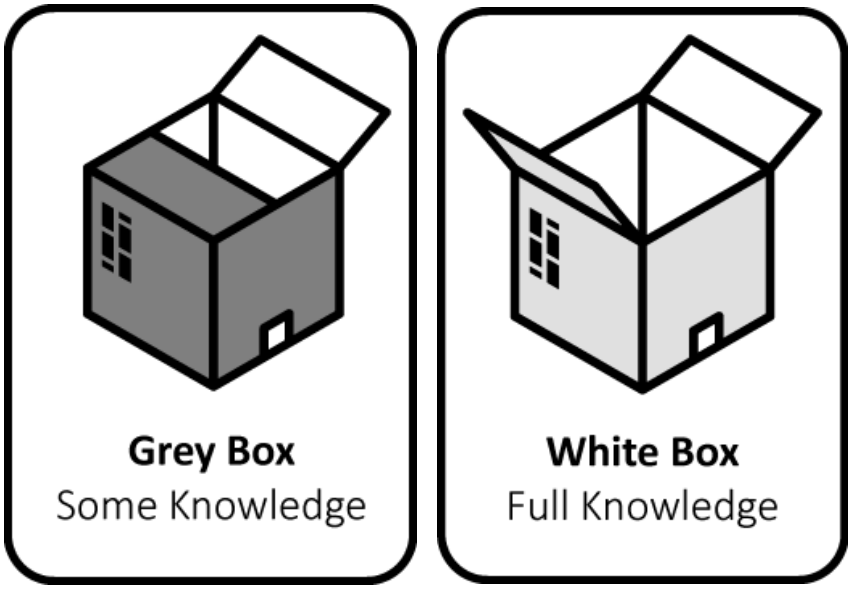
Rapportage Penetratietest



Klantnaam: Ministerie van Volksgezondheid, Welzijn en Sport

Datum: 01/09/2020

Versie: 1.0 - Definitief

Projectnaam: 20063 – Cholet

Disclaimer Penetratietest

NFIR B.V. voert de penetratietest uit volgens de huidige normen en methodologieën. Een beveiligingscontrole is echter een momentopname. NFIR B.V. aanvaardt geen aansprakelijkheid voor kwetsbaarheden die niet (algemeen) bekend waren op het moment van het uitvoeren van de beveiligingsaudit.

Copyright © 2020 NFIR BV

Handelsnamen

NFIR en het NFIR-logo zijn handelsmerken van NFIR B.V. Alle andere handelsmerken in dit document zijn eigendom van de vermelde partijen.

Over NFIR

NFIR is een specialist in cyberbeveiliging. Bij NFIR zijn cyber-engineers in dienst: Hackers die weten wat ze doen, testers die de werking van ICT-infrastructuren en software begrijpen, privédetectives die, indien nodig, onderzoeken kunnen uitvoeren en adviseurs/consultants die u het juiste advies kunnen geven of die u kunnen ondersteunen bij een incident.

POB-vergunning

Het ministerie van Justitie en Veiligheid heeft NFIR een vergunning afgegeven, waardoor NFIR haar werkzaamheden mag uitvoeren. Deze vergunning betreft de POB-vergunning. De POB-vergunning dekt het verwerken van strafrechtelijke gegevens, waarmee NFIR in aanraking kan komen bij het uitvoeren van haar diensten.

Het POB-licentienummer van NFIR is: 1672.

Document beheer

Klant	Ministerie van Volksgezondheid, Welzijn en Sport
Classificatie	Publiek
Projectnaam	20063 – Cholet
Begindatum pentest	10 augustus 2020
Einddatum pentest	28 augustus 2020
Versie	1.0 - Definitief
Auteurs	M.R. van Geelen – OSCP, eWPT D. de Weille – OSCP, eWPT R. Verdoes – OSCP, OSCE, eCPPT, eWPTx, eWPT M.B.J. Koek G.P. de Vries D. Miltenburg

Versie informatie

Versie	Datum	Auteur	Beschrijving
0.1	10/08/2020	M.R. van Geelen	Draftversie
0.2	14/08/2020	M.R. van Geelen D. de Weille R. Verdoes M.B.J. Koek	Informatie toegevoegd
0.3	17/08/2020	M.R. van Geelen D. de Weille R. Verdoes G.P. de Vries	Informatie toegevoegd
0.4	18/08/2020	M.R. van Geelen R. Verdoes M.B.J. Koek	Informatie toegevoegd
0.5	19/08/2020	M.R. van Geelen	Informatie toegevoegd
0.6	20/08/2020	D. Miltenburg	Review uitgevoerd
0.7	21/08/2020	M.R. van Geelen	Review verwerkt
0.8	24/08/2020	M.R. van Geelen	Bevindingen hertest verwerkt
0.8.5	28/08/2020	M.R. van Geelen	Bevindingen hertest verwerkt
0.9	28/08/2020	M.R. van Geelen	Conceptversie
1.0	01/09/2020	M.R. van Geelen D. Miltenburg	Definitieve versie

Distributielijst

Versie	Naam	Organisatie	Verspreidingsmethode
0.9	Brenno de Winter	Ministerie van VWS	Deelportaal
1.0	Brenno de Winter	Ministerie van VWS	Deelportaal

Contact informatie

Naam	NFIR B.V.
Adres	Verlengde Tolweg 2 2517 JV Den Haag Nederland
Telefoonnummer	+31 (0) 88 – 323 02 05
E-mail	info@nfir.nl

Managementsamenvatting

Het Ministerie van Volksgezondheid, Welzijn en Sport (WVS) heeft NFIR verzocht om een penetratietest uit te voeren op de infrastructuur (API), Content-Delivery-Network (CDN) en de mobiele applicaties behorende tot CoronaMelder. De penetratietest vond plaats van 10 augustus 2020 tot en met 28 augustus 2020. Deze gehele periode omvat het testen van de gedefinieerde scopeobjecten, het rapporteren van de bevindingen en het aansluitend uitvoeren van een hertest waarbij gecontroleerd is of de gevonden kwetsbaarheden zijn opgelost. Bij het uitvoeren van de penetratietest is gebruik gemaakt van publieke standaarden- en testmethodieken voor het uitvoeren van de penetratietest:

- OWASP Web Security Testing Guide (WSTG)
- OWASP Mobile Security Testing Guide (MSTG)
- Penetration Testing Execution Standard (PTES)

Voor het berekenen van de scores van de gevonden kwetsbaarheden is de Common Vulnerability Scoring System (CVSS) versie 3.1 toegepast. Bij het berekenen van de score is rekening gehouden met de volgende, door het Ministerie van VWS aangeleverde, CIA-bepaling:

Scope:	Confidentiality	Integrity	Availability
Alle scopeobjecten	Hoog	Hoog	Hoog

Tabel 1 - Scope - CIA scores

Bevindingen penetratietest

In totaal zijn tijdens de uitvoering van de penetratietest 25 kwetsbaarheden vastgesteld. Tabel 2 toont per risicoclassificatie het aantal bevindingen van de uitgevoerde penetratietest. In de laatste kolom is het aantal bevindingen opgenomen dat is opgelost.

Classificatie	Score	Bevindingen API/Meldportaal	Bevindingen App - iOS	Bevindingen App - Android	Opgeloste bevindingen
Kritiek	9.0 – 10.0	0	0	0	0
Hoog	7.0 – 8.9	1	0	0	1
Gemiddeld	4.0 – 6.9	8	2	1	7
Laag	0.1 – 3.9	2	2	2	2
Informatief	0.0	1	2	4	1

Tabel 2 – Bevindingen

NFIR adviseert om de nog niet opgeloste kwetsbaarheden op te lossen en een her-test uit te voeren om te verifiëren of de kwetsbaarheden zijn opgelost.

Inhoudsopgave

Bevindingenlijst

1 Inleiding

Ministerie van Volksgezondheid, Welzijn en Sport heeft NFIR verzocht om een penetratietest uit te voeren en heeft hiervoor de desbetreffende scope aangeleverd. De penetratietest vond plaats van 10 augustus 2020 tot en met 28 augustus 2020. Deze gehele periode omvat zowel het testen als het rapporteren.

1.1 Doel

Het doel van een penetratie test is kwetsbaarheden vinden binnen de afgesproken scope en de daar bijbehorende infrastructuur.

Er zijn drie mogelijke aanvalsperspectieven om technische beveiligingsrisico's of misbruik van een IT-infrastructuur, web/mobiele applicatie, website, en API's in kaart te brengen. Die manieren omvatten een Crystal Box (ook wel White Box genoemd), Grey Box of een Black Box beveiligingsonderzoek. Met een beveiligingsonderzoek op basis van het Crystal Box-principe wordt alle informatie van tevoren gedeeld. Dit in tegenstelling tot een Black box beveiligingsonderzoek, waarbij vooraf geen informatie wordt gedeeld. Een tussenvorm is de Grey Box penetratie test, waarbij de onderzoekers beperkte inloggegevens en -informatie ter beschikking hebben. Deze drie soorten pentesten kunnen plaatsvinden gedurende een vooraf afgesproken aantal uren, ook wel bekend als een Timeboxed penetratie test.

Door Ministerie van VWS is gevraagd om een penetratietest uit te voeren op het API/meld-portal als een Timeboxed Grey Box variant. Daarnaast is door het Ministerie van VWS gevraagd om een penetratietest uit te voeren op de mobiele applicaties voor iOS en Android waarbij een Timeboxed White Box-variant is gehanteerd.

1.2 Scope

De pentest omvat een Timeboxed Grey Box en White Box-onderzoek, waarbij NFIR 140 uur heeft besteed aan het onderzoek en de rapportage. Bij het onderzoek is vooraf informatie verkregen en zijn voor het Grey Box onderdeel gebruikersaccounts ontvangen.

In de onderstaande tabellen wordt de scope voor de pentest weergegeven.

1.2.1 Applicaties

Scope:	Beschrijving:
CoronaMelder voor iOS	Buildnummer 1.0.1 – a6b1876
CoronaMelder voor iOS (no SSL pinning)	Buildnummer 1.0.0 – 644f133
CoronaMelder voor Android	Buildnummer 0.4.1 – 67885-57e9ac2
CoronaMelder voor Android (no SSL pinning)	Buildnummer 0.4.1 – 254abe4

Tabel 3 - Scope – Applicaties

1.2.2 Hostnames, IP-adressen

Scope:	Beschrijving:
coronamelder-portal.nl	Meldportaal CoronaMelder
coronamelder-api.nl	API-endpoint CoronaMelder
coronamelder-dist.nl	Distributie-domein CoronaMelder
productie.coronamelder-dist.nl	Distributie-eindpoint CoronaMelder
82.201.43.165 82.201.43.164 217.194.124.5 217.194.124.3 195.121.65.164 217.194.124.4 82.201.43.163 82.201.43.172	Scope – verstrekte IP-adressen

Tabel 4 - Scope – Hostnames, IP-adressen

1.2.3 Gebruikersaccounts

Scope:	Beschrijving:
Gebruikersnaam: m.vangeelen@nfir.nl	Account 1 Meld Portaal
Gebruikersnaam: g.devries@nfir.nl	Account 2 Meld Portaal
Gebruikersnaam: COVP\geelenm	Account 1 Beheer omgeving
Gebruikersnaam: COVP\vriesg	Account 2 Beheer omgeving

Tabel 5 - Scope – Gebruikersaccounts

1.2.4 CIA-scores

De onderstaande CVSS CIA scores zijn gebruikt om de CVSS-score te berekenen voor de gevonden kwetsbaarheden. Deze CIA-scores zijn door opdrachtgever verstrekt.

Scope:	Confidentiality	Integrity	Availability
Alle scopeobjecten	Hoog	Hoog	Hoog

Tabel 6 - Scope - CIA scores

Aanvallen op de beschikbaarheid van gegevens, zogenaamde Distributed Denial of Service (DDoS) -aanvallen, worden niet uitgevoerd. Als tijdens het veiligheidsonderzoek kwetsbaarheden worden aangetroffen die ertoe leiden dat systemen niet meer beschikbaar zijn, wordt hier direct melding van gemaakt. Daarnaast zal dit worden meegenomen in het rapport.

2 Methodiek

2.1 Procedure

Om een succesvolle beveiligingsaudit uit te voeren, gebruikt NFIR verschillende methoden voor het testen van informatiebeveiliging. De twee belangrijkste standaarden zijn de Penetration Execution Standard (PTES) en het Open Web Application Security Project (OWASP). Door gebruik te maken van deze normen, zorgt NFIR voor een succesvol en grondig uitgevoerd veiligheidsonderzoek.

Er zijn zeven fasen tijdens een penetratietest. Deze zeven fasen zijn:

- Fase 1: Intelligence Gathering
- Fase 2: Threat Modeling
- Fase 3: Vulnerability Analysis
- Fase 4: Exploitation
- Fase 5: Post-Exploitation
- Fase 6: Reporting
- Fase 7: Re-audit

2.1.1 Fase 1: Intelligence Gathering

Deze fase bestaat uit het verzamelen van zoveel mogelijk informatie uit beschikbare bronnen. Dit kunnen openbare bronnen (OSINT) zijn, zoals de WHOIS-database, de gebruikte DNS-servers, (sub)-domeinnamen, e-mailadressen en databases met gelekte wachtwoorden. Tevens kan informatie worden aangeleverd door de opdrachtgever, zoals netwerktekeningen en een IP-nummerplan. Deze beschikbare bronnen hoeven niet noodzakelijkerwijs deel uit te maken van de van tevoren geïdentificeerde scope.

2.1.2 Fase 2: Threat Modelling

Gedurende deze fase wordt de informatie gewaardeerd en wordt daarmee vastgesteld welke informatie relevant is voor de penetratietest. U kunt hierbij denken aan het identificeren van waardevolle informatie, uitdenken van een aanvalsmethodiek en onderzoeken van de bedreigingen.

2.1.3 Fase 3: Vulnerability Analysis

Nadat alle informatie is verzameld, wordt in deze fase gezocht naar kwetsbaarheden in systemen en applicaties. Hierbij wordt gebruik gemaakt van tooling die automatisch zoekt naar bekende kwetsbaarheden. Daarnaast wordt door een ethische hacker op een creatieve wijze handmatig gezocht en gekeken naar kwetsbaarheden. Tijdens deze fase wordt gebruik gemaakt van diverse internationale standaarden zoals OWASP Top 10, PTES en OWASP MSTG.

2.1.4 Fase 4: Exploitation

Tijdens de exploitatie fase is toegang verkrijgen tot het systeem het doel. De reeds verzamelde informatie wordt gebruikt om op een zorgvuldige wijze aanvallen uit te voeren. Deze aanvallen hebben als doel de geïdentificeerde kwetsbaarheden uit de vorige fase te bevestigen.

2.1.5 Fase 5: Post-Exploitation

In de post-exploitatie fase wordt vastgesteld wat de waarde is van het gecompromitteerde systeem. Dit is afhankelijk van de gevonden data en of deze bruikbaar is om het netwerk verder te compromitteren.

2.1.6 Fase 6: Reporting

Alle bevindingen zullen worden samengebracht in een compleet en helder uitgewerkt rapport. Dit rapport bevat een beschrijving van de bevindingen, een scoresysteem (CVSS) waarbij de kwetsbaarheden een classificatie krijgen, de mogelijke impact van de kwetsbaarheden en aanbevelingen die uw organisatie helpen met het oplossen van de gevonden kwetsbaarheden.

2.1.7 Fase 7: Re-audit

Op basis van de aanbevelingen kunnen de gevonden kwetsbaarheden door uw eigen organisatie (of externe partij) worden opgelost. Zodra de kwetsbaarheden zijn opgelost, wordt NFIR veelal gevraagd dit te controleren met een re-audit (hertest). Er wordt dan onderzocht en gerapporteerd of de kwetsbaarheden daadwerkelijk zijn opgelost. Op deze manier bent u verzekerd van een onpartijdig en scherp oordeel over de aangebrachte verbeteringen. U kunt de hertest rapportage bijvoorbeeld gebruiken om externe partijen (afnemers, partners, auditors, etc.) te overtuigen van de technische weerbaarheid van uw systemen en applicaties. Een hertest kan alleen worden begroot na voltooiing van de initiële penetratietest.

2.2 CVSS v.3.1

Het Common Vulnerability Scoring System versie 3.1 [https://www.first.org/cvss/calculator/3.1], afgekort tot het CVSS-risicomodel, wordt gebruikt om de ernst van een kwetsbaarheid te bepalen. Dit model wordt gebruikt door NFIR om beveiligingslekken te classificeren.

Het scoresysteem werkt op basis van acht verschillende basisparameters, die samen de risicoscore bepalen. Die basisparameters zijn:

1. Het aanvalsoppervlak waarop de aanval wordt uitgevoerd.
2. De complexiteit van de uitvoering van de aanval, voor een aanvaller.
3. De privileges (bijvoorbeeld accounts) die nodig zijn om de aanval uit te voeren.
4. De vorm van interactie met het slachtoffer die al dan niet nodig is om de aanval uit te voeren.
5. Het beheer van de 'scope' door een, al dan niet, externe partij die beslissingen neemt.
6. De gevolgen voor de vertrouwelijkheid van het aangevallen systeem.
7. De gevolgen voor de integriteit van het aangevallen systeem.
8. De impact op de beschikbaarheid van het aangevallen systeem.

Deze parameters vormen samen een score van nul (0) tot tien (10), waarbij tien de hoogste vorm van kwetsbaarheid vertegenwoordigt. Deze scores geven een classificatie op basis van vijf categorieën: informatief, laag, gemiddeld, hoog en kritiek, zoals te zien is in de onderstaande tabel.

Classificatie	Score
Kritiek	9.0 – 10.0
Hoog	7.0 – 8.9
Gemiddeld	4.0 – 6.9
Laag	0.1 – 3.9
Informatief	0.0

Tabel 7 - Classificaties

Binnen de CVSS-methode worden zogenaamde vector strings toegepast, die per kwetsbaarheid worden toegekend op basis van de bovengenoemde parameters. De vector string begint met het label "CVSS:" en een numerieke weergave van de huidige versie, "3.1". Metrische informatie volgt in de vorm van een reeks metrieken, waarbij elke maat wordt voorafgegaan door een schuine streep, "/", die als een scheidingsteken fungeert. De metrieken en overeenkomstige waarde wordt weergegeven in de verkorte vorm, met daartussen een dubbele punt. Deze vector strings kunnen gebruikt worden om te herleiden waarop de scores gebaseerd zijn. Dit herleiden kan eenvoudig gereproduceerd worden door op de vectorstring te klikken of door de vectorstring uit dit rapport in te voeren achter de volgende URL: https://www.first.org/cvss/calculator/3.1#

3 Verloop penetratietest

Onderstaande tabel geeft een overzicht van het verloop van de uitvoering en de rapportage van de penetratietest.

Datum:	Tijd	Uitleg:
(za) 08/08/2020	15:00	Start voorbereidende werkzaamheden
(za) 08/08/2020	18:00	Smartphone (Android) ingericht
(ma) 10/08/2020	09:00	Penetratietest gestart
(ma) 10/08/2020	09:15	Port scans gestart (top 10000 ports)
(ma) 10/08/2020	09:30	SSL-scans gestart
(ma) 10/08/2020	12:20	Kwetsbaarheden uitgeschreven
(ma) 10/08/2020	15:00	2e portscan gestart (1-65535)
(ma) 10/08/2020	09:00	Gestart met testen van API
(ma) 10/08/2020	09:45	Gestart met testen van Meldportaal
(ma) 10/08/2020	10:00	Gestart met testen van CDN
(ma) 10/08/2020	10:00 – 11:00	Problemen met het Meldportaal, deze werkt niet.
(di) 11/08/2020	09:00	Analyse van API-calls
(wo) 12/08/2020	08:00 – 17:00	Inloggegevens verkregen voor meldportaal, meldportaal functioneel
(wo) 12/08/2020	10:15 – 10:30	Android TEK-upload getest, sleutels succesvol geüpload
(wo) 12/08/2020	10:15 – 10:30	iOS TEK-upload getest, sleutels succesvol geüpload
(wo) 12/08/2020	14:00 – 15:00	Analyse van de APK-bestanden
(wo) 12/08/2020	14:00 – 17:00	Android App penetratietest, gebruik gemaakt van MSTG-checklist.
(wo) 12/08/2020	14:00 – 17:00	iOS App penetratietest, gebruik gemaakt van MSTG-checklist.
(do) 13/08/2020	09:00 – 17:00	Vervolg Android/iOS App penetratietest, gebruik gemaakt van MSTG-checklist.
(do) 13/08/2020	09:00 – 17:00	Vervolg API/Meldportaal penetratietest, gebruik gemaakt van WSTG-checklist.

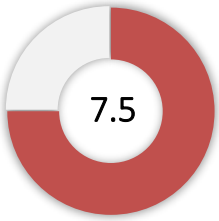
(do) 13/08/2020	12:00 – 15:00	Vaststellen ExposureNotification-werking in relatie tot apps
(vr) 14/08/2020	09:00 – 17:00	Vervolg Android/iOS App penetratietest, gebruik gemaakt van MSTG-checklist
(vr) 14/08/2020	09:00 – 17:00	Vervolg API/Meldportaal penetratietest, gebruik gemaakt van WSTG-checklist.
(ma) 17/08/2020	09:00 – 17:00	Vervolg API/Meldportaal penetratietest, gebruik gemaakt van WSTG-checklist.
(ma) 17/08/2020	09:00 – 17:00	Uitwerken bevindingen in rapportage.
(di) 18/08/2020	09:00 – 17:00	Uitwerken bevindingen in rapportage.
(wo) 19/08/2020	09:00 – 17:00	Uitwerken bevindingen in rapportage
(do) 20/08/2020	16:00 – 17:00	Review
(vr) 21/08/2020	10:00 – 12:00	Review afgerond
(vr) 21/08/2020	12:00 – 14:00	Review verwerkt
(ma) 24/08/2020	14:00 – 22:00	Hertest bevindingen
(vr) 28/08/2020	09:00 – 13:00	Hertest bevindingen
(ma) 31/08/2020	12:00 – 14:30	Reactie opdrachtgever verwerkt

Tabel 8 - Verloop penetratietest

4 Bevindingen penetratietest – Grey Box

4.1 Bevindingen – API/Meldportaal

4.1.1 Hoog

1. HTTP Security Headers – Meerdere headers	
Hosts: coronamelder-portal.nl coronamelder-api.nl productie.coronamelder-dist.nl	 Opgelost **CVSS Score:** Hoog
CVSS Vector String: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H	

Kwetsbaarheid:
Voor de hosts zijn geen of niet alle HTTP Security headers ingesteld. Door het instellen van deze headers kunnen webbrowsers diverse kwetsbaarheden voorkomen. De volgende headers zijn gecontroleerd:

- **HTTP Strict Transport Security (HSTS):** Header die het gebruik van HTTPS forceert.
- **X-Frame-Options:** Geeft een melding aan de browser of de site in een frame geladen mag worden of niet. Met deze header kunnen clickjacking aanvallen voorkomen worden, aangezien de site niet in een frame geladen kan worden. Als een aanvaller de site wel in een frame zou kunnen laden, kan de aanvaller de gebruiker onbedoeld omleiden naar een malafide website.
- **X-Content-Type-Options:** Voorkomt dat een MIME-sniffing-aanval kan worden uitgevoerd, waarmee een aanvaller een bestand kan verbergen als een ander bestandstype. Hierdoor kan een malafide bestand bijvoorbeeld worden geüpload met een jpg-extensie. Doordat het bestand de kenmerken heeft van een legitiem, uitvoerbaar bestand, wordt de applicatie in sommige gevallen opgeslagen en uitgevoerd.
- **Content-Security-Policy:** Header die een site beschermt tegen Cross-site scripting (XSS) aanvallen. Door het definiëren van een lijst met toegestane content, kan je voorkomen dat de browser kwaadwillende code kan laden van een externe bron. Met de kwaadwillende code kunnen bijvoorbeeld cookies van gebruikers worden afgevangen en gebruikt om mee in te loggen, of om malware te injecteren in de website.
- **Referrer-Policy:** Header die controleert welke informatie door een site opgehaald mag worden van een andere bron.
- **Feature-Policy:** Header waarmee controle is over welke functies en API's gebruikt kunnen worden in de browser.

Er is vastgesteld dat bij bovengenoemde hosts een of meerdere van deze zes security headers niet zijn ingesteld.

Bevestiging:
Voor alle hosts binnen de scope zijn de security headers gecontroleerd door middel van een script, zie onderstaand schermafdruk:

```
kali@kali:~/securityheaders$ python3 securityheaders.py --max-redirects 5 https://coronamelder-portal.nl
Header 'x-frame-options' is missing ... [ WARN ]
Header 'strict-transport-security' contains value 'max-age=31536000; includeSubDomains; preload' ... [ OK ]
Header 'access-control-allow-origin' is missing ... [ OK ]
Header 'content-security-policy' is missing ... [ WARN ]
Header 'x-xss-protection' is missing ... [ WARN ]
Header 'x-content-type-options' is missing ... [ WARN ]
Header 'x-powered-by' contains value 'ASP.NET' ... [ WARN ]
Header 'server' contains value 'Microsoft-IIS/10.0' ... [ WARN ]
HTTPS supported ... [ OK ]
HTTPS valid certificate ... [ OK ]
```

Figuur 1 - Security Headers gecontroleerd

De resultaten zijn opgenomen in de bijlage: 6.5.4 HTTP Security Headers

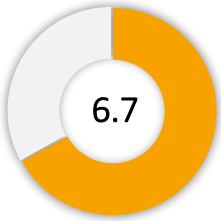
Mogelijke Impact:
De kwetsbaarheden die ontstaan door het ontbreken van de juiste headers, kunnen ertoe leiden dat een aanvaller handelingen uit kan voeren die schadelijk kunnen zijn voor de host of de gebruiker.

Aanbeveling:
Geadviseerd wordt om alle security headers welke van toepassing zijn op de specifieke host(s) toe te passen.

4.1.2 Gemiddeld

2. TLS v1.0 en v1.1 protocol ondersteuning (WSTG-CRYP-01)

Host:
https://productie.coronamelder-dist.nl



CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:C/C:L/I:L/A:N/CR:H/IR:H/AR:H

CVSS Score: Gemiddeld

Kwetsbaarheid:
De host ondersteunt de onveilige TLS-versies 1.0 of 1.1. Deze protocollen zijn sinds maart 2020 end-of-life en worden daarom niet meer ondersteund door de meest gebruikte browsers.

Het CDN-endpoint (https://productie.coronamelder-dist.nl/) wordt volgens de publieke documentatie[1] door de CoronaMelder-applicatie voor iOS en Android gebruikt om publiek beschikbare informatie op te halen, waaronder bijvoorbeeld Diagnosis Keys (DKs), die apart voorzien zijn van een digitale handtekening.

Bevestiging:
Een SSLscan is uitgevoerd op de host 'productie.coronamelder-dist.nl', wat leidde tot de volgende output:

```
  Supported Server Cipher(s):
Preferred TLSv1.2  128 bits  ECDHE-RSA-AES128-GCM-SHA256   Curve P-256 DHE 256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-GCM-SHA384   Curve P-256 DHE 256
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-SHA256       Curve P-256 DHE 256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-SHA384       Curve P-256 DHE 256
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE 256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE 256
Accepted  TLSv1.2  128 bits  AES128-GCM-SHA256
Accepted  TLSv1.2  256 bits  AES256-GCM-SHA384
Accepted  TLSv1.2  128 bits  AES128-SHA256
Accepted  TLSv1.2  256 bits  AES256-SHA256
Accepted  TLSv1.2  128 bits  AES128-SHA
Accepted  TLSv1.2  256 bits  AES256-SHA
Preferred TLSv1.1  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE 256
Accepted  TLSv1.1  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE 256
Accepted  TLSv1.1  128 bits  AES128-SHA
Accepted  TLSv1.1  256 bits  AES256-SHA
Preferred TLSv1.0  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE 256
Accepted  TLSv1.0  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE 256
Accepted  TLSv1.0  128 bits  AES128-SHA
Accepted  TLSv1.0  256 bits  AES256-SHA
```

Figuur 2 - TLS 1.0 & 1.1 - productie.coronamelder-dist.nl

Uit bovenstaande scanresultaten blijkt dat de TLS-versies 1.0 en 1.1 ondersteund worden.

Mogelijke Impact:
Een aanvaller kan een Man-in-the-Middle (MiTM) aanval uitvoeren, in combinatie met bekende kwetsbaarheden in TLS v1.0, om zo het dataverkeer mogelijk te kunnen ontdoen van versleuteling.

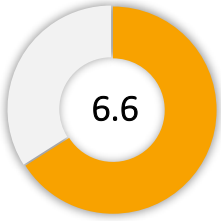
Aanbeveling:
Geadviseerd wordt om de ondersteuning versie 1.0 en 1.1 van TLS uit te schakelen.

[1] https://github.com/minvws/nl-covid19-notification-app-coordination/blob/master/architecture/Solution%20Architecture.md#phase-3-publishing-the-keys

Reactie opdrachtgever (31 augustus 2020):
Deze bevinding is een verwachtte bevinding bij het ontwerp. Daarom hebben we een tweede mechanisme ingericht dat een extra laag met ondertekening levert, waardoor de signaleerde risico's in zijn geheel zijn ondervangen. Sterker nog: door de inzet van HSM's wordt een hoger niveau van integriteit bereikt dan gebeurd zou zijn vertrouwend op de CDN-leverancier (niet afhankelijk van een gekozen leverancier).

3. JWT Token blijft tot 3 uur geldig na uitloggen (WSTG-SESS-06)

Host:
coronamelder-portal.nl



CVSS Vector String:
CVSS:3.1/AV:A/AC:H/PR:N/UI:R/S:U/C:N/I:H/A:N/CR:H/IR:H/AR:H

Opgelost

CVSS Score: Gemiddeld

Kwetsbaarheid:
De JWT token van een gebruiker blijft geldig voor een periode van 3 uur, vanaf het moment dat er uitgelogd is van de applicatie via de uitlogfunctionaliteit. Hierdoor kan er gedurende deze 3 uur opnieuw van deze JWT-token gebruik worden gemaakt om toegang te verkrijgen tot de applicatie.

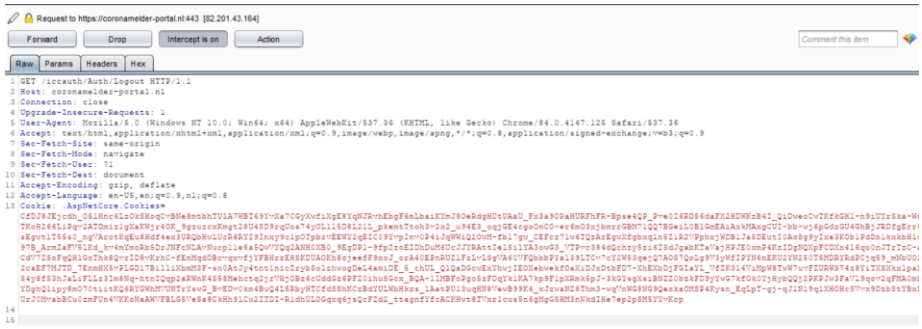
Bevestiging:
Er is een actieve sessie met het gebruikersaccount mvangeelen2@ggdghor.nl:

Figuur 3 – coronamelder-portal.nl - Actieve Sessie

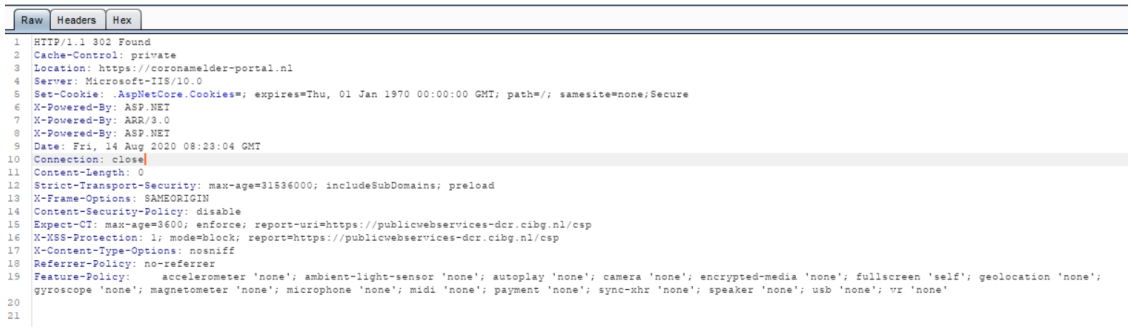
Tijdens het inlogproces is via het volgende verzoek de JWT-token opgevangen:

```
GET
/auth?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJleHAiOjE1OTc0MDMyNzMsImlkIjoiNTk1NzQzODgzNjY3MzAwODM1IiwiYWNjZXNzX3Rva2VuIjoiX2g1dnZtZ1l3TVNPY1d6c2hzSW95USIsIm5hbWUiOiJNaXNjaGEgdmFuIEdlZWxlbiJ9.8ktRuwG7vqvT5cCHI0MKXl8YzdAU-J2UOfe_GsI6hBY HTTP/1.1
Host: coronamelder-portal.nl
[...]
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,nl;q=0.8
```

Vervolgens is op ‘logout’ geklikt en zijn het volgende verzoek en antwoord te zien in de applicatie genaamd Burp Suite, dat wordt gebruikt voor het testen en analyseren van de veiligheid van webapplicaties:



Figuur 4 - Logout Request in Burp



Figuur 5 - Logout Response in Burp

In de browser wordt het inlogscherm opnieuw zichtbaar. Vervolgens wordt het eerder opgevangen verzoek (/auth) opnieuw bezocht:

Figuur 6 – Auth-url opnieuw bezocht

Na het klikken op ‘Inloggen’ is weer op de applicatie ingelogd, zonder dat hiervoor een gebruikersnaam, wachtwoord en 2fa token benodigd is.

Figuur 7 - Sessie is weer ingelogd na importeren van de JWT-token

Mogelijke Impact:

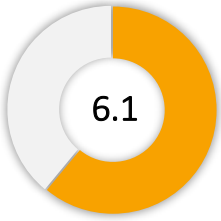
Het is mogelijk dat een aanvaller die de JWT-token van de gebruiker heeft verkregen, deze JWT-token opnieuw gebruikt om zo toegang te verkrijgen tot de applicatie.

Aanbeveling:

Geadviseerd wordt om ervoor te zorgen dat bij elke login nieuwe JWT-tokens worden gebruikt. Zodra er uitgelogd wordt binnen de applicatie, moet de bijbehorende JWT-token geïnvalideerd worden.

4. Denial of Service - Secure Client-Initiated Renegotiation

Hosts:
coronamelder-api.nl
coronamelder-portal.nl



Opgelost

CVSS Score: Gemiddeld

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L/CR:H/IR:H/AR:H

Kwetsbaarheid:
Secure Client-Initiated Renegotiation maakt het mogelijk om op een veilige manier te onderhandelen tussen de client en de host tijdens SSL-connecties. Een aanvaller kan honderden van deze onderhandelingen starten, en zo de host onbereikbaar maken door middel van een Denial of Service (DoS).

Bevestiging:
Door middel van de tool testssl op coronamelder-api.nl, wordt geconstateerd dat 'Secure Client-Initiated Renegotiation' kwetsbaar is voor Denial of Service:

```
Secure Client-Initiated Renegotiation     VULNERABLE (NOT ok), DoS threat
```

Deze kwetsbaarheid geldt ook voor coronamelder-portal.nl.

Mogelijke Impact:
Een aanvaller kan deze kwetsbaarheid gebruiken om het endpoint mogelijk onbereikbaar te maken. Hierdoor is het voor gebruikers niet meer mogelijk om de host te bereiken.

Aanbeveling:
Geadviseerd wordt om de webserver te updaten naar de laatste ondersteunde versie, of indien mogelijk, Secure Client-Initiated Renegotiation uit te schakelen.

5. Kwetsbare ciphers waargenomen

Host:
productie.coronamelder-dist.nl

CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

CVSS Score: Gemiddeld

Kwetsbaarheid:
De host stelt kwetsbare ciphers beschikbaar, welke mogelijk kwetsbaar zijn voor een kwetsbaarheid genaamd BEAST – deze kwetsbaarheid draagt het CVE-nummer CVE-2011-3389.

Bevestiging:
Door middel van de tool testssl op productie.coronamelder-dist.nl, wordt geconstateerd dat de server mogelijk kwetsbaar is voor een BEAST-aanval, omdat verouderde ciphers worden toegestaan:

```
POODLE, SSL (CVE-2014-3566)                not vulnerable (OK), no SSLv3
support
TLS_FALLBACK_SCSV (RFC 7507)               Downgrade attack prevention
supported (OK)
SWEET32 (CVE-2016-2183, CVE-2016-6329)     not vulnerable (OK)
FREAK (CVE-2015-0204)                      not vulnerable (OK)
DROWN (CVE-2016-0800, CVE-2016-0703)       not vulnerable on this host and
port (OK)
                                            make sure you don't use this
certificate elsewhere with SSLv2 enabled services

https://censys.io/ipv4?q=651F22FBEA81B72F3B2168CA7E5227602215233A7FDCE195FC
273BDBD5DB81A3 could help you to find out
LOGJAM (CVE-2015-4000), experimental       not vulnerable (OK): no DH EXPORT
ciphers, no DH key detected with <= TLS 1.2
BEAST (CVE-2011-3389)                      TLS1: ECDHE-RSA-AES128-SHA ECDHE-
RSA-AES256-SHA AES128-SHA AES256-SHA
VULNERABLE -- but also supports higher protocols TLSv1.1 TLSv1.2 (likely
mitigated)
```

Uit het bovenstaande blijkt dat de verouderde ciphers ECDHE-RSA-AES128-SHA, ECDHE-RSA-AES256-SHA AES128, en SHA AES256-SHA worden toegestaan.

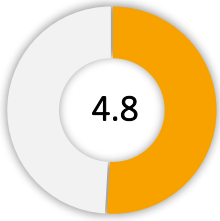
Mogelijke Impact:
Een aanvaller die in staat is om de verbinding tussen de gebruiker en de server te onderscheppen, kan mogelijk de verbinding ontdoen van versleuteling en de inhoud van de verstuurde- en ontvangen berichten inzien.

Aanbeveling:
Om de kwetsbaarheid te mitigeren, wordt geadviseerd om de kwetsbare ciphers (ECDHE-RSA-AES128-SHA, ECDHE-RSA-AES256-SHA AES128, en SHA AES256-SHA) uit te schakelen.

Reactie opdrachtgever (31 augustus 2020):
Dit is al in de ontwerpfase als potentiële dreiging onderkend. Door aanvullende ondertekening te bieden, wordt dit probleem ondervangen. Het inzetten van HSM's biedt hier de oplossing om dit probleem volledig te mitigeren en een hoger niveau van beveiliging te bieden dan een dienstverlener had geboden.

6. Niet-versleutelde HTTP-verbinding (WSTG-CRYP-03)

Host:
http://coronamelder-api.nl



Opgelost

CVSS Score: Gemiddeld

CVSS Vector String:
CVSS:3.1/AV:A/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

Kwetsbaarheid:
De webserver ondersteunt HTTP. Hierdoor wordt data over een niet-versleutelde verbinding naar de webserver gestuurd.

Bevestiging:
Door te navigeren naar http://coronamelder-api.nl/ stuurt de webserver een valide response. Om erachter te komen of ook gevoelige data uitgewisseld wordt, kan er een POST-request worden gemaakt naar het endpoint /v1/register:

```
POST /v1/register HTTP/1.1
Host: coronamelder-api.nl
[...]
Connection: close
Content-Length: 3

a=1
```

De webserver stuurt vervolgens een response met onder andere een geldig labConfirmationId:

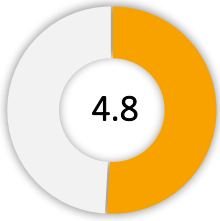
```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Server: Microsoft-IIS/10.0
[...]
X-Powered-By: ASP.NET
Date: Mon, 10 Aug 2020 10:12:09 GMT
Connection: close
Content-Length: 171

{"labConfirmationId":"35U-
33B","bucketId":"h6kHGR1yR+WBs6hrvrm1SaNLqvWyaqkvhN1RcRIxRSk=","confirmatio
nKey":"vsVisJ54GmvOSTWgHtffM7/nXuxpGxw8mE8df9jZWvo=","validity":58640}
```

Mogelijke Impact:
Indien geforceerd kan worden dat de app verbinding maakt over HTTP in plaats van HTTPS, stelt dit een derde in staat om het netwerkverkeer, waarover mogelijk gevoelige data wordt uitgewisseld, af te luisteren

Aanbeveling:
Geadviseerd wordt om HTTP uit te schakelen of al het verkeer door te sturen naar de HTTPS equivalent, in acht nemend dat de Strict-Transport-Security header correct geconfigureerd is.

7. Testomgeving vindbaar via Google (WSTG-INFO-01)

Host:
https://test.coronamelder-portal.nl



Opgelost

CVSS Score: Gemiddeld

CVSS Vector String:
CVSS:3.1/AV:A/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

Kwetsbaarheid:
De testomgeving van het meldportaal is beschikbaar via internet. Daarnaast is deze testomgeving geïndexeerd door Google.

Bevestiging:
Door op Google te zoeken naar: "coronamelder-portal.nl" wordt één resultaat gevonden. Het geïndexeerde resultaat betreft het subdomein "test.coronamelder-portal.nl", de testomgeving van het portaal.

Mogelijke Impact:
De testomgeving van het meldportaal is publiekelijk vind- en benaderbaar. Mogelijk zijn er voor een aanvallers kwetsbaarheden te vinden in de testomgeving die later in de productieomgeving misbruikt kunnen worden.

Aanbeveling:
Geadviseerd wordt om een bestand genaamd robots.txt aan te maken in de hoofdmap van de website, waarin aangegeven wordt welke pagina's geïndexeerd mogen worden door zoekmachines. Additioneel wordt geadviseerd om de test-omgeving alleen vanaf vertrouwde omgevingen beschikbaar te maken.

8. Informatie in foutmeldingen en headers (WSTG-INFO-02)

Hosts:
https://coronamelder-api.nl
https://coronamelder-portal.nl
https://productie.coronamelder-dist.nl

CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N/CR:H/IR:H/AR:H

Opgelost

CVSS Score: Gemiddeld

Kwetsbaarheid:
De webserver geeft veel informatie vrij bij foutmeldingen, waardoor het aanvalsoppervlak voor een potentiële aanvaller wordt vergroot.

Bevestiging:
Door naar een willekeurige pagina te navigeren stuurt de webserver de volgende headers mee in de response:

```
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
X-Powered-By: ARR/3.0
```

Daarnaast wordt er een enigszins uitgebreide foutmelding getoond bij het volgende POST-verzoek:

```
POST /v1/postkeys?sig=<script>alert()</script> HTTP/1.1
Host: coronamelder-api.nl
[...]
Connection: close
Content-Length: 3

a=1
```

De foutmelding beschrijft wat er is misgegaan:

```
HTTP/1.1 400 Bad Request
Content-Type: application/problem+json; charset=utf-8
[...]
Connection: close
Content-Length: 235

{"type":"https://tools.ietf.org/html/rfc7231#section-6.5.1","title":"One or
more validation errors occurred.","status":400,"traceId":"|6893094d-
4a5ad87d2b40cb1f.","errors":{"sig":["The value '<script>alert()</script>'
is not valid."]}}
```

Zo blijkt uit bovenstaande tabel dat de reactie van de server informatie bevat over hoe geprobeerd is om de verzonden handtekening te verwerken.

Mogelijke Impact:
Het kan voor een aanvaller nuttig zijn om precieze versienummers en de interne structuur van de gebruikte software te weten, om zo zijn aanval gerichter en mogelijk succesvoller te kunnen maken.

Aanbeveling:
Deze informatie kan nuttig zijn in ontwikkel- en testomgevingen, echter wordt er voor de live omgeving geadviseerd om een generieke foutmelding in te stellen in het geval dat er iets is misgaat.

In IIS kan dit gedaan worden door een <httpErrors>-element te definiëren in het bestand ApplicationHost.config (of per applicatie in het desbetreffende web.config-bestand). Meer informatie is te vinden op https://docs.microsoft.com/en-us/iis/configuration/system.webserver/httperrors/

Schakel ook de headers met versie-informatie uit. De volgende configuratie-opties kunnen daarvoor worden gebruikt in IIS:

```
<configuration>
 <system.web>
  <httpRuntime enableVersionHeader="false" />
 </system.web>
 <system.webServer>
  <httpProtocol>
   <customHeaders>
    <remove name="X-Powered-By" />
   </customHeaders>
  </httpProtocol>
  <rewrite>
   <outboundRules>
    <rule name="Remove IIS version" patternSyntax="Wildcard">
     <match serverVariable="RESPONSE_SERVER" pattern="*" />
     <action type="Rewrite" value="Microsoft-IIS" />
    </rule>
   </outboundRules>
  </rewrite>
 </system.webServer>
</configuration>
```

Daarmee wordt de X-Powered-By header verwijderd en wordt het versienummer verwijderd uit de Server-header.

9. Sessie is niet gebonden aan IP-adres

Host:

https://coronamelder-portal.nl

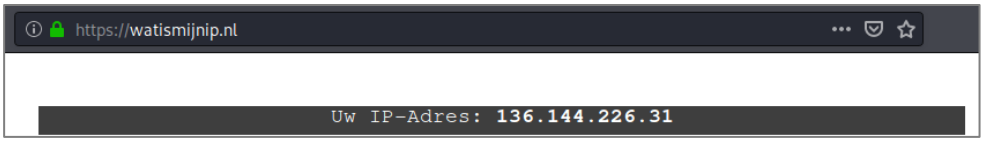
CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Gemiddeld

Kwetsbaarheid:
De webapplicatie kijkt niet of voor een actieve sessie het publieke IP-adres gewijzigd is. De sessie blijft actief en er wordt niet gevraagd om extra validatie.

Bevestiging:
Er is een sessie geopend richting het web portaal. Vervolgens wordt door middel van een VPN ons publieke IP-adres gewijzigd:

Figuur 8 - IP adres vooraf



Figuur 9 - IP adreswijziging

De sessie blijft actief, zie onderstaande schermafdruk:

Figuur 10 - Sessie actief na wijzigingen publieke IP-adres

Mogelijke Impact:
Het is mogelijk dat een aanvaller die de cookie van de gebruiker heeft verkregen, deze cookie opnieuw gebruikt om toegang te verkrijgen tot de applicatie. De applicatie controleert daarbij namelijk niet of het publieke IP-adres van de gebruiker is gewijzigd.

Aanbeveling:
Geadviseerd wordt om de gebruiker opnieuw te laten inloggen nadat het publieke IP-adres is gewijzigd.

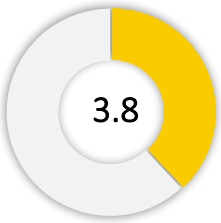
Reactie opdrachtgever (31 augustus 2020):
Dit probleem is onderkend, maar complex om op te lossen. Omdat het daadwerkelijk uitvoeren van een succesvolle aanvallen complexer is en daarna vervolgens maar beperkte impact zal zijn in de executie. Zelfs als het lukt om in te loggen is de impact beperkt. Zelfs na het doorzetten van sleutels is het veel werk om daadwerkelijk een melding te activeren.

Het probleem wordt echter wel opgelost en is daarom aan de back log toegevoegd.

4.1.3 Laag

10. Angular Development modus ingeschakeld

Host:
coronamelder-portal.nl



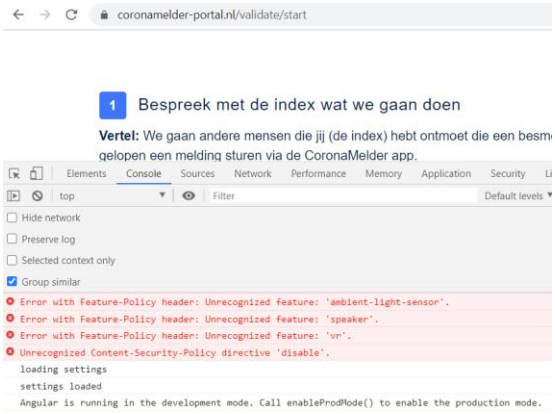
CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:N/A:N/CR:H/IR:H/AR:H

Opgelost

CVSS Score: Laag

Kwetsbaarheid:
De webapplicatie die gehost wordt op coronamelder-portal.nl is geconfigureerd om in de 'development modus' te worden uitgevoerd.

Bevestiging:
Er is genavigeerd naar de webpagina https://coronamelder-portal.nl waarna in de Console-functionaliteit van de browser is waargenomen dat de webapplicatie de volgende waardes rapporteert:



Figuur 11 - Angular is running in the development mode-notificatie

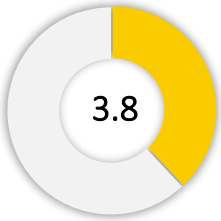
Op basis van bovenstaande waargenomen meldingen in de browser console, is vastgesteld dat de Angular applicatie rapporteert dat deze zich in de development-mode bevindt.

Mogelijke Impact:
Een aanvaller die toegang heeft tot de applicatie kan mogelijk informatie verkrijgen vanuit de development modus van Angular.

Aanbeveling:
Geadviseerd wordt om de developer modus van AngularJS uit te schakelen op de productieomgeving.

11. Access-Control-Allow-Origin configuratie (WSTG-CLNT-07)

Hosts:
https://productie.coronamelder-dist.nl/*
https://coronamelder-portal.nl/iccauth
- /CaregiversPortalApi/v1/labverify
- /CaregiversPortalApi//v1/labconfirm



CVSS Vector String:
CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N/CR:H/IR:H/AR:H

Opgelost

CVSS Score: Laag

Kwetsbaarheid:
Cross Origin Resource Sharing (CORS) is een HTML5-technologie die moderne webbrowsers de mogelijkheid geeft om beperkingen te versoepelen die standaard zijn geïmplementeerd door het Same Origin-beleid.

De 'Access-Control-Allow-Origin'-header is onveilig geconfigureerd wanneer deze is ingesteld op '*' of null, omdat de header er dan voor zorgt dat elk domein het toestaat om cross-domein verzoeken uit te voeren en de reacties uit te lezen.

Bevestiging:
De website https://productie.coronamelder-dist.nl/ is bezocht met de Google Chrome browser, waarna de volgende instelling van de header is waargenomen in de reactie van de server (HTTP 200):

```
HTTP/1.1 200 OK
Content-Type: text/plain
Vary: Accept-Encoding
X-Powered-By: ASP.NET
Date: Mon, 17 Aug 2020 19:17:10 GMT
[...]
Access-Control-Allow-Origin: *
[...]
```

Figuur 12 – Header 'Access-Control-Allow-Origin' met waarde '' waargenomen*

Op basis van bovenstaande waargenomen header, is vastgesteld dat de 'Access-Control-Allow-Origin'-header geconfigureerd is met de waarde '*' omdat de header er dan voor zorgt dat elk domein toestaat cross-domein verzoeken uit te voeren en te HTTP-responses uit te lezen. Dit geldt ook voor de andere benoemde URL's.

Mogelijke Impact:
Een aanvaller zou deze misconfiguratie mogelijk kunnen misbruiken door inhoud op te halen uit een toepassing die geen standaardverificatiemechanismen gebruikt.

Aanbeveling:
Geadviseerd wordt om de 'Access-Control-Allow-Origin' header te configureren om alleen bekende en vertrouwde domeinen op een 'toegestane lijst' te zetten, of om de header te verwijderen.

4.1.4 Informatief

12. Invoervalidatie onvolledig (WSTG-BUSL-01)

Host:
https://coronamelder-api.nl/v1/postkeys

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

Opgelost

CVSS Score: Informatief

Kwetsbaarheid:
De API controleert de invoer niet op tekens die een speciale betekenis hebben in HTML en JavaScript.

Bevestiging
Door '<script>alert()</script>' als waarde voor de parameter sig mee te geven in het POST-request naar /v1/postkeys, antwoordt de webserver met een error 400 (Bad Request).

```
POST /v1/postkeys?sig=<script>alert()</script> HTTP/1.1
Host: coronamelder-api.nl
[...]
Content-Length: 3

a=1
```

Daarnaast is een foutmelding te lezen waarin beschreven staat wat er mis is. Hierin wordt de bovenstaande waarde van de parameter sig exact gereflecteerd.

```
HTTP/1.1 400 Bad Request
Content-Type: application/problem+json; charset=utf-8
[...]

{"type":"https://tools.ietf.org/html/rfc7231#section-6.5.1","title":"One or
more validation errors occurred.","status":400,"traceId":"|6893094d-
4a5ad87d2b40cb1f.","errors":{"sig":["The value '<script>alert()</script>'
is not valid."]}}
```

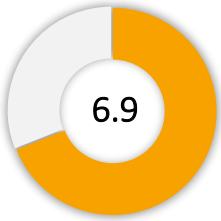
Mogelijke Impact:
De API geeft alleen responses waarin de gebruikersinvoer voorkomt met de header Content-Type: application/json (of application/problem+json), waardoor er geen Cross-Site Scripting (XSS) optreedt. Doordat de X-Content-Type-Options header eveneens niet correct wordt meegegeven, is XSS alsnog mogelijk in een aantal specifieke browsers.

Aanbeveling:
Geadviseerd wordt om alle gebruikersinvoer te filteren op karakters die een speciale betekenis hebben in HTML en JavaScript, zoals <, >, ", en '. Deze kunnen worden verwijderd uit de invoer. Als het noodzakelijk is om het gebruik van HTML-karakters toe te staan in bepaalde velden, moet de applicatie die de data weergeeft in een webpagina deze karakters vervangen door hun HTML-encodings (<, >, ", ') als ze op schermen worden weergegeven. Voor meer informatie verwijzen we naar de OWASP XSS-preventie cheat sheet op https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

5 Bevindingen penetratietest – White Box

5.1 Bevindingen – Mobiele applicatie – iOS

5.1.1 Gemiddeld

13. Verouderde SSL-bibliotheek aanwezig (MSTG-NETWORK-6)	
Scope: CoronaMelder app voor iOS (buildnummer 1.0.0 – 644f133)	 Opgelost **CVSS Score:** Gemiddeld
CVSS Vector String: CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A:H/E:P/RL:O/RC:C/CR:H/IR:H/AR:H	

Kwetsbaarheid:
De COVID-19 Notificatie applicatie voor Apple iOS (buildnummer 1.0.0 – 644f133) maakt voor het verwerken van data vanaf internet gebruik van een verouderde externe software-bibliotheek (OpenSSL versie 1.1.1d). Deze software- bibliotheek is mogelijk kwetsbaar voor een Denial-of-Service (DoS) aanval. Deze kwetsbaarheid draagt het CVE-nummer: CVE-2020-1967.

Bevestiging:
Er is gekeken naar de gepubliceerde broncode van de CoronaMelder app voor iOS[2], daarbij is specifiek gekeken naar het gebruik van externe softwarebibliotheken.

Daarbij is waargenomen dat de versie van de software-bibliotheek OpenSSL, een verouderde versie betreft (1.1.1d).

Mogelijke Impact:
Een aanvaller die in staat is om een man-in-the-middle (MitM)-aanval uit te voeren, kan mogelijk door middel van deze kwetsbaarheid een Denial-of-Service (DoS) van de mobiele applicatie veroorzaken.

Aanbeveling:
Geadviseerd wordt om de nieuwe versie van OpenSSL (vanaf OpenSSLversie 1.1.1g) te gebruiken, om de aanwezige kwetsbaarheid te verhelpen.

[2]Bron: https://github.com/minvws/nl-covid19-notification-app-ios/tree/master/vendor/OpenSSL-for-iPhone, geraadpleegd op 17-08-2020

14. Cache aanwezig in app-container (MSTG-PLATFORM-10)

Scope:
CoronaMelder app voor iOS
(buildnummer 1.0.0 – 644f133)

CVSS Vector String:
CVSS:3.1/AV:P/AC:H/PR:H/UI:R/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Gemiddeld

Kwetsbaarheid:
De COVID-19 Notificatie applicatie voor Apple iOS (buildnummer 1.0.0 – 644f133) maakt voor het verwerken van data-sleutels gebruik van verschillende mappen binnen de applicatie-container. Na het verwerken van deze sleutels, blijven de cache-bestanden aanwezig binnen de applicatie-container.

Bevestiging:
Met het SFTP-protocol is ingelogd op de mobiele telefoon, waarna vervolgens is genavigeerd naar de applicatie-container van de mobiele applicatie (`/private/var/mobile/Containers/Data/Application/[UUID`[3]`]`).

Daarbij zijn de volgende bestandsmappen waargenomen welke mogelijk cache-informatie bevatten (zoals vanaf de server gedownloade gepubliceerde TEKs):

- `/tmp`
- `/Library`
  - `/Library/Caches/`
  - `/Library/Caches/store/`
  - `/Library/Caches/nl.rijksoverheid.en/`
  - `/Library/Caches/nl.rijksoverheid.en/fsCachedData/`
- `/Documents`
  - `/Documents/store/`

Mogelijke Impact:
Een aanvaller, met fysieke toegang tot de telefoon en bijbehorende toegangscode, die in staat is toegang te verkrijgen tot het bestandssysteem van de individuele telefoon (door middel van jailbreaking), kan mogelijk zien welke publieke sleutels er door de mobiele applicatie verwerkt zijn.

Aanbeveling:
Geadviseerd wordt om, nadat de verwerking van de publieke data is afgerond, de bestandsmappen waarin cache-informatie opgeslagen staat te legen.[4]

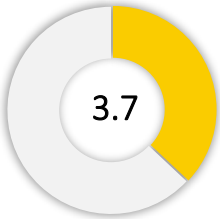
Reactie opdrachtgever (31 augustus 2020):
Dit betreft sleutels die publiek beschikbaar zijn en worden opgehaald. De openbare informatie vormt daarmee geen probleem. Het is wel een punt dat we toevoegen aan de back log om op te lossen.

[3] Universally Unique Identifier (UUID) – deze identifier is bij elke installatie anders, in het geval van deze installatie 'C0CB95A6-9B3B-4520-B6A3-0FCB380AAACF'

5.1.2 Laag

15. Jailbreak detectie niet aanwezig (MSTG-RESILIENCE-1)

Scope:
CoronaMelder app voor iOS
(buildnummer 1.0.0 – 644f133)



CVSS Vector String:
CVSS:3.1/AV:P/AC:H/PR:H/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

CVSS Score: Laag

Kwetsbaarheid:
De COVID-19 Notificatie applicatie voor Apple iOS (buildnummer 1.0.0 – 644f133) detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, voorzien is van een zogenoemde "jailbreak".

Bevestiging:
De applicatie is geïnstalleerd op een iPhone X met iOS versienummer 13.6 welke voorzien is van een jailbreak. Na de installatie van de applicatie op de mobiele telefoon is de applicatie met succes gestart:

Figuur 13 - Applicatie uitgevoerd op iPhone X (gejailbreakt)

Mogelijke Impact:
Een aanvaller, die fysieke toegang heeft tot een mobiele telefoon en deze succesvol kan jailbreaken, kan mogelijk toegang krijgen tot data die door de applicatie worden bijgehouden.

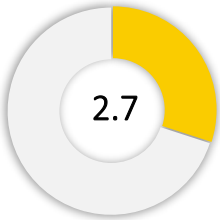
Aanbeveling:
Geadviseerd wordt om een vorm van jailbreak detectie toe te passen die de gebruiker waarschuwt dat de mobiele telefoon gejailbreakt is, of het uitvoeren van de applicatie blokkeert. Meer informatie over het toepassen van controles kan worden gevonden op: https://github.com/OWASP/owasp-mstg/blob/master/Document/0x06j-Testing-Resiliency-Against-Reverse-Engineering.md#jailbreak-detection-mstg-resilience-1

4 https://developer.apple.com/library/archive/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemOverview/FileSystemOverview.html#//apple_ref/doc/uid/TP40010672-CH2-SW4 - Table 1-1

Reactie opdrachtgever (31 augustus 2020):
Dit is een bewuste keuze. Door telefoons met jailbreak af te sluiten, sluiten we een doelgroep uit. Ook biedt het hebben van een jailbreak mensen de mogelijkheid te controleren dat de app niks anders is dan de open-sourcebroncode laat zien.

16. Reverse Engineering Tools detectie (MSTG-RESILIENCE-4)

Scope:
CoronaMelder app voor iOS
(buildnummer 1.0.0 – 644f133)



CVSS Vector String:
CVSS:3.1/AV:P/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Laag

Kwetsbaarheid:
De COVID-19 Notificatie applicatie voor Apple iOS (buildnummer 1.0.0 – 644f133) detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, reverse engineering tools zoals Frida bevat.

Bevestiging:
De applicatie is geïnstalleerd op een telefoon met Apple iOS versie 13.6. Op deze telefoon staat Frida, Substrate, Cycript en SSL Kill switch geïnstalleerd. Vervolgens wordt de COVID-19 notificatie applicatie opgestart. Er komt geen melding dat deze software actief is.

Mogelijke Impact:
Een aanvaller, die in staat is om op de mobiele telefoon bepaalde applicaties op te starten, kan reverse engineering uitvoeren op de applicatie en zo mogelijk vertrouwelijke informatie achterhalen.

Aanbeveling:
Geadviseerd wordt om een vorm van reverse engineering tooling detectie toe te passen die de gebruiker waarschuwt dat bepaalde tools actief zijn op de mobiele telefoon, of het uitvoeren van de applicatie blokkeert. Meer informatie over het toepassen van deze controles kan worden gevonden op: https://github.com/OWASP/owasp-mstg/blob/master/Document/0x06j-Testing-Resiliency-Against-Reverse-Engineering.md

***Reactie opdrachtgever (31 augustus 2020)**:*
Dit is een bewuste keuze. Gebruikers mogen de werking analyseren en dat in de gaten houden.

5.1.3 Informatief

17. Emulator detectie niet aanwezig (MSTG-RESILIENCE-5)

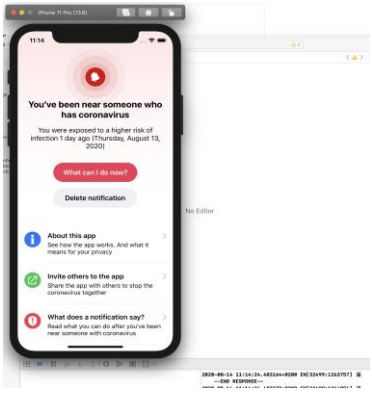
Scope:
CoronaMelder app voor iOS
(GitHub versie - buildnummer 1.0.2)

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
De COVID-19 Notificatie applicatie voor Apple iOS (GitHub versie - buildnummer 1.0.2) detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, een zogenoemde emulator betreft.

Bevestiging:
De applicatie is geïnstalleerd op een virtuele telefoon (emulator) waarop Apple iOS versie 13.6 geïnstalleerd staat. Na de installatie van de applicatie op de mobiele telefoon is de applicatie met succes gestart:



Figuur 14 - Applicatie gestart in iOS Emulator

Mogelijke Impact:
Een aanvaller, die in staat is om de mobiele applicatie in een emulator te laten werken, kan mogelijk aanvallen uitvoeren op de integriteit van de applicatie, welke normaal niet mogelijk zijn op een echte telefoon.

Aanbeveling:
Geadviseerd wordt om een vorm van emulator detectie toe te passen die de gebruiker waarschuwt dat de mobiele telefoon geroot is, of het uitvoeren van de applicatie blokkeert. Meer informatie over het toepassen van deze controles kan worden gevonden op https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05j-Testing-Resiliency-Against-Reverse-Engineering.md#testing-emulator-detection-mstg-resilience-5

Reactie opdrachtgever (31 augustus 2020):
Dit is een bewuste keuze, waardoor beveiligingsonderzoekers hun werk ook volledig kunnen uitvoeren.

18. Informatie in schermafdruk (MSTG-STORAGE-9)

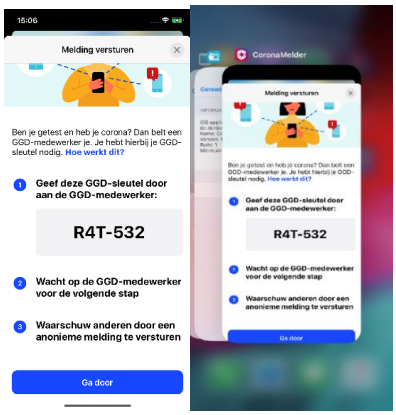
Host:
CoronaMelder app voor iOS
(buildnummer 1.0.0 – 644f133)

CVSS Vector String:
CVSS:3.1/AV:P/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
iOS apparaten bevatten de mogelijkheid om een schermafdruk te maken terwijl de applicatie op de achtergrond actief is. Hiermee kan mogelijk gevoelige data worden opgeslagen in een schermafdruk.

Bevestiging:
De COVID-19 notificatie applicatie wordt geopend en het tijdelijke wachtwoord wordt getoond, zie schermafdruk links:



Figuur 15 - GGD-sleutel in applicatie

Vervolgens is de applicatie switcher geopend, waarbij het tijdelijke wachtwoord nog steeds leesbaar is, zie schermafdruk rechts.

Mogelijke Impact:
Een ongeautoriseerde gebruiker kan mogelijk toegang verkrijgen tot het iOS toestel en hierbij de corona melder applicatie bekijken. Door de opzet van de COVID-19 notificatie applicatie is het risico echter laag.

Aanbeveling:
Geadviseerd wordt om de GGD-sleutel niet zichtbaar te maken wanneer de applicatie op de achtergrond actief is.

Reactie opdrachtgever (31 augustus 2020):
Omdat sommige gebruikers moeite hebben om te communiceren, kan een schermafdruk ze helpen. De risico's worden beperkt door het inzetten van wetgeving om misbruik hiervan te voorkomen en te bestraffen.

5.2 Bevindingen – Mobiele applicatie - Android

5.2.1 Gemiddeld

19. Cache aanwezig in app-container (MSTG-PLATFORM-10)	
Scope: CoronaMelder app voor Android - v0.4.1-1-prod-release	 Opgelost **CVSS Score:** Gemiddeld
CVSS Vector String: CVSS:3.1/AV:P/AC:H/PR:H/UI:R/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H	

Kwetsbaarheid:
De CoronaMelder applicatie maakt voor het verwerken van de gepubliceerde TEK-sleutels gebruik van verschillende mappen binnen de applicatiecontainer. Na het verwerken van deze sleutels, blijven de cache-bestanden aanwezig binnen de applicatie-container.

Bevestiging:
Met het ADB-protocol is ingelogd op de mobiele telefoon, waarna vervolgens genavigeerd is naar de applicatie-container van de mobiele applicatie.

Daarbij zijn de volgende bestandsmappen waargenomen welke mogelijk cache-informatie bevatten:

- `/data/user/0/nl.rijksoverheid.en/cache/http/`
- `/data/data/nl.rijksoverheid/cache/http/`

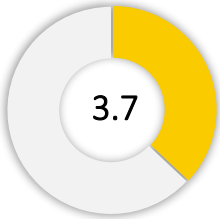
Mogelijke Impact:
Een aanvaller, met fysieke toegang tot de telefoon en bijbehorende toegangscode, die in staat is toegang te verkrijgen tot het bestandssysteem van de individuele telefoon (door middel van rooting), kan mogelijk zien welke publieke sleutels er door de mobiele applicatie verwerkt zijn.

Aanbeveling:
Geadviseerd wordt om de bestandsmappen waarin cache-informatie opgeslagen staat te legen, nadat de verwerking van de publieke data is afgerond.

5.2.2 Laag

20. Root detectie niet aanwezig (MSTG-RESILIENCE-1)

Scope:
CoronaMelder app voor Android - v0.4.1-1-prod-release

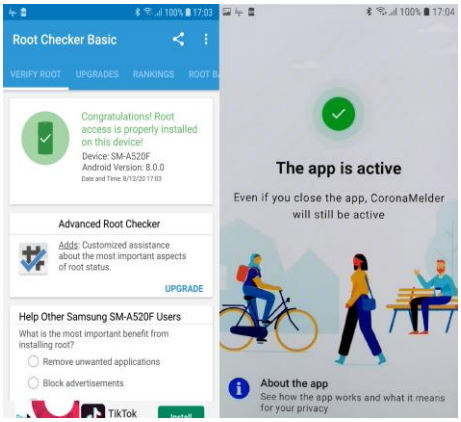


CVSS Vector String:
CVSS:3.1/AV:P/AC:H/PR:H/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

CVSS Score: Laag

Kwetsbaarheid:
De CoronaMelder applicatie voor Google Android detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, voorzien is van een zogenoemde “root-toegang”.

Bevestiging:
De applicatie is geïnstalleerd op een fysieke telefoon waarop Google Android versienummer 10.0.0 geïnstalleerd staat. Deze telefoon is geroot. Na de installatie van de applicatie op de mobiele telefoon is de applicatie met succes gestart:



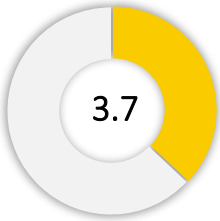
Mogelijke Impact:
Een aanvaller, die fysieke toegang heeft tot een mobiele telefoon en deze succesvol kan rooten, kan mogelijk toegang krijgen tot data die door de applicatie wordt bijgehouden.

Aanbeveling:
Geadviseerd wordt om een vorm van root detectie toe te passen die de gebruiker waarschuwt dat de mobiele telefoon geroot is, of het uitvoeren van de applicatie blokkeert. Meer informatie over het toepassen van controles kan worden gevonden op: https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05j-Testing-Resiliency-Against-Reverse-Engineering.md#user-content-testing-root-detection

Reactie opdrachtgever (31 augustus 2020):
We blokkeren het gebruik van een telefoon, waarop root-toegang verkregen is niet. Hierdoor is het mogelijk om de werking en de code te verifiëren. Ook willen we het rooten van een telefoon niet bestraffen of mensen de mogelijkheid ontnemen om de app te kunnen gebruiken.

21. Reverse Engineering Tools detectie (MSTG-RESILIENCE-4)

Host:
CoronaMelder app voor Android -
v0.4.1-1-prod-release



CVSS Vector String:
CVSS:3.1/AV:P/AC:H/PR:H/UI:R/S:U/C:L/I:L/A:N/CR:H/IR:H/AR:H

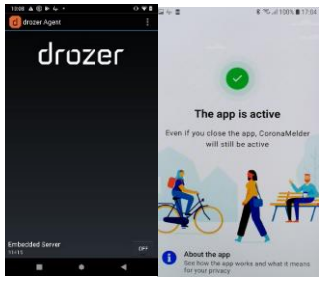
CVSS Score: Laag

Kwetsbaarheid:
De CoronaMelder applicatie voor Google Android detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, reverse engineering tools zoals Frida en Drozer bevat.

Bevestiging:
De applicatie is geïnstalleerd op een telefoon met Google Android versie 10.0. Op deze telefoon is de agent voor Drozer en een Frida-server geïnstalleerd en actief gemaakt:

```
# ps -ef | grep frida| awk '{print $1 " " $8}'
root frida-server
root frida-helper-32
```

Figuur 16 - Frida-server is actief



Figuur 17 - Drozer is actief

Vervolgens is de CoronaMelder applicatie gestart en kwam er geen melding dat er reverse engineering software actief was noch werd de uitvoer van de applicatie geblokkeerd.

Mogelijke Impact:
Een kwaadwillende, die in staat is om op de mobiele telefoon bepaalde applicaties op te starten, kan reverse engineering uitvoeren op de applicatie en zo mogelijk vertrouwelijke informatie achterhalen.

Aanbeveling:
Geadviseerd wordt om een vorm van reverse engineering tooling detectie toe te passen die de gebruiker waarschuwt dat bepaalde tools actief zijn op de mobiele telefoon, of het uitvoeren van de applicatie blokkeert. Meer informatie over het toepassen van deze controles kan worden gevonden op: https://github.com/OWASP/owasp-mstg/blob/master/Document/0x06j-Testing-Resiliency-Against-Reverse-Engineering.md

Reactie opdrachtgever (31 augustus 2020):
Reverse engineering is nadrukkelijk niet verboden. Dit betreft een open-sourceproject waar de broncode beschikbaar is. Hoe de app werkt is geen geheim. Juist het Kerckhoff's-principe achten wij zeer belangrijk.

5.2.3 Informatief

22. Emulator detectie niet aanwezig (MSTG-RESILIENCE-5)

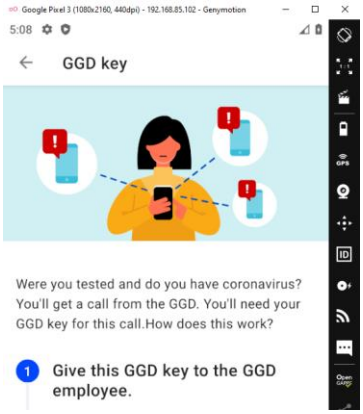
Scope:
CoronaMelder app voor Android - v0.4.1-1-prod-release

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
De CoronaMelder applicatie voor Google Android detecteert niet of het apparaat waarop de applicatie geïnstalleerd is, een zogenoemde emulator betreft.

Bevestiging:
De CoronaMelder applicatie is geïnstalleerd op een virtuele Google Pixel 3 telefoon (emulator) waarop Google Android versienummer 10.0.0 geïnstalleerd staat. Voor de emulatie is gebruik gemaakt van de Genymotion. Na de installatie van de applicatie op de mobiele telefoon is de applicatie met succes gestart:



Mogelijke Impact:
De mogelijkheid van het emuleren van de CoronaMelder applicatie maakt het debuggen en reverse engineering van de applicatie aanzienlijk makkelijker. Een aanvaller kan hierdoor eenvoudiger mogelijke kwetsbaarheden vinden.

Aanbeveling:
Geadviseerd wordt om een vorm van emulator detectie toe te passen die het uitvoeren van de applicatie in een emulator blokkeert. Meer informatie over het toepassen van deze controles kan worden gevonden op https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05j-Testing-Resiliency-Against-Reverse-Engineering.md#testing-emulator-detection-mstg-resilience-5

Reactie opdrachtgever (31 augustus 2020):
Inherent aan open-sourcesoftware is het kunnen achterhalen van de werking van het systeem of het uitvoeren in een emulator. Dat is geen functionaliteit die we willen verbieden gelet op het open karakter. Wel zal de werking pas vol tot zijn recht komen als in de emulator ook de GAEN-laag is toegevoegd.

23. Voorspelbare willekeurigheid (MSTG-CRYPTO-6)

Scope:
CoronaMelder app voor Android - v0.4.1-1-prod-release

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
De CoronaMelder applicatie voor Google Android maakt twee keer gebruik van de voorspelbare Random()-functie[5]. Deze functie is minder willekeurig (en daarmee minder veilig) dan de SecureRandom()-functie[6].

Bevestiging:
In de broncode is twee keer de verwijzing naar de Random()-functie geconstateerd:

```
63    private fun generatePadding(size: Long): String {
64        return (0 until size)
65            .map { CHARACTER_SET.random() }
66            .joinToString("")
67    }
```

Tabel 9 - \api\src\main\java\nl\rijksoverheid\en\api\PaddedRequestInterceptor.kt

```
263    private fun generateDecoyBucketId(size: Int): String {
264        val source =
           "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890"
265        return (1..size)
266            .map { source.random() }
267            .joinToString("")
268    }
```

Tabel 10 - \app\src\main\java\nl\rijksoverheid\en\labtest\LabTestRepository.kt

Mogelijke Impact:
Onder bepaalde omstandigheden kan deze zwakte de versleuteling van de gegevens van de mobiele applicatie op basis van willekeurigheid in gevaar brengen. Vanuit het huidige perspectief kan niet bepaald worden of deze gegevens een cryptografische context hebben. Desalniettemin wil NFIR het gebruik van deze Random()-functie kenbaar maken.

Aanbeveling:
Geadviseerd wordt om beide Random()-functies te vervangen met de SecureRandom() equivalent.

Reactie opdrachtgever (31 augustus 2020):
Dit heeft onze aandacht en wordt als item aan de back log toegevoegd.

[5] https://developer.android.com/reference/java/util/Random.html

[6] https://developer.android.com/reference/java/security/SecureRandom

24. WAKE_LOCK-permissie (MSTG-PLATFORM-1)

Scope:
CoronaMelder app voor Android - v0.4.1-1-prod-release

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
De CoronaMelder applicatie voor Google Android maakt gebruik van de WAKE_LOCK-permissie. Deze permissie zorgt ervoor dat de processor actief blijft en het scherm niet gesluimerd kan worden. In de applicatie is geen functionaliteit aangetroffen waarvoor deze permissie noodzakelijk is.

Bevestiging:
Met Apktool is het apk-bestand van de CoronaMelder applicatie gedecompileerd. In het bestand AndroidManifest.xml s de WAKE_LOCKpermissie aangetroffen:

AndroidManifest.xml

```
1  <?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="
2      <uses-feature android:name="android.hardware.bluetooth_le" android:required
3      <uses-feature android:name="android.hardware.bluetooth"/>
4      <uses-permission android:name="android.permission.INTERNET"/>
5      <uses-permission android:name="android.permission.BLUETOOTH"/>
6      <uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>
7      <uses-permission android:name="android.permission.WAKE_LOCK"/>
8      <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
9      <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
10     <application android:allowBackup="false" android:appComponentFactory="andro
11         <activity android:name="nl.rijksoverheid.en.MainActivity">
12             <intent-filter>
```

Figuur 18 - Permissies in AndroidManifest.xml

Mogelijke Impact:
De CoronaMelder applicatie kan ervoor zorgen dat de processor actief blijft en het scherm niet gesluimerd kan worden. Daardoor kan de batterij van de smartphone sneller leeg raken.

Aanbeveling:
Geadviseerd wordt om te bekijken of de WAKE_LOCK-permissie noodzakelijk is voor de werking van deze app. Verwijder indien mogelijk deze permissie uit de app.

Reactie opdrachtgever (31 augustus 2020):
Het omgaan met de slaapfunctionaliteit is iets wat momenteel in behandeling is vanuit een breder perspectief.

25. Apparaat beveiligingsbeleid opties (MSTG-STORAGE-11)

Scope:
CoronaMelder app voor Android
-v0.4.1-1-prod-release

CVSS Vector String:
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N/CR:H/IR:H/AR:H

CVSS Score: Informatief

Kwetsbaarheid:
De applicatie controleert niet of één van de volgende beveiligingsinstellingen aanwezig zijn:

- Pincode of wachtwoord om het apparaat te ontgrendelen
- USB Debugging
- Apparaat Encryptie

Het risico dat een ongeautoriseerde gebruiker toegang kan verkrijgen tot het apparaat is hiermee verhoogd.

Bevestiging:
Bovenstaande beveiligingsinstellingen zijn op het testtoestel niet ingesteld. De applicatie kan succesvol worden gestart dat een notificatie getoond wordt over de beveiligingsinstellingen van het apparaat.

Mogelijke Impact:
Een kwaadwillende kan mogelijk toegang verkrijgen tot het Android toestel en hierbij de CoronaMelder applicatie bekijken.

Aanbeveling:
Geadviseerd wordt om te bekijken of bovenstaande beveiligingsinstellingen vereist zijn om de applicatie te gebruiken.

Reactie opdrachtgever (31 augustus 2020):
Dit is inherent aan het open ontwikkelproces.

6 Bijlage

6.1 Verklarende woordenlijst

Term:	Uitleg:
Android	Android is een besturingssysteem wat ontwikkeld- en onderhouden wordt door Google
API	API staat voor Application Programming Interface (API) en is een verzameling van definities op basis waarvan een computerprogramma kan communiceren met een ander programma of onderdeel (meestal in de vorm van bibliotheken).
BladeRF	Gespecialiseerde hardware voor het versturen of ontvangen van signalen variërend van 47Mhz tot 6GHz
Bluetooth	Draadloze communicatietechnologie voor het versturen van apparaat informatie over korte afstanden
Bluetooth LE	Draadloze communicatietechnologie voor het versturen van apparaat informatie over korte afstanden, speciaal gemaakt voor laag energieverbruik
Buildnummer	Een buildnummer wordt gebruikt om bij te houden hoeveel keer een bepaalde software gecompileerd is.
Content-Security-Policy header	Header die een site beschermt tegen Cross-site scripting (XSS) aanvallen. Door het definiëren van een lijst met toegestane content, kan worden voorkomen dat de browser kwaadwillende code kan laden van een externe bron. Met de kwaadwillende code kunnen bijvoorbeeld cookies van gebruikers worden afgevangen en worden gebruikt om mee in te loggen of kan malware worden geïnjecteerd in de website.
CVSS	CVSS staat voor Common Vulnerability Scoring System en wordt gebruikt om een risicoprofiel te kunnen berekenen voor een kwetsbaarheid.
Emulator	Een emulator is een hardware of softwareprogramma dat een computersysteem in staat stelt om functies te imiteren van andere systemen.
ExposureNotification-Framework	Een door Google & Apple gepubliceerd raamwerk welke het mogelijk maakt om ExposureNotification-advertisements te versturen en ontvangen via het gebruik van Bluetooth Low Energy (LE).
Feature-Policy header	Header waarmee controle is over welke functies en API's gebruikt kunnen worden in de browser.

HSM	Hardware security module
HTTP	HTTP staat voor HyperText Transfer Protocol en is een techniek die wordt gebruikt voor communicatie tussen gebruikers en servers (websites).
HTTPS	HTTPS staat voor HyperText Transfer Protocol Secure en is een techniek die wordt gebruikt voor het veilig communiceren tussen gebruikers en servers.
iOS	iOS is een besturingssysteem wat ontwikkeld- en onderhouden wordt door Apple
MSTG	MSTG staat voor Mobile Security Testing Guide en is een methodiek die specifiek gebruikt wordt voor het uitvoeren van penetratietesten op mobile applicaties.
NMAP	Nmap, is een afkorting van Network Mapper, een open source tool voor het scannen van open poorten en kwetsbaarheden.
PTES	PTES staat voor Penetration Testing Execution Standard en is een methodiek die gebruikt wordt voor het uitvoeren van penetratietesten
Referrer-Policy header	Header die controleert welke informatie door een site opgehaald mag worden van een andere bron.
RPI	Rolling Proximity Identifier
TEK	Temporary Exposure Key
WSTG	WSTG staat voor Web Security Testing Guide en is een methodiek die specifiek gebruikt wordt voor het uitvoeren van penetratietesten op webomgevingen.
X-Content-Type-Options header	Voorkomt dat een MIME-sniffing-aanval kan worden uitgevoerd waarmee een aanvaller een bestand kan verhullen als een ander bestandstype. Hierdoor kan een malafide uitvoerbaar bestand bijvoorbeeld worden geüpload met een jpg-extensie. Doordat het bestand de kenmerken heeft van een uitvoerbaar bestand, wordt het bestand in de sommige gevallen vervolgens opgeslagen en uitgevoerd.
X-Frame-Options header	Geeft een melding aan de browser of de site in een frame geladen mag worden of niet. Hierdoor kunnen clickjacking aanvallen worden voorkomen, aangezien de site niet in een frame geladen kan worden. Een aanvaller zou een dergelijke aanval kunnen gebruiken om een malafide website te laden in een frame, waardoor de gebruiker onbedoeld wordt omgeleid.

Tabel 11 - Verklarende woordenlijst

6.2 Hulpmiddelen

Door het gebruik van hulpmiddelen zijn scantaken geautomatiseerd en is de infrastructuur geïdentificeerd. Hieronder staan de gebruikte hulmiddelen, inclusief versienummers:

Hulpmiddel:	Versie:
Apksigner	30.0.3
Apktool	2.4.1
Burp Suite Professional	2020.7
Drozer	2.4.4
Frida	12.10.4
Fridump	0.1
Ghidra	9.1.2
Gobuster	3.0.1-0
Hashcat	6.0.0
Kali Linux	2020.1a/2020.3
JiaoXianjun/BTLE	n/a
Mac OS X	10.15.6
MobSF	3.0
Mozilla Firefox	79.0
Nessus Professional	8.10.0
Nikto	1:2.1.6
Nmap	7.80
Nuand BladeRF	Revision 2
SSLScan	2.0.0
SSLTest	3.1dev
Xcode	11.6
Windows	10 Pro
Ubuntu	20.04 LTS

Atom	1.50.0
Sublime Text 3	Build 3211
Visual Studio Code	1.48.1

Tabel 12 - Gebruikte hulpmiddelen software

6.3 Ontvangen bestanden

NFIR heeft de volgende bestanden ontvangen en gebruikt voor deze penetratie test:

Bestandsnaam	Beschrijving	SHA1-Hash
covid-notificatie-v0.4.1-1-prod-release.apk	CoronaMelder App voor Android (No SSL pinning)	a58f4022bba76306d2bfb99b4c2c0a4f016da7fc
covid-notificatie-v0.4.1-67885-prod-release.apk	CoronaMelder App voor Android	7b03d4e53024aab6cc17ce992a891d0c500f8189
EN Disable SSL Pinning\EN.ipa	CoronaMelder App voor iOS (No SSL pinning)	cfb1a9594d244842f3d01611b70d192eaa9b0a0c
EN SSL Pinning\EN.ipa	CoronaMelder App voor iOS (SSL pinning)	5f182db1f788403e6789be28980f353cfc02831e
EN Disable SSL Pinning\EN.ipa	CoronaMelder App voor iOS (No SSL pinning) hertest	06d14aa4e649250fe16749508e78ad92e4750e84
EN SSL Pinning\EN.ipa	CoronaMelder App voor iOS (SSL pinning) hertest	525a6c8360f6e670da6afea5cd63266b3f8215ef
20200729-VWS-CIBG-CoronaMelder App-V0.15-LLD.pdf	Netwerk overzicht (incl. bijbehorende IP-adressen)	596345c98fa0f783a66a66b8df751f1445db0fe8

Tabel 13 - Ontvangen bestanden

6.4 Accounts verwijderen

Onderstaande accounts zijn aangemaakt tijdens de uitgevoerde test.

Account	Beschrijving
Gebruikersnaam: m.vangeelen@nfir.nl	Account 1 Meld Portaal
Gebruikersnaam: g.devries@nfir.nl	Account 2 Meld Portaal
Gebruikersnaam: COVP\geelenm	Account 1 Beheer omgeving
Gebruikersnaam: COVP\vriesg	Account 2 Beheer omgeving

Tabel 14 - Accounts verwijderen

NFIR adviseert om te controleren of de accounts verwijderd zijn of dit alsnog te doen.

6.5 Uitgevoerde scans

6.5.1 WHOIS-registratiegegevens

6.5.1.1 WHOIS coronamelder-api.nl[7]

Domeinnaam	coronamelder-api.nl
Status	Actief
Houder	Rijksoverheid
Administratieve contactpersoon	domeinnaam@minaz.nl
Registar	Rijksoverheid Buitenhof 34 2513AH Den Haag Netherlands
Technische contactpersoon	domeinnaam@minaz.nl
DNSSEC	Ja
Domeinnaamservers	ns1.rijksoverheidnl.nl ns3.rijksoverheidnl.org ns2.rijksoverheidnl.eu ns4.rijksoverheidnl.com
Registratiedatum	2020-07-22
Datum laatste wijziging	2020-07-23

Tabel 15 - WHOIS-registratiegegevens coronamelder-api.nl

[7] Bron: https://www.sidn.nl/whois/?q=coronamelder-api.nl, waargenomen op 18-08-2020

6.5.1.2 WHOIS coronamelder-portal.nl[8]

Domeinnaam	coronamelder-portal.nl
Status	Actief
Houder	Rijksoverheid
Administratieve contactpersoon	domeinnaam@minaz.nl
Registar	Rijksoverheid Buitenhof 34 2513AH Den Haag Netherlands
Technische contactpersoon	domeinnaam@minaz.nl
DNSSEC	Ja
Domeinnaamservers	ns1.rijksoverheidnl.nl ns3.rijksoverheidnl.org ns2.rijksoverheidnl.eu ns4.rijksoverheidnl.com
Registratiedatum	2020-07-22
Datum laatste wijziging	2020-07-22

Tabel 16 - WHOIS-registratiegegevens coronamelder-portal.nl

[8] Bron: https://www.sidn.nl/whois/?q=coronamelder-portal.nl, waargenomen op 18-08-2020

6.5.1.3 WHOIS coronamelder-dist.nl[9]

Domeinnaam	coronamelder-dist.nl
Status	Actief
Houder	Rijksoverheid
Administratieve contactpersoon	domeinnaam@minaz.nl
Registar	Rijksoverheid Buitenhof 34 2513AH Den Haag Netherlands
Technische contactpersoon	domeinnaam@minaz.nl
DNSSEC	Ja
Domeinnaamservers	ns01.is.nl ns02.is.nl ns03.is.nl
Registratiedatum	2020-07-20
Datum laatste wijziging	2020-08-17

Tabel 17 - WHOIS-registratiegegevens coronamelder-dist.nl

[9] Bron: https://www.sidn.nl/whois/?q=coronamelder-dist.nl, waargenomen op 18-08-2020

6.5.2 NMAP-scans

6.5.2.1 NMAP-scan coronamelder-api.nl

IP address	Open port	Service
82.201.43.165	80/tcp 443/tcp	http – Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP) https - Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)

Tabel 18 - NMAP scan coronamelder-api.nl

6.5.2.2 NMAP-scan coronamelder-portal.nl

IP address	Open port	Service
82.201.43.164	443/tcp	https - Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)

Tabel 19 - NMAP scan coronamelder-portal.nl

6.5.2.3 NMAP-scan productie.coronamelder-dist.nl

IP address	Open port	Service
195.121.65.164	443/tcp	https - Varnish http accelerator

Tabel 20 - NMAP scan productie.coronamelder-dist.nl

6.5.2.4 NMAP-scan 217.194.124.5

IP address	Open port	Service
217.194.124.5	443/tcp	https - Cisco ASA SSL VPN

Tabel 21 - NMAP scan 217.194.124.5

6.5.2.5 NMAP-scan 217.194.124.4

IP address	Open port	Service
217.194.124.4	443/tcp	https - Cisco ASA SSL VPN

Tabel 22 - NMAP scan 217.194.124.4

6.5.3 SSL-scans

6.5.3.1 SSL-scan coronamelder-api.nl

```
Version: 1.11.13-static
OpenSSL 1.0.2-chacha (1.0.2g-dev)

Connected to 82.201.43.165

Testing SSL server coronamelder-api.nl on port 443 using SNI name
coronamelder-api.nl

  TLS Fallback SCSV:
Server supports TLS Fallback SCSV

  TLS renegotiation:
Session renegotiation not supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.2 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.0 not vulnerable to heartbleed

  Supported Server Cipher(s):
Preferred TLSv1.2  256 bits  ECDHE-RSA-AES256-GCM-SHA384   Curve P-384 DHE
384
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-GCM-SHA256   Curve P-384 DHE
384

  SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
RSA Key Strength:    2048

Subject:  coronamelder-api.nl
Altnames: DNS:coronamelder-api.nl
Issuer:   KPN BV PKIoverheid Organisatie Server CA - G3

Not valid before: Jul 27 10:10:33 2020 GMT
Not valid after:  Jul 27 10:10:33 2021 GMT
```

6.5.3.2 SSL-scan coronamelder-portal.nl

```
Version: 1.11.13-static
OpenSSL 1.0.2-chacha (1.0.2g-dev)

Connected to 82.201.43.164

Testing SSL server coronamelder-portal.nl on port 443 using SNI name
coronamelder-portal.nl

  TLS Fallback SCSV:
Server supports TLS Fallback SCSV

  TLS renegotiation:
Session renegotiation not supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.2 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.0 not vulnerable to heartbleed

  Supported Server Cipher(s):
Preferred TLSv1.2  256 bits  ECDHE-RSA-AES256-GCM-SHA384   Curve P-384 DHE
384
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-GCM-SHA256   Curve P-384 DHE
384

  SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
RSA Key Strength:    2048

Subject:  coronamelder-portal.nl
Altnames: DNS:coronamelder-portal.nl
Issuer:   KPN BV PKIoverheid Organisatie Server CA - G3

Not valid before: Jul 27 10:10:41 2020 GMT
Not valid after:  Jul 27 10:10:41 2021 GMT
```

6.5.3.3 SSL-scan coronamelder-productie.coronamelder-dist.nl

```
Version: 1.11.13-static
OpenSSL 1.0.2-chacha (1.0.2g-dev)

Connected to 195.121.65.164

Testing SSL server productie.coronamelder-dist.nl on port 443 using SNI name
productie.coronamelder-dist.nl

  TLS Fallback SCSV:
Server supports TLS Fallback SCSV

  TLS renegotiation:
Secure session renegotiation supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.2 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.0 not vulnerable to heartbleed

  Supported Server Cipher(s):
Preferred TLSv1.2  128 bits  ECDHE-RSA-AES128-GCM-SHA256   Curve P-256 DHE
256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-GCM-SHA384   Curve P-256 DHE
256
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-SHA256       Curve P-256 DHE
256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-SHA384       Curve P-256 DHE
256
Accepted  TLSv1.2  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE
256
Accepted  TLSv1.2  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE
256
Accepted  TLSv1.2  128 bits  AES128-GCM-SHA256
Accepted  TLSv1.2  256 bits  AES256-GCM-SHA384
Accepted  TLSv1.2  128 bits  AES128-SHA256
Accepted  TLSv1.2  256 bits  AES256-SHA256
Accepted  TLSv1.2  128 bits  AES128-SHA
Accepted  TLSv1.2  256 bits  AES256-SHA
Preferred TLSv1.1  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE
256
Accepted  TLSv1.1  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE
256
Accepted  TLSv1.1  128 bits  AES128-SHA
Accepted  TLSv1.1  256 bits  AES256-SHA
Preferred TLSv1.0  128 bits  ECDHE-RSA-AES128-SHA          Curve P-256 DHE
256
Accepted  TLSv1.0  256 bits  ECDHE-RSA-AES256-SHA          Curve P-256 DHE
256
Accepted  TLSv1.0  128 bits  AES128-SHA
Accepted  TLSv1.0  256 bits  AES256-SHA

  SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
RSA Key Strength:    2048

Subject:  productie.coronamelder-dist.nl
Altnames: DNS:productie.coronamelder-dist.nl
Issuer:   KPN BV PKIoverheid Organisatie Server CA - G3

Not valid before: Jul 29 08:40:15 2020 GMT
Not valid after:  Jul 29 08:40:15 2021 GMT
```

6.5.4 HTTP Security Headers

Voor de domeinen van de CoronaMelder-omgeving, welke in scope waren tijdens het onderzoek, zijn de http-security headers gecontroleerd. Hieronder staat een schematische weergave voor welke domeinen de headers ingesteld zijn:

✓= Geen kwetsbaarheid aangetroffen, ✗ = Kwetsbaarheid gevonden, N/a = Niet van toepassing

URL / IP-adres	HTTP Strict-Transport-Security	Content-Security-Policy	X-Frame-Options	X-Content-Type-Options	Referrer-Policy	Feature-Policy
https://coronamelder-portal.nl (82.201.43.164)	✓	✗	✗	✗	✗	✗
https://coronamelder-api.nl (82.201.43.165)	✓	✗	✗	✗	✗	✗
https://productie.coronamelder-dist.nl (195.121.65.164)	✗	✗	✗	✗	✗	✗

Tabel 23 - HTTP Security Headers

6.6 ExposureNotification Framework – Apple iOS

6.6.1 Situatieschets

Voor het testen van de door opdrachtgever verstrekte mobiele applicatie (CoronaMelder) is gekeken naar het ExposureNotification Framework. Daarbij is gebruik gemaakt van echte mobiele telefoons (Apple iPhones, 7 en X) als testtelefoons.

Tijdens de periode van de penetratietest is gekeken naar mogelijke artefacten die gegeneerd worden op bestandssysteem niveau en die (mogelijk) te relateren zijn aan het gebruik van het ExposureNotification-framework.

Om dit te bereiken is de verstrekte versie van de mobiele applicatie geïnstalleerd en geactiveerd op de testtelefoons. De beschreven artefacten zijn waargenomen in iOS **versie 13.6** (vrijgegeven door Apple op 15 juli 2020).

6.6.2 Waargenomen werking - Opslag

Middels het SSH-protocol is ingelogd op de testtelefoon, waarna met het commando 'lsof'[10] is gekeken naar de op de telefoon actief geraadpleegde bestanden met de zoekterm 'ExposureNotification':

```
lsof | grep 'ExposureNotification'
```

Daarbij is waargenomen dat het uitvoerbare bestand `'/usr/bin/bluetoothd'` de volgende mappen raadpleegde:

- `/private/var/mobile/Library/ExposureNotification/Exposure/`
  `/private/var/mobile/Library/ExposureNotification/Advertisements/`
- `/private/var/containers/Shared/SystemGroup/56ADA82B-9837-475C-BF5B-9DD8D7E40EBE/Library/ExposureNotification/`

Binnen deze mappen werden de volgende bestanden waargenomen als in gebruik door de het bovengenoemde uitvoerbare bestand:

- `en_advertisements.db`
- `en_exposure.sqlite`
- `DetectionHistory.plist`

[10] https://linux.die.net/man/8/lsof

Advertisements database (RPI)

Op basis van de bovenstaande reeks bestanden die door het uitvoerbare bestand 'bluetoothd' werden benaderd, is het volgende bestand aangetroffen op het bestandssysteem:

Bestandsnaam	en_advertisements.db
Databasestructuur	SQLLite
Bestandslocatie	/private/var/mobile/Library/ExposureNotification/Advertisements/

Tabel 24 - Bestandsinformatie (en_advertisements.db)

Bij het inladen van het SQLLite bestand is de volgende tabel waargenomen:

SQLLite tabel: en_advertisements

Sleutel	Type	Vermoedelijke betekenis
rpi	BLOB	Rolling Proximity Identifier (RPI)
encrypted_aem	BLOB	Associated Encrypted Metadata (encrypted)
timestamp	INTEGER	Timestamp (Linux epoch)
scan_interval	INTEGER	Scan interval
rssi	INTEGER	Received signal strength indication
max_rssi	INTEGER	Received signal strength indication (max value)
saturated	BOOLEAN	Onbekend
counter	INTEGER	Onbekend

Tabel 25 - Tabelstructuur en_advertisements

Daarbij zijn de volgende waardes waargenomen in de database:

	rpi	encrypted_aem	timestamp	scan_interval	rssi	max_rssi	saturated	counter
1	BLOB	BLOB	1595257168	305	-77	-70	0	17
2	BLOB	BLOB	1595257473	304	-77	-70	0	15
3	BLOB	BLOB	1595257778	304	-78	-69	0	17
4	BLOB	BLOB	1595258075	296	-77	-69	0	21
5	BLOB	BLOB	1595253227	304	-38	-38	0	9

Figuur 19 - Waargenomen informatie (SQLLite database)

Exposure database

Op basis van de bovenstaande reeks bestanden die door het uitvoerbare bestand 'bluetoothd' werden benaderd, is het volgende bestand aangetroffen op het bestandssysteem:

Bestandsnaam	en_exposure.sqlite
Databasestructuur	SQLLite
Bestandslocatie	/private/var/mobile/Library/ExposureNotification/Exposure/

Tabel 26 - en_exposure.sqlite file

Bij het inladen van het SQLLite bestand zijn de volgende tabellen waargenomen:

1. SQLLite tabel: advertisements

Sleutel	Type	Vermoedelijke betekenis
rpi	BLOB	Rolling Proximity Identifier (RPI)
encrypted_aem	BLOB	Associated Encrypted Metadata (encrypted)
timestamp	INTEGER	Timestamp (Linux epoch)
scan_interval	INTEGER	Scan interval
rssi	INTEGER	Received signal strength indication
max_rssi	INTEGER	Received signal strength indication (max value)
saturated	BOOLEAN	Onbekend
counted	INTEGER	Onbekend
tek_id	INTEGER	Onbekend

Tabel 27 - Tabelstructuur advertisements

2. SQLLite tabel: kvs

Sleutel	Type	Vermoedelijke betekenis
ROWID	INTEGER	Row-id
key	TEXT	Onbekend
value		Onbekend
type	INTEGER	Onbekend
mod_date	REAL	Vermoedelijk: modificatie datum
expiration_date	REAL	Vermoedelijk: verloop datum

Tabel 28 - Tabelstructuur kvs

3. SQLLite tabel: teks

Sleutel	Type	Vermoedelijke betekenis
ROWID	INTEGER	Row-id
region_id	TEXT	Vermoedelijk: Regio-ID
app_bundle_id	TEXT	Vermoedelijk: Applicatie identifier[11]
key	BLOB	Onbekend
start	INTEGER	Onbekend
period	INTEGER	Onbekend
end	INTEGER	Onbekend
onset_days	INTEGER	Onbekend
report_type	INTEGER	Onbekend
transmission_risk	INTEGER	Onbekend

Tabel 29 - Tabelstructuur teks

[11] https://pspdfkit.com/guides/ios/current/faq/finding-the-app-bundle-id/

Detection History

Op basis van de bovenstaande reeks bestanden die door het uitvoerbare bestand 'bluetoothd' werden benaderd, is het volgende bestand geïdentificeerd op het bestandssysteem:

Bestand: DetectionHistory.plist

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<array>
	<dict>
		<key>time</key>
		<real>616915930.742652</real>
		<key>hash</key>
		<data>
		kktyPo7CvsCTRikX0c4rzncV136cwY5r7Gu7SalWILs=
		</data>
		<key>matchCount</key>
		<integer>0</integer>
		<key>keyCount</key>
		<integer>3</integer>
		<key>bundleID</key>
		<string>nl.rijksoverheid.en</string>
	</dict>
[...]
```

6.6.3 Waargenomen werking – Bluetooth Low Energy (BLE)

Met het gebruik van gespecialiseerde hardware (Nuand BladeRF revision 2) – is gekeken naar de Bluetooth Low Energy (BLE)-signalen welke na installatie en activatie van de door opdrachtgever verstrekte applicatie worden verstuurd.

Daarbij is een Nuand BladeRF aangesloten op onderzoeksapparatuur via het USB-protocol (versie 3.0), waarbij het apparaat vervolgens geconfigureerd is en gereed is gemaakt voor gebruik.

Waargenomen BLE-advertisements

Vervolgens is de applicatie BTLE_RX gecompileerd[12] en is het volgende commando uitgevoerd:

```
btle_rx > iPhone_7_advertisements.txt
```

Gebruikmakend van de applicatie 'btle_rx' zijn met de standaard instellingen (45dB) op een afstand van 2 meter in een periode van ongeveer 15 minuten 1859 BLE-advertisements van de testtelefoon ontvangen welke deels of volledig aan het formaat voldoen van een ExposureNotification-advertisement, waarvan 253 als invalide gerapporteerd zijn door de applicatie (CRC check).

Vervolgens zijn op een afstand van 4 meter in een periode van ongeveer 15 minuten 595 BLE-advertisements van de testtelefoon ontvangen die deels of volledig aan het formaat voldoen van een ExposureNotification-advertisement, waarvan 142 invalide geclassificeerd zijn (CRC check).

Vanaf het moment van het activeren van de Bluetooth functionaliteit op de telefoon werden door de testtelefoon zogenoemde 'advertisements' verstuurd waarbij de willekeurige alfanumerieke waarde '`029b71f3b69f`' ingesteld werd als het Advertiser -Address (AdvA):

Ch[13]	AA[14]	ADV[15]	AdvA[16]	Data
37	8e89bed6	ADV_NONCONN_IND	029b71f3b69f	02011a03036ffd17166ffd77f71cb9b51406e854b759f1531f1a68176753d5

Tabel 30 - Voorbeeld verstuurde advertisements

Na een periode van ongeveer 15 minuten werd een ander Advertiser Address (AdvA) waargenomen, de alfanumerieke waarde '`223f47158ec8`':

Ch	AA	ADV	AdvA	Data
37	8e89bed6	ADV_NONCONN_IND	223f47158ec8	02011a03036ffd17166ffdd797aeb1841993c68b116775c3ab59c601d55fed

Tabel 31 - Vervolg voorbeeld verstuurde advertisements

Het formaat van de waargenomen data lijkt overeen te komen met de publiek gepubliceerde Bluetooth standaard voor ExposureNotifications[17].

[12] https://github.com/JiaoXianjun/BTLE

[13] Channel

[14] Access Address

[15] Bluetooth Core Specification 5.2, page 2874, paragraph 2.3.1.3 - ADV_NONCONN_IND - https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=478726

[16] Advertisers Address

[17] https://covid19-static.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-BluetoothSpecificationv1.2.pdf?

6.7 ExposureNotification Framework – Android

6.7.1 Situatieschets

Voor het testen van de door opdrachtgever verstrekte mobiele applicatie (CoronaMelder) is gekeken naar het ExposureNotification Framework. Daarbij is gebruik gemaakt van mobiele telefoons (Samsung S5 en LG Nexus) als testtelefoons.

Tijdens de periode van de penetratietest is gekeken naar mogelijke artefacten welke gegeneerd worden op bestandssysteem niveau en welke (mogelijk) te relateren zijn aan het gebruik van het ExposureNotification-framework.

Om dit te bereiken is de verstrekte versie van de mobiele applicatie geïnstalleerd en geactiveerd op de testtelefoons. De beschreven artefacten zijn waargenomen in Android-versie 10 met Google Play Services versie 20200714 (OpenGApps).

6.7.2 Waargenomen werking - Opslag

Middels het SSH-protocol is ingelogd op de testtelefoon, waarna met het commando 'lsof'[18] is gekeken naar de op de telefoon actief geraadpleegde bestanden met de zoekterm 'app_contact':

```
lsof | grep 'app_contact'
```

Daarbij is waargenomen dat het de applicatie '`com.google.android.gms.persistent`' de volgende mappen geraadpleegd werden:

- `/data/data/com.google.android.gms/app_contact-tracing-contact-record-db/`
- `/data/data/com.google.android.gms/app_contact-tracing-self-tracing-key-db/`

Met publiek beschikbare tooling is het niet gelukt om de verschillende aanwezige bestanden in te lezen.

[18] https://linux.die.net/man/8/lsof

6.7.3 Waargenomen werking – Bluetooth Low Energy (BLE)

Met het gebruik van gespecialiseerde hardware (Nuand BladeRF revision 2) – is gekeken naar de Bluetooth Low Energy (BLE)-signalen die na installatie en activatie van de door opdrachtgever verstrekte applicatie worden verstuurd.

Daarbij is een Nuand BladeRF aangesloten op onderzoeksapparatuur via het USB-protocol (versie 3.0), waarbij het apparaat vervolgens geconfigureerd is en gereed is gemaakt voor gebruik.

Waargenomen BLE-advertisements

Vervolgens is de applicatie BTLE_RX gecompileerd[19] en is het volgende commando uitgevoerd:

```
btle_rx > Android_advertisements.txt
```

Gebruikmakend van de applicatie 'btle_rx' zijn met de standaard instellingen (45dB) op een afstand van 2 meter in een periode van ongeveer 15 minuten 1306 BLE-advertisements van de testtelefoon ontvangen die deels of volledig aan het formaat voldoen van een ExposureNotification-advertisement, waarvan 194 als invalide gerapporteerd zijn door de applicatie (CRC check).

Vervolgens zijn op een afstand van 4 meter in een periode van ongeveer 15 minuten 646 LE-advertisements van de testtelefoon ontvangen welke deels of volledig aan het formaat voldoen van een ExposureNotification-advertisement, waarvan 418 invalide geclassificeerd zijn (CRC check).

De willekeurige alfanumerieke waarde '`42b5c85e43dc`' werd waargenomen als het Advertiser-Address (AdvA):

Ch[20]	AA[21]	ADV	AdvA[22]	Data
38	8e89bed6	ADV_SCAN_IND	42b5c85e43dc	03036ffd17166ffd065cc573aa4dbd4908ffab3eb59cc680f5debff8

Tabel 32 - Voorbeeld verstuurde advertisements

Na een periode van ongeveer 15 minuten werd een ander Advertiser Address (AdvA) waargenomen, de alfanumerieke waarde '`515ad24b5377`':

Ch	AA	ADV	AdvA	Data
38	8e89bed6	ADV_SCAN_IND	515ad24b5377	03036ffd17166ffdf5875b992c1d5e8e78fcbad7d69805c0ecebdaff

Tabel 33 - Vervolgvoorbeeld verstuurde advertisements

Het formaat van de waargenomen data lijkt overeen te komen met de publiek gepubliceerde Bluetooth standaard voor ExposureNotifications[23].

[19] https://github.com/JiaoXianjun/BTLE

[20] Channel

[21] Access Address

[22] Advertiser Address

[23] https://covid19-static.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-BluetoothSpecificationv1.2.pdf?

6.8 OWASP WSTG v4.1 – Checklist

✓= Geen kwetsbaarheid aangetroffen, ✗ = Kwetsbaarheid gevonden, N/a = Niet van toepassing

WSTG-ID	Information Gathering	Resultaat
WSTG-INFO-01	Conduct Search Engine Discovery and Reconnaissance for Information Leakage	✗
WSTG-INFO-02	Fingerprint Web Server	✗
WSTG-INFO-03	Review Webserver Metafiles for Information Leakage	✓
WSTG-INFO-04	Enumerate Applications on Webserver	✓
WSTG-INFO-05	Review Webpage Comments and Metadata for Information Leakage	✓
WSTG-INFO-06	Identify Application Entry Points	✓
WSTG-INFO-07	Map Execution Paths Through Application	✓
WSTG-INFO-09	Fingerprint Web Application Framework	✓
WSTG-INFO-09	Fingerprint Web Application	✓
WSTG-INFO-10	Map Application Architecture	✗

WSTG-ID	Configuration and Deploy Management Testing	Resultaat
WSTG-CONF-01	Test Network Infrastructure Configuration	✓
WSTG-CONF-02	Test Application Platform Configuration	✓
WSTG-CONF-03	Test File Extensions Handling for Sensitive Information	✓
WSTG-CONF-04	Backup and Unreferenced Files for Sensitive Information	✓
WSTG-CONF-05	Enumerate Infrastructure and Application Admin Interfaces	✓
WSTG-CONF-06	Test HTTP Methods	✓
WSTG-CONF-07	Test HTTP Strict Transport Security	✗
WSTG-CONF-08	Test RIA Cross Domain Policy	N/a
WSTG-CONF-09	Test File Permission	N/a
WSTG-CONF-10	Test for Subdomain Takeover	✓
WSTG-CONF-11	Test Cloud Storage	N/a

WSTG-ID	Identity Management Testing	Resultaat
WSTG-IDNT-01	Test Role Definitions	N/a
WSTG-IDNT-02	Test User Registration Process	N/a
WSTG-IDNT-03	Test Account Provisioning Process	✓
WSTG-IDNT-04	Testing for Account Enumeration and Guessable User Account	✓
WSTG-IDNT-05	Testing for Weak or Unenforced Username Policy	✓

WSTG-ID	Authentication Testing	Resultaat
WSTG-ATHN-01	Testing for Credentials Transported over an Encrypted Channel	✓
WSTG-ATHN-02	Testing for Default Credentials	✓
WSTG-ATHN-03	Testing for Weak Lock Out Mechanism	✓
WSTG-ATHN-04	Testing for Bypassing Authentication Schema	✓
WSTG-ATHN-05	Testing for Vulnerable Remember Password	N/a
WSTG-ATHN-06	Testing for Browser Cache Weakness	✓
WSTG-ATHN-07	Testing for Weak Password Policy	✓
WSTG-ATHN-08	Testing for Weak Security Question Answer	N/a
WSTG-ATHN-09	Testing for Weak Password Change or Reset Functionalities	N/a
WSTG-ATHN-10	Testing for Weaker Authentication in Alternative Channel	✓

WSTG-ID	Authorization Testing	Resultaat
WSTG-ATHZ-01	Testing Directory Traversal - File Include	✓
WSTG-ATHZ-02	Testing for Bypassing Authorization Schema	✓
WSTG-ATHZ-03	Testing for Privilege Escalation	N/a
WSTG-ATHZ-04	Testing for Insecure Direct Object References	✓

WSTG-ID	Session Management Testing	Resultaat
WSTG-SESS-01	Testing for Bypassing Session Management Schema	✓
WSTG-SESS-02	Testing for Cookies Attributes	✓
WSTG-SESS-03	Testing for Session Fixation	✓
WSTG-SESS-04	Testing for Exposed Session Variables	✓
WSTG-SESS-05	Testing for Cross Site Request Forgery	✓
WSTG-SESS-06	Testing for Logout Functionality	✗
WSTG-SESS-07	Test Session Timeout	✓
WSTG-SESS-08	Testing for Session Puzzling	✓

WSTG-ID	Input Validation Testing	Resultaat
WSTG-INPV-01	Testing for Reflected Cross Site Scripting	✓
WSTG-INPV-02	Testing for Stored Cross Site Scripting	✓
WSTG-INPV-03	Testing for HTTP Verb Tampering	✓
WSTG-INPV-04	Testing for HTTP Parameter pollution	✓
WSTG-INPV-05	Testing for SQL Injection	✓
WSTG-INPV-06	Testing for LDAP Injection	N/a
WSTG-INPV-07	Testing for XML Injection	N/a
WSTG-INPV-08	Testing for SSI Injection	N/a
WSTG-INPV-09	Testing for XPath Injection	N/a
WSTG-INPV-10	IMAP/SMTP Injection	N/a
WSTG-INPV-11	Testing for Code Injection	✓
WSTG-INPV-12	Testing for Command Injection	✓
WSTG-INPV-13	Testing for Buffer overflow	✓
WSTG-INPV-14	Testing for Incubated Vulnerabilities	✓

WSTG-INPV-15	Testing for HTTP Splitting/Smuggling	✓
WSTG-INPV-16	Testing for HTTP Incoming Requests	✓
WSTG-INPV-17	Testing for Host Header Injection	✓
WSTG-INPV-18	Testing for Server Side Template Injection	N/a

WSTG-ID	Error Handling	Resultaat
WSTG-ERRH-01	Analysis of Error Codes	✓
WSTG-ERRH-02	Analysis of Stack Traces	✓

WSTG-ID	Cryptography	Resultaat
WSTG-CRYP-01	Testing for Weak SSL TLS Ciphers Insufficient Transport Layer Protection	✗
WSTG-CRYP-02	Testing for Padding Oracle	✓
WSTG-CRYP-03	Testing for Sensitive Information Sent Via Unencrypted Channels	✗
WSTG-CRYP-04	Testing for Weak Encryption	✓

WSTG-ID	Business Logic Testing	Resultaat
WSTG-BUSL-01	Test Business Logic Data Validation	✗
WSTG-BUSL-02	Test Ability to Forge Requests	✓
WSTG-BUSL-03	Test Integrity Checks	✓
WSTG-BUSL-04	Test for Process Timing	✓
WSTG-BUSL-05	Test Number of Times a Function Can be Used Limits	N/a
WSTG-BUSL-06	Testing for the Circumvention of Work Flows	✓
WSTG-BUSL-07	Test Defenses Against Application Misuse	✓
WSTG-BUSL-08	Test Upload of Unexpected File Types	N/a
WSTG-BUSL-09	Test Upload of Malicious Files	✓

WSTG-ID	Client Side Testing	Resultaat
WSTG-CLNT-01	Testing for DOM based Cross Site Scripting	✓
WSTG-CLNT-02	Testing for JavaScript Execution	✓
WSTG-CLNT-03	Testing for HTML Injection	✓
WSTG-CLNT-04	Testing for Client Side URL Redirect	✓
WSTG-CLNT-05	Testing for CSS Injection	✓
WSTG-CLNT-06	Testing for Client Side Resource Manipulation	✓
WSTG-CLNT-07	Test Cross Origin Resource Sharing	✗
WSTG-CLNT-08	Testing for Cross Site Flashing	✓
WSTG-CLNT-09	Testing for Clickjacking	✓
WSTG-CLNT-10	Testing WebSockets	N/a
WSTG-CLNT-11	Test Web Messaging	N/a
WSTG-CLNT-12	Test Local Storage	✓
WSTG-CLNT-13	Testing for Cross Site Script Inclusion	✓

6.9 iOS - OWASP MSTG v1.1.3.1 - Checklist

6.9.1 Mobile Application Security Requirements

✓= Geen kwetsbaarheid aangetroffen, ✗ = Kwetsbaarheid gevonden, N/a = Niet van toepassing

MSTG-ID	1. Data Storage and Privacy	Resultaat
MSTG-STORAGE-1	System credential storage facilities need to be used to store sensitive data, such as PII, user credentials or cryptographic keys.	✓
MSTG-STORAGE-2	No sensitive data should be stored outside of the app container or system credential storage facilities.	✓
MSTG-STORAGE-3	No sensitive data is written to application logs.	✓
MSTG-STORAGE-4	No sensitive data is shared with third parties unless it is a necessary part of the architecture.	✓
MSTG-STORAGE-5	The keyboard cache is disabled on text inputs that process sensitive data.	N/a
MSTG-STORAGE-6	No sensitive data is exposed via IPC mechanisms.	✓
MSTG-STORAGE-7	No sensitive data, such as passwords or pins, is exposed through the user interface.	✓
MSTG-STORAGE-8	No sensitive data is included in backups generated by the mobile operating system.	✓
MSTG-STORAGE-9	The app removes sensitive data from views when moved to the background.	✗
MSTG-STORAGE-10	The app does not hold sensitive data in memory longer than necessary, and memory is cleared explicitly after use.	✓
MSTG-STORAGE-11	The app enforces a minimum device-access-security policy, such as requiring the user to set a device passcode.	N/a
MSTG-STORAGE-12	The app educates the user about the types of personally identifiable information processed, as well as security best practices the user should follow in using the app.	✓
MSTG-STORAGE-13	No sensitive data should be stored locally on the mobile device. Instead, data should be retrieved from a remote endpoint when needed and only be kept in memory.	N/a
MSTG-STORAGE-14	If sensitive data is still required to be stored locally, it should be encrypted using a key derived from hardware backed storage which requires authentication.	✓

MSTG-STORAGE-15	The app's local storage should be wiped after an excessive number of failed authentication attempts.	N/a

MSTG-ID	Cryptography	Resultaat
MSTG-CRYPTO-1	The app does not rely on symmetric cryptography with hardcoded keys as a sole method of encryption.	✓
MSTG-CRYPTO-2	The app uses proven implementations of cryptographic primitives.	✓
MSTG-CRYPTO-3	The app uses cryptographic primitives that are appropriate for the particular use-case, configured with parameters that adhere to industry best practices.	N/a
MSTG-CRYPTO-4	The app does not use cryptographic protocols or algorithms that are widely considered deprecated for security purposes.	✓
MSTG-CRYPTO-5	The app doesn't re-use the same cryptographic key for multiple purposes.	N/a
MSTG-CRYPTO-6	All random values are generated using a sufficiently secure random number generator.	✓

MSTG-ID	Authentication and Session Management	Resultaat
MSTG-AUTH-1	If the app provides users access to a remote service, some form of authentication, such as username/password authentication, is performed at the remote endpoint.	N/a
MSTG-AUTH-2	If stateful session management is used, the remote endpoint uses randomly generated session identifiers to authenticate client requests without sending the user's credentials.	N/a
MSTG-AUTH-3	If stateless token-based authentication is used, the server provides a token that has been signed using a secure algorithm.	N/a
MSTG-AUTH-4	The remote endpoint terminates the existing session when the user logs out.	N/a
MSTG-AUTH-5	A password policy exists and is enforced at the remote endpoint.	N/a
MSTG-AUTH-6	The remote endpoint implements a mechanism to protect against the submission of credentials an excessive number of times.	N/a

MSTG-AUTH-7	Sessions are invalidated at the remote endpoint after a predefined period of inactivity and access tokens expire.	N/a
MSTG-AUTH-8	Biometric authentication, if any, is not event-bound (i.e. using an API that simply returns "true" or "false"). Instead, it is based on unlocking the keychain/keystore.	N/a
MSTG-AUTH-9	A second factor of authentication exists at the remote endpoint and the 2FA requirement is consistently enforced.	N/a
MSTG-AUTH-10	Sensitive transactions require step-up authentication.	N/a
MSTG-AUTH-11	The app informs the user of all sensitive activities with their account. Users are able to view a list of devices, view contextual information (IP address, location, etc.), and to block specific devices.	N/a
MSTG-AUTH-12	Authorization models should be defined and enforced at the remote endpoint.	✓

MSTG-ID	Network Communication	Resultaat
MSTG-NETWORK-1	Data is encrypted on the network using TLS. The secure channel is used consistently throughout the app.	✓
MSTG-NETWORK-2	The TLS settings are in line with current best practices, or as close as possible if the mobile operating system does not support the recommended standards.	✓
MSTG-NETWORK-3	The app verifies the X.509 certificate of the remote endpoint when the secure channel is established. Only certificates signed by a trusted CA are accepted.	✓
MSTG-NETWORK-4	The app either uses its own certificate store, or pins the endpoint certificate or public key, and subsequently does not establish connections with endpoints that offer a different certificate or key, even if signed by a trusted CA.	✓
MSTG-NETWORK-5	The app doesn't rely on a single insecure communication channel (email or SMS) for critical operations, such as enrollments and account recovery.	N/a
MSTG-NETWORK-6	The app only depends on up-to-date connectivity and security libraries.	✗

MSTG-ID	Platform Interaction	Resultaat
MSTG-PLATFORM-1	The app only requests the minimum set of permissions necessary.	✓
MSTG-PLATFORM-2	All inputs from external sources and the user are validated and if necessary sanitized. This includes data received via the UI, IPC mechanisms such as intents, custom URLs, and network sources.	N/a
MSTG-PLATFORM-3	The app does not export sensitive functionality via custom URL schemes, unless these mechanisms are properly protected.	✓
MSTG-PLATFORM-4	The app does not export sensitive functionality through IPC facilities, unless these mechanisms are properly protected.	N/a
MSTG-PLATFORM-5	JavaScript is disabled in WebViews unless explicitly required.	N/a
MSTG-PLATFORM-6	WebViews are configured to allow only the minimum set of protocol handlers required (ideally, only https is supported). Potentially dangerous handlers, such as file, tel and app-id, are disabled.	N/a
MSTG-PLATFORM-7	If native methods of the app are exposed to a WebView, verify that the WebView only renders JavaScript contained within the app package.	N/a
MSTG-PLATFORM-8	Object deserialization, if any, is implemented using safe serialization APIs.	✓
MSTG-PLATFORM-9	The app protects itself against screen overlay attacks. (Android only)	N/a
MSTG-PLATFORM-10	A WebView's cache, storage, and loaded resources (JavaScript, etc.) should be cleared before the WebView is destroyed.	✗
MSTG-PLATFORM-11	Verify that the app prevents usage of custom third-party keyboards whenever sensitive data is entered.	N/a

MSTG-ID	Code Quality and Build Settings	Resultaat
MSTG-CODE-1	The app is signed and provisioned with a valid certificate, of which the private key is properly protected.	✓
MSTG-CODE-2	The app has been built in release mode, with settings appropriate for a release build (e.g. non-debuggable).	✓
MSTG-CODE-3	Debugging symbols have been removed from native binaries.	N/a
MSTG-CODE-4	Debugging code and developer assistance code (e.g. test code, backdoors, hidden settings) have been removed. The app does not log verbose errors or debugging messages.	N/a
MSTG-CODE-5	All third-party components used by the mobile app, such as libraries and frameworks, are identified, and checked for known vulnerabilities.	N/a
MSTG-CODE-6	The app catches and handles possible exceptions.	N/a
MSTG-CODE-7	Error handling logic in security controls denies access by default.	N/a
MSTG-CODE-8	In unmanaged code, memory is allocated, freed and used securely.	N/a
MSTG-CODE-9	Free security features offered by the toolchain, such as bytecode minification, stack protection, PIE support and automatic reference counting, are activated.	✓

6.9.2 Resiliency Against Reverse Engineering

MSTG-ID	Impede Dynamic Analysis and Tampering	Resultaat
MSTG-RESILIENCE-1	The app detects, and responds to, the presence of a rooted or jailbroken device either by alerting the user or terminating the app.	✗
MSTG-RESILIENCE-2	The app prevents debugging and/or detects, and responds to, a debugger being attached. All available debugging protocols must be covered.	N/a
MSTG-RESILIENCE-3	The app detects, and responds to, tampering with executable files and critical data within its own sandbox.	✓
MSTG-RESILIENCE-4	The app detects, and responds to, the presence of widely used reverse engineering tools and frameworks on the device.	✗
MSTG-RESILIENCE-5	The app detects, and responds to, being run in an emulator.	✗
MSTG-RESILIENCE-6	The app detects, and responds to, tampering the code and data in its own memory space.	N/a
MSTG-RESILIENCE-7	The app implements multiple mechanisms in each defense category (8.1 to 8.6). Note that resiliency scales with the amount, diversity of the originality of the mechanisms used.	N/a
MSTG-RESILIENCE-8	The detection mechanisms trigger responses of different types, including delayed and stealthy responses.	✗
MSTG-RESILIENCE-9	Obfuscation is applied to programmatic defenses, which in turn impede de-obfuscation via dynamic analysis.	✓

MSTG-ID	Device Binding	Resultaat
MSTG-RESILIENCE-10	The app implements a 'device binding' functionality using a device fingerprint derived from multiple properties unique to the device.	N/a

MSTG-ID	Impede Comprehension	Resultaat
MSTG-RESILIENCE-11	All executable files and libraries belonging to the app are either encrypted on the file level and/or important code and data segments inside the executables are encrypted or packed. Trivial static analysis does not reveal important code or data.	✓
MSTG-RESILIENCE-12	If the goal of obfuscation is to protect sensitive computations, an obfuscation scheme is used that is both appropriate for the particular task and robust against manual and automated de-obfuscation methods, considering currently published research. The effectiveness of the obfuscation scheme must be verified through manual testing. Note that hardware-based isolation features are preferred over obfuscation whenever possible.	✓

MSTG-ID	Device Binding	Resultaat
MSTG-RESILIENCE-13	As a defense in depth, next to having solid hardening of the communicating parties, application level payload encryption can be applied to further impede eavesdropping.	✓

6.10 Android - OWASP MSTG v1.1.3.1 - Checklist

6.10.1 Mobile Application Security Requirements

✓= Geen kwetsbaarheid aangetroffen, ✗ = Kwetsbaarheid gevonden, N/a = Niet van toepassing

MSTG-ID	2. Data Storage and Privacy	Resultaat
MSTG-STORAGE-1	System credential storage facilities need to be used to store sensitive data, such as PII, user credentials or cryptographic keys.	✓
MSTG-STORAGE-2	No sensitive data should be stored outside of the app container or system credential storage facilities.	✓
MSTG-STORAGE-3	No sensitive data is written to application logs.	✓
MSTG-STORAGE-4	No sensitive data is shared with third parties unless it is a necessary part of the architecture.	✓
MSTG-STORAGE-5	The keyboard cache is disabled on text inputs that process sensitive data.	N/a
MSTG-STORAGE-6	No sensitive data is exposed via IPC mechanisms.	N/a
MSTG-STORAGE-7	No sensitive data, such as passwords or pins, is exposed through the user interface.	✓
MSTG-STORAGE-8	No sensitive data is included in backups generated by the mobile operating system.	✓
MSTG-STORAGE-9	The app removes sensitive data from views when moved to the background.	✗
MSTG-STORAGE-10	The app does not hold sensitive data in memory longer than necessary, and memory is cleared explicitly after use.	✓
MSTG-STORAGE-11	The app enforces a minimum device-access-security policy, such as requiring the user to set a device passcode.	✗
MSTG-STORAGE-12	The app educates the user about the types of personally identifiable information processed, as well as security best practices the user should follow in using the app.	✓
MSTG-STORAGE-13	No sensitive data should be stored locally on the mobile device. Instead, data should be retrieved from a remote endpoint when needed and only be kept in memory.	N/a
MSTG-STORAGE-14	If sensitive data is still required to be stored locally, it should be encrypted using a key derived from hardware backed storage which requires authentication.	✓

MSTG-STORAGE-15	The app's local storage should be wiped after an excessive number of failed authentication attempts.	N/a

MSTG-ID	Cryptography	Resultaat
MSTG-CRYPTO-1	The app does not rely on symmetric cryptography with hardcoded keys as a sole method of encryption.	✓
MSTG-CRYPTO-2	The app uses proven implementations of cryptographic primitives.	✓
MSTG-CRYPTO-3	The app uses cryptographic primitives that are appropriate for the particular use-case, configured with parameters that adhere to industry best practices.	N/a
MSTG-CRYPTO-4	The app does not use cryptographic protocols or algorithms that are widely considered deprecated for security purposes.	✓
MSTG-CRYPTO-5	The app doesn't re-use the same cryptographic key for multiple purposes.	N/a
MSTG-CRYPTO-6	All random values are generated using a sufficiently secure random number generator.	✓

MSTG-ID	Authentication and Session Management	Resultaat
MSTG-AUTH-1	If the app provides users access to a remote service, some form of authentication, such as username/password authentication, is performed at the remote endpoint.	N/a
MSTG-AUTH-2	If stateful session management is used, the remote endpoint uses randomly generated session identifiers to authenticate client requests without sending the user's credentials.	N/a
MSTG-AUTH-3	If stateless token-based authentication is used, the server provides a token that has been signed using a secure algorithm.	N/a
MSTG-AUTH-4	The remote endpoint terminates the existing session when the user logs out.	N/a
MSTG-AUTH-5	A password policy exists and is enforced at the remote endpoint.	N/a
MSTG-AUTH-6	The remote endpoint implements a mechanism to protect against the submission of credentials an excessive number of times.	N/a

MSTG-AUTH-7	Sessions are invalidated at the remote endpoint after a predefined period of inactivity and access tokens expire.	N/a
MSTG-AUTH-8	Biometric authentication, if any, is not event-bound (i.e. using an API that simply returns "true" or "false"). Instead, it is based on unlocking the keychain/keystore.	N/a
MSTG-AUTH-9	A second factor of authentication exists at the remote endpoint and the 2FA requirement is consistently enforced.	N/a
MSTG-AUTH-10	Sensitive transactions require step-up authentication.	N/a
MSTG-AUTH-11	The app informs the user of all sensitive activities with their account. Users are able to view a list of devices, view contextual information (IP address, location, etc.), and to block specific devices.	N/a
MSTG-AUTH-12	Authorization models should be defined and enforced at the remote endpoint.	✓

MSTG-ID	Network Communication	Resultaat
MSTG-NETWORK-1	Data is encrypted on the network using TLS. The secure channel is used consistently throughout the app.	✓
MSTG-NETWORK-2	The TLS settings are in line with current best practices, or as close as possible if the mobile operating system does not support the recommended standards.	✓
MSTG-NETWORK-3	The app verifies the X.509 certificate of the remote endpoint when the secure channel is established. Only certificates signed by a trusted CA are accepted.	✓
MSTG-NETWORK-4	The app either uses its own certificate store, or pins the endpoint certificate or public key, and subsequently does not establish connections with endpoints that offer a different certificate or key, even if signed by a trusted CA.	✓
MSTG-NETWORK-5	The app doesn't rely on a single insecure communication channel (email or SMS) for critical operations, such as enrollments and account recovery.	N/a
MSTG-NETWORK-6	The app only depends on up-to-date connectivity and security libraries.	✓

MSTG-ID	Platform Interaction	Resultaat
MSTG-PLATFORM-1	The app only requests the minimum set of permissions necessary.	✗
MSTG-PLATFORM-2	All inputs from external sources and the user are validated and if necessary sanitized. This includes data received via the UI, IPC mechanisms such as intents, custom URLs, and network sources.	N/a
MSTG-PLATFORM-3	The app does not export sensitive functionality via custom URL schemes, unless these mechanisms are properly protected.	✓
MSTG-PLATFORM-4	The app does not export sensitive functionality through IPC facilities, unless these mechanisms are properly protected.	✓
MSTG-PLATFORM-5	JavaScript is disabled in WebViews unless explicitly required.	N/a
MSTG-PLATFORM-6	WebViews are configured to allow only the minimum set of protocol handlers required (ideally, only https is supported). Potentially dangerous handlers, such as file, tel and app-id, are disabled.	N/a
MSTG-PLATFORM-7	If native methods of the app are exposed to a WebView, verify that the WebView only renders JavaScript contained within the app package.	N/a
MSTG-PLATFORM-8	Object deserialization, if any, is implemented using safe serialization APIs.	✓
MSTG-PLATFORM-9	The app protects itself against screen overlay attacks. (Android only)	N/a
MSTG-PLATFORM-10	A WebView's cache, storage, and loaded resources (JavaScript, etc.) should be cleared before the WebView is destroyed.	✗
MSTG-PLATFORM-11	Verify that the app prevents usage of custom third-party keyboards whenever sensitive data is entered.	N/a

MSTG-ID	Code Quality and Build Settings	Resultaat
MSTG-CODE-1	The app is signed and provisioned with a valid certificate, of which the private key is properly protected.	✓
MSTG-CODE-2	The app has been built in release mode, with settings appropriate for a release build (e.g. non-debuggable).	✓
MSTG-CODE-3	Debugging symbols have been removed from native binaries.	✓
MSTG-CODE-4	Debugging code and developer assistance code (e.g. test code, backdoors, hidden settings) have been removed. The app does not log verbose errors or debugging messages.	N/a
MSTG-CODE-5	All third party components used by the mobile app, such as libraries and frameworks, are identified, and checked for known vulnerabilities.	N/a
MSTG-CODE-6	The app catches and handles possible exceptions.	N/a
MSTG-CODE-7	Error handling logic in security controls denies access by default.	N/a
MSTG-CODE-8	In unmanaged code, memory is allocated, freed and used securely.	N/a
MSTG-CODE-9	Free security features offered by the toolchain, such as byte-code minification, stack protection, PIE support and automatic reference counting, are activated.	✓

6.10.2 Resiliency Against Reverse Engineering

MSTG-ID	Impede Dynamic Analysis and Tampering	Resultaat
MSTG-RESILIENCE-1	The app detects, and responds to, the presence of a rooted or jailbroken device either by alerting the user or terminating the app.	✗
MSTG-RESILIENCE-2	The app prevents debugging and/or detects, and responds to, a debugger being attached. All available debugging protocols must be covered.	N/a
MSTG-RESILIENCE-3	The app detects, and responds to, tampering with executable files and critical data within its own sandbox.	✓
MSTG-RESILIENCE-4	The app detects, and responds to, the presence of widely used reverse engineering tools and frameworks on the device.	✗
MSTG-RESILIENCE-5	The app detects, and responds to, being run in an emulator.	✗
MSTG-RESILIENCE-6	The app detects, and responds to, tampering the code and data in its own memory space.	N/a
MSTG-RESILIENCE-7	The app implements multiple mechanisms in each defense category (8.1 to 8.6). Note that resiliency scales with the amount, diversity of the originality of the mechanisms used.	✗
MSTG-RESILIENCE-8	The detection mechanisms trigger responses of different types, including delayed and stealthy responses.	✗
MSTG-RESILIENCE-9	Obfuscation is applied to programmatic defenses, which in turn impede de-obfuscation via dynamic analysis.	✓

MSTG-ID	Device Binding	Resultaat
MSTG-RESILIENCE-10	The app implements a 'device binding' functionality using a device fingerprint derived from multiple properties unique to the device.	N/a

MSTG-ID	Impede Comprehension	Resultaat
MSTG-RESILIENCE-11	All executable files and libraries belonging to the app are either encrypted on the file level and/or important code and data segments inside the executables are encrypted or packed. Trivial static analysis does not reveal important code or data.	N/a
MSTG-RESILIENCE-12	If the goal of obfuscation is to protect sensitive computations, an obfuscation scheme is used that is both appropriate for the particular task and robust against manual and automated de-obfuscation methods, considering currently published research. The effectiveness of the obfuscation scheme must be verified through manual testing. Note that hardware-based isolation features are preferred over obfuscation whenever possible.	N/a

MSTG-ID	Device Binding	Resultaat
MSTG-RESILIENCE-13	As a defense in depth, next to having solid hardening of the communicating parties, application level payload encryption can be applied to further impede eavesdropping.	N/a

NFIR B.V.
Verlengde Tolweg 2
2517 JV Den Haag
Telefoon: 088 - 323 02 05
info@nfir.nl
https://www.nfir.nl