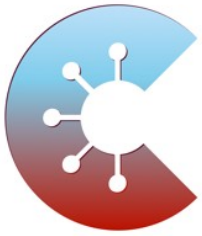

APPVISORY by mediaTest digital

APPVISORY PrePentest

# Corona-Warn-App (iOS)

**Durchschnitt CVSS 4.0 (medium)**

**Höchster CVSS 6.0 (medium)**

## App-Informationen

| | |
|---|---|
| • Hersteller | Robert Koch-Institut |
| • Name | Corona-Warn-App |
| • Betriebssystem | iOS |
| • Version | 1.3.0 |
| • Erscheinungsdatum | 04.09.2020 |
| • BundleIdentifier | de.rki.coronawarnapp |
| • Benötigte OS-Version | iOS 13.5 oder höher |
| • Store-Link | https://apps.apple.com/de/app/corona-warn-app/id1512595757 |
| • Store-Kategorie | Gesundheit und Fitness |

# Inhaltsverzeichnis

# 1 Testergebnis

| Corona-Warn-App | Version 1.3.0 | iOS |
|---|---|---|
| **Datenschutzverstöße** | | |
| • kein Verstoß festgestellt. | | |
| | Einstufung | **unbedenklich** |
| **Datensicherheitsverstöße** | | |
| • kein Verstoß festgestellt. | | |
| | Einstufung | **unbedenklich** |
| **AGB und Datenschutzerklärung** | | |
| • Keine Überprüfung der Datenschutzerklärung vorgenommen | | |
| | Einstufung | **nicht verfügbar** |
| **SSL/TLS Analyse** | | |
| • eine Man-In-The-Middle-Attacke war erfolgreich | | |
| | Einstufung | **bedenklich** |

## 1.1 Zusätzliche Funktionalitäten

Mobile Applikationen können über Drittanbieter erweiterte Funktionalitäten bieten. Dabei umfasst die Bandbreite neben In-App-Käufen zur Funktionserweiterung oder Entfernung von Werbung auch die externe Speicherung von Nutzerdaten in Cloud-Speicherdiensten, das Teilen von Meldungen über soziale Netzwerke oder die Registrierung eines Nutzer-Accounts über ein bereits bestehendes Profil eines sozialen Netzwerks.

### 1.1.1 In-App-Käufe

In der Applikation sind keine In-App-Käufe möglich.

### 1.1.2 Externe Cloud-Dienste

Die Applikation ermöglicht keine Sicherung von Daten über externe Cloud-Dienste.

Über Cloud-Dienste ist es theoretisch möglich, firmeninterne und möglicherweise geheime, sicherheitsrelevante oder anderweitig schützenswerte Daten manuell oder automatisch auf fremde, unbekannte Server eines Drittanbieters zu übertragen. Die Serverstandorte können im Ausland liegen. Es besteht die Möglichkeit eines unkontrollierten Datenabflusses.

### 1.1.3 Externe Login-Dienste

Die Applikation ermöglicht keinen verkürzten Registrierungs- und Loginprozess über externe Login-Dienste.

### 1.1.4 Social Network Sharing

Die App ermöglicht kein Sharing von Inhalten und Nutzerkommentaren über soziale Netzwerke.

### 1.1.5 Verwendung Webviews

Die Applikation verwendet keine Webviews.

# 2 Durchgeführte Analysevektoren

| TLS analysierbar? | ja |
|---|---|

| Sniffing Item | Kategorie | Übertragung festgestellt | Verschlüsselung |
|---|---|---|---|
| ETag | Metadata | **ja** | **ja** |
| meta-cwa-hash | Metadata | **ja** | **ja** |

**Hinweis:** Grundsätzlich wird empfohlen, die Kommunikation zwischen Applikation und Servern ausschließlich über verschlüsselte Verbindungen vorzunehmen.

# 3 Testergebnisse

In diesem Kapitel sind die Ergebnisse des Application Security Audits zusammengefasst. Das zugrunde liegende Testverfahren kann in der aktuell gültigen mediaTest digital Prüfspezifikation [separate Anlage] nachgelesen werden. Getestet wurde im Testlabor der mediaTest digital GmbH.

Das Testverfahren beinhaltet keine Aussagen über Funktionalität und Design sowie Performance und Stabilität der getesteten Applikation. Aussagen über Sicherheitslücken in angeschlossenen Backend-Systemen können lediglich in geringem Umfang getroffen werden.

Die vollständige technische Dokumentation kann im Kapitel 4 eingesehen werden. Dort finden sich ausführliche Informationen über Datenverbindungen und Datentransfer.

## 3.1 Test Setup

Der Testdurchlauf wurde im Testlabor der mediaTest digital GmbH durch einen Application Security Analyst durchgeführt. Getestet wurde auf einem physikalischen Testgerät. Grundsätzlich ist sowohl ein hoher Verbreitungsgrad des Testgerätes ebenso gegeben wie die Verwendung einer aktuellen Version des Betriebssystems.

- Testdatum 09. September 2020

| | |
|---|---|
| • Testgerät | iPhone 7 (iOS 13.5.1) |
| • Testumgebung | mediaTest digital Testlabor |
| • Prüfspezifikation | mediaTest digital Prüfspezifikation rev. 2.0 |

## 3.2 Tracking-Dienste

Während des Testlaufs wurde die Applikation hinsichtlich eingebundener Module und Bibliotheken von Drittanbietern untersucht. Diese Module bieten erweiterte Funktionalitäten (z.B. durch die Bereitstellung von Kartenmaterial), erlauben die Auslieferung von Werbung oder erheben Nutzerstatistiken.

Die Untersuchung des von der Applikation erzeugten Datenverkehrs während des Testlaufs sowie einer statischen Analyse zeigt die Einbindung folgender externer Dienste.

| Tracker | Beschreibung |
|---|---|
| • Keine Funde | |

## 3.3 Virenanalyse

Während der Virenanalyse über die virustotal API durchläuft die Applikationsdatei über 50 Virenscanner. So kann eine, je nach Ergebnis gewichtete Aussage, über eine potenzielle Bedrohung durch Viren oder Trojaner getroffen werden.

| | |
|---|---|
| • Dateiname | de.rki.coronawarnapp.ipa |
| • SHA256 Hashwert | ee80f8a34ac94eed46d0996ed09e324cfb989cfcb8801838b337ec49cc8d70c3 |
| • Virenscan-Ergebnis | **Funde: 0 von 61** |

## 3.4 Berechtigungsanalyse

Die Berechtigungsanalyse basiert auf einer statischen Analyse während des Testlaufs. Aufgerufene Berechtigungen werden – soweit möglich – dynamisch durch einen Application Security Analyst verifiziert. Schlägt eine manuelle Verifizierung fehl, wird die Berechtigung als „potenziell genutzt“ eingestuft.

Berechtigungen werden in mehrere Schweregrade eingestuft, die sich am Zugriff der Applikation auf Nutzerdaten orientieren:

| | |
|---|---|
| • system | Höchstes Risiko. Die Applikation muss mit dem identischen Zertifikat signiert sein wie die Applikationen auf Betriebssystemebene. Bei Aufruf der Berechtigung durch die Applikation wird der Nutzer **nicht** nach seiner Zustimmung gefragt. |
| • riskant | Hohes Risiko. Die Applikation erhält Zugriff auf Nutzerdaten oder weitreichende Funktionalitäten des Betriebssystems. Bei Aufruf der Berechtigung durch die Applikation wird der Nutzer nach seiner Zustimmung gefragt. |
| • normal | Niedriges Risiko. Die Applikation erhält isolierten Zugriff auf Applikationsebene und kann nicht auf Nutzerdaten oder weitreichende Funktionalitäten des Betriebssystems zugreifen. Bei Aufruf der Berechtigung durch die Applikation wird der Nutzer nach seiner Zustimmung gefragt. |
| • selbst erstellt | Durch den Entwickler selbst erstellte Berechtigung. Diese dient der Funktionalität der Applikation und erlaubt keinen Zugriff auf Nutzerdaten oder weitreichende Funktionalitäten des Betriebssystems. |

Die Applikation fordert die folgenden Berechtigungen an.

| Berechtigung | Beschreibung | Einstufung | Nutzung | Notwendig |
|---|---|---|---|---|
| Kamera | Erlaubt einer App den Zugriff auf die Kamera des Gerätes und die möglichkeit Fotos und Videos aufzunehmen | • riskant | Ja | ja |
| Bluetooth | Erlaubt einer App den Zugriff auf die Bluetoothfunktion des Gerätes | • riskant | Ja | ja |
| Mitteilungen | Erlaubt einer App den Zugriff auf die Push-Mitteilungs Funktion. | • normal | Ja | ja |

Die Berechtigungsanalyse zeigt die Verwendung von **3** Berechtigungen, die zur Verwendung aller Funktionen der Applikation benötigt werden. Die Applikation ruft nur die Berechtigungen ab, die für den Funktionsumfang notwendig sind. Die Applikation wird nicht als überprivilegiert eingestuft.

## 3.5 Binärdatei-Analyse

| Problematik | Schweregrad | Standards | Beschreibung |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Binary make use of insecure API(s) | High | **CVSS V2:** 6 (medium)<br>**CWE:** CWE-676 - Use of Potentially Dangerous Function<br>**OWASP Top 10:** M7: Client Code Quality<br>**OWASP MASVS:** MSTG-CODE-8 | The binary may contain the following insecure API(s) _fopen, _memcpy, _strlen. |
| Binary make use of malloc function | High | **CVSS V2:** 2 (low)<br>**CWE:** CWE-789 - Uncontrolled Memory Allocation<br>**OWASP Top 10:** M7: Client Code Quality<br>**OWASP MASVS:** MSTG-CODE-8 | The binary may use _malloc function instead of calloc. |
| Binary make use of Logging function | Info | **CVSS V2:** 7.5 (high)<br>**CWE:** CWE-532 - Insertion of Sensitive Information into Log File<br>**OWASP MASVS:** MSTG-STORAGE-3 | The binary may use _NSLog function for logging. |

## 3.6 Risikobewertung

Anhand des aufgezeichneten Datenverkehrs während des Testlaufs sowie auf Basis einer statischen Analyse werden Sicherheitslücken und damit potentielle Risiken überprüft, die mit der Nutzung der Applikation einhergehen können.

Die Auswertung des Testlaufs erfolgt halb-automatisiert und wird durch einen Application Security Analyst vorgenommen. Im Zuge einer Kernfunktionalitäts- und Plausibilitätsprüfung wurde dabei besonderer Wert auf Datenübermittlungen und Berechtigungen gelegt, die zur Ausführung der primären Funktionen der Applikation nicht unbedingt notwendig sind oder für den Nutzer keinen Mehrwert bieten.

Weiterführende Details zu während des Testlaufs festgestellten Datenschutz- und Datensicherheitsverstößen werden im Kapitel 4 in der Datenübertragungsanalyse aufgelistet. Die Handlungsempfehlungen bieten für den Entwickler der Applikation eine erste Hilfestellung, identifizierte Probleme zu beheben.

### 3.6.1 Datenschutzverstöße

In diesem Unterkapitel werden die Datenschutzverstöße aufgelistet und eine Handlungsempfehlung abgegeben. Als Datenschutzverstoß wird die Übertragung eines personenbezogenen[1] oder anderweitig sensiblen Datums an Dritte definiert.

| | |
|---|---|
| Keine Funde festgestellt | siehe **-** / **-** |

**Handlungsempfehlung:**

### 3.6.2 Datensicherheitsverstöße

In diesem Unterkapitel werden die Datensicherheitsverstöße aufgelistet und eine Handlungsempfehlung abgegeben. Als Datensicherheitsverstoß wird die Übertragung von Daten über eine unverschlüsselte Verbindung definiert.

| | |
|---|---|
| Keine Funde festgestellt | siehe **-** / **-** |

**Handlungsempfehlung:**

**HINWEIS:**

**Handlungsempfehlung:** Kein Handlungsbedarf

# 4 Verbindungsanalyse

In diesem Kapitel werden ausführliche Informationen über die während des Testlaufs aufgezeichneten Datenübertragungen bereitgestellt.

## 4.1 Analyse Serververbindungen

In diesem Unterkapitel werden die Server aufgelistet, zu denen die Applikation während des Testlaufs Verbindungen aufgebaut hat.

| Nr. | Servername | IP-Adresse | Port | Land | Betreiber |
|---|---|---|---|---|---|
| **V1** | https://svc90.main.px.t-online.de | 87.140.208.122 | 443 | DE | T-Online |

[1]

### 4.1.1 Auszug Datenübertragungsprotokoll

#### 4.1.1.1 Etag und meta-cwa-hash an t-online.de

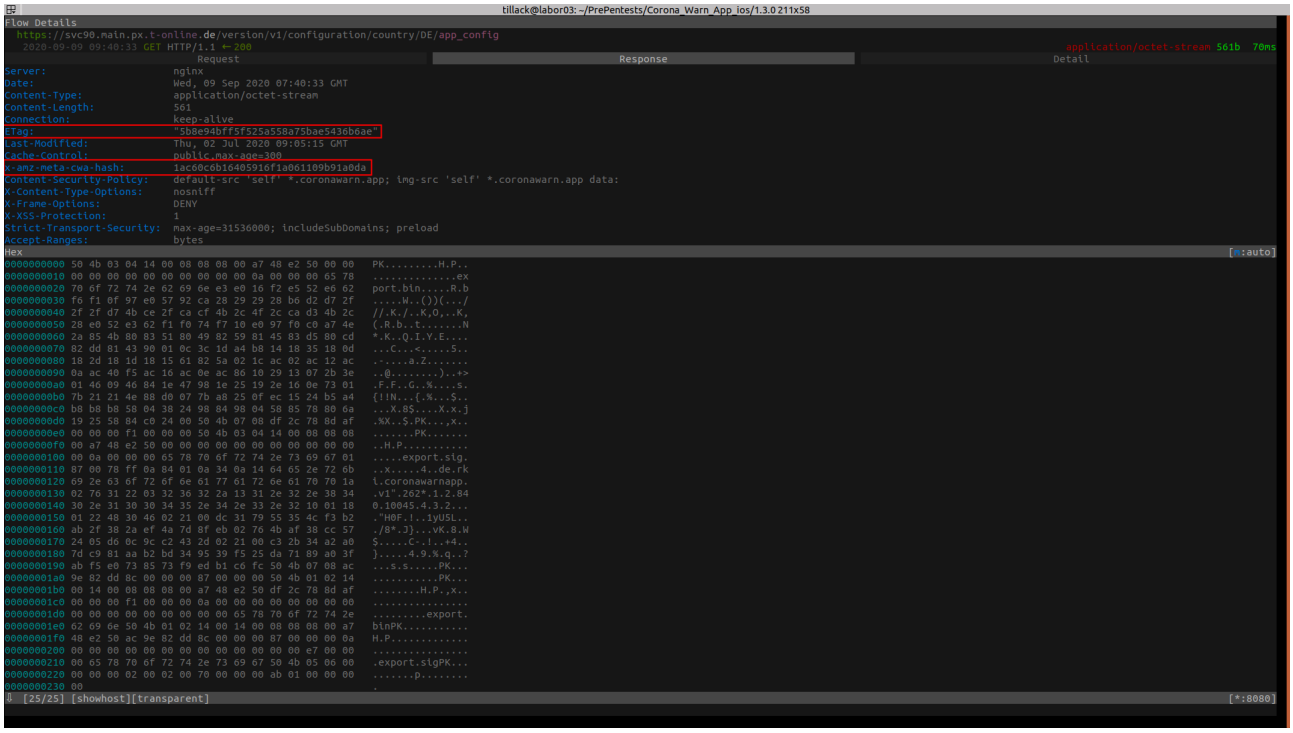


*Abbildung 1.1* Etag und meta-cwa-hash werden an t-online.de übertragen

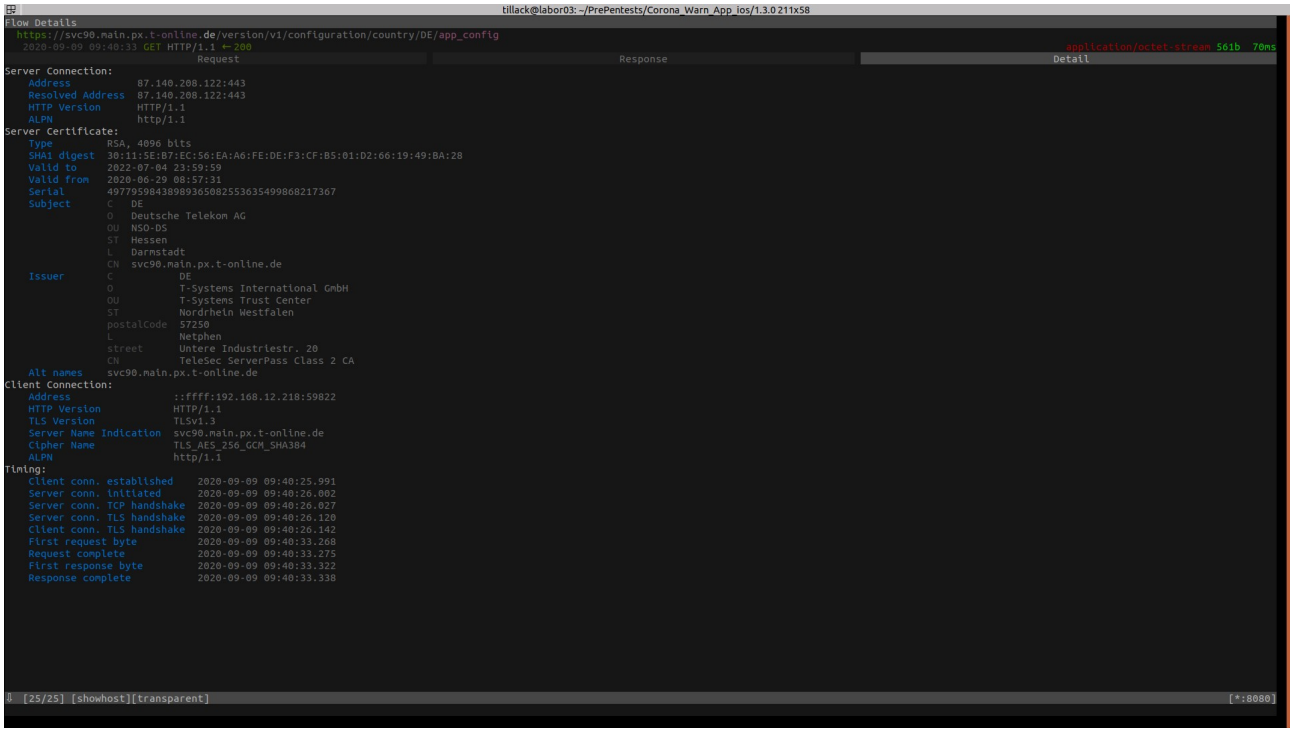


*Abbildung 1.2* Server-Details t-online.de

# 5 Mitigationstechniken

C-Based Toolchain Hardening sind Techniken um sicheren und zuverlässigen Code zu garantieren und bei dem kompilieren des Programmcodes helfen, bestimme Fehlerquellen in den Programmiersprachen Objective-C und Swift zu verhindern.

| | |
|---|---|
| Position-independent-Executable | **Enabled** |

*Eine Position-independent-Executable (PIE) kann an einer beliebigen Stelle im Speicher ausgeführt werden, ohne diese vorher zu modifizieren. Dies unterscheidet sich vom absoluten Code, welcher an einem bestimmten Ort geladen werden muss, um ordnungsgemäß zu funktionieren und load-time locatable Code der von einem Linker oder Program-loader vor der Ausführung modifiziert werden muss, so dass er von einer bestimmten Adresse im Speicher ausgeführt werden kann.*

| | |
|---|---|
| Automatic-Reference-Counting | **Enabled** |

*Automatic-Reference-Counting (ARC) - Unter Reference Counting versteht man in der Programmierung eine Technik zur Verwaltung der Anzahl der Verweise (Referenzen oder Zeiger) auf ein bestimmtes Objekt. Das Ziel ist dabei, zu erkennen, wann ein Objekt nicht mehr benötigt wird und gelöscht werden kann. Referenzzählung ist eine Möglichkeit zur automatischen Speicherbereinigung.*

| | |
|---|---|
| Stack-Canary | **Enabled** |

*Sprachen wie C erlauben aufgrund ihres Designs nicht immer die Überprüfung der Feldgrenzen (Beispiel: gets). Die Compiler müssen andere Wege gehen: Sie fügen zwischen der Rücksprungadresse und den lokalen Variablen Platz für eine Zufallszahl (auch „Canary" genannt) ein. Beim Programmstart wird diese Zahl ermittelt, wobei sie jedes Mal unterschiedliche Werte annimmt. Der erforderliche Code wird vom Compiler automatisch generiert. Vor dem Verlassen des Programms über die Rücksprungadresse fügt der Compiler Code ein, der die Zufallszahl auf den vorgesehenen Wert überprüft. Wurde sie geändert, ist auch der Rücksprungadresse nicht zu trauen. Das Programm wird mit einer entsprechenden Meldung abgebrochen. Bei jedem Unterprogrammaufruf wird die Zufallszahl in den dafür vorgesehenen Bereich geschrieben.*

# 6 Summary

Die Nutzung von Trackern oder Analyse-Tools konnte nicht festgestellt werden. Die Jailbreak-detection ist rudimentär implementiert. Gesammelte Daten werden möglicherweise nur lokal verarbeitet, da die volle Funktionsweise der Applikation uns während des Testlaufes nicht zur Verfügung stand (QR-Code). Es ist nicht absehbar was mit den gespeicherten Daten passiert, da die Kernfunktion der Applikation von dem jeweiligen Systemhersteller (Apple und Google) zur Verfügung gestellt wird.