www.nviso.eu

Coronalert Application

Public Report

Interfederal Committee Testing & Tracing

1 October 2020

About NVISO

NVISO is a cyber security consulting firm with offices in Belgium (Brussels) and Germany (Frankfurt, Munich). NVISO is exclusively focused on cyber security services, and has extensive expertise in security-critical industries such as financial services, government & defense and the technology sector. NVISO's people are recognized experts and actively present at major security conferences and teach at Universities, High schools and the SANS Institute: expertise and knowledge transfer is part of our DNA.

Disclaimer

The delivery to you and your use of the present document, is entirely subject to the present disclaimer. For users of this document with whom we have a written agreement, the below provisions will apply in addition to the relevant provisions in our agreement, and where and to the extent the below provisions would be in conflict with the terms of our agreement, the below provisions will take precedence.

Content - This document contains an overview of the security assessment as performed for Interfederal Committee Testing & Tracing on the Coronalert Application v1.0. This assessment was executed during the development of the application in August and September 2020 and thus contains a snapshot of the environment on that point in time. Changes after 18/09/2020 have not been assessed. The document does not deal with, and we therefore do not accept any liability or responsibility for, any other matter except as expressly set out in this document. Therefore, this document is not intended to be exhaustive, meaning that other security weaknesses than those discovered by us can exist. The document is made on a "point-in-time" basis which means it does not relate to nor makes any statement about, and we therefore do not accept any liability or responsibility for, any event taking place outside of the timeframe within which we delivered our services and as may be indicated in this report. This document does not constitute an official security certificate or attestation report.

Our liability - To the maximum extent permitted by law, we exclude all representations, warranties and guarantees not explicitly set out in this document. Without limiting the generality of the foregoing, we do not represent, warrant or guarantee that the product(s) or service(s) as described in this document is/are reliable, secure, safe or otherwise will comply with the user's needs. We explicitly exclude our liability for any event beyond our control and for indirect or consequential losses and/or damages including liability for loss or damage to income, profits, revenue, use, production, anticipated savings, business, contracts, commercial opportunities or goodwill or for any loss or corruption of data, databases or software. The limitations as set out before govern all liabilities which may arise on a contractual or non-contractual basis, whatever the legal theory invoked.

General - If a section of this disclaimer is determined by any court or other competent authority to be unlawful and/or unenforceable, the other sections of this disclaimer continue in effect. If any unlawful and/or unenforceable section would be lawful or enforceable if part of it were deleted, that part will be deemed to be deleted, and the rest of the section will continue in effect. This disclaimer will be governed by and construed in accordance with the laws of Belgium, and any disputes relating to this disclaimer will be subject to the exclusive

jurisdiction of the courts of Belgium. In this disclaimer, "we" and "us" refers to NVISO BV, with company number 0723.542.596.

1. Executive summary

Context

In August and September 2020, NVISO was tasked by the Interfederal Committee Testing & Tracing with the assessment of the security of the Coronalert mobile app, covering both the iOS and Android mobile applications and the supporting cloud (AWS) services. The application is based on the German COVID-19 application and uses the Google Apple Exposure Notification (GAEN) system to track prolonged proximity to other users who have installed the application. The operating system sends and receives Bluetooth beacons which can then be requested and shared by the application via cloud services.

Approach

The objective of the security assessment was the identification of potential security issues that could impact the confidentiality, integrity or availability of the application's data. Due to the nature of the application, a major area of focus were the security controls that ensure data confidentiality and enforce privacy. The engagement consisted of the following activities:

- General security review of the application and its backend based on the OWASP MASVS and ASVS;
- Security configuration review of the cloud services used by the application based on the CIS Benchmarks;
- Validation of compliance with privacy rules as determined by the Interfederal Committee Testing & Tracing and the Belgian Data Protection Authority;
- Verification of the implementation of the custom polling protocol, as this is a custom component that is not included in the German COVID-19 application.

These activities were performed throughout the development of the application, in order to ensure a rapid detection and resolution of security issues. Important to note is that our efforts are limited in time and performed on a specific version of the applications (refer to scope below).

The outcome of our activities were reported as *security vulnerabilities*, where the risk resulting from each vulnerability was evaluated based on the CVSS (Common Vulnerability Scoring System). The following section briefly describes open vulnerabilities.

Results

Application-level reviews

Throughout our assessment, 1 high-risk, 3 medium-risk and 3 low-risk vulnerabilities were identified. As these were reported during the development phase, 1 high-risk, 2 medium-and 2 low risk vulnerability have already been remediated. As a result, 1 medium-risk and 1 low-risk vulnerability is still present in the application at the date of this report's publication. These will be mitigated in future developments of the applications.

These issues can be summarized as follows:

- [Medium-risk] Since the application does not remove the status of a received COVID 19 test, simply opening the application will show the result of a COVID 19 test of the owner of the phone (Finding #A04, medium-risk). In case the phone is lost or stolen and found in an unlocked state or without any authentication required for unlocking the phone, anyone could identify if the owner of the phone was infected with the COVID 19 virus, impacting the owner's privacy. In addition, the application could be wrongly used to act as attest the phone's holder COVID-19 infection state.

- [Low-risk] The configuration of the TLS services used to transmit traffic between the mobile application and the cloud back-end should be updated to fully meet security best practices (Finding #A05, low-risk). In the unlikely event the traffic is captured from a single mobile phone and the cloud services (a man-in-the-middle scenario), the traffic may possibly be reversed to a readable format given sufficient time and computer power and thus be read by an unauthorized party who managed to intercept and decrypt the traffic originating from the targeted phone.

Cloud-level Review

The cloud review consisted of a verification of 86 security controls defined in the industry standard CIS Benchmarks. Out of these 86 controls, 16 were insufficiently implemented or required an improvement. As these were reported during the development phase, 5 of these security controls have been additionally implemented and verified by NVISO and 7 have been implemented by IXOR, only 4 controls have not been implemented yet and have a security impact as per their CVSS rating. The medium risk (#IAM03) and low risk (#DP01) will be implemented by Ixor.

These vulnerabilities can be summarized as follows:

- [Medium-risk] In the unlikely event the service account's credentials are leaked or stolen, they can be abused to read, modify and delete data and services within the cloud environment from any location without any network restriction in place. (Finding #IAM03, Medium-risk).
- [Low-risk] In the unlikely event someone would be able to capture traffic between the AWS components within AWS, it might be the data is not transmitted in an encrypted manner as it is not enforced to be enabled by the AWS configuration. At the moment of our review the encryption was applied though (finding #DP01).
- [Low-risk] The availability of the cloud environment could be jeopardized as there is a redundant VPN (Finding #BC01, low-risk) connection towards the Sciensano back-end which is not automatically failing over and relies on a manual intervention. In addition, the advanced AWS DDoS protection is not enabled on the environment (Finding #NET03, low-risk). The latter is however mitigated to limit the amount of requests per application-user to a pre-defined amount and only allow data to be transmitted from within the European Union (including Switzerland).

Summary of our recommendations

Security recommendations were issued throughout the project, and a number of vulnerabilities were resolved as indicated above. Some vulnerabilities remain, and the following actions would contribute to their reduction:

- Remove the test result of the application user from the application after a certain number of hours to limit the time the results remain visible in the application (Mitigating finding #A04);
- Configure TLS services according to industry best practices and remove support for older ciphers and TLS versions (Mitigating finding #A05);

- Restrict API calls from outside the AWS environment by adding network access conditions (Mitigating finding #IAM03);
- Enforce encryption between the different AWS components (Mitigating finding #DP01);
- Enable automated VPN tunnel failover towards the cloud environment from the Sciensano back-end (Mitigating finding #BC01).

2. Contents

1. Executive summary 3
Context 3
Results 3
Application-level reviews 3
Cloud-level Review 4
Summary of our recommendations 4
2. Contents 6
3. Scope of the project 7
Application-level reviews 7
Cloud-level Review 8
4. Project Approach 9
Application-level reviews 9
Cloud-level Review 13
Reporting 13
5. Assessment results overview 14
Application-level reviews 14
Cloud-level Review 16

3. Scope of the project

Application-level reviews

The following applications were reviewed as part of this assessment based on which the vulnerabilities are described:

Platform	Package name	Type	Version
iOS	be.sciensano.coronalertbe	Staging	1.0.2 (Build version 50)
Android	be.sciensano.coronalertbe	Staging	1.4.8 (Build 68)
Android	be.sciensano.coronalertbe	Production	1.4.8 (Build 67)

Testing was performed on the Staging build, with additional static checks on the Production binary. For iOS, the Staging binary was used for Production checks as it was said to be identical to the Production build, with the exception of the targeted endpoints.

After reviewing the builds mentioned above a final validation has been done on the production builds (version 1.6.0) with a specific focus on the identified vulnerabilities.

Platform	Package name	Type	Version
iOS	be.sciensano.coronalertbe	Staging	1.0.2 (Build version 50)
Android	be.sciensano.coronalertbe	Staging	1.4.8 (Build 68)
Android	be.sciensano.coronalertbe	Production	1.4.8 (Build 67)

The following URLs were reviewed as part of this assessment:

URL	IP	Type	Description
c19-verification-tst.ixor.be	18.195.200.203	Test	The verification server
c19-submission-tst.ixor.be	18.194.235.151	Test	The submission server
c19distcdn-tst.ixor.be	13.227.219.67	Test	CDN with submitted TEKs
c19-verification-stg.ixor.be	52.28.172.216	Staging	The verification server
c19-submission-stg.ixor.be	52.29.127.227	Staging	The submission server
c19distcdn-stg.ixor.be	13.227.219.8	Staging	CDN with submitted TEKs
	94.143.186.17	Production	IPsec Endpoint
c19distcdn-prd.ixor.be	54.192.86.110	Production	CDN with submitted TEKs
c19-verification-prd.ixor.be	52.57.107.232	Production	The verification server
c19-submission-prd.ixor.be	3.124.88.228	Production	The submission server

Testing was performed on the Test and Staging environment. External infrastructure scans have been replicated on Production endpoints.

In case multiple IPs were available for a single domain, one of the IPs was selected as being representative.

Cloud-level Review

The cloud configuration review of the AWS environment consisted of an assessment of resources and services that are hosted on the AWS Production account installed by Ixor to run the Coronalert application.

The following AWS services have been reviewed as part of the assessment:

IAM	CloudTrail	CloudWatch	S3
VPC	EC2	EBS	KMS
Config	Security Hub	GuardDuty	Inspector
CloudFront	ELBv2	ECS (Fargate)	RDS
ACM	Shield	WAF	Resources Manager
System Manager	VPN	SNS	

Out of scope

The following items were out of scope for this assessment:

- The generation and storage of Bluetooth tokens and TEKs. These are generated by either Google or Apple (GAEN) and are not under the control of the application. As soon as the application receives the TEKs, they are considered in scope.
- The security of the medical professional's connection to Sciensano, the applications used by Sciensano, the internal infrastructure of Sciensano and the e-forms solution used by Sciensano
- Licensing verification of 3rd party software libraries
- Maintainability of the code base, including coding style and code quality
- A source code review of the used native libraries (SQLite)
- A full review of the implementation and decisions made by the German COVID-19 application team
- Distributed Denial of Service attacks (DDoS)
- The specific configuration of the Operating System of EC2 instances, applications and containers

4. Project Approach

The general approach is described below, followed by a more specific explanation of the different aspects of the assessment.

Application-level reviews

General approach used during development

The development of the application and the accompanying backend was performed in several sprints by Ixor and Devside. Sprints are documented on the Devside Jira and tickets are assigned to different sprints. At the end of each sprint, the following actions were taken:

- Consult all Jira tickets marked as 'done' for the previous sprint
- Map the Jira tickets to pull requests on the Android, iOS and backend repositories whenever a code change was committed
- Perform dynamic testing on the features developed during the sprint, whenever both frontend and backend were available and aligned
- Verify the new implementation against the MASVS and ASVS

For each identified issue, a new Jira ticket was created and assigned to the relevant parties. Missing security controls were only reported in case there was no Jira ticket to implement the missing security control planned for one of the future sprints.

As a result, the issues documented in this report are only those which would not have been identified without the security assessment.

Web application assessment approach

The Coronalert application was assessed according to the latest version of the OWASP Application Security Verification Standard (ASVS v4.0.1). This assessment was performed manually from an unauthenticated perspective and included the testing categories noted below. Each category represents a control objective in the OWASP ASVS:

Authentication	Session management	Access control
Input validation	Stored cryptography	Error handling
Data protection	Communication	Malicious code
Business logic	File & resource	API
	Configuration flaws	

Infrastructure assessment approach

For each server defined in the scope, the following scans were performed:

Type	Range
TCP	All ports
UDP	Top 1k ports according to nmap

For each identified service, an automated vulnerability assessment is executed, coupled with a manual validation of any identified issues.

Mobile application assessment approach

The mobile applications were assessed according to the OWASP Mobile Application Security Verification Standard (MASVS) Level 2 (v1.2). The table below provides an overview of the different chapters included in the OWASP Mobile Application Security Verification Standard:

Data Storage	Cryptography	Authentication and Session
Authentication Management	Session Management	Network Communication
Platform Interaction	Code Quality	Build Settings

All tests were performed from a white-box perspective, i.e. NVISO had access to the underlying mobile application source code and documentation.

Denial of Service assessment approach

On 4 September 2020, tests were executed to test the limitations and the saturation point of the coronalert servers (including CDN). The tests were divided as follows, each of these three types of test were executed for both the CDN and the backend:

- Geographical limit: only EU countries and Switzerland can send requests and receive either a 200-OK or 204-No Content HTTP response, as opposed to a 403-Forbidden response for other countries outside of this region.
- IP-based rate limiting:
    - CDN: one IP address can send a maximum of 16.000 requests per 5 minutes, equally resulting in 403 responses after the limit has been reached.
    - Backend: one IP address can send 100 requests per 5 minutes.
- Load test: ramping up the number of requests to determine the saturation point of the server, considering a maximum of 50.000 requests per second.

Validation of identified threat model risks

The following identified risks from the threat model assessment are taken into account where appropriate:

Abuse case	Scope	In scope	Reason
As a malicious user, I should not be able to find a vulnerable version of a third party library used in the mobile application or backend services when reviewing the open source codebase. A vulnerable version is a version that has known, public vulnerabilities that can be used to manipulate the normal flow of the application.	iOS/Android	Yes	
As a malicious user, after capturing a large amount of BLE tokens generated by the COVID-19 Alert applications around me, I should not be able to replay this traffic on a large scale in a successful attempt to halt or degrade the	GAEN	No	The GAEN implementation is

normal working of the application and/or mobile device.			not in scope of this assessment.[1]
As a malicious user, I should not be allowed to create multiple COVID-19 test requests toward the e-forms functionality with a value of d=1 and duplicates of already known R1 values in a short amount of time. If I do succeed in passing these COVID-19 test requests, an alert should be triggered that monitors for abnormal behavior patterns in the upload of COVID-19 test request from the same system with already known R1 values.	e-forms	No	The e-forms functionality is not in scope of this assessment.
As a malicious user, it should not be possible to send multiple requests in a short time span with an originating IP outside of Belgium and Switzerland towards the e-forms functionality.	Infrastructure	No	This threat is validated in the Distributed Denial of Service (DDoS) test.
As a malicious user, it should not be possible to send a maliciously crafted string through the e-forms functionality in any of the available fields (t0,R1) and have it passed unsanitized from the Sciensano system to the AWS Aurora datastore where a backend service or batch job uses this as unsanitized input in the program flow.	Backend	No	The e-forms functionality is not in scope of this assessment.
As a malicious user, I should not be able to upload malicious data during a TEK key upload after a positive test result, that is then used without sanitization by the backend program logic.	Backend	Yes	
As a malicious user, It should not be possible to derive the private values (R1,R0,K,t0,t1) from the mobile application datastore through the use of malware installed on the system.	iOS/Android	Yes	
As a malicious user, I should not be able to send maliciously crafted strings towards the API endpoints coming from a non-mobile endpoint without being logged or noticed through an alert.	Backend	No	This is validated in the AWS configuration review assessment.
As a malicious user, I should not be able to go through the open sourced codebase and uncover secrets in the form of e.g. hardcoded connection credentials that are still actively used in the normal operation of the production environment	iOS/Android Backend	Yes	

[1] Google and Apple have identified that the tokens submitted are only valid for approximately 30 minutes, effectively mitigating this attack. NVISO did not verify this behavior.

Business logic validation

The functional requirements as documented by the Interfederal Committee Testing & Tracing have been translated into the following abuse cases:

Abuse case	Scope
As a malicious user, I cannot successfully submit my TEKs if I do not have a positive test result	Backend
As a malicious user, I cannot submit my TEKs several times for a single positive test result	Backend
As a malicious user, I cannot submit my TEKs based on another user’s positive test result (i.e. tokens to identify the test are high entropy and are correctly checked)	Backend
As a malicious user, I cannot modify the TEKs which are sent to the server when sharing them after a positive test result	Backend
As a malicious user, I cannot send more than 14 TEKs to the server in one request	Backend

Privacy requirements validation

The privacy requirements as documented by the Interfederal Committee Testing & Tracing and the Data Privacy Impact Assessment (DPIA) have been translated into the following abuse cases:

Abuse case	Scope
As an attacker, I cannot obtain today's TEK of another user	Backend iOS/Android
As an attacker, I cannot acquire the test result of another user from the backend (i.e. tokens to identify the test are high entropy and are correctly checked)	Backend
As an attacker, I cannot retrieve personal information about any user of the application	iOS/Android
As an attacker with access to encrypted network traffic, I cannot deduce the result of another user's test based on the network traffic (i.e. no data should be deducible from encrypted packets and DNS queries)	Backend iOS/Android
As an attacker with access to encrypted network traffic, I cannot deduce if another user has been tested and is waiting for the results	iOS/Android
As an attacker with access to encrypted network traffic, I cannot deduce if the result of another user's test has been delivered	iOS/Android
As an attacker with access to another user's device, I cannot deduce if the user has been tested	iOS/Android
As an attacker with access to another user's device, I cannot deduce if the user has been tested positive	iOS/Android
As an attacker with access to another user's application backup, I cannot deduce personal information about the user	iOS/Android

As a malicious user, it is not be possible to derive the private values (R1,R0,K,t0,t1) from the mobile application datastore through the use of malware installed on the system.	iOS/Android

Cloud-level Review

Cloud Configuration assessment approach

The assessment was based on the AWS security baseline checklist created by NVISO, which includes CIS AWS Configuration Benchmark (v1.2.0)[2] checks, AWS Well-Architected Framework[3] Security Pillar recommendations and additional controls. Additional configuration verifications were performed to make sure that the services implement the best practices as suggested by AWS documentation and based on NVISO expertise.

In addition, the assessment considered specific requirements documented and provided by the customer regarding the default configuration for the AWS services such as VPN configuration, S3 configuration, encryption, etc.

Reporting

A security review of the new iterations of the application was executed each sprint. As the application was in a state of constant development, issues were not reported in case they were already known by the development team, and thus scheduled to be implemented in a subsequent sprint.

The issues reported in the overview of this report include both remediated and non-remediated issues that were identified either during the sprints, or during the final assessment after the last sprint, which were not yet known to the development team.

Finally, in addition to security vulnerabilities stated in this report, the MASVS, ASVS and cloud configuration reviews contain controls that do not pose a direct risk according to the CVSS rating to the security of the application or the data handled by the application. Only issues with a direct security impact are reported in this document.

[2] CIS AWS Configuration Benchmarks: https://www.cisecurity.org/benchmark/amazon_web_services/

[3] AWS Well-Architected Framework: https://aws.amazon.com/architecture/well-architected/

5. Assessment results overview

The table below provides a full overview of all identified findings, while the subsequent section includes full details on all identified findings.

Application-level reviews

ID	Finding Name	Risk	CVSS	Scope	Status
A01	Valid TEKs marked as invalid	**High**	7.7	Backend	Resolved
A02	Uploading of real TEKs can be identified in encrypted traffic due to incorrect padding	**Medium**	6.8	Android iOS	Resolved
A03	Positive result can be deduced from encrypted traffic	**Medium**	6.8	Backend	Resolved
A04	Application does not remove test result after upload of TEKs	**Medium**	5.9	Android iOS	Active
A05	Weak SSL/TLS configuration	**Low**	3.7	Backend	Active
A06	Delivery of encrypted test result can be identified in rare cases	**Low**	3.7	Android, iOS	Resolved
A07	Submission of fake TEKs limited to 6hr window each day	**Low**	3.7	Android	Resolved

Denial of Service Results

Test	CDN	Backend
Geographic limitation	**Pass**	**Pass**
IP Rate limitation	**Pass**	**Pass**
Load testing	**Pass**	**Pass**

Business logic validation

Abuse case	Scope	Result
As a malicious user, I cannot successfully submit my TEKs if I do not have a positive test result	Backend	**Pass**
As a malicious user, I cannot submit my TEKs several times for a single positive test result	Backend	**Pass**
As a malicious user, I cannot submit my TEKs based on another user's positive test result (i.e. tokens to identify the test are high entropy and are correctly checked)	Backend	**Pass**
As a malicious user, I cannot modify the TEKs which are sent to the server when sharing them after a positive test result	Backend	**Pass**

As a malicious user, I cannot send more than 14 TEKs to the server in one request	Backend	Pass

Privacy requirements validation

The privacy requirements as documented by the Interfederal Committee Testing & Tracing have been translated into the following abuse cases:

Abuse case	Scope	Result
As an attacker, I cannot obtain today's TEK of another user	iOS	Pass
	Android	Pass
As an attacker, I cannot acquire the test result of another user from the backend (i.e. tokens to identify the test are high entropy and are correctly checked)	iOS	Pass
	Android	Pass
As an attacker, I cannot retrieve personal information about any user of the application	iOS	Pass
	Android	Pass
As an attacker with access to encrypted network traffic, I cannot deduce the result of another user's test based on the network traffic (i.e. no data should be deducible from encrypted packets and DNS queries)	iOS	A02*, A07*
	Android	
	Backend	A03*
As an attacker with access to encrypted network traffic, I cannot deduce if another user has been tested and is waiting for the results	iOS	A06*
	Android	A06*
As an attacker with access to encrypted network traffic, I cannot deduce if the result of another user's test has been delivered	iOS	Pass
	Android	Pass
As an attacker with access to another user's device, I cannot deduce if the user has been tested	iOS	A04
	Android	
As an attacker with access to another user's device, I cannot deduce if the user has been tested positive	iOS	A04
	Android	
As an attacker with access to another user's application backup, I cannot deduce the personal information about the user	Android	Pass
	iOS	Pass

* This issue has been resolved during development

Threat model risks

Only the in-scope threats are listed below

Abuse case	Scope	Result
As a malicious user, I should not be able to find a vulnerable version of a third party library used in the mobile application or backend services when reviewing the open source codebase. A vulnerable version is a version that has known, public vulnerabilities that can be used to manipulate the normal flow of the application.	iOS	Pass
	Android	Pass
As a malicious user, I should not be able to upload malicious data during a TEK key upload after a positive test result, that is then used without sanitization by the backend program logic.	Backend	Pass
As a malicious user, It should not be possible to derive the private values (R1,R0,K,t0,t1) from the mobile application datastore through the use of malware installed on the system.	iOS	Pass
	Android	Pass
As a malicious user, I should not be able to go through the open sourced codebase and uncover secrets in the form of e.g. hardcoded connection credentials that are still actively used in the normal operation of the production environment	iOS	Pass
	Android	Pass
	Backend	Pass

Cloud-level Review

The identified findings are mentioned below and are structured according to the AWS security baseline categories.

Identity and Access Management

ID	Finding	Overall Risk	CVSS	Status
IAM01	Root user without MFA enabled	High	8.1	Resolved
IAM02	IAM policies allowing full administrative privileges are created	Medium	5.9	Resolved
IAM03	AWS API calls from service accounts or IAM users are not restricted	**Medium**	5.5	Active

Business Continuity

ID	Finding	Overall Risk	CVSS	Status
BC01	AWS VPN not redundant	**Low**	3.7	Active

Data Protection

ID	Finding	Overall Risk	CVSS	Status
DP01	S3 security controls disabled	**Low**	3.3	Active

Networking

ID	Finding	Overall Risk	CVSS	Status
NET03	Advanced DDoS protection is not enabled	**Low**	3.7	Active

Configuration management

ID	Finding	Overall	CVSS	Status
CM01	Sensitive data exposed to EC2 instance user	Low	2.0	Resolved

Thank you for your confidence in NVISO.

Brussels

Guimardstraat 8 rue Guimard
1040 Brussels
www.nviso.be

Frankfurt

Holzgraben 5
60313 Frankfurt am Main
www.nviso.de

München

Herzogspitalstr. 24
80331 München
www.nviso.de

www.nviso.eu

nviso