Managementsamenvatting

Broncodeonderzoek DigiD-backend

in opdracht van

Logius

Logius-DigiD-backend-SUMMARY-v1.1.pdf
Auteurs: MV, TT, MK, RM
Versie 1.1
4 december 2023

Over Secura

Sinds 2000 helpt Secura bedrijven, (zorg-)organisaties en (lokale) overheden bij het identificeren, verminderen en voorkomen van IT-beveiligingsrisico's. Dit doen we door het uitvoeren van audits, waaronder hoogwaardige beveiligingsonderzoeken- en beoordelingen, consultancy, penetratietests en certificering van producten en diensten.

Sinds 2021 is Secura onderdeel van Bureau Veritas (BV), wat betekent dat Bureau Veritas de meerderheid van aandelen in bezit heeft. Bureau Veritas is een beursgenoteerde onderneming (Euronext: BVI), gespecialiseerd in testen, inspecteren en certificeren. Bureau Veritas is opgericht in 1828, heeft meer dan 75.000 werknemers en is aanwezig in 140 landen. Secura vormt de hoeksteen binnen de cyberbeveiligingsstrategie van Bureau Veritas.

Op Europees niveau draagt Secura bij aan de ECSO (European Cyber Security Organization) om internationale samenwerking en standaardisatie te bevorderen (ook gekoppeld aan de EU Cyber Security Act). Op nationaal niveau participeert Secura in Cyberveilig Nederland. Secura is als een van de weinige bedrijven door de Nederlandse overheid geaccrediteerd voor het Baseline Security Product Assessment (BSPA) schema. Secura heeft in 2021 als eerste bedrijf in Nederland het keurmerk ontvangen voor 'Centrum voor Criminaliteitspreventie en Veiligheid (CCV) gecertificeerd pentest bedrijf'. Dit betekent dat onze pentesten worden uitgevoerd door gekwalificeerd personeel via een gestructureerde aanpak wat resulteert in een helder leesbaar rapport.

Secura hecht ook waarde aan het delen van kennis en ervaring door onze klanten te helpen met effectieve trainingen. Bovendien bieden wij een programma (SAFE®) aan dat is gericht op het overbruggen van de kloof tussen bewustwording (awareness) en gedrag. Daarnaast ontwikkelen wij krachtige tools zoals de Secure File Exchange (SFE): een gebruiksvriendelijk platform om gevoelige informatie (zoals onze rapportages) op de meest veilige manier te delen.

Secura B.V.

Vestdijk 59
5611 CA EINDHOVEN
The Netherlands

Herikerbergweg 15
1101 CN AMSTERDAM
The Netherlands

T +31 (0)40 23 77 990

E info@secura.com

W https://www.secura.com

DOCUMENTMANAGEMENT

Reviewers

Naam	Functie	Datum	Versie
Ralph Moonen	Technical Director	2023-10-09	0.1

Wijzigingen

Versie	Datum	Initialen	Aanpassingen
0.1	2023-10-09	MK	Initiële versie
1.0	2023-10-10	MK	Definitieve versie
1.1	2023-12-04	MK	Feedback Logius verwerkt

INHOUDSOPGAVE

1. MANAGEMENTSAMENVATTING

In de periode van 31 juli tot en met 31 augustus 2023 heeft Secura op verzoek van Logius een onderzoek uitgevoerd op de broncode van de DigiD-backend. Dit onderzoek heeft zich gericht op de beveiligingsaspecten van de software. De achterliggende reden hiervoor is dat Logius het voornemen heeft de broncode vrij te geven danwel publiek te maken in het kader van een Woo-verzoek.

Op 28 september 2023 is versie 1.0 van het onderzoeksrapport opgeleverd aan Logius. Na goedkeuring van dat rapport door Logius is deze managementsamenvatting opgesteld. Op 4 december is de reactie van Logius toegevoegd aan dit document.

1.1. Samenvatting van de resultaten

Op basis van de aangeleverde broncode heeft Secura vastgesteld dat er diverse kwetsbaarheden bestaan in de software.

Er zijn twee hoge risico's vastgesteld.

Het eerste hoge risico betreft de aanwezigheid van cryptografische sleutels die niet publiek bekend zouden moeten worden. Het betreft geen sleutels die worden gebruikt in een productieomgeving.

Het tweede hoge risico betreft de mogelijkheid om DigiD-gebruikers te misleiden door een phishing-link te maken die lijkt alsof het daadwerkelijk naar DigiD leidt, maar in werkelijkheid automatisch doorverwijst naar een website van de aanvaller. Aanvallers gebruiken dit soort kwetsbaarheden om DigiD-inloggegevens en persoonsgegevens van gebruikers te ontfutselen. De achterliggende oorzaak hiervan is ontbrekende of incorrecte controle op gebruikersinvoer.

Daarnaast zijn vijf lage risico's vastgesteld, waarvan er hier drie worden uitgelicht.

Ten eerste wordt onvertrouwde gebruikersinvoer zonder validatie of normalisatie doorgestuurd naar interne systemen, die buiten de scope van dit onderzoek vallen. Dit doet zich voor bij DigiD Hoog, dat voor identiteitscontrole interne systemen raadpleegt. Om de kans op manipulatie van deze systemen te verkleinen wordt aanbevolen de gebruikersinvoer reeds in de DigiD-backend zelf te valideren en/of normaliseren, en er niet op te vertrouwen dat de interne systemen dat zelf doen.

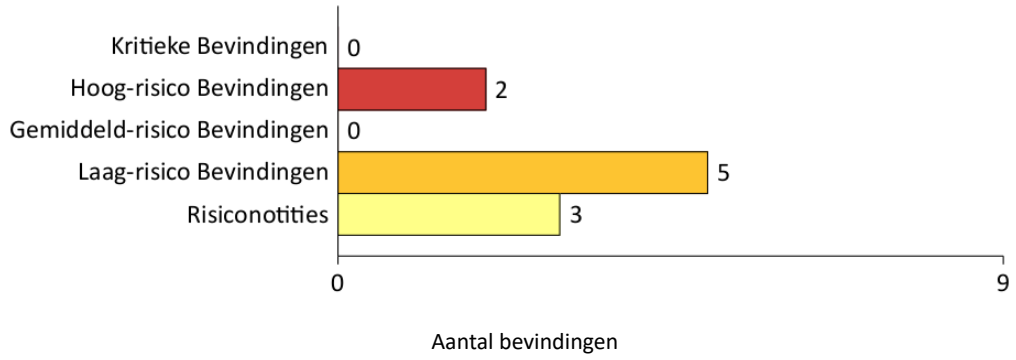
Ten tweede bestaan er zwakheden in de wijze waarop DigiD omgaat met wachtwoorden. Zo controleert de applicatie niet of wachtwoorden al eens elders op het internet zijn gelekt, en worden wachtwoorden niet altijd opgeslagen volgens de laatste aanbevolen cryptografische standaarden. Aanbevolen wordt het aantal iteraties van het gebruikte sleutelderivatie-algoritme te verhogen conform de laatste inzichten.

Ten derde bevat de logging van de applicatie persoonsgegevens zoals Burger Service Nummers (BSNs). Deze logging kan mogelijk door beheerders of andere automatische verwerkers worden ingezien en er bestaat het risico dat de logging op ongecontroleerde plekken terecht komt waardoor pesoonsgegevens kunnen uitlekken.

Naast de twee hoge risico's en vijf lage risico's bevat het onderzoeksrapport drie risiconotities en tien aandachtspunten.

Risicoinschatting bevindingen

In totaal heeft Secura twee hoge risico's, vijf lage risico's en drie risiconotities geïdentificeerd.



Aanbevelingen

Toen de broncode van de DigiD-app openbaar werd gemaakt was hier veel belangstelling voor. De verwachting is dat dit voor de backend ook zal gelden. Aangezien de tijdens dit onderzoek geïdentificeerde zwakheden bij vrijgave van de broncode ook door anderen gevonden zullen worden, adviseert Secura om in ieder geval alle bevindingen in dit rapport te evalueren en waar nodig op te lossen voordat vrijgave plaatsvindt.

Voor technische details inzake het oplossen van de bevindingen wordt verwezen naar de bevindingen en aanbevelingen in het onderzoeksrapport.

1.2. Strategische aanbevelingen

De bevinding van de doorverwijzing met een phishing-link is een kwetsbaarheid die bij een regulier broncode-onderzoek gevonden had kunnen worden. Het feit dat dit niet is gebeurd kan wijzen op een onvolledige eerder uitgevoerde review, of op ongeteste wijzigingen in de applicatie. In beide gevallen adviseert Secura daarom het test-regime van DigiD te evalueren.

Daarnaast wijzen de bevindingen op cryptografisch gebied op een ontwikkelgeschiedenis van DigiD die niet geheel gelijk heeft gelopen met de state-of-the-art inzichten in cryptografische algoritmes. Secura adviseert hier meer aandacht aan te besteden in de toekomst, en de cryptografische implementaties altijd in de vorm van broncode-onderzoek te laten testen, en niet alleen in de vorm van een externe penetratietest.

1.3. Reactie Logius

Secura heeft de volgende reactie ontvangen van Logius:

> *Secura heeft in totaal vijf risico's uitgelicht in de managementsamenvatting, waarvan er twee een hoog en drie een laag risico vormen.*
>
> *Het eerste hoge risico dat door Secura wordt benoemd, is de aanwezigheid in de code van cryptografische sleutels die niet openbaar moeten worden. Dit wordt een risico als deze sleutels in de productieomgeving worden gebruikt. Dit is niet het geval, zoals Secura ook opmerkt: de cryptografische sleutels worden alleen voor testen gebruikt. Om verwarring te voorkomen, is besloten om de sleutels niet te publiceren. Bij een eerder Woo-besluit over broncode hebben we ook geen sleutelmateriaal gepubliceerd en om de werking van software na te gaan, is de vrijgave van de cryptografische sleutels niet nodig.*
>
> *Secura noemt als tweede hoge risico een kwetsbaarheid waarmee een phishing-link gemaakt kan worden die lijkt te verwijzen naar de website van DigiD, maar in werkelijkheid leidt naar de website van de aanvaller. We hebben geconstateerd dat dit risico daadwerkelijk kan leiden tot misbruik en zijn blij dat we deze kwetsbaarheid vóór publicatie hebben kunnen oplossen. Veel dank daarvoor.*

Bij de geconstateerde lage risico's hebben we onderstaande opmerkingen:

- *Secura heeft geconstateerd dat er onvertrouwde gebruikersinvoer zonder validatie of normalisatie wordt doorgestuurd naar interne systemen. Omdat die systemen hiervoor zelf verantwoordelijk zijn, zal dit in de praktijk geen probleem vormen. Om daarvan niet afhankelijk te zijn, nemen we de aanbeveling van Secura over de invoer eerst zelf te schonen. We zullen hier bij doorontwikkeling van de code extra aandacht aan besteden.*
- *Een ander geïdentificeerd laag risico is de wijze waarop DigiD met wachtwoorden omgaat. Het eerste punt daarbij is dat niet gecontroleerd wordt of wachtwoorden eerder op het internet gelekt zijn, bijvoorbeeld naar aanleiding van een hack of een datalek. We kiezen hier bewust niet voor, omdat het kiezen van een juist wachtwoord voor de gebruiker dan erg omslachtig en ingewikkeld wordt. We zullen wel bij gelegenheid de cryptografische sterkte van de wachtwoordopslag verbeteren.*
- *Het derde lage risico dat Secura toelicht betreft het loggen van Burgerservicenummers (BSN's). DigiD gebruikt het BSN om de gebruiker te identificeren, dat is een essentieel onderdeel van de werking ervan. De logging van de BSN's is nodig om DigiD veilig te houden. Zo kunnen we voorkomen dat burgers het slachtoffer worden van misbruik of oneigenlijk gebruik van hun DigiD-account.*

Rest me u te danken voor deze code-review. Logius doet er alles aan om te voorkomen dat openbaarmaking leidt tot onvoorziene veiligheidsrisico's. Deze review door Secura is daarbij een belangrijke maatregel.

1.4. Relatie met OWASP ASVSv4

De controls die in dit hoofdstuk staan beschreven zijn afkomstig uit ASVSv4[1].

Hoewel ASVS redelijk uitputtend is in de typen kwetsbaarheden die in webapplicaties kunnen bestaan, kan het lastig zijn om onderlinge verbanden tussen kwetsbaarheden te herkennen binnen de context van een specifieke applicatie. Ook omvat ASVS veel controls die bedoeld zijn voor defense-in-depth. Om die reden heeft Secura een selectie gemaakt van controls die relevant zijn voor deze situatie, waardoor het onderzoek doelgericht in zijn werk gaat. Niet elke bevinding die tijdens een onderzoek wordt gedaan is noodzakelijk in verband te brengen met een ASVS-control.

Norm	Beschrijving	Bevinding	Referentie
2.1.6	Verify that password change functionality requires the user's current and new password.	Ja	Risiconotitie 1
2.1.7	Verify that passwords submitted during account registration, login, and password change are checked against a set of breached passwords either locally (such as the top 1,000 or 10,000 most common passwords which match the system's password policy) or using an external API. (...)	Ja	Laag-risico Bevinding 1
2.2.1	Verify that anti-automation controls are effective at mitigating breached credential testing, brute force, and account lockout attacks. (...)	-	
2.2.5	Verify that where a Credential Service Provider (CSP) and the application verifying authentication are separated, mutually authenticated TLS is in place between the two endpoints.	-	
2.3.1	Verify system generated initial passwords or activation codes SHOULD be securely randomly generated, SHOULD be at least 6 characters long, and MAY contain letters and numbers, and expire after a short period of time. These initial secrets must not be permitted to become the long term password.	-	

loopt door op de volgende pagina...

[1] https://owasp.org/www-project-application-security-verification-standard/

Tabel 1.1 (vervolgd)

Norm	Beschrijving	Bevinding	Referentie
2.4.1	Verify that passwords are stored in a form that is resistant to offline attacks. Passwords SHALL be salted and hashed using an approved one-way key derivation or password hashing function. Key derivation and password hashing functions take a password, a salt, and a cost factor as inputs when generating a password hash.	Ja	Aandachtspunt 1
2.4.2	Verify that the salt is at least 32 bits in length and be chosen arbitrarily to minimize salt value collisions among stored hashes. For each credential, a unique salt value and the resulting hash SHALL be stored.	-	
2.4.3	Verify that if PBKDF2 is used, the iteration count SHOULD be as large as verification server performance will allow, typically at least 100,000 iterations.	Ja	Laag-risico Bevinding 2
2.4.4	Verify that if bcrypt is used, the work factor SHOULD be as large as verification server performance will allow, with a minimum of 10.	-	
2.4.5	Verify that an additional iteration of a key derivation function is performed, using a salt value that is secret and known only to the verifier. Generate the salt value using an approved random bit generator (SP 800-90Ar1) and provide at least the minimum security strength specified in the latest revision of SP 800-131A. The secret salt value SHALL be stored separately from the hashed passwords (e.g., in a specialized device like a hardware security module).	-	
2.7.6	Verify that the initial authentication code is generated by a secure random number generator, containing at least 20 bits of entropy (typically a six digital random number is sufficient).	-	
2.8.2	Verify that symmetric keys used to verify submitted OTPs are highly protected, such as by using a hardware security module or secure operating system based key storage.	-	
2.8.3	Verify that approved cryptographic algorithms are used in the generation, seeding, and verification of OTPs.	-	
2.9.1	Verify that cryptographic keys used in verification are stored securely and protected against disclosure, such as using a Trusted Platform Module (TPM) or Hardware Security Module (HSM), or an OS service that can use this secure storage.	-	
2.9.2	Verify that the challenge nonce is at least 64 bits in length, and statistically unique or unique over the lifetime of the cryptographic device.	-	
2.9.3	Verify that approved cryptographic algorithms are used in the generation, seeding, and verification.	-	
3.2.2	Verify that session tokens possess at least 64 bits of entropy.	-	
3.2.4	Verify that session tokens are generated using approved cryptographic algorithms.	-	
7.1.1	Verify that the application does not log credentials or payment details. Session tokens should only be stored in logs in an irreversible, hashed form.	-	

loopt door op de volgende pagina...

Tabel 1.1 (vervolgd)

Norm	Beschrijving	Bevinding	Referentie
11.1.4	Verify that the application has anti-automation controls to protect against excessive calls such as mass data exfiltration, business logic requests, file uploads or denial of service attacks.	-	
11.1.6	Verify that the application does not suffer from ”Time Of Check to Time Of Use”(TOCTOU) issues or other race conditions for sensitive operations.	-	

Tabel 1.1: ASVS-controls: Verificatielogica authenticatieprotocollen

Norm	Beschrijving	Bevinding	Referentie
2.6.1	Verify that lookup secrets can be used only once.	-	
2.6.2	Verify that lookup secrets have sufficient randomness (112 bits of entropy), or if less than 112 bits of entropy, salted with a unique and random 32-bit salt and hashed with an approved one-way hash.	-	
2.6.3	Verify that lookup secrets are resistant to offline attacks, such as predictable values.	-	
5.5.2	Verify that the application correctly restricts XML parsers to only use the most restrictive configuration possible and to ensure that unsafe features such as resolving external entities are disabled to prevent XML eXternal Entity (XXE) attacks.	-	
13.2.6	Verify that the message headers and payload are trustworthy and not modified in transit. (...)	-	

Tabel 1.2: ASVS-controls: Veiligheid SAML-artifactresolutie

Norm	Beschrijving	Bevinding	Referentie
2.10.4	Verify passwords, integrations with databases and third-party systems, seeds and internal secrets, and API keys are managed securely and not included in the source code or stored within source code repositories. (...)	Ja	Hoog-risico Bevinding 1
6.4.1	Verify that a secrets management solution such as a key vault is used to securely create, store, control access to and destroy secrets.	-	

Tabel 1.3: ASVS-controls: Geheimen in broncode(-geschiedenis)

Norm	Beschrijving	Bevinding	Referentie
5.2.3	Verify that the application sanitizes user input before passing to mail systems to protect against SMTP or IMAP injection.	-	

loopt door op de volgende pagina...

Tabel 1.4 (vervolgd)

Norm	Beschrijving	Bevinding	Referentie
5.2.4	Verify that the application avoids the use of eval() or other dynamic code execution features. Where there is no alternative, any user input being included must be sanitized or sandboxed before being executed.	-	
5.2.5	Verify that the application protects against template injection attacks by ensuring that any user input being included is sanitized or sandboxed.	-	
5.3.1	Verify that output encoding is relevant for the interpreter and context required. (...)	-	
5.3.2	Verify that output encoding preserves the user's chosen character set and locale, such that any Unicode character point is valid and safely handled.	-	
5.3.3	Verify that context-aware, preferably automated - or at worst, manual - output escaping protects against reflected, stored, and DOM based XSS.	Ja	Risiconotitie 2
5.3.4	Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks.	-	
5.3.5	Verify that where parameterized or safer mechanisms are not present, context-specific output encoding is used to protect against injection attacks, such as the use of SQL escaping to protect against SQL injection.	-	
5.3.6	Verify that the application protects against JSON injection attacks, JSON eval attacks, and JavaScript expression evaluation.	-	
5.3.7	Verify that the application protects against LDAP injection vulnerabilities, or that specific security controls to prevent LDAP injection have been implemented.	-	
5.3.8	Verify that the application protects against OS command injection and that operating system calls use parameterized OS queries or use contextual command line output encoding.	-	
5.3.9	Verify that the application protects against Local File Inclusion (LFI) or Remote File Inclusion (RFI) attacks.	-	
5.3.10	Verify that the application protects against XPath injection or XML injection attacks.	-	
7.3.1	Verify that all logging components appropriately encode data to prevent log injection.	-	

Tabel 1.4: ASVS-controls: Injectiekwetsbaarheden

Norm	Beschrijving	Bevinding	Referentie
5.1.1	Verify that the application has defenses against HTTP parameter pollution attacks, particularly if the application framework makes no distinction about the source of request parameters (GET, POST, cookies, headers, or environment variables).	-	

loopt door op de volgende pagina...

Tabel 1.5 (vervolgd)

Norm	Beschrijving	Bevinding	Referentie
5.1.2	Verify that frameworks protect against mass parameter assignment attacks, or that the application has countermeasures to protect against unsafe parameter assignment, such as marking fields private or similar.	Ja	Laag-risico Bevinding 4 en Aandachtspunt 2
5.1.5	Verify that URL redirects and forwards only allow destinations which appear on an allow list, or show a warning when redirecting to potentially untrusted content.	Ja	Hoog-risico Bevinding 2
5.2.6	Verify that the application protects against SSRF attacks, by validating or sanitizing untrusted data or HTTP file metadata, such as filenames and URL input fields, and uses allow lists of protocols, domains, paths and ports.	-	
6.1.1	Verify that regulated private data is stored encrypted while at rest, such as Personally Identifiable Information (PII), sensitive personal information, or data assessed likely to be subject to EU's GDPR.	-	
6.2.1	Verify that all cryptographic modules fail securely, and errors are handled in a way that does not enable Padding Oracle attacks.	-	
6.2.2	Verify that industry proven or government approved cryptographic algorithms, modes, and libraries are used, instead of custom coded cryptography.	-	
6.2.3	Verify that encryption initialization vector, cipher configuration, and block modes are configured securely using the latest advice.	-	
6.2.4	Verify that random number, encryption or hashing algorithms, key lengths, rounds, ciphers or modes, can be reconfigured, upgraded, or swapped at any time, to protect against cryptographic breaks. (C8)	-	
6.2.5	Verify that known insecure block modes (i.e. ECB, etc.), padding modes (i.e. PKCS#1 v1.5, etc.), ciphers with small block sizes (i.e. Triple-DES, Blowfish, etc.), and weak hashing algorithms (i.e. MD5, SHA1, etc.) are not used unless required for backwards compatibility.	-	
6.2.6	Verify that nonces, initialization vectors, and other single use numbers must not be used more than once with a given encryption key. The method of generation must be appropriate for the algorithm being used.	-	
6.2.7	Verify that encrypted data is authenticated via signatures, authenticated cipher modes, or HMAC to ensure that ciphertext is not altered by an unauthorized party.	-	
6.2.8	Verify that all cryptographic operations are constant-time, with no 'short-circuit' operations in comparisons, calculations, or returns, to avoid leaking information.	-	
6.3.1	Verify that all random numbers, random file names, random GUIDs, and random strings are generated using the cryptographic module's approved cryptographically secure random number generator when these random values are intended to be not guessable by an attacker.	-	

loopt door op de volgende pagina...

Tabel 1.5 (vervolgd)

Norm	Beschrijving	Bevinding	Referentie
6.3.2	Verify that random GUIDs are created using the GUID v4 algorithm, and a Cryptographically-secure Pseudo-random Number Generator (CSPRNG). GUIDs created using other pseudo-random number generators may be predictable.	-	
7.1.2	Verify that the application does not log other sensitive data as defined under local privacy laws or relevant security policy.	Ja	Laag-risico Bevinding 3
9.2.1	Verify that connections to and from the server use trusted TLS certificates. Where internally generated or self-signed certificates are used, the server must be configured to only trust specific internal CAs and specific self-signed certificates. All others should be rejected.	-	
9.2.2	Verify that encrypted communications such as TLS is used for all inbound and outbound connections (...)	-	
9.2.3	Verify that all encrypted connections to external systems that involve sensitive information or functions are authenticated.	-	
14.2.1	Verify that all components are up to date, preferably using a dependency checker during build or compile time.	Ja	Aandachtspunt 3

Tabel 1.5: ASVS-controls: Overige normen

2. BESCHRIJVING VAN DE OPDRACHT

Dit onderzoek is gebaseerd op de offerte met referentie 'P501137 DVo Analyserisico broncode Secura' waarin de volgende beschrijving van de opdracht is opgenomen:

> *In het kader van de Wet Open Overheid (Woo) heeft Logius een verzoek ontvangen om de broncode van DigiD openbaar te maken. Voor de broncode van de DigiD-app is dat inmiddels gebeurd (zie https://github.com/MinBZK/woo-besluit-broncode-digid-app).*
>
> *Deze code is openbaar gemaakt na een interne redactieslag voor het verwijderen van aperte security- en privacy-problemen en een beperkte aanvullende externe review. Omdat de broncode van een app door de-compilatie toch vrij eenvoudig te reversed-engineeren valt, is voor deze tamelijk lichte aanpak gekozen.*
>
> *Voor de backend-code van DigiD ligt dat anders. Deze code is altijd vertrouwelijk behandeld en het vrijgeven kan kwetsbaarheden aan het licht brengen die door kwaadwillende actoren zouden kunnen worden misbruikt. Logius heeft er daarom voor de backend-code voor gekozen de meest risicovolle delen eerst door een deskundige partij te laten beoordelen op mogelijke security-problemen. (...)*

2.1. Scope van het onderzoek

De scope van het onderzoek zoals oorspronkelijk gedefinieerd door Logius is weergegeven in tabel 2.1.

Identificatie	Beschrijving
`digid_x-master.zip`	Broncode DigiD X (master-branch, juli 2023)
`digid_app-master.zip`	Broncode DigiD App (master-branch, juli 2023)
`digid_saml-master.zip`	Broncode DigiD SAML (master-branch, juli 2023)

Tabel 2.1: Scope zoals oorspronkelijk gedefinieerd

Alvorens met het onderzoek te beginnen had Logius bevestigd dat deze scope tijdens de beoordeling aangehouden kon worden.

2.2. Vooraf verstrekte informatie

Logius heeft Secura de informatie in tabel 2.2 verstrekt.

Identificatie	Beschrijving
`CodeAssessment.7z`	Broncode van DigiD X, DigiD App en DigiD SAML
`digid_shared_lib-master.zip`	Broncode van de DigiD shared lib library
`digid_utils-master.zip`	Broncode van de DigiD Utils library
`DigiD_X_routes.txt`	Uitvoer van 'ruby routes' voor DigiD X

Tabel 2.2: Verstrekte informatie

Deze informatie zal na afronding van dit project door Secura op een veilige wijze worden vernietigd (zie paragraaf A.6 op pagina 13 voor meer informatie).

2.3. Doel van het onderzoek

Het doel van dit onderzoek was het onafhankelijk vaststellen van het beveiligingsniveau van de DigiD broncode, het opsporen van kwetsbaarheden hierin die niet of niet volledig naar voren komen uit grey-box of black-box applicatieonderzoeken, en het aandragen van mogelijke verbeteringen.

2.4. Rapportage

De managementsamenvattingen en technische samenvattingen dienen als overzicht van de onderzoeksresultaten voor respectievelijk het algemeen en technisch management. In de daaropvolgende hoofdstukken worden de gedetailleerde resultaten beschreven, ondersteund door reproduceerbare bevindingen. Deze hoofdstukken zijn bedoeld om technisch personeel te begeleiden in het reproduceren en mitigeren van de gevonden problemen.

2.5. Onderzoeksmethode

Dit onderzoek is uitgevoerd als een crystal-box-applicatieonderzoek met code-inspectie. Door middel van automatische tests is de beveiliging van de gebruikte standaardsoftware onderzocht op publiek bekende kwetsbaarheden en configuratiefouten. In de volgende stap is de applicatie door middel van handwerk, ondersteund door tools onderzocht. Hierbij is broncode gebruikt om kwetsbaarheden op te sporen.

Bij een onderzoek volgens de crystal-box-methode is er beschikking over uitgebreide informatie over de interne werking van de software, zoals de broncode, configuratiebestanden en logbestanden van de software en interne documentatie.

Na analyse van de resultaten is dit rapport opgesteld.

2.6. Beperkingen

Een beveiligingsonderzoek biedt waardevol inzicht met betrekking tot de IT-beveiliging van het doelsysteem of -applicatie. Een dergelijk onderzoek is echter slechts een momentopname en biedt geen garantie voor de veiligheid van de IT-omgeving en data. Voortdurend worden er nieuwe aanvalstechnieken ontwikkeld en ontdekt. Daarnaast kan een geringe aanpassing aan de IT-omgeving eenvoudig nieuwe kwetsbaarheden introduceren. Minstens zo belangrijk als het technologische aspect is de rol van processen, procedures en de menselijke factor in informatiebeveiliging. Dit rapport geeft slechts een overzicht van gevonden kwetsbaarheden en is daarmee niet bedoeld als een garantie.

A. TESTAANPAK EN ACHTERGROND

A.1. Onderzoeksmethoden

Om het hierboven beschreven doel te bereiken gebruikt Secura een methodologie die is afgeleid van het Open Web Application Security Project (OWASP), het Information Systems Security Assessment Framework (ISSAF) en van de Open Source Security Testing Methodology Manual (OSSTMM). Deze voorziet in een correcte uitvoering van, onder andere, de volgende projecten:

- Technische beveiligingsbeoordeling;
  (deze kan betrekking hebben op infrastructuur, netwerken en netwerkcomponenten, systemen, applicaties en combinaties ervan)
  - Black-box *(met minimale informatie vooraf en zonder credentials)*;
  - Gray-box *(gebruikmakend van vooraf overhandigde credentials)*;
  - Crystal-box *(met volledige toegang tot informatie over het te onderzoeken systeem)*;
- Code-inspectie;
- Kwetsbaarhedenscan *(ook wel aangeduid als Vulnerability scan)*.

A.2. Wijzigingen van systemen

We raden aan eventuele wijzigingen in de omgeving ten behoeve van dit onderzoek terug te draaien. Hierbij valt aan de infrastructuurkant te denken aan wijzigingen in firewall-inrichting of het aanpassen van IDS/IPS-instellingen. Bij applicaties worden vaak gebruikers aangemaakt en gegevens ingevoerd.

A.3. Normenkader

Secura heeft de inrichting en het beheer van de te onderzoeken systemen beoordeeld ten opzichte van best practices, zoals die golden voor het type systeem gedurende het onderzoek.

A.4. Classificatie van bevindingen

Risico wordt bepaald als het product van de kans op het optreden van gevolgen en de impact van deze gevolgen.

In deze paragraaf staat beschreven hoe kans en impact bepaald worden.

A.4.1. Kans

De kans van optreden wordt beïnvloed door een aantal factoren, zoals de benodigde (technische) kennis om een kwetsbaarheid te misbruiken en de al dan niet publiekelijke beschikbaarheid van programma's (exploits) om een kwetsbaarheid uit te buiten. Ook historische gegevens kunnen gebruikt worden om een kans zo goed mogelijk in te schatten. Onze classificaties houden rekening met het niveau van toegang tot het systeem, het niveau van de gebruikte technieken en de kennis die beschikbaar, of noodzakelijk, is voor de aanvaller. De classificatie van de kans is opgenomen in tabel A.1 op de pagina hierna.

Classificatie	Omschrijving
Laag	Vereist de specifieke inspanning van een bekwame aanvaller.
Gemiddeld	Vereist een gemiddelde vaardigheid en kennis, zoals de mogelijkheid om eenvoudige scripts of programma's te schrijven.
Hoog	Vereist een beginners- of gemiddelde vaardigheid, een minimale inspanning en het gebruik van algemeen beschikbare kennis of hulpmiddelen.

Tabel A.1: Kans op misbruik

A.4.2. Impact

De impact is het resultaat van het optreden van de gebeurtenis. Deze impact betreft onder andere financiële schade en imagoschade. Bij gebrek aan gedetailleerde kennis van de bedrijfsvoering kunnen we de impact alleen bepalen vanuit het standpunt van de techniek. De daadwerkelijke business impact zal door de klant moeten worden ingeschat. De classificatie van de (technische) impact van gevolgen is opgenomen in tabel A.2.

Classificatie	Omschrijving
Laag	Heeft zeer geringe negatieve invloed op vertrouwelijkheid, integriteit en/of beschikbaarheid van de onderzochte omgeving.
Gemiddeld	Heeft beperkte negatieve invloed op vertrouwelijkheid, integriteit en/of beschikbaarheid van de onderzochte omgeving.
Hoog	Heeft grote negatieve invloed op vertrouwelijkheid, integriteit en/of beschikbaarheid van de onderzochte omgeving.

Tabel A.2: (Technische) impact van gevolgen

A.4.3. Risico

Op basis van de inschatting voor beide factoren kan het risico zelf bepaald worden. Hieronder wordt een matrix getoond die het resulterende risico aangeeft.

De risico's worden ingedeeld in de volgende niveaus: kritiek, hoog, gemiddeld, laag en notitie.

We raden aan alle risico's te verhelpen, maar de kritieke, hoge en gemiddelde risico's in het bijzonder. Voor kritieke risico's adviseren we desbetreffende applicatie of infrastructuur direct onbereikbaar te maken en het risico onmiddellijk te verhelpen. Daarnaast is het goed om na te gaan of hier al misbruik van gemaakt is. Voor hoge risico's raden we aan het risico zo snel mogelijk te verhelpen.

Een voorbeeld van een risicomatrix is opgenomen in tabel A.3 op de pagina hierna.

		Impact		
		Laag	Gemiddeld	Hoog
Kans	Hoog	Gemiddeld	Hoog	Kritiek
	Gemiddeld	Laag	Gemiddeld	Hoog
	Laag	Risiconotitie	Laag	Gemiddeld

Tabel A.3: Voorbeeld van een risicomatrix

A.5. Aandachtspunten

Naast risico's hanteert Secura ook het begrip aandachtspunt.

Aandachtspunten zijn geen beveiligingsproblemen, maar veelal functionele problemen. Het wel of niet verbeteren van aandachtspunten wordt overgelaten aan de opdrachtgever.

A.6. Dossier

Tijdens het onderzoek zijn uitgebreide logbestanden gemaakt. Deze informatie is gebruikt voor het opstellen van deze rapportage en is opgeslagen in het onderzoeksdossier. Deze informatie wordt voor een beperkte periode beveiligd opgeslagen op de systemen van Secura. Indien vooraf overeengekomen, kunnen deze logbestanden ter beschikking gesteld worden.

Om te voldoen aan de gegevensbeschermingsregelgeving en de geldende ISO27002-controles, verwijdert Secura klantgegevens, met uitzondering van het rapport en voorstel, uiterlijk twee maanden na het aanleveren van het eindrapport. Het rapport en het voorstel worden binnen 15 maanden na voltooiing van het project gearchiveerd. De maximale bewaartermijn is zeven jaar. Dit is specifiek vereist voor zogenaamde Assurance Audits.

B. GEBRUIKTE AFKORTINGEN

In ons vakgebied wordt vaak gebruikgemaakt van afkortingen. Niet altijd is meteen duidelijk waar een bepaalde afkorting voor staat. Vandaar dat we in deze appendix proberen om een overzicht te geven van de in dit rapport gebruikte afkortingen[1].

ASVS	Application Security Verification Standard
BSN	Burgerservicenummer
BSPA	Baseline Security Product Assessment
CCV	Centrum voor Criminaliteitspreventie en Veiligheid
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
ISSAF	Information Systems Security Assessment Framework
IT	Information Technology
OSSTMM	Open Source Security Testing Methodology Manual
OWASP	Open Web Application Security Project
SFE	Secure File Exchange

[1] In veel gevallen zal hier een Engelse “verklarende beschrijving” zijn opgenomen.