderlichen Informationen, noch eine klare Trennung zwischen der Rolle als Auftragsverarbeiter und jener als Verantwortlicher.

Besorgniserregend ist insbesondere folgende Klausel: „*You grant Uepaa the right to use, reproduce and distribute Your Use Data and the End User Data in connection with Your use and the End User's use of the Service. Such data will be used to measure, customize, and improve Our Service*." Dies würde eine Nutzung von Daten der Betroffenen für Zwecke der Uepaa erlauben und stünde wohl im Widerspruch mit der Rollenverteilung nach DSGVO.

Der Auftragsverarbeitungsvertrag („AVV") zwischen Accenture und Uepaa wurde uns nur in zwei kurzen Auszügen vorgelegt, da es sonst unter eine Vertraulichkeitsklausel fällt.[26] Die im AVV definierten Zwecke umfassen dabei nur die Kernfunktionalität (Auffinden und Kommunikation zweier Geräte). Der AVV ist damit enger gefasst als die Klausel in der Datenschutzinformation des ÖRK. Eine Nutzung wie in der Datenschutzinformation angedeutet ist damit rechtlich unzulässig.

➔ **Eine Angleichung der Verarbeitungszwecke zwischen AVV und Datenschutzinformation ist dringend erforderlich. Aktuell verlinkt die Datenschutzinformation des ÖRK auf eine Datenschutzrichtlinie der Uepaa, die falsche (da durch den AVV ausgeschlossene) Verarbeitungszwecke angibt.**

Google-Konzern („Google")

Google (bzw Teile des Google-Konzerns) wird für die Bereitstellung der Dienste Nearby und Firebase Cloud Messaging herangezogen. Nearby wird zur Abwicklung des manuellen

[24] Klarstellung in einem E-Mail von Accenture vom 14.04.2020
[25] http://p2pkit.io/pdf/160718_PrivacyPolicy.pdf (abgerufen am 18.04.2020)
[26] E-Mail von Accenture vom 14.04.2020

„digitalen Handshakes“ verwendet, Firebase Cloud Messaging für den Versand von Push-Notifications für den Fall, dass ein Kontaktpartner des Users eine Covid-19-Erkrankung, den Verdacht einer solchen Erkrankung oder eine Entwarnung bekannt gegeben hat.

Irritierend ist, dass Google in Punkt 5.3.2 und Punkt 5.3.4 der Datenschutzinformation als Auftragsverarbeiter genannt ist, in der Datenschutz-Folgenabschätzung aber mitunter als eigenständiger Verantwortlicher.

Auf Nachfrage teilte Accenture bezüglich Google Nearby mit, dass Google „*kein Auftragsverarbeiter sondern von den UserInnen genutzter Service am Endgerät*“ sei. Firebase Cloud Messaging ist wiederum als Auftragsverarbeiter des ÖRK tätig.[27] Hier ist eine Klarstellung bzw. Korrektur der Datenschutzinformation erforderlich.

Insbesondere ist damit auch der pauschale Verweis auf die Privacy Policy von Google unzutreffend, da diese Policy für Fälle gilt, in denen Google datenschutzrechtlich Verantwortlicher ist. Nach dem Konzept des ÖRK wäre diese Policy lediglich für Google Nearby anwendbar. Für Firebase Cloud Messaging wäre hingegen die Datenschutzrichtlinie des ÖRK, also des Verantwortlichen, relevant. Zwischen dem ÖRK bzw. Accenture und Google ist hingegen der abgeschlossene AVV das relevante Dokument.

➔ **Eine klare Trennung der beiden Google-Dienste und der jeweils Verantwortlichen in der Datenschutzinformation ist sicherzustellen. Nur Dienstleister, die tatsächlich Auftragsverarbeiter im Sinne des Artikel 4(8) DSGVO sind, dürfen auch als solche bezeichnet werden.**

Apple Inc. („Apple“)

Firebase Cloud Messaging leitet Push Notifications für iOS-Geräte wiederum an den „Apple Push Notification Service“ weiter. Die Rolle dieses Dienstes (Auftragsdatenverarbeiter oder eigenständiger Verantwortlicher) ist unklar. Accenture hat eine Nachfrage dazu bis dato nicht beantwortet.

➔ **Eine Klarstellung in der Datenschutzinformation ist dringend erforderlich.**

Anmerkung zur Nutzung von US-Anbietern

Generell wurde kritisiert, dass das ÖRK primär US-amerikanische Anbieter nutzt, die insbesondere auch unter US-Überwachungsgesetze (wie z.B. den Cloud Act, EO 12.333 oder FISA 702) fallen können. Eine Pflicht zur Herausgabe nach US-Recht ist dabei unabhängig vom Status des Unternehmens als Verantwortlicher oder Auftragverarbeiter. Gerade Daten zu einer globalen Pandemie sind auch für US-Geheimdienste höchst relevant und erfüllen Tatbestände nach US-Überwachungsgesetzen.

Bei der Nutzung von iOS und Android besteht leider aufgrund der praktischen Verbreitung dieser Betriebssysteme für Smartphones aktuell keine Alternative. Es ist jedoch nicht davon

[27] E-Mail von Accenture vom 14.04.2020.

auszugehen, dass lokal in der App gespeicherte Daten von diesen Konzernen abgezogen werden und weiterverarbeitet werden.

Online-Dienste wie Microsoft Azure, Google Firebase Cloud Messaging, Google Nearby, Amazon Web Services via p2pkit oder die Apple Push Notifications erhalten zumindest gewissen Metadaten der User und könnten bei anderen Systemarchitekturen durchaus minimiert oder gänzlich vermieden werden.

Auch wenn diese Verarbeitung nach aktueller Rechtslage (insbesondere nach dem EU-US-Privacy Shield) legal ist, scheint die Einbindung von US-Diensten, auch wegen potentiellen Konflikten zwischen der DSGVO und US-Recht, durchaus praktisch problematisch. Eine Abkehr von diesen Auftragsverarbeitern ist daher dringend zu empfehlen.

➔ **Die Verwendung alternativer Auftragsverarbeiter, die nicht unter US-Gesetze fallen, ist zu empfehlen.**

3.1.3. Andere User als Datenquelle und -empfänger

Zu bedenken ist letztlich, dass im Zusammenhang mit der Benutzung der *Stopp Corona App* auch zu einem Datentransfer an die einzelnen User kommt. Über deren Endgeräte werden User-IDs (siehe Punkt 3.2.2.) anderer User erhoben, wenn ein „digitaler Handshake" vorgenommen oder Verdachtsfälle, Erkrankungen oder Entwarnungen gemeldet werden. Bei diesen User-IDs handelt es sich um (wenngleich stark pseudonymisierte) personenbezogene Daten.[28]

3.1.4. „Gesundheitsbehörden" und Bezirksverwaltungsbehörden als Datenempfänger?

In Punkt 5.2 der Datenschutzinformation wird die „[...] *möglicherweise gesetzlich erforderliche Übermittlung von Informationen über konkrete Infektionsfälle an die Gesundheitsbehörden auf deren Verlangen* [...]" genannt und auf § 10 DSG (Datenverarbeitung im Katastrophenfall) verwiesen. Zudem könne „[...] *auf Verlangen der Bezirksverwaltungsbehörden eine Pflicht des Verantwortlichen zur Auskunftserteilung über Verdachtsfälle und Infektionen nach § 5 Abs. 3 Epidemiegesetz 1950 bestehen* [...]".

Zwar wären derartige Datenübermittlungen gemäß Artikel 6(1)(c) und Artikel 9 (2)(i) DSGVO durchaus gerechtfertigt, allerdings lässt die Datenschutzinformation offen, ob solche Übermittlungen tatsächlich stattfinden, bzw. ob diese überhaupt technisch vorgesehen und möglich sind; eine Schnittstelle für diesbezügliche Datentransfers besteht nicht. Unklar ist zudem, welche konkreten Stellen mit „Gesundheitsbehörden" gemeint sind.[29]

Im Lichte des Artikel 12(1) und 13(1)(e) DSGVO empfiehlt sich hier eine Spezifizierung der konkreten Empfänger oder zumindest der Empfängerkategorien. Da das ÖRK nach eigenen

[28] Siehe im Detail Punkt 3.
[29] § 10 DSG spricht von „Verantwortlichen des öffentlichen Bereichs und Hilfsorganisationen".

Angaben zu den einzelnen Usern nur eine zufallsgenerierte User-ID sowie bei Meldung einer Covid-19-Erkrankung/eines Erkrankungsverdachts eine Telefonnummer speichert, ist zuletzt auch fraglich, wie gut nutzbar diese Daten für die genannten öffentlichen Stellen tatsächlich wären.

➔ **Es wäre wünschenswert jene Daten die konkret technisch beauskunftet werden können und die im österreichischen Recht bekannten Fälle der Beauskunftung klar zu benennen.**

3.2. Verarbeitete Daten, korrespondierende Verarbeitungszwecke und Rechtsgrundlagen (Artikel 5(1)(b), 6 und 9 DSGVO)

3.2.1. Download und Installation der App

Für die Benutzung der App ist grundsätzlich keine Angabe personenbezogener Daten (wie Name oder Geburtsdatum) des Users notwendig. Bei Installation der *Stopp Corona App* wird dem User eine zufallsgenerierte statische Kennzahl (Unique Identifier ID, „UUID“) zugewiesen, die den User bzw. sein Endgerät individualisiert und als pseudonymes Datum (siehe Punkt 3.3.) zu qualifizieren ist.

Weiters wird ein asymmetrisches Schlüsselpaar auf dem Endgerät des Users erstellt, wobei der öffentliche Schlüssel zum Austausch beim „digitale Handshake“ und der private Schlüssel zur Entschlüsselung erhaltener Nachrichten (Verdachts- oder Erkrankungsmeldung, Entwarnung) dient.

Zweck dieser Verarbeitungen ist die Ermöglichung des verschlüsselten Informationsaustausches mit anderen Endgeräten. Das ÖRK stützt sich für diese Verarbeitungen auf die Rechtsgrundlage der (ausdrücklichen) Einwilligung gemäß Artikel 6(1)(a) und Artikel 9(2)(a) DSGVO. Der User wird beim erstmaligen Öffnen der App um die Abgabe folgender Einwilligung ersucht:

> *„Ich willige ein, dass das Österreichische Rote Kreuz (ÖRK) meine personenbezogenen Daten (eindeutige Kennzahl [ID], die IP-Adresse meines Endgeräts, meine Telefonnummer, Verdachtserhebung und Meldung meiner COVID-19 Erkrankung [=Gesundheitsdaten]) zum Zweck der schnellen Unterbrechung der Corona-Infektionskette verarbeitet. Darüber hinaus wird meine pseudonyme ID mit meinen Intensiv-Kontakten zum Zweck der späteren Warnung vor einer ärztlich bestätigten Infektionsgefährdung ausgetauscht.“*

Den Ausführungen im Bericht zur Datenschutz-Folgenabschätzung, wonach diese Einwilligung das Kopplungsverbot (Artikel 7(4) DSGVO) nicht verletzt, ist beizupflichten. Zwar ist die Benutzbarkeit der App an die Erteilung der Einwilligung geknüpft ist, allerdings

beschränkt sich die Einwilligung - soweit ersichtlich - auf das für die ordnungsgemäße Nutzung der App Erforderliche.[30]

Unklar ist jedoch, wozu eine Einwilligung zur Verarbeitung der IP-Adressen des Users eingeholt wird. Der Datenschutzinformation ist nicht zu entnehmen, zu welchem Zweck und auf welche Weise eine Verarbeitung der IP-Adresse durch das ÖRK erfolgt (jedenfalls zur Abwicklung der Kommunikation). Es wird lediglich festgehalten, dass weder Uepaa noch Accenture die IP-Adresse speichern. Auch die Datenschutz-Folgenabschätzung und die FAQs liefern keine genauere Erklärung für die Einholung der spezifischen Einwilligung.

Zu erwähnen ist, dass bereits vor Einholung der Einwilligung Datenverarbeitungen stattfinden: Es erfolgt ein Download von Konfigurations-Informationen und von Infection-Messages (Meldungen von Verdachtsfällen, Erkrankungen und Entwarnungen) von Microsoft Azure sowie die Registrierung bei Google Firebase. Diese Verarbeitungen werden in der Einwilligungserklärung bzw. der darin verlinkten Datenschutzinformation nicht genannt; auch in den Nutzungsbedingungen erfolgt kein Hinweis darauf. Hier ist eine Korrektur der Datenschutzinformation nötig.

Zu bedenken ist, dass für diese Verarbeitungen nicht zwingend eine Einwilligung erforderlich sein dürfte. Nachdem hierfür keine Verarbeitung besonderer Datenkategorien im Sinne des Artikel 9(1) DSGVO stattfindet kann die Verarbeitung auf ander Tatbestände des Artikel 6(1) DSGVO gestützt werden.[31]

➔ **Es ist klarzustellen, zu welchem Zweck IP-Adressen verarbeitet werden und auf welche Rechtsgrundlage jene Datenverarbeitungen gestützt werden, die bereits vor Erteilung der Einwilligung erfolgen.**

3.2.2. Erfassungsvorgang („digitaler Handshake“)

Beim automatischen „digitalen Handshake“ (nur zwischen Android-Geräten möglich) werden über das p2pkit von Uepaa zufallsgenerierte Kennzahlen erstellt („User-ID“;[32] nicht zur verwechseln mit der statischen UUID) und gemeinsam mit einer Zeitinformation, dem Gerätemodell, sowie Betriebssysteminformationen an die Server der Uepaa übermittelt. Nachdem die ausreichende räumliche Nähe zwischen zwei Geräten bestätigt wird (via Bluetooth und Wifi Direct), erfolgt ein Austausch der User-IDs zwischen den beiden involvierten Geräten, wobei hierfür das unter Punkt 3.2.1. genannte asymmetrische Schlüsselpaar zur Anwendung gelangt. Die User-ID des Kontaktpartners wird lokal auf dem Endgerät des Users gespeichert.

Beim manuellen „digitalen Handshake“, wird Google Nearby anstelle von p2pkit verwendet, um Daten auszutauschen, wobei hierbei Bluetooth, WLAN und auch Ultraschall-Signale eingesetzt werden können.

---

[30] Bericht zur Datenschutz-Folgenabschätzung , Seite 34, 35.
[31] Eine vertragliche Grundlage besteht bereits vor Einholung der Einwilligung: Den Allgemeinen Nutzungsbedingungen zufolge beginnt „*die Lizenzvereinbarung über die Nutzung der Softwarekopie der App [...] mit dem Download und endet mit der Löschung der App auf dem Endgerät des Users*“.
[32] Diese User-IDs werden alle 14 Tage automatisch geändert.

Der Zweck dieser Verarbeitungen besteht in der technischen Erfassung von Kontaktpersonen eines jeden Users, wodurch ein Kontakt-Tagebuch erstellt wird, auf dessen Basis in weiterer Folge Benachrichtigungen in zwei Richtungen erfolgen können: Zum einen, falls der User selbst an Covid-19 erkrankt, den Verdacht einer Erkrankung hat oder eine Entwarnung gibt, zum anderen falls dies bei einem seiner erfassten Kontakte der Fall ist (hierzu im Detail unter Punkt 3.2.3.) Als Rechtsgrundlage stützt sich das ÖRK auch hier auf die unter Punkt 3.2.1. genannte Einwilligung.

3.2.3. Vorfall (Meldung von Verdachtsfällen, Erkrankungen und Entwarnungen)

Wird ein User anhand seiner (nicht verifizierbaren) Angaben im Symptom-Checker als Verdachtsfall einer Covid-19-Erkrankung qualifiziert („Verdachtsfall“, gelb), kann er dieses Ergebnis freiwillig über die App melden. Selbiges gilt, wenn ein ärztliches Attest dem User eine Covid-19-Erkrankung bescheinigt („Erkrankung“, rot), wobei auch hier keine Verifizierung der Angaben des Users vorgesehen ist. Schließlich kann ein User eine Entwarnung absetzen („Entwarnung“, grün), was insbesondere dazu dient, ein negatives Covid-19-Testergebnis bei gemeldeten Verdachtsfällen zu berücksichtigen.

Der Zweck dieser Verarbeitungen ist evident und besteht in der Meldung dieser Ereignisse an die Kontaktpersonen des Users innerhalb der letzten 54 Stunden, was über Google Firebase Cloud Messaging per Push-Notification erfolgt. Die Empfänger der Meldung sind sodann dazu angehalten (eigenverantwortlich) entsprechende Maßnahmen zu ergreifen, um die Infektionskette zu durchbrechen. Bemerkenswert ist, dass in den Allgemeinen Nutzungsbedingungen ein abweichender, bzw. deutlich unschärferer Benachrichtigungszeitraum ausgewiesen ist: „*Verständigt werden ausschließlich jene Kontaktpersonen, mit denen der Nutzer in den vergangenen 3 Kalendertagen in Kontakt war*.“ Hier empfiehlt sich eine Präzisierung.

Vor Abschicken einer Meldung wird der User dazu aufgefordert, eine Mobiltelefonnummer anzugeben, an welche sodann per SMS eine TAN versendet wird. Erst mit korrekter Eingabe der TAN, wird die Meldung abgesetzt. Der Verarbeitungszweck besteht hierbei in der Verhinderung von missbräuchlichen (absichtlichen falschen) Meldungen (Hemmschwelle durch Preisgabe der Telefonnummer), sowie in der Möglichkeit der Kontaktaufnahme durch das ÖRK für Hilfeleistungen.

Als Rechtsgrundlage für das Absetzen der Meldungen dient abermals die unter Punkt 3.2.1. genannte Einwilligung. In Missbrauchsfällen wird die Telefonnummer auf Basis Artikel 6(1)(f) DSGVO (Wahrung berechtigter Interessen) und Artikel 9(2)(f) DSGVO (Geltendmachung, Ausübung oder Verteidigung von Rechtsansprüchen) verarbeitet.

Vor der erstmaligen Benutzung des Symptom-Checkers wird jedoch eine gesonderte Einwilligung eingeholt:

> „*Ich willige ein, dass das Österreichische Rote Kreuz (ÖRK) meine personenbezogenen Daten (eindeutige Kennzahl [ID], die IP-Adresse meines Endgeräts, meine Telefonnummer, Verdachtserhebung und Meldung meiner*

COVID-19 Erkrankung [=Gesundheitsdaten]) zum Zweck der schnellen Unterbrechung der Corona-Infektionskette verarbeitet. Darüber hinaus wird meine pseudonyme ID mit meinen Intensiv-Kontakte zum Zweck der Warnung vor einer Infektionsgefährdung sowie zur späteren Entwarnung oder Bestätigung der Infektion ausgetauscht."

Diese Einwilligung unterscheidet sich nur im letzten Satz geringfügig von der unter Punkt 3.2.1. genannten, indem auch die Entwarnung oder Bestätigung einer Infektion erwähnt werden. Warum für die Verarbeitungen im Zusammenhang mit dem Symptom-Checker eine gesonderte Einwilligung eingeholt wird, ist nicht ersichtlich und geht aus den bereitgestellten Dokumenten nicht hervor. Durch die zweite Einwilligung wird keine zusätzliche Transparenz erreicht: Sie ist nahezu wortgleich mit der ersten Einwilligung; beide Einwilligungen verweisen auf dieselbe Datenschutzinformation. Die Vermeidung der Bündelung mehrere Einwilligungen kann ebenso nicht ausschlaggebend sein, da sich bereits die bei erstmaliger Benutzung eingeholte Einwilligung auf mehrere (und in weiten Teilen dieselben) Verarbeitungen bezieht. Letztlich ist auch keine gesonderte Widerrufbarkeit der zweiten Einwilligung erforderlich, da die vom User angegebenen Antworten (Gesundheitsdaten) nach Beendigung oder Abbruch des Fragebogens ohnehin unverzüglich verworfen werden. Hier wäre anzudenken, die Verarbeitungen im Zusammenhang mit dem Symptom-Checker in der Datenschutzinformation verständlicher darzulegen und lediglich eine einzige Einwilligungserklärung zu verwenden.

➔ **Die Notwendigkeit einer zweiten, gesonderten Einwilligung für den Symptom-Checker ist zu überdenken.**
➔ **Der Benachrichtgungszeitraum (54 Stunden) sollte in allen Dokumenten einheitlich aufscheinen.**

3.2.4. Statistiken

Zuletzt werden zu statistischen Zwecken die Anzahl der App-Installationen sowie die Anzahl der Handshakes und der Meldungen gemeinsam mit einer stundengenauen Zeitinformation aufgezeichnet.

Die Rechtsgrundlage dieser Verarbeitungen ist § 7(1)(2) DSG in Verbindung mit Artikel 9(2)(j) DSGVO. Anhand der Statistiken soll dass aggregierte Nutzungsverhalten der User und die Verteilung von Handshakes über den Tag eruiert werden. Ebenfalls soll ermittelt werden, ob der durchschnittliche User auf Warnungen reagiert.

Wie im technischen Teil dieses Berichts beschrieben, bestehen Bedenken ob die Umsetzung dieser Funktion dem Gebot der Datenminimierung (Artikel 5(1)(c) DSGVO) entspricht. Falls technisch möglich eine datensparsamere Umsetzung geboten.

➔ **Es bestehen Bedenken ob die Statistik dem Gebot der Datenminimierung entspricht.**

3.3. Pseudonymisierung und Datenminimierung (Artikel 5(1)(c))

Wie dargelegt (Punkt 3.2.1.) stellt der User dem ÖRK grundsätzlich keine Daten wie Name, Geburtsdatum. etc. bereit. Lediglich bei Abgabe einer Meldung (Verdachtsfall, Erkrankung, Entwarnung) muss eine Mobiltelefonnummer angegeben werden, um per SMS eine TAN zu empfangen, mit der die Meldung freigegeben wird.

Daten, die das ÖRK generiert, um den User eindeutig zu identifizieren oder die verschlüsselte Kommunikation mit anderen Usern zu ermöglichen (UUID, User-ID, asymmetrisches Schlüsselpaar) sind allesamt als pseudonyme Daten im Sinne des Artikel 4(5) DSGVO zu qualifizieren, die lediglich einen indirekten Personenbezug aufweisen.

Den Ausführungen im Bericht zur Datenschutzfolgenabschätzung und den Bemerkungen in den FAQs ist insofern zuzustimmen: In Anbetracht der Rechtsprechung des EuGH in der Rechtssache *Breyer*[33] ist trotz der geringen Wahrscheinlichkeit eines Rückschlusses auf die Person hinter den vom ÖRK vergebenen Identifikatoren von pseudonymen und nicht von anonymisierten Daten auszugehen. Eine uneingeschränkte Anwendbarkeit der DSGVO ist daher gegeben; die Verwendung pseudonymer Daten ist jedoch eine sinnvolle technische Maßnahme im Sinne des Artikel 32 DSGVO die zur Minimierung der Risiken der Datenverarbeitung beiträgt.

Die vom User allenfalls bekanntgegebene Telefonnummer ist indes als direkt personenbezogenes Datum zu betrachten, da der User damit unmittelbar kontaktiert werden kann. Zu beachten ist, dass nach den derzeit vorliegenden Informationen offenbar keine Verknüpfung der UUID mit der vom User zum Erhalt der TAN angegebenen Telefonnummer erfolgt.

Die Umsetzung der App, ohne dass für die Verwendung der Grundfunktionalitäten eine Bekanntgabe direkt personenbezogener Daten gefordert wird, ist unter Gesichtspunkten der Datenminimierung (Artikel 5(1)(c) DSGVO sehr begrüßenswert und trägt in dieser Hinsicht auch den Vorgaben von Privacy by Design gemäß Artikel 25 DSGVO Rechnung.

3.4. Speicherbegrenzung (Artikel 5(1)(e) DSGVO) und Datenlöschung durch den User

3.4.1. Widerruf der Einwilligung und sonstige Löschungen durch den User

Da der Großteil der Datenverarbeitungen auf Basis der Einwilligung des Users erfolgen, muss ein Widerruf der Einwilligung zu einer Löschung dieser Daten führen, es sei denn, es besteht eine andere Rechtsgrundlage, auf der die Daten verarbeitet werden (siehe Artikel 17(1)(b) DSGVO). Dies muss den vorliegenden Informationen zufolge die UUID, die

[33] EuGH 19.10.2016 C-582/14.

User-IDs und das asymmetrische Schlüsselpaar betreffen, nicht aber die im Zug einer Meldung erzeugten bzw. bekannt gegeben Daten (zu diesen siehe Punkt 3.2.3.).

Nicht transparent dargelegt wird in Einwilligungserklärung und der Datenschutzinformation, auf welche Art und Weise ein Widerruf erfolgen kann. Punkt 4.2. der Datenschutzinformation lässt sich lediglich entnehmen, dass eine Deinstallation bzw. Löschung der App einen Widerruf impliziert. Ein partieller Widerruf ist zudem dadurch möglich, dass der automatische „digitale Handshake" deaktiviert wird, wofür ein Button in der App besteht.

Diese Möglichkeit wird jedoch lediglich im Bericht zur Datenschutz-Folgenabschätzung erörtert,[34] findet sich jedoch nicht in der Datenschutzinformation, die insofern zu ergänzen wäre. Dem Bericht zur Datenschutz-Folgenabschätzung zufolge kann die Einwilligung zudem „*ab der Übermittlung der COVID-19 Krankmeldung oder Verdachtsmeldung [...] beim Datenschutzbeauftragten des Verantwortlichen per E-Mail, Telefon oder Post jederzeit widerrufen werden.*"

Vollkommen unklar ist, wie sich ein derartiger, an den Datenschutzbeauftragten gerichteter Widerruf auf die grundlegende Funktionalität der App (die diesfalls ja am Endgerät der Users verbleibt) auswirkt. Ebenso unklar ist, wie sich der Betroffene gegenüber dem ÖRK identifizieren soll (siehe hierzu im Detail Punkt 3.6.). Im Sinne einer größtmöglichen Transparenz wäre die Datenschutzinformation entsprechend zu ergänzen. Die Modalitäten eines Widerrufs sollten zudem bereits in der/den Einwilligungserklärung(en) angeführt sein.

Missverständlich ist zudem folgende Stelle in Punkt 6. der Datenschutzinformation: „*In der Regel sind Ihre personenbezogenen Daten für die Dauer der Nutzung nur in der App gespeichert. Sie können die Löschung dieser Daten jederzeit auf Ihrem Endgerät selbst durchführen.*" Zum einen wären bei Verwendung dieser Formulierung auch die Ausnahmen von der genannten Regel anzuführen. Zum anderen suggeriert der Satz dem User, er könne einzelne Datenobjekte (wie z.B. einzelne Handshakes) manuell von seinem Endgerät entfernen. Dies ist, soweit ersichtlich, jedoch nur durch Deinstallation bzw. Löschung der gesamten App möglich.

➔ **Die Modalitäten und Auswirkungen eines Widerrufs der erteilten Einwilligungen sollten verständlich dargestellt werden. Missverständliche Formulierungen, die eine manuelle Löschung einzelner Daten in Aussicht stellen, sollten umgeschrieben werden.**

3.4.2. Datenlöschungen abseits vom Widerruf / Speicherfristen

Neben der Möglichkeit des Users, durch einen Einwilligungswiderruf die Löschung personenbezogener Daten zu erwirken, werden personenbezogenen Daten gemäß den vorliegenden Dokumenten nach Ablauf der folgenden Speicherfristen gelöscht:

[34] Bericht zur Datenschutz-Folgenabschätzung, Seite 31 - 35.

Die zufallsgenerierten User-IDs werden gemeinsam mit einer Zeitinformation, dem Gerätemodell, sowie Betriebssysteminformationen an Uepaa übermittelt, dort für bis zu 14 Tage gespeichert (Punkt 5.3.3. und Punkt 6. der Datenschutzinformation).

Wie lange einzelne „digitale Handshakes" gespeichert werden, geht aus der Datenschutzinformation hingegen nicht hervor. Jedoch ist auf Seite 83 des Berichts zur Datenschutz-Folgenabschätzung zu lesen: „*Die digitalen Handshakes des Benutzers sind für die letzten 7 Tage verfügbar und werden danach automatisch gelöscht.*"

Dies ist mit der realen Funktionsweise der App nicht in Einstimmung zu bringen. So waren Handshakes (egal ob manuell oder automatische Handshakes) weiterhin in der Auflistung in der App zu sehen. Keiner der Handshakes verschwand - auch nicht nach über 17 Tagen. Eine Benachrichtigung der älteren Handshakes über eine Verdachtsmeldung erfolgte jedoch nicht. Es scheint also, als ob die lokale Liste zwar keine Löschungsroutine inkludiert, jedoch nur die Kontaktdaten des relevanten Zeitraums informiert werden. Auf Nachfrage bei Accenture wurde versichert, dass die relevante Löschungsroutine im nächsten Release inkludiert wird.[35]

Daten die anlässlich der Meldung eines Verdachtsfalls oder einer Erkrankung übermittelt werden, bleiben für 30 Tage nach dem Absetzen dieser Meldung gespeichert. Dies umfasst insbesondere auch vom User angegebene Telefonnummern. Bestehen konkrete Anhaltspunkte für ein gesetzwidriges bzw. missbräuchliches Verhalten, werden die Daten für einen Zeitraum von bis zu drei Jahren nach dem Absetzen der Krankmeldung gespeichert (Punkt 4.4. und Punkt 6. der Datenschutzinformation). Punkt 7.1.6. der Datenschutzinformation weist zudem darauf hin, dass der Verarbeitung einer Telefonnummer innerhalb der 30-tägigen Speicherfrist gemäß Artikel 21(1) DSGVO widersprochen werden kann. Konsequenterweise muss dies auch während der möglichen dreijährigen Speicherung gelten, da auch hier Artikel 6(1)(f) DSGVO als Rechtsgrundlage herangezogen wird. Die Datenschutzinformation wäre diesbezüglich zu ergänzen.

Nach dem Ende der Epidemie sollen alle Daten gelöscht werden, wobei eingeräumt wird, dass ein Ende derzeit nicht absehbar ist.

Unklar ist, für welche Dauer IP-Adressen gespeichert werden. Wie unter Punkt 3.2.1. dargelegt, willigt der User in deren Verarbeitung ein, wobei der konkrete Verarbeitungszweck im Dunkeln bleibt.

- ➔ **Wie auch der Verarbeitungszweck, bleibt die Speicherdauer von IP-Adressen im Dunkeln. Hier ist nachzubessern.**
- ➔ **Auch ist die Speicherdauer der „digitalen Handshakes" in der Datenschutzinformation auszuweisen. Die App muss die Handshakes umgehend nach der Löschfrist auch tatsächlich löschen.**
- ➔ **Zuletzt muss aus der Datenschutzinfomation hervorgehen, dass ein Widerspruchsrecht gemäß Artikel 21 DSGVO bezüglich sämtlichen auf Basis berechtigter Interessen erfolgenden Verarbeitungen besteht.**

[35] E-Mail von Accenture vom 15.04.2020.

3.4.3. Zielkonflikt zwischen Speicherbegrenzung und Datenrichtigkeit

Die Aktuellen Speicherfristen der App sind (auch im internationalen Vergleich) sehr kurz gewählt. Es ist in der Praxis durchaus vorstellbar, dass Personen eine Erkrankung erst deutlich später als eine Infektion mit SARS-CoV-2 erkennen. Daten können zu diesem Zeitpunkt schon gelöscht sein. Es scheint daher, dass im Sinne der Datenrichtigkeit (siehe unten) durchaus auch eine differenzierte Speicherfrist möglich wäre. Hierbei könnte etwa der Zeitraum der relevanten Information je nach Einzelfall (inklusive Fortschritt der Krankheit) gewählt werden um ein korrektes Zeitfenster auswählen zu können.

3.5. Problematik der Datenrichtigkeit (Artikel 5(1)(d) DSGVO)

3.5.1. Allgemeines

Nach Artikel 5(1)(d) DSGVO muss der Verantwortliche (hier das ÖRK) auch für die Richtigkeit der Daten innerhalb der Datenanwendung sorgen. Insbesondere bei Informationen zur Gesundheit und Daten zur Verbreitung von SARS-CoV-2 ist hier ein hohes Maß an die Datenqualität anzulegen.

3.5.2. Konsequenzen der Information

In der Praxis ist etwa an die Strafdrohungen in §§ 178 und 179 StGB zu denken, die eine vorsätzliche bzw. fahrlässige Gefährdung von Menschen mit übertragbaren Krankheiten mit bis zu drei Jahren Freiheitsstrafe ahnden. Ein User der nach einer Information über die mögliche Ansteckung *keine* Quarantänemaßnahmen einleitet, würde sich wohl nach diesen Bestimmungen verantworten müssen, wenn sich später herausstellt, dass er im betreffenden Zeitraum SARS-CoV-2-war. Auch eine zivilrechtliche Haftung wäre denkbar (etwa wenn ein Arbeitnehmer trotz Warnung weiter arbeitet und damit ein Unternehmen zum Stillstand bringt). User müssen sich daher wohl (zumindest) indirekt den Informationen, die in der App angezeigt werden, beugen.

Bei einer Fehlinformation würde der User sich hingegen ohne Grund in seiner Freiheit beschränken, da er im Zweifel von einer korrekten Information ausgehen muss und diese (auch wegen der anonymität der Information) nicht überprüfen kann.

Zusammenfassend ergeben sich mitunter massive Eingriffe in die Rechte der User bei Fehlinformationen durch die App.

3.5.3. Differenzierte Information

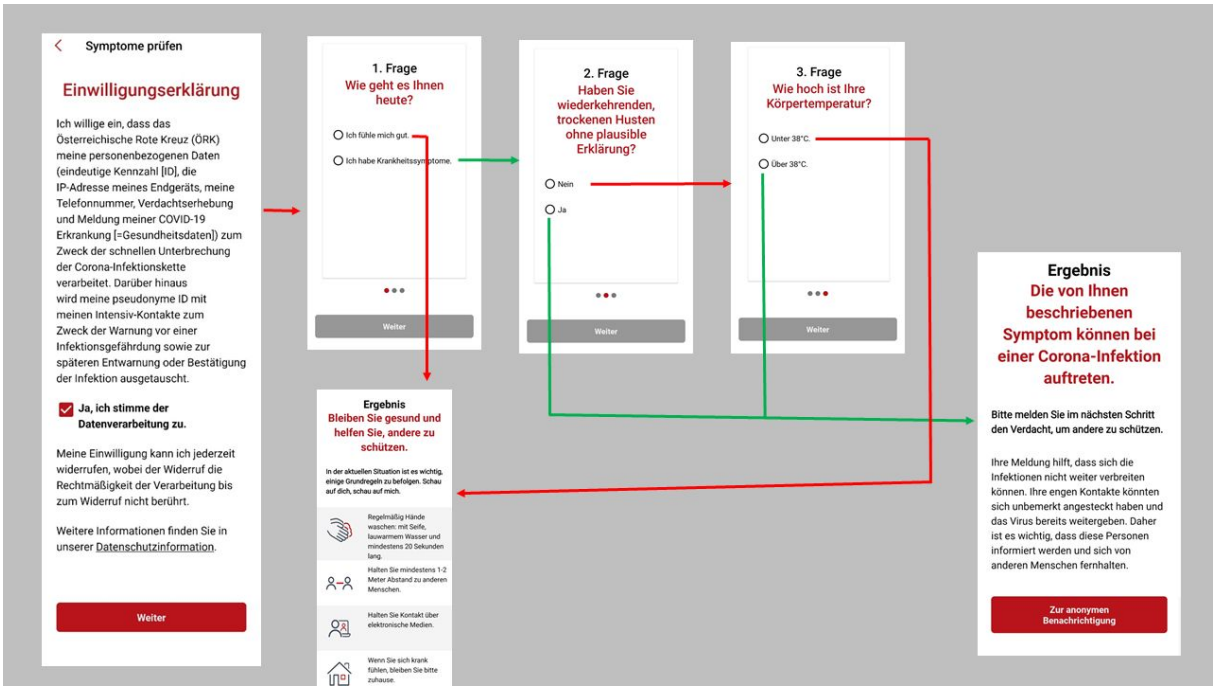
Positiv anzumerken ist das Konzept des ÖRK, hier eine differenzierte Meldung nach (1) Verdachtsfall, (2) bestätigte Erkrankung (positiver Test und) und (3) Entwarnung zu ermöglichen. Mit diesen verschiedenen Informationen kann ein User tatsächlich korrekte

Informationen übermitteln bzw. eine falsche Information sogar korrigieren. Andere Konzepte gehen bisher nur von einer Warnung aus, die auch nicht mehr korrigiert werden kann.

3.5.4. Fehlende Verifizierung der Userangaben

Wenn eine Person *de facto* nicht SARS-CoV-2-positiv ist, die Kontaktperson aber eine entsprechende Information erhält, sind die verarbeiteten Daten nicht „sachlich richtig" im Sinne des Artikel 5(1)(d) DSGVO. Nachdem das ÖRK es allein den Usern überlässt, eine Warnung, Positiv-Meldung oder Entwarnung einzugeben, bestehen Bedenken ob in der Praxis eine Datenrichtigkeit vom ÖRK gewährleistet werden kann:

Aktuell führt die Logik des Selbst-Tests (siehe unten) schon bei der positiven Beantwortung der Fragen nach „trockenem Husten" oder „hohem Fieber" zur Anweisung, dass der User die Warn-Funktion auslösen soll. Eine weitere Überprüfung durch eine dritte Stelle erfolgt nicht. User, die der inneren Überzeugung sind, dass sie SARS-CoV-2-positiv sind werden angesichts der Fragestellung wohl regelmäßig eine der Fragen positiv beantworten und werden dann unmittelbar zur Auslösung der Warnung ermutigt.



Selbst zur Spitzenzeit der Corona-Ausbreitung, in der nur sehr restriktiv und gezielt getestet werden konnte, waren nur etwa 20% der Tests positiv. Aktuell liegt dieser Wert bei etwa 15%. Es ist daher davon auszugehen, dass die Bevölkerung im großen Ausmaß eine Ansteckung bloß vermutet, sich eine solche jedoch nicht bewahrheitet. Nach Auskunft des ÖRK, wird genau eine solche „hohe Sensitivität" (also der Einschluss von möglichst vielen Personen) angestrebt um sicherzustellen, dass jeder Verdachtsfall und Kontakt überprüft wird.[36]

[36] E-Mail vom ÖRK vom 19.04.2020.

Die Logik der App verlangt auch keine anschließende Testung oder Entwarnung vom User. User können also nach einer abgesetzten Warnung durchaus keine weiteren Schritte setzen. Nach Auskunft des ÖRK wurde daher auch die vorgeschlagene Quarantäne auf 7 Tage beschränkt, da hier davon auszugehen sei, dass entweder ein Test erfolgt ist oder der User darauf vergessen hat: „*Eine Verdachtsmeldung, die sich nicht innerhalb von 7 Tagen* [...] *bestätigt oder verwirft wurde vermutlich vergessen.* [...]“.[37] Es stellt sich damit die Frage ob Fehlinformationen, die zu einem massiven Eingriff in die Rechte des Gewarnten führt, bei der derzeitigen Logik nicht wissentlich in Kauf genommen werden.

3.5.5. Angemessene Maßnahmen?

Artikel 5(1)(d) DSGVO verlangt keine absolute Richtigkeit von Daten, aber das Ergreifen von „angemessene Maßnahmen“ zur Sicherstellung von richtigen Daten. Angesichts der massiven Konsequenzen einer Falschinformation (*de facto* Freiheitsentzug) ist hier ein hoher Maßstab anzusetzen.

Die Verifizierung des Users über eine Telefonnummer (TAN) scheint hier zumeist ungeeignet, da übervorsichtigen Usern kein Vorwurf gemacht werden kann und selbst bei Missbrauch nicht klar ist wie dieser, bei einer für den Empfänger anonyme Information, aufgedeckt werden sollte.

Andere Konzepte basieren etwa auf “Tokens” (also ein Code) die von einer dritten Partei (Gesundheits-Hotline, Arzt, etc.) zur Verfügung gestellt werden, wenn ein Verdacht realistisch ist, bzw. wenn ein Test positiv ausgefallen ist. Dieser “Token” wären dann notwendig, um die Benachrichtigung über die App auszusenden. Hierbei ergäbe sich auch ein weiterer Vorteil: Diese Nummern können zufällig vergeben werden, was eine Registrierung der User mit einer Handynummer vermeiden würde, bei gleichzeitig massiv erhöhtem Missbrauchsschutz.

Der Eingriff in die Rechte des Gewarnten könne beispielsweise auch durch eine konkretere Information und damit eine erhöhte Planungssicherheit minimiert werden. So könne mit einer Warnung auch das konkrete Datum eines Tests und das Datum des zu erwartenden Testresultats mitgesendet werden statt einer pauschalen (und damit wohl oft falschen) Frist von sieben Tagen anzuzeigen. Auch dieses Datum könnte in einem Token der bei Vereinbarung des Test-Termins ausgegeben wird codiert werden.

Damit wäre sichergestellt, dass der Verdachtsfall (1) einen Test-Termin ausgemacht hat, (2) seine Symptome zumindest einem Dritten geschildert hat und (3) der Gewarnte eine korrektere Quarantänefirst durch die App angezeigt bekommt.

Gemäß Informationen des ÖRK[38] scheint die geplante Containment 2.0-Strategie zur epidemiologischen Eindämmung der Krankheit eine hohe Zahl von falsch positiven Meldungen in Kauf zu nehmen. Demzufolge wird es für vertretbar gehalten eine größere Anzahl an Menschen durch die App in Quarantäne zu schicken, auch wenn davon 90% nicht infiziert sind, anstatt die gesamte Bevölkerung ähnlichen Maßnahmen zu unterziehen. Es

[37] E-Mail von Accenture vom 15.04.2020
[38] E-Mail des ÖRK vom 19.04.2020.

wurden jedoch auch Möglichkeiten wie ein wiederholter Hinweis nach einer Warnung auch eine Entwarnung zu schicken oder ein kürzere Quarantänezeit im Fall einer Warnung bei schnellerer Testung ins Spiel gebracht.

➔ **Auch wenn es klar außerhalb unserer Expertise liegt epidemiologische Konzepte zu bewerten, scheinen durchaus „angemessen Maßnahmen" (im Sinne des Artikel 5(1)(c) DSGVO) zu bestehen, um falsche Informationen soweit wie möglich zu vermeiden oder möglichst schnell zu berichtigen, die bisher nicht ergriffen wurden.**

3.5.6. Dauer der Benachrichtigung

Im Testfall dauerte die Benachrichtigung über einen Verdacht etwa 15 Minuten und wurde durch eine Pop-Up am Gerät der Kontaktperson proaktiv angezeigt. Bei einer darauf folgenden Entwarnung ging über eine Stunde keine Nachricht am Gerät der Kontaktperson ein. Erst nachdem die App abermals geöffnet wurde, schien die Entwarnung auf.

Nach Information von Accenture sollte eine Information innerhalb einer Stunde am Gerät sichtbar sein. Es würde eine „silent push" verwendet. Bei längeren Fristen müsse ein technischen Problem vorliegen.[39]

➔ **Nachdem User mitunter fundamentale Entscheidungen treffen müssen, ist eine umgehende Übermittlung der Informationen sicherzustellen. Es ist zu empfehlen, die Übermittlungszeiten in der Datenschutzrichtlinie oder in den FAQs anzuführen.**

3.5.7. Einzelfallentscheidung gemäß Artikel 22 DSGVO?

In diesem Zusammenhang ist auch anzudenken ob die Information durch die App nicht auch eine „automatisierte Einzelfallentscheidung" nach Artikel 22 DSGVO darstellt. Beim aktuellen Stand der App ist das Vorliegen einer „Entscheidung" jedoch mangels einer gewissen Logik durch die App selbst zu verneinen. Noch scheint die App primär eine neutrale Nachrichtenvermittlung darzustellen.

Sobald jedoch die App eine (durchaus zu begrüßende) genauere Auswahl trifft, die diverse Faktoren wie die Nähe, Dauer und Zeit einer Infektion einbindet und damit eine Auswahl der gewarnten Kontaktpersonen trifft, wäre diese Frage weiter zu beleuchten. In ihrer Guidance on Apps supporting the fight against COVID 19 pandemic in relation to data protection hat die Europäische Kommission ebenfalls auf dieses Problem hingewiesen[40].

3.6. Geltendmachung von Betroffenenrechten

Punkt 7 der Datenschutzinformation listet - wie Artikel 13/14 DSGVO dies vorschreiben, die den Usern als betroffene Personen zustehenden Rechte gemäß Artikel 15 ff DSGVO auf.

[39] E-Mail von Accenture vom 15.04.2020
[40] Siehe Seite 7 https://ec.europa.eu/info/sites/info/files/5_en_act_part1_v3.pdf (abgerufen am 19.04.2020).

Ohne im Detail auf diese Rechte einzugehen, stellt sich die zentrale Frage, ob und wie ein User dem ÖRK gegenüber diese Rechte tatsächlich ausüben kann:

Die bei Installation zugewiesene eigene UUID oder die bei Handshakes verwendeten User-IDs sind dem User nicht bekannt; weder werden sie ihm in der App angezeigt, noch gibt es eine Möglichkeit diese zu exportieren. Andere Identifikatoren (wie Name, Geburtsdatum, Adresse oder Geräte-ID des Endgeräts) sind dem ÖRK nicht bekannt; auch werden bekanntgegebene Telefonnummern nicht mit der UUID des Users verknüpft.

Es stellt sich daher die Frage, wie sich ein User dem ÖRK gegenüber eindeutig identifizieren kann, um seine Rechte gemäß Artikel 15 ff DSGVO auszuüben. Sowohl die Datenschutzinformation als auch der Bericht der Datenschutz-Folgenabschätzung liefern keine Antwort auf diese Frage; in Punkt 7.1.8. der Datenschutzinformation wir lediglich der Datenschutzbeauftragte als Kontaktstelle für Betroffenenrechte genannt und eine E-Mail- und Postadresse, sowie eine Telefonnummer bereitgestellt.

Gemäß Artikel 11(2) DSGVO finden die Betroffenenrechte der Artikel 15 ff DSGVO keine Anwendung, wenn der Verantwortliche nachweisen kann, dass er nicht in der Lage ist, eine betroffene Person zu identifizieren. Das ÖRK kann sich jedoch nicht auf diese Bestimmung stützen, da mit der statischen UIID ein unveränderlicher Identifikator besteht, anhand dessen eine eindeutige Identifikation eines jeden Users möglich ist. Das Problem liegt hier einzig und allein darin, dass die eigene UUID dem User nicht zugänglich ist.

Gemäß Artikel 12(2) DSGVO ist das ÖRK verpflichtet, betroffenen Person die Ausübung ihrer Rechte gemäß den Artikeln 15 bis 22 DSGVO zu erleichtern. Ein Zugriff auf die eigene UUID ist daher unbedingt zu ermöglichen. Wird dies nicht umgesetzt, besteht zudem ein gravierendes Problem bezüglich der Grundsätze der Transparenz und der Verarbeitung nach Treu und Glauben gemäß Artikel 5(1)(a) DSGVO, da den Usern in der Datenschutzinformation eine unproblematische Möglichkeit zur Ausübung ihrer Betroffenenrechte suggeriert wird.

Diese Identifizierungsproblematik stellt sich zudem analog bei der Möglichkeit, einen Widerruf gegenüber dem Datenschutzbeauftragten zu erheben (siehe oben, Punkt 3.4.1.)

Generell ist anzudenken, dem User die Ausübung seiner Rechte auf Auskunft (Artikel 15 DSGVO) und Datenübertragbarkeit (Artikel 20 DSGVO) dadurch zu ermöglichen, dass jederzeit ein Export des derzeitigen lokalen Ist-Datenbestands der App möglich ist (insbesondere Download UUID, verwendeter User-IDs, Meldungen und Kontakte der letzten 54 Stunden).

➔ **Zumindest die eigene UUID ist dem User ersichtlich zu machen oder eine alternative Möglichkeit der eindeutigen Identifizierung zu schaffen, um die Ausübung von Betroffenenrechten und den Widerruf erteilter Einwilligungen zu ermöglichen. Andernfalls drohen Rechtsschutzdefizite, da sich kein User praktisch zur Ausübung seiner Rechte identifizieren kann.**

4. Dokumenthistorie

Version	Datum	Anmerkungen
1.0	21.04.2020	finale Version