Grip op Secure Software Development (SSD)

Het testen van de (versie 2.0) Beveiligingseisen voor (web)applicaties

Versie: 1.0

Opdrachtgever	A. Reuijl	CIP
Auteurs	Werkgroep SSD testen	

Classificatie	Publiek
Status	CIP categorie 'Becommentarieerde Practice'
Datum	21 november 2016
Filenaam	Testen Grip op SSD beveiligingseisen v1.0.docx.docx

© Centrum voor Informatiebeveiliging en Privacybescherming.
Voort dit werk geldt een Creative Commons Naamsvermelding GelijkDelen 4.0 Internationaal-licentie verleend door het CIP. Zie http://creativecommons.org/licenses/by-sa/4.0/

Voorwoord en leeswijzer

Het is voor organisaties een uitdaging om als opdrachtgever te komen tot veilige IT-diensten, waarbij de gebruikte software voldoet aan de beveiligingsvereisten. Om te komen tot veilige software moet al tijdens het voortbrengingsproces sturing gegeven worden aan de IT-projecten en het doorvoeren van veranderingen op bestaande systemen. Uitbesteding van ontwikkeling, onderhoud en beheer aan meerdere externe leveranciers maakt dit sturingsvraagstuk extra complex. Over en weer zijn er onuitgesproken verwachtingen rondom informatiebeveiliging en privacybescherming. De methode "Grip op SSD" geeft aan hoe de opdrachtgever sturing kan geven en verwachtingen kan expliciteren en de uitvoering kan bewaken.

Een belangrijk onderdeel van de besturing is het gebruik van een hanteerbaar aantal normen. Binnen de methode "Grip op SSD" beschrijft het document "Beveiligingseisen voor (web)applicaties" deze normen. Voor u liggen de testbeschrijvingen voor deze normen. De testbeschrijvingen zijn uitgesplitst naar de verschillende fasen in het voortbrengingsproces. Hoe de testen kunnen worden ingezet in het proces om te komen tot veilige software en hoe met de resultaten moet worden omgegaan staat beschreven in de methode "Grip op SSD".

We beseffen ons dat het opleveren van de testbeschrijvingen geen garantie is voor goede testen. Het testen van software op zwakheden in de beveiliging is namelijk enerzijds kennisintensief, waarbij kennis moet worden gedeeld en anderzijds een formeel proces, waarbij afspraken, taken en verantwoordelijkheden duidelijk zijn afgebakend, vastgelegd en worden nageleefd. Om de kans op succes te vergroten zijn een 9-tal principes beschreven met de belangrijkste aandachtspunten die het testen op beveiliging tot een succes kunnen maken.

Dit document is tot stand gekomen door nauwe samenwerking tussen verschillende partijen binnen de overheid en het bedrijfsleven. Deze samenwerking is tot stand gekomen in de SSD-practitionersgroep. De auteurs willen met name hun dank uitspreken voor de ondersteuning door diegenen die een actieve bijdrage hebben geleverd bij het samenstellen van dit document:

Hans Baylé, CGI;
Olaf Bontes, Ministerie van Defensie;
Steven Eringfeld, Belastingdienst;
Elger Jonker; Centric;
Ad Kint, CIP;
Marcel Koers, CIP;
Marinus Kuivenhoven, Sogeti;
Reinier Landsman, Cert2connect;
Brenda Langedijk, SIG;
Bob Ranzijn, Capgemini;
Rene Slikker, Ministerie van Defensie;
David Vaartjes, Securify;
Willem Vollemans, Valori.

Amsterdam, 21 november 2016

Inhoud

Voorwoord en leeswijzer 2

1. Inleiding 4
1.1. De scope: het testen van beveiligingseisen voor (web)applicaties van "Grip op SSD" 4
1.2. De doelgroep van dit document 4
1.3. De governance bij het testen 5
1.4. De betrokken partijen 5
1.5. Verantwoordelijkheid voor de compliancy aan de eisen 6

2. Software beveiligingstesten 7
2.1. De principes van Software beveiligingstesten 7
2.2. Testen en toetsen 8
2.3. Testvormen 8
2.4. Gehanteerd structuur bij de beschrijving van de testen 9

3. Testen van de beveiligingseisen voor de (web)applicatie 11
3.1. SSD-1: Hardening van technische componenten 11
3.2. SSD-2: Gebruik van veilige cookies of apps 15
3.3. SSD-3: Beveiliging van mobile code 18
3.4. SSD-4: Sessie versleuteling 20
3.5. SSD-5: Vaststellen identiteit van een externe gebruiker 24
3.6. SSD-6: Vaststellen identiteit van een interne gebruiker 28
3.7. SSD-7: Functiescheiding 32
3.8. SSD-8: Least Privilege 36
3.9. SSD-9: Registreren van Unsuccessful Login Attempts 39
3.10. SSD-10: Concurrent Session Control 41
3.11. SSD-11: System Use Notification 43
3.12. SSD-12B: Session termination 45
3.13. SSD-13: Onweerlegbaarheid 47
3.14. SSD-14: Borgen van Sessie Authenticiteit 50
3.15. SSD-15: Scheiding van Presentatie, Applicatie en Gegevens 52
3.16. SSD-17: Beheerinterface 54
3.17. SSD-18: HTTP validatie 56
3.18. SSD-19: Invoer normalisatie 60
3.19. SSD-20: Codering van dynamische onderdelen 62
3.20. SSD-21: Geparametriseerde queries 64
3.21. SSD-22: Invoervalidatie 67
3.22. SSD-23: File includes 69
3.23. SSD-24: Beperking van te versturen HTTP-headers 71
3.24. SSD-25: Beperken van te tonen HTTP-header informatie 73
3.25. SSD-26: HTTP-methoden 75
3.26. SSD-27: Error handling 77
3.27. SSD-28: Commentaar(regels) 79
3.28. SSD-29 Directory Listing 80
3.29. SSD-30: Applicatie logging 82
3.30. SSD-31: Standaard stack 86

1. Inleiding

Dit document beschrijft de reviews en testen die nodig zijn om te bepalen of software aan de Grip op SSD beveiligingseisen voldoet. De bij Grip op SSD gekozen aanpak om te bepalen of software veilig is, richt zich daarmee op de beveiligingskwaliteit van de software zelf. Dit in tegenstelling tot veel normenkaders, zoals ISO27000, die zich richten op de interne processen van de leverancier. Deze aanpak samen met de normen en de testbeschrijvingen geven de opdrachtgever en de opdrachtnemer voor de bouw van de software de handvaten om in een vroeg stadium binnen het voortbrengingsproces grip te krijgen op de beveiligingskwaliteit van de software.

De beschrijvingen van de reviews en testen zijn uitgesplitst naar de verschillende fasen in het voortbrengingsproces. Per beveiligingseis moet zekerheid over meerdere zogenaamde indicatoren verkregen worden bij het testen. Per indicator is aangegeven of er een review plaats moet vinden (aangegeven met een R), er een handmatige test is (H) en of de test ook geautomatiseerd mogelijk is (A). Vervolgens worden de testen toegelicht aan de hand van een generiek omschreven testgeval. Dit betekent dat de testomschrijvingen niet programmeertaal- of casespecifiek zijn, maar een aanzet wordt gegeven om te komen tot specifieke testgevallen.

1.1. De scope: het testen van beveiligingseisen voor (web)applicaties van "Grip op SSD"

De testbeschrijvingen hebben als scope de beveiligingseisen voor (web)applicaties van "Grip op SSD". De testbeschrijvingen vormen daarmee een aanvulling op de beveiligingseisen die van toepassing zijn bij de ontwikkeling en aanschaf van (web)applicaties. Samen met de methode "Grip op SSD", waarin de aanpak "hoe grip erop te krijgen" is beschreven, wordt met de eisen de opdrachtgever een complete oplossing geboden om tot veilige software te komen. De testen beperken zich tot de applicatielaag van een systeem, tenzij vanuit de applicatieontwikkeling de realisatie van de eis afhankelijk is van de configuratie van bijvoorbeeld de serverlaag. In die situaties kan een test en daarmee controle gevraagd worden van een onderdeel dat zich buiten de applicatielaag bevindt. Dit zal zich alleen voordoen bij de productieomgeving of de daarop gelijkende testomgeving(en).

1.2. De doelgroep van dit document

Dit document is opgesteld om de dialoog tussen de ontwikkelaars en de opdrachtgever over de uit te voeren testen te ondersteunen. Het helpt de opdrachtgever om te kunnen aangeven waarop gelet moet worden tijdens het testen. Om uiteindelijk de testgevallen zelf te kunnen maken is een verdiepingsslag vereist. Veelal is deze verdiepingsslag afhankelijk van de programmeertaal /-omgeving en van de gebruikte testmiddelen. De OWASP testing guide[1] vormt daarvoor een nuttig hulpmiddel.

[1] OWASP Testing Guide is gratis te downloaden: https://www.owasp.org/index.php/OWASP_Testing_Project

1.3. De governance bij het testen

Het testen vindt plaats binnen de governance structuur, zoals die is beschreven in de methode 'Grip op SSD'. Dit betekent dat ten aanzien van de gestelde beveiligingseisen het principe 'pas toe of leg uit[2]' geldt. Belangrijk is steeds dat de genomen maatregelen en de risico's die geaccepteerd worden steeds inzichtelijk zijn en aansluiten op de "risk appetite[3]" van de opdrachtgever en dus bewaakt wordt in een governance proces. Door deze aanpak van 'pas toe of leg uit' ontstaat een risicogebaseerde governance.

1.4. De betrokken partijen

De betrokken partijen vormen een keten, waarbij het drievlaksmodel van Looijen is gehanteerd. In het drievlaksmodel worden drie vormen van beheer onderscheiden: functioneel beheer, applicatiebeheer en technisch beheer.

Bij het testen van de beveiligingseisen betrokken rollen en partijen zijn:

- **Functioneel beheer**
  Functioneel beheer is namens de gebruikersorganisatie verantwoordelijk voor het realiseren van de functionaliteit van een applicatie, waarbij deze applicatie optimaal moet aansluiten bij het bedrijfsproces. Functioneel beheer treedt op als opdrachtgever van de afgenomen (ICT-)dienst en dus als opdrachtgever voor applicatie- en technisch beheer.
  Het model gaat er dus vanuit dat het functioneel beheer door de opdrachtgever in eigen beheer wordt gehouden. De opdrachtgever is binnen dit model ook de ontvangende partij, daarom is bij het testen niet het onderscheid ontvangende partij en opdrachtgever aangehouden.
- **Applicatiebeheer**
  Onder applicatiebeheer wordt verstaan het ontwikkelen en onderhouden van de applicatie. Hierbij worden de instandhouding en aanpassing van de applicatieprogrammatuur en de structuur van de gegevensverzamelingen geregeld.
  Het applicatiebeheer ligt bij de interne of externe softwareleverancier, die het ontwerp, de ontwikkeling, het testen en vaak ook het implementeren verzorgt.
- **Technisch beheer**
  Het technisch beheer is verantwoordelijk voor de beschikbaarheid van de (ICT-)dienst en instandhouding van de technische infrastructuur. Op die infrastructuur stelt het technisch beheer ook de applicaties en de bestanden beschikbaar. De technisch beheerder heeft de zorg voor de totale technische infrastructuur (hardware, inclusief de netwerk- en werkplekinfrastructuur, systeemprogrammatuur, ontwikkelhulpmiddelen, enz.).
  De hostingpartij draagt zorg voor het technisch beheer.

Een testmanager heeft hierbij een coördinerende rol. Al dan niet in overleg met een projectmanager en een security officer moet hij de kwaliteit van het op te leveren product (applicatie) waarborgen / aantonen. Als er geen testmanager is, ligt deze rol bij een security officer.

[2] Zie Grip op SSD, Beveiligingseisen voor (web)applicaties, https://www.cip-overheid.nl/downloads/grip-op-ssd/

[3] Zie Grip op SSD, De methode, https://www.cip-overheid.nl/downloads/grip-op-ssd/

1.5. Verantwoordelijkheid voor de compliancy aan de eisen

De onderlinge verantwoordelijkheden zijn van belang bij het testen. Daarom citeren we nogmaals de verantwoordelijkheden, zoals beschreven in paragraaf 2.2.4 van het document "SSD Beveiligingseisen v 2.0":

"De primaire verantwoordelijkheid voor het implementeren van de beschreven eis ligt bij de interne of externe softwareleverancier. Als onderdeel van de oplevering van een release, die klaar is voor productie, dient deze leverancier te rapporteren over de wijze waarop aan de gestelde eisen is voldaan.
De controle en toezicht op de werking en implementatie van de gestelde beveiligingseisen is belegd bij de leverancier, de hostingpartij en de opdrachtgever. In de praktijk zijn bij de opdrachtgever een test team en een security-team verantwoordelijk voor de controle en het toezicht."

Deze controle gebeurt, afhankelijk van de eis, op één van de volgende manieren:

Controlemomenten	**Opdr. gever**	**Appl. Ontw.**	**Hosting-partij**
Governance binnen "Grip op SSD", inclusief de beveiligingseisen	V		
Controle of het ontwerp aan de beveiligingseisen voldoet	V	A	
Controle van de code op compliancy met de beveiligingseisen en de rapportage hierover aan de opdrachtgever		V	A
Controle van de werking van de specifieke functionaliteit	V	A	
Controle door te verifiëren of de applicatie kwetsbaar is voor aanvallen en de rapportage hierover aan de opdrachtgever		V	V
Controle door te verifiëren of de configuratie van de applicatie voldoet aan de beveiligingseisen en hierover te rapporteren aan de applicatieontwikkelaar en de opdrachtgever		A	V
Toetsen van de compliancy-rapportage aan de beveiligingseisen	V		

V= verantwoordelijk, A = adviserend

2. Software beveiligingstesten

2.1. De principes van Software beveiligingstesten

Het testen van software op zwakheden in de beveiliging is enerzijds kennisintensief, waarbij kennis moet worden gedeeld en anderzijds een formeel proces, waarbij afspraken, taken en verantwoordelijkheden duidelijk zijn afgebakend, vastgelegd en worden nageleefd.
De onderstaande principes beschrijven de belangrijkste aandachtspunten die het testen op security tot een succes maken.

1. **Beveiligingstesten kunnen beperkt de aanwezigheid van zwakheden en maatregelen aantonen.**
   Testsets zijn gericht op het testen op zwakheden en de aanwezigheid van maatregelen. Er bestaat geen testsets voor alle mogelijke zwakheden en alle mogelijke combinaties van data en scenario's.

2. **Beveiligingstesten hebben tot doel antwoord op risico's die optreden bij gebruik.**
   De testsets zijn niet alleen afhankelijk van het type, de aard en vertrouwelijkheid van de toepassing, maar ook van de context, waarin het wordt toegepast. Testsets zijn daarom altijd gebaseerd op een risicoanalyse.

3. **Beveiligingstesten maken onderdeel uit van de SDLC.**
   Hoe eerder de testen in de Software Development Life Cycle (SDLC) plaatsvinden hoe effectiever en efficiënter tijd en middelen ingezet kunnen worden om de software veiliger te maken. (Gebreken kunnen veel gemakkelijker en goedkoper opgelost worden.)

4. **Het beveiligen maakt onderdeel uit van het ontwerpproces.**
   Het definiëren van beveiligingseisen, en het toetsen en testen daarop start (voorafgaand aan het programmeren) al in de ontwerpfase ('Security by Design').
   De testsets ontwikkelen mee met de ontwikkeling van het product en de bedreigingen/risico's.

5. **Rapportage over de bevindingen zijn het startpunt van de verbeteringen.**
   Verbeteringen zijn de doorvertaling van de bevindingen.
   De opvolging van de bevindingen is alleen gewaarborgd door het monitoren van in de rapportage beschreven verbeterpunten.
   De rapportage geeft focus aan nieuwe testsets.

6. **Beveiligingstesten zijn alleen mogelijk in de juiste organisatorische ophanging.**
   Testen worden uitgevoerd of door een onafhankelijke partij onafhankelijk van het team dat de software ontwikkeld of er is procesmatig (bij voorkeur geautomatiseerd) gewaarborgd dat de bevindingen daadwerkelijk worden opgevolgd.

7. **Testen vinden onafhankelijk van het project plaats.**
   Governance gaat verder dan de projectscope. Rapportages worden centraal door de beveiligingsorganisatie (bijvoorbeeld binnen een SOC) beheerd, zodat zij in de gehele SDLC kunnen dienen als input van testsets.

8. **Testen gebeurt door het inbrengen van geldige en ongeldige en onverwachte condities.**
   Het programma moet de juiste foutmeldingen genereren wanneer een ongeldige testconditie optreedt en moet de juiste resultaten genereren wanneer de test geldig is.

9. **De softwarestack wordt tijdens de test niet gewijzigd.**
   Door de software en zo ook de gehele softwarestack niet te wijzigen tijdens de uitvoering van de test wordt voorkomen dat de set van ontworpen testgevallen niet aansluit op de gewijzigde software.

2.2. Testen en toetsen

Bij het testen van de software wordt gecontroleerd of de software voldoet aan de functionele eisen, de kwaliteitseisen en de beveiligingseisen. Testen en toetsen is een standaard onderdeel van ieder ontwikkelproces. In het kader van SSD moet dit proces worden uitgebreid met het expliciet testen op het voldoen aan de beveiligingseisen. Door deze testen en toetsen vooraan in het ontwikkelproces kunnen eventuele afwijkingen vroeg in het ontwikkelproces tegen relatief lage kosten worden opgelost en worden hoge kosten verderop in het ontwikkelproces voorkomen.

Voor testen en toetsen is een pragmatische insteek nodig. Hierbij geldt als kanttekening dat een 100 % test alleen mogelijk is tegen hoge kosten. Zo een volledige test geeft alleen de zekerheid dat wordt voldaan aan de gestelde beveiligingseisen, maar bevestigt niet dat het systeem daadwerkelijk volkomen veilig is.

Het verdient aanbeveling om bij testen en toetsen van de ontwikkelde software ook de relatie met zowel het onderliggende platform als de eventuele gekoppelde interne en externe systemen te toetsen. Die systemen vallen weliswaar niet binnen de scope van de testfase, maar spelen wel een rol bij het mitigeren van de beveiligingsrisico's van een applicatie. Een ketting is zo sterk als zijn zwakste schakel.

In de SDLC kan zowel tijdens de acceptatietest als in de productiefase een pentest worden ingezet om zowel de software als de hiermee geassocieerde infrastructuur na te gaan op kwetsbaarheden. Het voordeel hiervan is dat hiermee een werkelijke kwaadwillende hacker voor bijna 100% kan worden gesimuleerd. Meestal worden de gevonden en geverifieerde kwetsbaarheden alleen gerapporteerd door de pentester en niet uitgebuit. Het rapport beschrijft voor elke bevinding het ontstane risico en geeft advies hoe de gevonden kwetsbaarheden kunnen worden opgelost.

Indien een test, naar het oordeel van de opdrachtgever, door zijn aard of complexiteit beter alleen door de leverancier of een derde partij kan worden uitgevoerd, dan kan de opdrachtgever hiervoor kiezen en verifieert (toetst) de opdrachtgever de testresultaten. Veelal worden deze testen uitgevoerd door de leverancier. De resultaten worden overlegd aan de opdrachtgever, die deze voor de beveiligingseisen laat verifiëren.

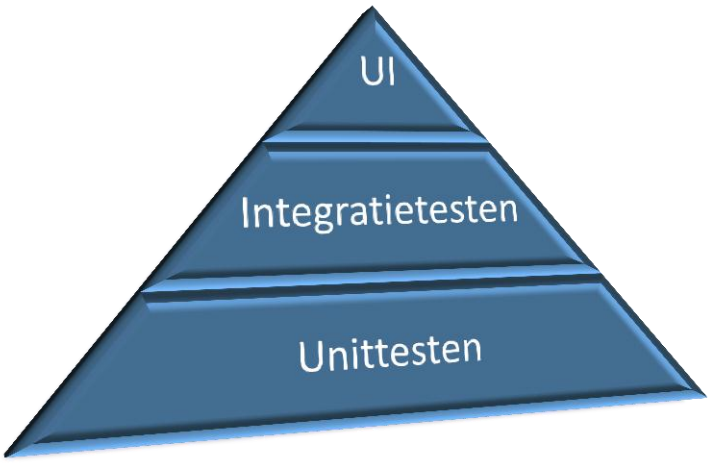
Eventuele afwijkingen worden vastgelegd in een afwijkingsrapportage en worden meegenomen in de risicoacceptatie en verwerkt in het SSD dashboard, zoals beschreven in de methode 'Grip op SSD'.

2.3. Testvormen

Dit document heeft niet tot doel alle testvormen te beschrijven. Het document beschrijft echter wel twee testvormen; unittesten en (user)interfacetesten[4].
Met unittesten worden softwaremodules getest. Unittesten worden dan ook doorgaans door ontwikkelaars geschreven en gebruikt. Unittests zijn doorgaans eenvoudig aan het ontwikkelingsproces toe te voegen en geven zodoende vroeg in de totstandkoming van de applicatie

[4] Ook wel gebruikersacceptatietesten genaamd.

mogelijke afwijkingen aan. Tevens bieden unittests de mogelijkheid om geautomatiseerd op bijvoorbeeld het overschrijden van grenswaarden te testen.
Userinterfacetesten worden met name uitgevoerd wanneer de applicatie gereed is. Deze testen worden dan ook doorgaans door of namens de opdrachtgever uitgevoerd. Het oplossen van de afwijking kost daardoor meer als dat in een eerdere fase gebeurd, bijvoorbeeld in de unittest. Deze unittesten kunnen of handmatig of geautomatiseerd worden uitgevoerd.
In de Test Automation Pyramid is de relatie gelegd tussen de unittesten en de userinterfacetesten. De piramide geeft aan dat de unittesten de basis vormen van het testen en dat de userinterfacetesten in aantal minder worden uitgevoerd.



2.4. Gehanteerd structuur bij de beschrijving van de testen

Bij het beschrijven van de testen van de SSD beveiligingseisen wordt gebruik gemaakt van tabellen. Deze systematische manier van beschrijven zorgt ervoor dat het document kort en leesbaar is gebleven.
De tabellen en conventies die zijn gebruikt worden hier uitgelegd.

Er wordt in de beschrijving drie tabellen gebruikt:

1. Tabel met de SSD beveiligingseis;
2. Tabel met de wijze van testen;
3. Tabel met de beschrijvingen van de testen.

Tabel met de SSD beveiligingseis

Deze tabel bevat de SSD beveiligingseis, zoals die in 'Grip op SSD, SSD beveiligingseisen' is beschreven.

SSD-1: Hardening van technische componenten van de SSD eis	
Criterium	Het criterium beschrijft de SSD-beveiligingseis, waaraan voldaan moet worden. In de eis staan indicatoren onderstreept die getest moeten worden.
Doelstelling	De reden waarom de norm gehanteerd wordt.
Risico	Het risico dat de aanleiding vormt om de norm te hanteren.

Tabel met de wijze van testen

Deze tabel wordt gebruikt voor iedere afzonderlijk sleutelwoord in de SSD eis..

SSD-1: Hardening van technische componenten van de SSD eis							
	indicatoren	I	O	B	S	A	P
/01	De indicator, zoals die in het criterium is beschreven.						
/01.01	Toetsbare/Testbare eis om te bepalen of aan de indicator is voldaan.		R	A	H	R	A
/01.02							

De letters IOBSAP staan voor de fase in het voortbrengingsproces: Initiatie, Ontwerp, Bouw, Systeemtest, Acceptatietest en Productie.
Per fase wordt angegeven of een review (R) moet plaatsvinden, geautomatiseerd (A) of handmatig (H) getest moet worden of dat er geen test plaatsvindt. In het laatste geval blijft het veld leeg ( ).

Tabel met de beschrijvingen van de testen

Hierin worden toelichtingen gegeven op de eis. De spelregels van deze toelichting worden onder het voorbeeld weergegeven.

fase	Toelichting /01.01
Initiatie	Uitleg of *verwijzing naar een uitleg*
Ontwerp	Uitleg of *verwijzing naar een uitleg*
Bouw	Uitleg of *verwijzing naar een uitleg*
Systeemtest	Uitleg of *verwijzing naar een uitleg*
Accept. Test	Uitleg of *verwijzing naar een uitleg*
Productie	Uitleg of *verwijzing naar een uitleg*

Spelregels toelichtingstabel:

1. De testbeschrijving beschrijft een inleiding tot het HOE. De testbeschrijvingen hebben dus niet het doel volledig (en niet specifiek voor een programmeertaal) te beschrijven hoe de test uitgevoerd moeten worden (zie verder paragraaf 1.2).
2. De testbeschrijvingen zijn zelfstandig leesbaar. Herhaling van (grote stukken) teksten zijn voorkomen door het gebruik van referenties.
3. Voor meer detailinfo over het 'hoe getest moet worden' wordt verwezen naar externe bronnen. Hierbij is als beperking gehanteerd dat de bronnen "open source" moeten zijn en is gebruik gemaakt van gerenommeerde (toekomstvaste) organisaties, zoals OWASP en NIST
4. In een testbeschrijving in een fase is geen herhaling van de testbeschrijving uit een voorgaande fase gegeven. Als allebei bijvoorbeeld een R(eview) hebben in de tabel en er voor de te beschrijven fase geen aanvullende info wordt gegeven, dan wordt een verwijzing gegeven: *"zie …".*
5. De beschrijving is er steeds naar gestreefd een herkenbare toegevoegde waarde te leveren ten opzichte van hetgeen al beschreven is. Als er geen inhoudelijke toevoeging gegeven kan worden is het veld leeg.
6. Een eis binnen een indicator die op een bepaalde fase slaat en daardoor niet van toepassing is op andere fasen. Voor die andere fasen wordt dan ook geen testbeschrijving gegeven.

3. Testen van de beveiligingseisen voor de (web)applicatie

3.1. SSD-1: Hardening van technische componenten

SSD-1: Hardening van technische componenten	
Criterium (wie en wat)	De software voldoet aan het hardeningbeleid. De software en het platform zijn daartoe geconfigureerd volgens de bijbehorende hardeningrichtlijn. Het configureren is procesmatig en procedureel ingericht.
Doelstelling (waarom)	De geleverde dienst beveiligen tegen misbruik via zwakheden door het beperken van de functionaliteit tot hetgeen noodzakelijk is voor het correct functioneren van de diensten.
Risico	Aansprakelijkheid in geval van incidenten, doordat (verwijtbaar) niet voldaan wordt aan de actuele en gangbare beveiligingspraktijk, aangezien vermijdbare kwetsbaarheden niet zijn gedeactiveerd.

Testvormen en de beschrijving van de testen:

/01 hardeningbeleid

SSD-1: Hardening van technische componenten							
	indicatoren	I	O	B	S	A	P
/01	hardeningbeleid						
/01.01	Er zijn voorschriften voor hardening en patching van ICT-componenten.	R	R	R	R	R	R

fase	Toelichting /01.01
Initiatie	Stel vast dat het beleid en de afspraken voor hardening en patching door de ontwikkelaar en de hostingpartij zijn gedocumenteerd.
Ontwerp	Stel vast dat gedocumenteerd is welke protocollen, servicesen accounts nodig zijn, inclusief de onderbouwing van de noodzaak/functionaliteit. Controleer of de noodzaak/functionaliteit vanuit 'business requirements' oogpunt noodzakelijk is. Stel vast dat van alle software de handtekening van ontwikkelaar bevat, zodat de bron van de software vastligt. Overigens bieden de moderne besturingssystemen deze controle standaard, zodat er nooit software van een vreemde kan draaien.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

/02 hardeningrichtlijn

SSD-1: Hardening van technische componenten							
	indicatoren	I	O	B	S	A	P
/02	hardeningrichtlijn						
/02.01	De softwareleverancier stelt de software beschikbaar met een actueel overzicht van de noodzakelijke protocollen, services, digitale handtekeningen en accounts.			R	R	R	

SSD-1: Hardening van technische componenten								
/02.02	Tijdens de hosting zijn alleen de noodzakelijke protocollen, services, software en accounts actief, andere protocollen, services, software en accounts zijn gedeactiveerd of verwijderd.				H	H A	A
/02.03	Tijdens de hosting zijn de beveiligingsconfiguraties van netwerkservices en protocollen aangepast conform richtlijnen.				H	H A	A

fase	Toelichting /02.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat de leverancier een overzicht heeft gegeven en dat dit uit de beheerdersdocumentatie valt op te maken. Stel vast dat de inhoud van het overzicht in lijn is met hetgeen in het ontwerp (zie /01.01) hierover is vastgelegd.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	

fase	Toelichting /02.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat er op het systeem geen andere services en/of protocollen zijn opgestart, of andere accounts actief zijn dan die in het overzicht (zie /02.01) zijn vastgelegd. Stel vast dat alleen software met de juiste handtekeningen uitgevoerd wordt.
Accept. Test	Gebruik een scanner zoals nmap om te controleren welke services actief zijn (welke protocollen actief zijn). Of log in op het systeem om te bepalen welke services actief zijn. Voer een controle uit op de poortconfiguratie van de (virtuele) server. Vergelijk de resultaten met de lijst uit /02.01 (overzicht van de leverancier met de vereiste protocollen) en rapporteer over de gevonden verschillen. Stel vast dat de server uitsluitend software kan uitvoeren met de juiste digitale handtekeningen: bij voorkeur door de configuratie te inspecteren of anders door te pogen een betrouwbare (portable) testapplicatie met afwijkende handtekening uit te voeren.
Productie	*Zie vorige fase*

fase	Toelichting /02.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat op het systeem of deze geconfigureerd is volgens de richtlijnen.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase* Stel aanvullend vast dat de juiste instellingen door het gebruik van een scanner zoals nmap om te controleren welke netwerkservices actief zijn en welke protocollen op de netwerkservices worden doorgegeven. Indien een service actief is of en protocol wordt doorgegeven dan moet die in het overzicht (zie

	/02.01) zijn vastgelegd.

/03 procesmatig en procedureel

SSD-1: Hardening van technische componenten							
	indicatoren	I	O	B	S	A	P
/03	procesmatig en procedureel						
/03.01	ICT-componenten zijn (aantoonbaar) volgens de instructies en procedures van de leverancier ingericht.					R	R
/03.02	ICT-componenten ondersteunen alleen die services die vanuit het ontwerp noodzakelijk zijn, andere services zijn gedeactiveerd of verwijderd.					H	H A
/03.03	ICT-componenten zijn voorzien van de meest recente, relevante patches.				H	H	A
/03.04	Alleen ICT-componenten die voldoen aan de hardeningrichtlijnen zijn in productie genomen.				R	R	R
/03.05	Wanneer niet voorkomen kan worden dat de inrichting van een ICT-component afwijkt van de hardeningrichtlijn, dan is dit gedocumenteerd, gerapporteerd en geaccordeerd.					R	R
/03.06	Periodiek wordt getoetst of de in productie zijnde ICT-componenten niet meer dan de vanuit het ontwerp noodzakelijke services bieden (statusopname). Afwijkingen worden hersteld of gedocumenteerd (zie /01.05).						R

fase	Toelichting /03.01
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat de feitelijke configuratie conform de instructies en procedures van de leverancier zijn uitgevoerd.
Productie	*Zie vorige fase*

fase	Toelichting /03.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel aanvullend vast dat na configuratie volgens de instructies en procedures van de leverancier (zie /03.02) de juistheid van de instellingen door het gebruik van een scanner zoals nmap. Ga na of nmap services detecteert die niet noodzakelijk zijn.
Productie	*Zie vorige fase*

fase	Toelichting /03.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Controleer of de laatste patches zijn geïnstalleerd
Accept. Test	*Zie vorige fase*

Productie	*Zie vorige fase* Stel vast dat de laatste patches zijn geïnstalleerd. Hierbij kan gebruik gemaakt worden van de geautomatiseerde ondersteuning van de leverancier van de ontwikkelomgeving om te bepalen of alle patches zijn geïnstalleerd. Vaak kan ook een vulnerability scanner gebruikt worden om het patchlevel te verifiëren.

fase	Toelichting /03.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat alleen die componenten zijn geïnstalleerd die noodzakelijk zijn en die voldoen aan de hardeningsrichtlijnen. (Controleer of er geen toegangs-paden zijn die een 'achterdeurtje' (kunnen) vormen.)
Productie	*Zie vorige fase*

fase	Toelichting /03.05
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat een gedocumenteerde, gerapporteerde en geaccordeerde afwij-kingsrapportage aanwezig is, met daarin de reden geformuleerd waarom dit technisch gezien niet anders kan, indien er in /03.01 t/m /03/04 afwijkingen op de overzicht van de leverancier en daarmee op de hardeningsrichtlijn zijn geconstateerd,
Accept. Test	*Zie vorige fase*
Productie	Stel vast dat namens de opdrachtgever, bijvoorbeeld door een securityofficer, is getekend voor dit risico alvorens deze componenten in productie te nemen.

fase	Toelichting /03.06
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel het bestaan vast van rapportages waaruit blijkt dat de in /02 en /03.01 t/m /03.05 vereiste controles periodiek zijn uitgevoerd.

3.2. SSD-2: Gebruik van veilige cookies of apps

SSD-2 Gebruik van veilige cookies of apps	
Criterium (wie en wat)	In cookies en de applicatie op het front-end zijn de vertrouwelijke gegevens afgeschermd door toepassing van cryptografische technieken.
Doelstelling (waarom)	Vertrouwelijke gegevens mogen niet toegankelijk zijn voor onbevoegden.
Risico	Indien vertrouwelijke geachte gegevens (inclusief de authenticatiegegevens) niet zijn afgeschermd is hun vertrouwelijkheid niet gewaarborgd. De toegang tot de aangeboden dienst is in dat geval niet beperkt tot de geautoriseerde gebruikers.

Testvormen en de beschrijving van de testen:

/01 vertrouwelijke gegevens

SSD-2 Gebruik van veilige cookies of apps							
	indicatoren	I	O	B	S	A	P
/01	vertrouwelijke gegevens						
/01.01	De opdrachtgever specificeert de classificatie van de vertrouwelijkheid van de gegevens in de container/cookie.	R	R				
/01.02	Indien van een gegeven niet de classificatie van de vertrouwelijkheid is vastgesteld, wordt het gegeven (per default) versleuteld.		R				

fase	Toelichting /01.01
Initiatie	Stel vast dat voor de gegevens een classificatiestelsel bestaat en de gegevens zijn geclassificeerd. Per classificatie in het classificatiestelsel zijn de vereisten t.a.v. afscherming beschreven. De beschrijving hoeft daarbij nog niet in te gaan op de daarbij gebruikte technieken.
Ontwerp	Stel vast dat in het FO en het TO voor ieder in het FO en TO genoemde gegeven de classificatie is vastgelegd en vastgesteld. De beschrijving biedt de daarbij gebruikte technieken, zoals de wijze van afscherming per gegeven in de containers/cookies (HttpOnly en Secure en https en host en/of domein) voor alle (Request & Response) cookies.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat voor in het FO en TO nog niet genoemde gegevens (per default) de eis geldt dat deze door middel van versleuteling worden afgeschermd tegen compromittering.
Bouw	
Systeemtest	
Accept. Test	
Productie	

/02 cryptografische technieken

SSD-2 Gebruik van veilige cookies of apps							
	indicatoren	I	O	B	S	A	P
/02	cryptografische technieken						
/02.01	De vertrouwelijke gegevens in cookies/ gegevenscontainers zijn afgeschermd door versleuteling.			H	H	H	H
/02.02	De softwareleverancier stelt de software beschikbaar met bijbehorende configuratierichtlijnen.			R	R	R	
/02.03	Communicatie van vertrouwelijke gegevens (van en naar containers/ cookies) is versleuteld.			H	H	H	H A
/02.04	De hosting draagt zorg voor de configuratie conform de configuratierichtlijnen van de softwareleverancier.						H A

fase	*Toelichting /02.01*
Initiatie	
Ontwerp	
Bouw	Stel vast dat de bij het ontwerp vastgelegde versleuteltechniek wordt gebruikt op de vertrouwelijke gegevens. Stel vast dat enkel de in het ontwerp gekozen opslagmethode wordt gebruikt door de broncode te bekijken of alle opslaglocaties te bekijken. Opslaglocaties zijn o.a.: cookies, local storage, session storage, disk, manifest, app-cache, web-workers, online files, en vele anderen. Stel vast dat de versleuteling niet binnen aanzienlijke tijd wordt gebroken. Bijvoorbeeld doordat de versleuteling aan de FIPS normen voldoet (zie BIR Operationele Handreiking).
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	*Toelichting /02.02*
Initiatie	
Ontwerp	
Bouw	Stel vast dat de aanwezigheid van de configuratierichtlijnen en specifiek de keuzen die gemaakt zijn in de gehanteerde programmeertaal (bijvoorbeeld Java, generatie, …) en de gehanteerde IDE (Integrated Development Environment). Stel vast dat de configuratie en de keuzen daarbij de vereiste afscherming (door middel van versleuteling) biedt.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	*Toelichting /02.03*
Initiatie	
Ontwerp	
Bouw	Stel vast dat de software ingeregeld is conform de richtlijnen voor cryptografische maatregelen voor de in gebruik zijnde cookies/ gegevenscontainers, zoals beschreven door de leverancier.

Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	*Toelichting /02.04*
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat of de installatie conform de configuratiedocumentatie van de leveranciersleverancier heeft plaatsgevonden.

3.3. SSD-3: Beveiliging van mobile code

SSD-3: Beveiliging van mobile code	
Criterium (wie en wat)	Applicaties maken gebruik van geautoriseerde mobile code.
Doelstelling (waarom)	Compliance met dit criterium zorgt dat enkel ondertekende code wordt gebruikt, waarbij de bron wordt geverifieerd en het risico op kwaadaardige code wordt beperkt.
Risico	Mobile code, zoals plug-ins, kan virussen en malware bevatten. Zonder een beveiligingsmechanisme is een applicatie kwetsbaar zijn voor verscheidene vormen van aanvallen en virussen via mobile code.

Testvormen en de beschrijving van de testen:

/01 geautoriseerde mobile code

SSD-3: Beveiliging van mobile code							
	indicatoren	I	O	B	S	A	P
/01	geautoriseerde mobile code						
/01.01	De applicatie maakt geen gebruik van mobile code, waarvan de oorsprong en de veiligheid niet is vastgesteld.		R	H A	H A	H A	H A
/01.02	Voor het gebruik van de applicatie door de gebruiker wordt geen (downloaden van) mobile code vereist, waarvan de oorsprong en de veiligheid niet is vastgesteld.		R	H A	H A	H A	H A
/01.03	Indien gebruik van een mobile code vereist is waarvan de oorsprong of de veiligheid niet met zekerheid kan worden vastgesteld, wordt deze mobile code alleen in een besloten virtuele omgeving, mede gescheiden van vertrouwelijke informatie, uitgevoerd.		R	H	H	H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het ontwerp geen code op de server voorstelt, waarvan de oorsprong en de veiligheid niet is vastgesteld, door te controleren in het welke code er gebruikt wordt, incl. onweerlegbaarheid waar de code van afkomstig is.
Bouw	Stel vast dat alleen in het ontwerp genoemde vertrouwde code wordt gebruikt, bijvoorbeeld doordat de ontwikkelomgeving of het besturingssysteem onvertrouwde code weigert. Als dit niet zo is ingesteld, controleer dan dat code wordt gebruikt waarvan de oorsprong onweerlegbaar vertrouwd is.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat het ontwerp geen mobile code (te downloaden op de werkplek van de gebruiker) voorstelt waarvan de oorsprong en de veiligheid niet is vastgesteld.
Bouw	Stel vast dat de IDE (Integrated Development Environment) en/of het doelbe-

	sturingssysteem onvertrouwde code weigert. Als dit niet zo is ingesteld, controleer dan dat code wordt gebruikt waarvan de oorsprong onweerlegbaar vertrouwd is.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	Stel vast dat de noodzaak en de reden van het niet voorkomen dat deze oplossing gekozen is, is onderbouwd, inclusief de haalbaarheid van de oplossing. Overigens hebben op het moment van schrijven de nieuwe besturingssystemen allemaal een methode voor onweerlegbaarbeid te bepalen van programmacode: .dll, .so en andere applicatiecode. Voor scripts en tekstbestanden zoals javascript en html bestaat zoiets niet. Hiervoor zijn op de server alternatieve oplossingen beschikbaar middels checksums en monitoring tools. Hoewel applicatiecode in webomgevingen vaak Javascript en HTML genereert, is de uitkomst hiervan niet onweerlegbaar op de client. Er zijn op dit moment geen methoden die de inhoud van webpagina's op de client cryptografisch kunnen bewijzen. Stel dat de client een virus of plug-in heeft die de inhoud van de pagina aanpast (zoals adware of adblockers), wordt zowel de onweerlegbaarheid van code op de server en het beveiligd transport beiden omzeild.
Bouw	Stel vast dat wanneer code wordt gebruikt waarvan de oorsprong niet aangetoond vertrouwd is, er gebruik wordt gemaakt wordt van een virtuele omgeving die daadwerkelijk voldoende afscherming biedt. Stel hiertoe vast dat deze omgeving volgens de specificaties is geconfigureerd.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.4. SSD-4: Sessie versleuteling

SSD-4: Sessie versleuteling	
Criterium (wie en wat)	De applicatie past encryptie toe op de communicatie van vertrouwelijke gegevens over niet veilige netwerken.
Doelstelling (waarom)	Het waarborgen dat de vertrouwelijkheid, de integriteit en eventueel de onweerlegbaarheid van gegevensleveringen of transacties wordt geborgd.
Risico	Het ongeautoriseerd toegang tot diensten en onthullen of muteren van gegevens.

Testvormen en de beschrijving van de testen:

/01 encryptie

SSD-4: Sessie versleuteling							
	indicatoren	I	O	B	S	A	P
/01	encryptie						
/01.01	De softwareleverancier gebruikt alleen protocollen en cryptografische technieken die als veilig worden bestempeld.		R	R	R	R	R
/01.02	De applicatie versleutelt de communicatie tussen het serverdeel en het werkplekgedeelte van de applicatie.			H A	H A	H A	A
/01.03	De applicatie versleutelt de communicatie tussen het servers en de applicatie.			H A	H A	H A	A
/01.04	Bij de aanroep van de diensten is in de aanroep de vertrouwelijke informatie versleuteld.			H A	H A	H A	A
/01.05	De softwareleverancier stelt de software beschikbaar met bijbehorende configuratierichtlijnen.			R	R	R	
/01.06	De hosting draagt zorg voor de configuratie conform de configuratierichtlijnen van de softwareleverancier.						H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat er bronnen zijn verzameld die aangeven dat de voorgestelde cryptografische technieken als veilig worden bestempeld. Stel vast dat deze bronnen toonaangevend en recent zijn.
Bouw	Stel vast dat de gebruikte crypto-API's alleen gebruik maken van bij het ontwerp als veilig bestemde technieken Voor WCF zijn er diverse testpakketten beschikbaar. Een open bron om de controle uit te voeren is gepubliceerd door OWASP: https://www.owasp.org/images/6/6c/Attacking_WCF_Web_Services-Brian_Holyfield.pdf
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat onversleutelde verbindingen worden doorgestuurd naar de ver-

	sleutelde variant / dat onversleutelde verbindingen niet beschikbaar zijn. Dit kan bij http door een onbeveiligde verbinding op te zetten (bijvoorbeeld op poort 80 ipv 443). Voor TLS is de tooling verkrijgbaar via http://testssl.sh, http://www.ssllabs.com (voor systemen die benaderbaar zijn via het internet). Bij IMAP, POP3, NNTP en LDAP wordt gekeken of het STARTTLS commando wordt verstuurd nadat initieel een onveilige verbinding is opgezet. Daarbij kan ook nog worden gekeken in de DNS server of er sprake is van een TLSA record (DANE). Zie voor nadere informatie: https://en.wikipedia.org/wiki/Opportunistic_TLS Database vendors gebruiken regelmatig eigen protocollen. Het is vaak het eenvoudigst om hiervan de configuratie te controleren, hiervoor bestaat vanuit diverse leveranciers tooling. Bijvoorbeeld de Microsoft SQL Server Best Practice Analyzer. Of de database uberhaupt beveiligde verbindingen ondersteund is hier te zien: https://en.wikipedia.org/wiki/Comparison_of_relational_database_management_systems#Access_control Controleer bij https verbindingen of alle security-headers zijn toegepast middels een aanvraag naar de server.https://www.owasp.org/index.php/List_of_useful_HTTP_headers Controleer ook dat cookies de Secure Flag gebruiken, wat elders in dit document staat omschreven. Zie voor nadere informatie: https://www.owasp.org/index.php/Testing_for_SSL-TLS_(OWASP-CM-001) en https://www.owasp.org/index.php/Testing_for_weak_Cryptography Let hierbij ook op de stand van de techniek: van een encryptiemethode die in het verleden nog gangbaar was, kan vandaag een kwetsbaarheid zijn aangetoond: ga dit periodiek na. Controleer dat Forward Secrecy wordt gebruikt, zodat communicatie ook vertrouwelijk blijft nadat de sleutelbestanden zijn uitgelekt. Dit is ook te controleren door de eerder geadviseerde gereedschappen.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	*Zie /01.02*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	*Zie /01.02*
Systeemtest	*Zie vorige fase*

Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.05
Initiatie	
Ontwerp	
Bouw	Stel vast dat de aanwezigheid van de configuratierichtlijnen en specifiek de keuzen die gemaakt zijn in de gehanteerde programmeertaal (bijvoorbeeld Java, generatie, …) en de gehanteerde IDE (Integrated Development Environment). Stel vast dat de configuratie en de keuzen daarbij de vereiste afscherming (door middel van versleuteling) biedt.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.06
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de installatie conform /01.05 heeft plaatsgevonden. Stel periodiek vast dat de run-time omgeving nog steeds voldoet aan de vereisten van de leverancier (ofwel of de patches up-to-date zijn om de vereiste afscherming (zie /02.02) te bieden.) De geautomatiseerde test kan samen met die van SSD1/03.03 uitgevoerd worden.

/02 vertrouwelijke gegevens

SSD-4: Sessie versleuteling							
	indicatoren						
/02	vertrouwelijke gegevens	I	O	B	S	A	P
/02.01	De opdrachtgever specificeert de classificatie van de gegevens die worden uitgewisseld.	R	R				
/02.02	Per default geldt hierbij de classificatie, waarvoor versleuteling plaatsvindt.		R				

fase	Toelichting /02.01
Initiatie	Stel vast dat voor de gegevens een classificatiestelsel bestaat en de gegevens zijn geclassificeerd. Per classificatie in het classificatiestelsel zijn de vereisten t.a.v. afscherming beschreven. De beschrijving hoeft daarbij nog niet in te gaan op de daarbij gebruikte technieken.
Ontwerp	Stel vast dat in het FO en het TO voor ieder in het FO en TO genoemde gegeven de classificatie is vastgelegd en vastgesteld. De beschrijving biedt de daarbij gebruikte technieken, inclusief de uitgifte van de certificaten.
Bouw	
Systeemtest	
Accept. Test	

Productie	

fase	Toelichting /02.02
Initiatie	
Ontwerp	Stel vast dat voor in het FO en TO nog niet genoemde gegevens (per default) de eis geldt dat de sessies, waarbinnend deze gegevens worden uitgewisseld, door middel van versleuteling worden afgeschermd tegen compromittering.
Bouw	
Systeemtest	
Accept. Test	
Productie	

3.5. SSD-5: Vaststellen identiteit van een externe gebruiker

SSD-5: Vaststellen identiteit van een externe gebruiker	
Criterium (wie en wat)	(Web)applicaties stellen de identiteit van externe gebruikers vast op basis van een mechanisme voor identificatie en authenticatie, waarbij de authenticatiegegevens in een geconsolideerde authenticatievoorziening worden beheerd.
Doelstelling (waarom)	Het voorkomen van ongeautoriseerde externe toegang tot applicaties, zodat alleen geautoriseerde handelingen worden uitgevoerd en deze handelingen aantoonbaar herleidbaar zijn naar een natuurlijk persoon.
Risico	Het risico is misbruik van identiteit. De gevolgen kunnen zijn ongeautoriseerde onthulling van informatie, ongeautoriseerde modificatie of onterechte invoer van transacties uit naam van valide gebruikers.

Testvormen en de beschrijving van de testen:

/01 externe gebruikers

SSD-5: Vaststellen identiteit van een externe gebruiker							
	indicatoren	I	O	B	S	A	P
/01	externe gebruikers						
/01.01	Externe gebruikers zijn geregistreerde gebruikers, die niet aansluiten via het interne vertrouwde netwerk of de kantoorautomatisering.	R					
/01.02	De externe gebruiker heeft alleen toegang tot vooraf gedefinieerde en vastgelegde functionaliteit op basis van vastgelegde rollen.		R	R	H	H	H

fase	Toelichting /01.01
Initiatie	Stel vast dat voor externe gebruikers de rollen zijn beschreven en dat deze externe gebruikers aan de definitie voldoen aan de definitie, zoals omschreven in de toelichting van de SSD-5 beveiligingseis.
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat voor externe gebruikers per rol en benodigde rechten zijn beschreven bijvoorbeeld in de vorm van een autorisatiematrix.
Bouw	Stel door middel van een codereview vast dat de functionaliteit toe te kennen is per rol en naar de benodigde rechten.
Systeemtest	*Zie vorige fase*
Accept. Test	Stel vast dat de juiste rechten zijn toegekend en er geen rechten zijn toegekend die niet aan rollen zijn gekoppeld.
Productie	*Zie vorige fase*

/02 mechanisme voor identificatie en authenticatie

SSD-5: Vaststellen identiteit van een externe gebruiker							
	indicatoren	I	O	B	S	A	P
/02	mechanisme voor identificatie en authenticatie						
/02.01	De applicatie maakt uitsluitend gebruik van het door het opdrachtgever gespecificeerde mechanisme voor identificatie en authenticatie.	R	R	H	H	H	H
/02.02	De configuratie van de identificatie en authenticatievoorziening waarborgt dat de geauthentiseerde persoon inderdaad de geïdentificeerde persoon is.						H

fase	Toelichting /02.01
Initiatie	Stel vast dat voor de toegang tot de applicatie het mechanisme voor identificatie en voor authenticatie is gespecificeerd.
Ontwerp	Stel vast dat het proces van identificatie is vastgelegd en voldoet aan de vereisten die gelden voor de classificatie van de verwerkte gegevens. Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is gedefinieerd.
Bouw	Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is geïmplementeerd overeenkomstig het ontwerp
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /02.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de uitgifte van accounts alleen plaatsvindt op een wijze waarbij een account alleen wordt toegekend aan het voor het uit te geven account geïdentificeerde persoon. Stel vast dat er geen zwakke wachtwoorden worden toegestaan. Zie voor nadere informatie: OWASP Testing guide, Testing for Weak or unenforced username policy (OTG-IDENT-005)

/03 geconsolideerde authenticatievoorziening

SSD-5: Vaststellen identiteit van een externe gebruiker							
	indicatoren	I	O	B	S	A	P
/03	geconsolideerde authenticatievoorziening						
/03.01	Er wordt voor de authenticatie van burgers gebruik gemaakt van een federatieve voorziening binnen Nederland.	R	R	R	R	R	R
/03.02	Er wordt voor de authenticatie van interne- of externe medewerkers gebruik gemaakt van een bedrijfsbrede authenticatievoorziening.	R	R	R	R	R	R
/03.03	Het beheren en onderhouden van identiteiten en autorisatie vindt op onloochenbare wijze plaats.					R H	R H

SSD-5: Vaststellen identiteit van een externe gebruiker							
/03.04	De applicatie maakt gebruik van uniforme en flexibele[5] authenticatiemechanismen.		R	R	R	R	R

fase	Toelichting /03.01
Initiatie	Stel vast dat de keuze voor de federatieve voorziening is vastgelegd.
Ontwerp	Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is gedefinieerd.
Bouw	Ga na of gebruik gemaakt wordt van de vastgelegde authenticatievoorziening.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /03.02
Initiatie	Stel vast dat de bedrijfsbrede voorziening is vastgelegd.
Ontwerp	Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is gedefinieerd.
Bouw	Stel vast dat gebruik gemaakt wordt van de vastgelegde authenticatievoorziening.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /03.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat het beheren en onderhouden van identiteiten en van autorisaties plaatsvindt op onloochenbare wijze. Dit door na te gaan of het proces formeel is vastgelegd en de stappen en het resultaat zijn gelogd. Zie voor nadere informatie (OWASP Testing guide): Testing for Credentials Transported over an Encrypted Channel (OTG-AUTHN-001) Testing for default credentials (OTG-AUTHN-002) Testing for Weak lock out mechanism (OTG-AUTHN-003) Testing for bypassing authentication schema (OTG-AUTHN-004)
Productie	*Zie vorige fase*

fase	Toelichting /03.04
Initiatie	
Ontwerp	Stel vast dat gekozen is voor authenticatiemethodes die voldoen aan de industriestandaards, zoals SAML en WS-Trust
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	Stel vast dat gebruik gemaakt wordt van authenticatiemethodes die voldoen

[5] Het gaat hier om toekomstvastheid zodat mogelijke toekomstige risico's beperkt worden – bijvoorbeeld door technologiewijziging.

	aan de industriestandaards, zoals SAML en WS-Trust
Productie	*Zie vorige fase*

3.6. SSD-6: Vaststellen identiteit van een interne gebruiker

SSD-6 Vaststellen identiteit van een interne gebruiker	
Criterium (wie en wat)	(Web)applicaties stellen de identiteit van interne gebruikers vast op basis van een mechanisme voor identificatie en authenticatie, waarbij de authenticatie- en autorisatiegegevens in een geconsolideerde authenticatie- en autorisatievoorziening worden beheerd.
Doelstelling (waarom)	Het voorkomen van ongeautoriseerde interne toegang tot applicaties, zodat alleen geautoriseerde handelingen worden uitgevoerd en deze handelingen zijn aantoonbaar herleidbaar naar een natuurlijk persoon.
Risico	Het risico is misbruik van identiteit. De gevolgen kunnen zijn ongeautoriseerde onthulling van informatie, ongeautoriseerde modificatie of onterechte invoer van transacties uit naam van valide gebruikers.

Testvormen en de beschrijving van de testen:

/01 interne gebruikers

SSD-6 Vaststellen identiteit van een interne gebruiker							
	indicatoren	I	O	B	S	A	P
/01	interne gebruikers						
/01.01	Interne gebruikers zijn geregistreerde gebruikers, die aansluiten via het interne vertrouwde netwerk.	R					
/01.02	De interne gebruiker heeft alleen toegang tot vooraf gedefinieerde en vastgelegde functionaliteit en gegevens op basis van vastgelegde rollen.		R	R	H	H	H

fase	Toelichting /01.01
Initiatie	Stel vast dat voor interne gebruikers de rollen zijn beschreven.
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat voor interne gebruikers per rol en benodigde rechten zijn beschreven in de vorm van een autorisatiematrix. Stel vast dat de rollen functiescheiding mogelijk maken. Zie voor nadere informatie: OWASP Testing guide, Test Role Definitions (OTG-IDENT-001)
Bouw	Stel door middel van een codereview vast dat de functionaliteit toe te kennen is per rol en naar de benodigde rechten.
Systeemtest	*Zie vorige fase*
Accept. Test	Stel vast dat er niet meer rechten zijn toegekend dan in de autorisatiematrix is vastgelegd
Productie	*Zie vorige fase*

/02 mechanisme voor identificatie en authenticatie

SSD-6 Vaststellen identiteit van een interne gebruiker							
	indicatoren	I	O	B	S	A	P
/02	mechanisme voor identificatie en authenticatie						
/02.01	De applicatie maakt uitsluitend gebruik van het door het opdrachtgever gespecificeerde mechanisme voor identificatie en authenticatie.	R	R	H	H	H	H
/02.02	De configuratie van de identificatie en authenticatievoorziening waarborgt dat de geauthentiseerde persoon inderdaad de geïdentificeerde persoon is.						H

fase	Toelichting /02.01
Initiatie	Stel vast dat voor de toegang tot de applicatie het mechanisme voor identificatie en voor authenticatie is gespecificeerd.
Ontwerp	Stel vast dat het proces van identificatie is vastgelegd en voldoet aan de vereisten die gelden voor de classificatie van de verwerkte gegevens. Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is gedefinieerd.
Bouw	Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is geïmplementeerd overeenkomstig het ontwerp
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase* Stel vast dat het proces van identificatie is ingericht is conform het ontwerp.

fase	Toelichting /02.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de uitgifte van accounts alleen plaatsvindt op een wijze waarbij een account alleen wordt toegekend aan het voor het uit te geven account geïdentificeerde persoon. Stel vast dat voor de externe gebruikers minimaal 2-factor authenticatie plaatsvindt en er geen zwakke wachtwoorden worden toegestaan. Zie voor nadere informatie: OWASP Testing guide, Testing for Weak or unenforced username policy (OTG-IDENT-005)

/03 geconsolideerde authenticatie- en autorisatievoorziening

SSD-6 Vaststellen identiteit van een interne gebruiker							
	indicatoren	I	O	B	S	A	P
/03	geconsolideerde authenticatie- en autorisatievoorziening						
/03.01	Er wordt voor de authenticatie van externe- of interne medewerkers gebruik gemaakt van een bedrijfsbrede geconsolideerde authenticatie en autorisatievoorziening.	R	R	R	R	R	R

SSD-6 Vaststellen identiteit van een interne gebruiker							
/03.02	Het beheren en onderhouden van authenticatie- en autorisatiegegevens vindt op onloochenbare wijze plaats.					R	R
/03.03	De applicatie maakt gebruik van uniforme en flexibele[6] authenticatie- en autorisatiemechanismen		R	R	R	H	H
/03.04	Er is een actueel overzicht van accounts en de personen die daar gebruik van maken: - service-accounts; - beheeraccounts; - gebruikersaccount; (web)applicatie-accounts.					R	R
/03.05	In aanvulling op de centrale autorisatievoorziening kunnen binnen de applicatie meer fijnmazige autorisaties worden beheerd.		R	R	H	H	H

fase	Toelichting /03.01
Initiatie	Stel vast dat de federatieve voorziening is vastgelegd.
Ontwerp	Stel vast dat de koppeling tussen applicatie en de voorziening voor authenticatie is gedefinieerd.
Bouw	Stel vast dat gebruik gemaakt wordt / kan worden van de vastgelegde authenticatievoorziening.
Systeemtest	*Zie vorige fase*
Accept. Test	Stel vast dat gebruik gemaakt word van de vastgelegde authenticatievoorziening.
Productie	*Zie vorige fase*

fase	Toelichting /03.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat het beheren en onderhouden van identiteiten en van autorisatie plaatsvindt op onloochenbare wijze. Dit door na te gaan of het proces formeel is vastgelegd en de stappen en het resultaat zijn gelogd.
Productie	*Zie vorige fase*

fase	Toelichting /03.03
Initiatie	
Ontwerp	Stel vast dat gekozen is voor authenticatiemethodes die voldoen aan de industriestandaarden, zoals SAML en WS-Trust
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	Stel vast dat gebruik gemaakt wordt van authenticatiemethodes die voldoen aan de industriestandaarden, zoals SAML en WS-Trust. Zie voor nadere informatie (OWASP Testing guide): Testing for Credentials Transported over an Encrypted Channel (OTG-AUTHN-

[6] Het gaat hier om toekomstvastheid zodat mogelijke toekomstige risico's beperkt worden – bijvoorbeeld door technologiewijziging.

	001) Testing for default credentials (OTG-AUTHN-002) Testing for Weak lock out mechanism (OTG-AUTHN-003) Testing for bypassing authentication schema (OTG-AUTHN-004)
Productie	*Zie vorige fase*

fase	Toelichting /03.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat het beheren en onderhouden van identiteiten en van autorisatie plaatsvindt op onloochenbare wijze. Dit door na te gaan of het proces formeel is vastgelegd en de stappen en het resultaat zijn gelogd.
Productie	*Zie vorige fase*

fase	Toelichting /03.05
Initiatie	
Ontwerp	Stel vast dat in het ontwerp vastligt welke autorisaties niet in een centrale voorziening beheerd kunnen worden en wat de reden daarvoor is.
Bouw	Stel vast dat de binnen de applicatie ingestelde autorisaties conform het ontwerp zijn gedefinieerd.
Systeemtest	*Zie vorige fase*.
Accept. Test	Stel vast dat het beheren en onderhouden van identiteiten en van autorisaties op een eenduidige en daarmee onloochenbare wijze plaatsvindt. Dit door na te gaan of het proces en de autorisaties voor het beheren en onderhouden van identiteiten en van autorisatie formeel zijn vastgelegd en de stappen en het resultaat zijn gelogd.
Productie	*Zie vorige fase*

3.7. SSD-7: Functiescheiding

SSD-7: Functiescheiding	
Criterium (wie en wat)	De autorisaties van gebruikers (inclusief beheerders) binnen een (web)applicatie zijn zo ingericht dat autorisaties kunnen worden toegewezen aan organisatorische functies en scheiding van niet te verenigen autorisaties mogelijk is.
Doelstelling (waarom)	Het effectief en veilig inrichten van de toegang tot functionaliteit en (vertrouwelijke) gegevens.
Risico	Het ontbreken van een adequate implementatie van functiescheiding kan leiden tot een verhoogde kans op fraude of misbruik van bedrijfsmiddelen bij kritische of fraudegevoelige taken.

Testvormen en de beschrijving van de testen:

/01 autorisaties

SSD-7: Functiescheiding							
	indicatoren	I	O	B	S	A	P
/01	autorisaties						
/01.01	In de applicatie zijn de autorisaties op een beheersbare wijze geordend. De applicatie maakt daartoe gebruik van autorisatiegroepen.		R	R	R	R	R
/01.02	Bij het definiëren van autorisaties is extra aandacht voor accounts met hoge privileges, zoals functioneel beheeraccounts en platformaccounts. In het ontwerp en de bouw van de applicaties is hier rekening mee gehouden.		R	R	R	R	R
/01.03	Er bestaat een proces voor het definiëren, toekennen en onderhouden van de autorisaties; bij de oplevering de applicatie wordt in de configuratiebeschrijving hiervan gebruik gemaakt of naar verwezen.					R	R

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat in het ontwerp autorisatiegroepen zijn gedefinieerd, zonder dat hier rechten strijdig zijn. Stel vast dat de autorisatiegroepen gekoppeld zijn aan organisatorische rollen en (via de rollen aan) functies.
Bouw	Stel vast dat de functies van het systeem door middel van autorisatiegroepen toegewezen worden aan gebruikers en beheerders, zonder dat hier rechten strijdig zijn.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	Zie /01.01 Stel daarbij vast dat de autorisatie met betrekking tot accounts met hoge privileges, zoals functioneel beheeraccounts en platformaccounts optimaal is ingericht om de schade die bij een mogelijke aanval toegebracht kan worden zo

	beperkt mogelijk te laten zijn.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	Stel vast dat er een autorisatieproces is voor het definiëren, toekennen en onderhouden van de autorisaties opgesteld, waarbij de functiescheiding wordt bewaakt. Stel vast dat het proces beschreven is en geaccordeerd is door het management.
Ontwerp	Stel vast dat de in de applicatie toe te passen autorisaties aansluiten bij het autorisatieproces van de organisatie.
Bouw	
Systeemtest	
Accept. Test	Stel vast dat de periodieke controles op uitgegeven autorisaties plaatsvinden door de verantwoordelijke. Stel vast dat het in de applicatie gerealiseerde autorisatiemodel in de applicatie in overeenstemming is met het ontwerp. Stel vast dat in de configuratiebeschrijving verwezen wordt naar het proces van het toekennen en onderhouden van autorisaties.
Productie	*Zie vorige fase*

/02 organisatorische functies

SSD-7: Functiescheiding							
	indicatoren	I	O	B	S	A	P
/02	organisatorische functies						
/02.01	Op basis van taken, verantwoordelijkheden en bevoegdheden zijn in het ontwerp en de applicatie de verschillende functies, taken en autorisaties geïdentificeerd.	R	R	R			
/02.02	Er is een ingevulde autorisatiematrix, waarin aan de taken autorisaties zijn toegekend.		R	R	R	R	R

fase	Toelichting /02.01
Initiatie	Stel vast dat de toe te passen autorisaties in de applicaties aansluiten op de taken, verantwoordelijkheden en bevoegdheden van de organisatorische functies die gebruik maken van de applicatie.
Ontwerp	Stel vast dat de autorisaties zijn geïdentificeerd op basis van de taken, verantwoordelijkheden en bevoegdheden.
Bouw	*Zie vorige fase*
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /02.02
Initiatie	
Ontwerp	Stel vast dat er een goedgekeurde autorisatiematrix is als basis voor het ontwerp en toegepast is in de applicatie.

Bouw	Stel d.m.v. een codereview vast dat de autorisatiematrix conform het ontwerp geïmplementeerd is.
Systeemtest	Stel vast dat de autorisatie werkt volgens de goedgekeurde autorisatiematrix. Test voor alle autorisatieroepen dat: • uitsluitend de toegestane acties uitgevoerd kunnen worden; • voor alle acties dat deze niet door onbevoegden uitgevoerd kunnen worden.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

/03 scheiding van niet te verenigen autorisaties

SSD-7: Functiescheiding							
	indicatoren	I	O	B	S	A	P
/03	scheiding van niet te verenigen autorisaties						
/03.01	Op basis van taken, verantwoordelijkheden en bevoegdheden zijn de niet te verenigen taken en autorisaties geïdentificeerd.	R	R	R	R	R	R
/03.02	Er is een ingevulde autorisatiematrix, waarin de functiescheiding tot uitdrukking komt.		R	R	R	R	R
/03.03	Naast het bestaan van een gevulde matrix moet er een uitleg zijn over de ondersteuning van de functiescheiding.		R				
/03.04	Er bestaat een proces voor het definiëren en onderhouden van de autorisaties.					R	R

fase	Toelichting /03.01
Initiatie	Stel vast dat in de requirements documentatie voor de applicatie op basis van de voorziene taken, verantwoordelijkheden en bevoegdheden vastgesteld is welke taken en autorisaties niet te verenigen zijn.
Ontwerp	*Zie vorige fase*
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /03.02
Initiatie	
Ontwerp	Neem de controle op niet verenigbare taken en autorisaties mee in de test van /02.02
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /03.03
Initiatie	
Ontwerp	Stel vast dat er een beschrijving is hoe de onverenigbaarheid in het ontwerp is geborgd.
Bouw	
Systeemtest	

Accept. Test	
Productie	

fase	Toelichting /03.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	*Zie /01.03*
Productie	*Zie vorige fase*

3.8. SSD-8: Least Privilege

SSD-8 Least privilege	
Criterium (wie en wat)	De (web)applicatie dwingt de door de opdrachtgever voorgeschreven beperkende set van rechten en privileges af met alleen de voor de gebruiker noodzakelijke toegang.
Doelstelling (waarom)	Een medewerker krijgt alleen die rechten die nodig zijn voor de aan hem of haar opgelegde specifieke taken, zoals beschreven in de functieomschrijving.
Risico	Als een gebruiker over te hoge rechten beschikt kan daar bedoeld of onbedoeld misbruik van worden gemaakt. Tevens is er een risico dat bij compromittering van een gebruikersaccount of systeemaccount onnodige schade ontstaat indien dit account over te hoge rechten beschikt.

Testvormen en de beschrijving van de testen:

/01 rechten en privileges

SSD-8 Least privilege							
	indicatoren	I	O	B	S	A	P
/01	rechten en privileges						
/01.01	De opdrachtgever stelt, op basis van een risicoanalyse, vast welke taken strijdig zijn voor de gebruikers.	R					
/01.02	In het ontwerp is rekening gehouden met niet-verenigbare rechten.		R	R	R		
/01.03	De (web)applicatie zorgt via de toekenning van autorisaties dat niet meer dan de toegewezen rechten en privileges beschikbaar komen voor de gebruikers.			R H	R H	R H	R H
/01.04	Webapplicaties draaien op de onderliggende servers met "least privileges". Door de hostingpartij is de inperking van de rechten geïmplementeerd, zoals die door de softwareleverancier zijn meegegeven in de configuratiehandleiding.				H	H	H
/01.05	Webapplicaties en onderliggende servers zijn zodanig geconfigureerd dat ook security-gerelateerde events worden vastgelegd.		R	H	H	H	H

fase	Toelichting /01.01
Initiatie	Stel vast dat de organieke taken en - functies bekend zijn van diegenen die rechten op het systeem moeten krijgen en welke taken niet binnen één functie / gebruiker mogen worden uitgevoerd. Stel vast dat hierbij een risicoanalyse is uitgevoerd, waarbij een uitspraak is gedaan of taken wel of niet verenigbaar zijn of niet.
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat in het ontwerp rekening is gehouden met het minimaliseren van de autorisatie, waarbij op het systeem alleen die taken uitgevoerd kunnen

	worden die minimaal nodig zijn. Stel hiertoe vast dat geen rechten worden toebedeeld aan privileged en/of gevoelige functionaliteiten als dat niet vanuit de taken, bevoegdheden en verantwoordelijkheden noodzakelijk is. Stel vast dat dit - indien dit nodig is - geschiedt onder een applicatie account. Verifieer de volgende documentatie: - Security Architectuur: vergelijk architectuur met veiligheidseisen, verifieer applicatie heeft juiste eigenschappen. - Functioneel Ontwerp: verifieer hoe interfaces worden aangereopen, hoe de datastromen van en naar de interfaces worden verwerkt, hoe regels worden toegepast op de gegevens. - Technisch Ontwerp: verifieer implementatie van security in detail.
Bouw	Vraag een uitdraai van de applicatiebestanden (naam en CRUD-data) en (rol/taken lijst) autorisatielijsten. Stel vast dat de RBAC op de juiste en volledige wijze is geïmplementeerd. Stel vast dat hierbij de functies van het systeem door middel van autorisatiegroepen toegewezen kunnen worden aan functies, alleen als dit vanuit de taken, bevoegdheden en verantwoordelijkheden noodzakelijk is.
Systeemtest	*Zie vorige fase*
Accept. Test	
Productie	

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Vraag een uitdraai van de applicatiebestanden (naam en CRUD-data) en autorisatielijsten. Stel vast dat de RBAC op de juiste en volledige wijze is geïmplementeerd. Zie voor meer info: OWASP testing guide par. 4.6.2
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat in de geconfigureerde omgeving de applicatie draait met alleen die autorisaties die strikt noodzakelijk zijn om de taken uit te voeren conform de configuratiehandleiding. Zie voor meer info: OWASP testing guide par. 4.6.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.05
Initiatie	
Ontwerp	Stel in het ontwerp vast dat in ieder geval die gebeurtenissen worden gelogd, waarbij uitzonderingen op de op least privilige gebaseerde autorisaties nodig zijn. Ook autorisatie excepties moeten gelogd worden.
Bouw	Stel vast dat de te loggen gebeurtenissen worden weggeschreven naar een

	logfile conform ontwerp.
Systeemtest	Creëer een situatie die volgens het ontwerp gelogd moet worden en stel in de log vast dat deze inderdaad wordt gelogd. Maak per te loggen gebeurtenis een testgeval.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.9. SSD-9: Registreren van Unsuccessful Login Attempts

SSD-9 Registreren van Unsuccessful Login Attempts	
Criterium (wie en wat)	De (web)applicatie registreert mislukte login-pogingen.
Doelstelling (waarom)	Het tijdig signaleren van een aanval, zodat daarop actie ondernomen kan worden.
Risico	Indien mislukte login-pogingen niet worden gesignaleerd en bewaakt, is er een risico dat onbevoegden ongezien in de systemen kunnen binnendringen.

Testvormen en de beschrijving van de testen:

/01 mislukte login-pogingen

SSD-9 Registreren van Unsuccessful Login Attempts							
	indicatoren	I	O	B	S	A	P
/01	mislukte login-pogingen						
/01.01	Alleen daar waar geen gebruik gemaakt kan worden van centrale authenticatievoorzieningen, zoals beschreven in SSD 5 en SSD 6, vindt authenticatie en logging door de (web)applicatie plaats.		R				
/01.02	De (web)applicatie en de (web)server leggen mislukte login-pogingen vast in logbestanden.			H	H	H	H
/01.03	De logbestanden worden voor real time monitoring beschikbaar opengesteld voor de opdrachtgever of voor de door de opdrachtgever aangewezen derde partij.		R	H	H	H	H
/01.04	De opdrachtgever zorgt voor het monitoren en analyseren van de logrecords met mislukte login-pogingen om dreigingen adequaat aan te pakken.						H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat de uitzondering op het gebruik van de centrale authenticatievoorziening is vastgelegd en goedgekeurd.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	-
Ontwerp	Stel vast dat een logging mechanisme is ontworpen, waarbij mislukte inlogpogingen worden vastgelegd. De volgende attributen dienen minimaal vastgelegd te worden: • Applicatie • Userid • Source IP • Tijdstip • Actie
Bouw	Voer een aantal mislukte inlogpogingen uit en stel vast dat alle in het ontwerp

	gespecificeerde kenmerken worden gelogd.
Systeemtest	Zie vorige fase.
Accept. Test	Zie vorige fase.
Productie	Stel vast of alle gespecificeerde kenmerken juist worden vastgelegd.

fase	Toelichting /01.03
Initiatie	
Ontwerp	Stel vast dat de logbestanden real-time informatie bevatten, zodat real-time monitoring mogelijk is.
Bouw	Stel vast dat de logging real-time monitoring mogelijk maakt.
Systeemtest	Zie vorige fase.
Accept. Test	Zie vorige fase.
Productie	Zie vorige fase.

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat dreigingen/aanvallen worden opgemerkt. Stel vast of er duidelijk is welke actie er ondernomen moet worden, zodra een dreiging/aanval wordt opgemerkt.

3.10. SSD-10: Concurrent Session Control

SSD-10 Concurrent Session Control	
Criterium (wie en wat)	Een (web)applicatie is voorzien van Concurrent Session Control, die slechts één sessie per gebruiker toestaat, tenzij onderbouwd is dat meer noodzakelijk is.
Doelstelling (waarom)	Het gebruik van accounts en sessies van en met gebruikers op een beheersbare wijze laten verlopen, door het aantal sessies te beperken.
Risico	Het ongecontroleerd toestaan van meerdere gelijktijdige sessies verhoogt de kans op fouten door gebruikers en maakt de logica van de (web)applicaties ingewikkeld, doordat deze logica rekening moet houden met handelingen via meerdere sessies met dezelfde gebruiker.

Testvormen en de beschrijving van de testen:

/01 één sessie per gebruiker

SSD-10 Concurrent Session Control							
	indicatoren	I	O	B	S	A	P
/01	één sessie per gebruiker						
/01.01	De ontwerper stelt via een functionele analyse vast of er meer dan één gelijktijdige sessie van een gebruiker nodig is en hoe risico's ervan worden voorkomen.		R	R			
/01.02	De ontwerper legt de argumentatie vast indien er meer dan een sessie per gebruiker nodig blijkt te zijn.		R	R			
/01.03	De (web)applicatie limiteert het aantal gelijktijdige sessies per gebruiker conform de specificatie van de ontwerper.			H	H	H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat er maar één gelijktijdige sessie nodig is. Indien meerdere gelijktijdige sessies noodzakelijk wordt controle op /01.02 uitgevoerd.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat de functionele noodzaak is vastgelegd, inclusief een risicoanalyse en de beschrijving hoe de risico's zijn beperkt, indien er meerdere gelijktijdige sessies nodig zijn is
Bouw	Stel vast of de maatregelen die zijn gespecificeerd in de voorgaande fase zijn getroffen.
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.03

Initiatie	
Ontwerp	
Bouw	Voer de volgende test uit: 1. Initieer een sessie met de applicatie door in te loggen. 2. Initieer een tweede sessie met de applicatie door in te loggen met hetzelfde account vanaf een andere locatie (IP). 3. Herhaal stap 2 door totdat het eventueel ingestelde limiet wordt bereikt. Stel vast dat het opzetten van een extra sessie (limiet + 1) wordt geweigerd.
Systeemtest	Zie vorige fase.
Accept. Test	Zie vorige fase.
Productie	Zie vorige fase.

3.11. SSD-11: System Use Notification

SSD-11 System Use Notification	
Criterium (wie en wat)	De (web)applicatie toont tijdens het interactieve inlogproces aan de gebruiker een te conformeren melding, waarmee de gebruiker op de rechten en plichten wordt gewezen.
Doelstelling (waarom)	De melding wijst de gebruiker op zijn wettelijke plichten, zorgt voor een ontmoediging voor ongewenste handelingen en voor een juiste basis om te kunnen loggen en monitoren.
Risico	Het risico van het ontbreken van een System Use Notification is dat een kwaadwillende achteraf kan beweren dat deze niet op de hoogte was van de regels voor het correcte gebruik van de (web)applicatie, of bezwaar kan maken tegen het vastleggen van de uitgevoerde handelingen.

Testvormen en de beschrijving van de testen:

/01 te conformeren melding

SSD-11 System Use Notification							
	indicatoren	I	O	B	S	A	P
/01	te conformeren melding						
/01.01	De opdrachtgever stelt de tekst vast die wordt opgenomen in de System Use Notification.	R	R				
/01.02	De (web)applicatie toont de System Use Notification tijdens de interactieve inlogprocedure aan de gebruiker.			H	H	H	H
/01.03	De (web)applicatie zorgt dat de gebruiker een handeling moet verrichten om de System Use Notification van het scherm te verwijderen, bijvoorbeeld door een Enter te geven.			H	H	H	H

fase	Toelichting /01.01
Initiatie	Stel vast dat er in de opdracht een tekst is vastgesteld voor de System Use Notification.
Ontwerp	Stel vast dat in het ontwerp de System Use Notification is opgenomen.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat er een System Use Notification is geïmplementeerd
Systeemtest	Stel vast dat er tijdens de inlogprocedure een System Use Notification getoond wordt.
Accept. Test	Zie vorige fase.
Productie	Zie vorige fase.

fase	Toelichting /01.03
Initiatie	

Ontwerp	
Bouw	Stel vast dat de System Use Notification alleen door interactie met de gebruiker van het scherm verdwijnt.
Systeemtest	Zie vorige fase.
Accept. Test	Zie vorige fase.
Productie	Zie vorige fase.

3.12. SSD-12B: Session termination

SSD-12 Session termination	
Criterium (wie en wat)	De (web)applicatie beëindigt een sessie na een vooringestelde periode van inactiviteit van de gebruiker via automatische session termination,.
Doelstelling (waarom)	Het voorkomen dat een sessie slechts langer dan gedurende een beperkte tijd onbeheerd toegankelijk is voor andere personen, nadat de gebruiker de sessie onbeheerd heeft achtergelaten.
Risico	Het ontbreken van een session termination mechanisme kan als gevolg hebben dat de reeds geopende sessie door een kwaadwillende wordt misbruikt.

Testvormen en de beschrijving van de testen:

/01 vooringestelde periode

SSD-12 Session termination							
	indicatoren	I	O	B	S	A	P
/01	vooringestelde periode						
/01.01	Als default wordt 15 minuten aangehouden, tenzij de functionaliteit anders noodzakelijk maakt.			H	H	H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat de test van /02.01 de volgende test mee: Stel vast dat de session time-out conform de eis van 15 seconde is aangehouden.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

/02 automatische session termination

SSD-12 Session termination							
	indicatoren	I	O	B	S	A	P
/02	automatische session termination						
/02.01	De (web)applicatie activeert automatisch session termination na een door de opdrachtgever voorgeschreven tijdsinterval van inactiviteit.		R	H	H	H	H
/02.02	Session termination komt overeen met het uitloggen van de gebruiker en de (web)applicatie vernietigt daarbij dienovereenkomstig de sessie.		R	H	H	H	H

fase	Toelichting /02.01
Initiatie	
Ontwerp	Stel vast dat in het ontwerp de session termination is opgenomen.
Bouw	Voer de volgende test uit: 1. log aan op de applicatie 2. de gebruiker raakt muis/toetsenbord niet aan gedurende een tijd die lan-

	ger is dan de session timeout value Stel vast dat de gebruiker onomkeerbaar wordt uitgelogd. Meer info: OWASP ASVS 3.3
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	Stel door middel van een pentest vast dat bij een timeout de sessie wordt afgesloten als er geen gebruikers activiteit is geweest en de timeout tijd is verstreken? Meer info: OWASP ASVS 3.3

fase	Toelichting /02.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat het overschrijden van de timeout waarde van de sessie ook daadwerkelijk de sessie cookies reset, waarna bij her authenticatie een nieuwe sessie (nieuwe session cookies) wordt aangemaakt. Session cookies zijn zichtbaar in de HTTP response header. Meer info: OWASP ASVS 3.7 Op "back" klikken of in de proxy het oude sessie token gebruiken mag de gebruiker niet weer in de oude sessie terug brengen. Meer info: OWASP ASVS 3.7
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.13. SSD-13: Onweerlegbaarheid

SSD-13 Onweerlegbaarheid	
Criterium (wie en wat)	De (web)applicatie ondersteunt de onweerlegbaarheid voor daartoe aangewezen transacties via cryptografische technieken.
Doelstelling (waarom)	Het onweerlegbaar vastleggen van transacties, waarvan duidelijk moet zijn dat die door een specifieke gebruiker daadwerkelijk zijn uitgevoerd.
Risico	Indien een aangewezen transactie niet onweerlegbaar aan een persoon kan worden gekoppeld, kan die later zijn of haar betrokkenheid bij de transactie ontkennen.

Testvormen en de beschrijving van de testen:

/01 aangewezen transacties

SSD-13 Onweerlegbaarheid							
	indicatoren	I	O	B	S	A	P
/01	aangewezen transacties						
/01.01	Tijdens de risicoanalyse wordt door de opdrachtgever bepaald voor welke transacties onweerlegbaarheid nodig is. Denk hierbij onder andere aan het verwerken van financiele, persoon gerelateerde en andere vertrouwelijke gegevens.	R					
/01.02	Tijdens de ontwerpfase worden de aangewezen transacties nader gedefinieerd en wordt bepaald op welke wijze de onweerlegbaarheid wordt geïmplementeerd.		R				
/01.03	De (web)applicatie dwingt de onweerlegbaarheid af voor de aangewezen transacties.			H	H	H	H

fase	Toelichting /01.01
Initiatie	Stel vast dat duidelijk is vastgelegd voor welke operaties onweerlegbaarheid noodzakelijk is.
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat duidelijk is vastgelegd voor welke operaties onweerlegbaarheid noodzakelijk is en op welke wijze dit wordt bereikt.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.03
Initiatie	
Ontwerp	

Bouw	Stel vast dat voor alle benoemde operaties of het nagestreefde niveau van onweerlegbaarheid juist is geimplementeerd en of deze niet omzeilt kan worden zodat operaties achteraf alsnog weerlegd kunnen worden (er was immers een weg om deze te omzeilen, waardoor niet onomstotelijk kan worden vastgesteld dat deze uitsluitend door de gebruiker kan zijn uitgevoerd.).
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

/02 cryptografische technieken

SSD-13 Onweerlegbaarheid							
	indicatoren	I	O	B	S	A	P
/02	cryptografische technieken						
/02.01	De opdrachtgever stelt eisen aan cryptografische technieken, zodat die als veilig kunnen worden bestempeld voor het borgen van de onweerlegbaarheid.		R				
/02.02	De opdrachtgever specificeert op welke wijze het beheer van sleutels en certificaten wordt ingericht voor de gebruikte cryptografische techniek.		R				
/02.03	De (web)applicatie gebruikt voor de aangewezen transacties cryptografische techniek die aan de eisen van de opdrachtgever voldoet.			H	H	H	H
/02.04	De softwareleverancier legt de wijze van sleutelbeheer vast in de configuratiehandleiding.				R	R	
/02.05	De hostingpartij draagt zorg voor het beheer volgens de vastgelegde processen.						R

fase	Toelichting /02.01
Initiatie	
Ontwerp	Stel eerst vast voor welk doel er cryptografische technieken worden ingezet. Controleer vervolgens of de geselecteerde technieken wel passen bij de doelstelling. Ook dient vastgesteld te worden of de technieken en gekozen instellingen (algorithme, cipher, modus, key-size, etc.) aansluiten op geldende wet- en regelgeving en best-practices.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /02.02
Initiatie	
Ontwerp	Stel vast of er een eenduidige procedure is beschreven voor key-management.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /02.03

Initiatie	
Ontwerp	
Bouw	Stel vast of de in het ontwerp omschreven aanpak mbt cryptografische technieken juist zijn vertaald naar code en configuratie.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /02.04
Initiatie	Zie. Toelichting /02.02
Ontwerp	
Bouw	
Systeemtest	Stel vast dat sleutelbeheer in de configuratiehandleiding is opgenomen.
Accept. Test	*Zie vorige fase*
Productie	

fase	Toelichting /02.05
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat het proces voor sleutelbeheer is ingericht.

3.14. SSD-14: Borgen van Sessie Authenticiteit

SSD-14 Borgen van Sessie Authenticiteit	
Criterium (wie en wat)	De (web)applicatie hanteert bij de sessienummering op een onvoorspelbare wijze van nummeren en bij het uitloggen van de gebruiker wordt de sessie actief beëindigd.
Doelstelling (waarom)	Het voorkomen van misbruik van een nog openstaande sessie, die door de oorspronkelijke gebruiker niet meer wordt gebruikt.
Risico	Als een andere persoon een nog openstaande sessie kan oppakken, geeft dit de mogelijkheid van misbruik van de identiteit van de oorspronkelijke gebruiker.

Testvormen en de beschrijving van de testen:

/01 onvoorspelbare wijze van nummeren

SSD-14 Borgen van Sessie Authenticiteit							
	indicatoren	I	O	B	S	A	P
/01	onvoorspelbare wijze van nummeren						
/01.01	Een sessie-ID is voldoende sterk, namelijk een lang en random nummer, om deze onvoorspelbaar te maken.			H	R H	H	H
/01.02	De webapplicatie genereert steeds een nieuw random sessie-ID bij het inloggen en het opnieuw inloggen van een gebruiker.			H	H	H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat de Session-Id een unieke code genereert die voldoende van lengte is. Zie voor aanvullende informatie: https://www.owasp.org/index.php/Session_Management_Cheat_Sheet#Session_ID_Length)
Systeemtest	Stel vast dat in de code dat er Session-Id worden gemaakt met de juiste sterkte. Stel vast dat met de developer tools van de browser of het session ID uniek is door de sessie meerdere keren op te starten. Een sessie blijft hetzelfde bij een browser wanneer gebruik wordt gemaakt van de tabbladen. Voor iedere sessie moet een aparte browser instance gestart worden. Er zijn tools in de markt die de randomness van Sessie tokens kunnen analyseren.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat het Session-Id van het cookie wordt vernieuwd bij veranderen van een sessie of nieuwe authenticatie en dat daarbij een unieke nummer wordt gebruikt.
Systeemtest	*Zie vorige fase*

	Stel met de ontwikkel tools vast dat de Session-Id verandert, wanneer de authenticatie verandert.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

/02 actief beëindigd

SSD-14 Borgen van Sessie Authenticiteit								
	indicatoren	I	O	B	S	A	P
/02	actief beëindigd						
/02.01	De webapplicatie vernietigt aan de serverzijde actief de sessie bij het uitloggen van een gebruiker op de applicatie.			R	R H	R H	R H

fase	Toelichting /02.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat de applicatie actie onderneemt om de sessie te beëindigen door de sessie tokens ongeldig te maken of te resetten. Controleer dat de Session-Id 'expire' tijd kort genoeg is geprogrammeerd. Alle sessies moeten gebruik maken van absolute time-out. Nadat de sessie beëindigd is moet de gebruiker zich weer identificeren. Stel vast dat actief gemonitord wordt of een sessie verlopen is of een gebruiker uitgelogd is. Dit kan door de controle of gebruik is gemaakt van de volgende functies: • J2EE: Functies zoals "HttpSession.invalidate(), zie http://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpSession.html#invalidate() • ASP.NET : "Session.Clear()" en "Session.Abandon()"[7] Controleer daarbij dat in de web.config is opgenomen : <configuration> <system.web> <sessionState timeout="2"></sessionState> </system.web> </configuration> • PHP
Systeemtest	Stel vast dat de sessies verlopen na de ingestelde tijd en dat de gebruiker een begrijpelijke melding krijgt. Stel vast dat het cookie verwijdert is.
Accept. Test	
Productie	

[7] De methode abandon() zorgt ervoor dat een flag geplaatst wordt die pas behandeld wordt aan het einde van de page_request. De gebruiker kan dus gewoon op de pagina doorwerken met sessie objecten nadat de abandon() methode is aangeroepen. Zodra de verwerking van de pagina compleet is wordt de sessie verwijderd.

3.15. SSD-15: Scheiding van Presentatie, Applicatie en Gegevens

SSD-15 Scheiding van presentatie, applicatie en gegevens	
Criterium (wie en wat)	De architectuur van een (web)applicatie is gebaseerd op een gelaagde structuur door de presentatie-laag, de applicatielaag en de gegevens te scheiden, zodat de lagen beschermd kunnen worden binnen de netwerkzones.
Doelstelling (waarom)	Het waarborgen van de vertrouwelijkheid en/of de integriteit van de applicatielogica en/of gegevens.
Risico	Het compromitteren van de applicatielogica en/of gegevens.

Testvormen en de beschrijving van de testen:

/01 gelaagde structuur

SSD-15 Scheiding van presentatie, applicatie en gegevens							
	indicatoren	I	O	B	S	A	P
/01	gelaagde structuur						
/01.01	De scheiding van de presentatie, applicatie en gegevens maken de waarborging van de vertrouwelijkheid en/of de integriteit van de applicatielogica en/of gegevens mogelijk.		R	R			
/01.02	Bedrijfskritische applicatielogica, vertrouwelijke gegevens of gegevens waarvan de onweerlegbaarheid moeten worden gewaarborgd worden buiten de DMZ opgeslagen.		R	R	R	R	R
/01.03	In de configuratiebeschrijving staat beschreven in welke zone welke delen van de applicatie komen.				R	R	R
/01.04	De hostingpartij past voor de zones security regimes toe, die gelden voor die specifieke zone.						R

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast door middel van een design review of er een gelaagd / gesegmenteerd applicatieontwerp is toegepast. Denk hierbij aan: webapplicaties ontkoppeld, gescheiden DB model, MVC principes.
Bouw	*Zie vorige fase*
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast waar welke gegevens worden opgeslagen en of de zone passend is voor de classificatie/aard van de gegevens. Minimaal worden geen vertrouwelijke gegevens in de DMZ opgeslagen.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat in de configuratiebeschrijving staat beschreven in welke zone welke delen van de applicatie komen.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat voor de zones security regimes worden toegepast, die gelden voor die specifieke zone.

3.16. SSD-17: Beheerinterface

SSD-17 Beheerinterface	
Criterium (wie en wat)	Beheeractiviteiten vinden plaats via een van de standaard gebruikersinterface gescheiden beheerinterface.
Doelstelling (waarom)	Het voorkomen van onbedoeld gebruik of misbruik van de beheerfunctionaliteit.
Risico	Door het ontbreken van afdoende afscherming kunnen gewone gebruikers beheerdersautorisaties verwerven of kan de beheerder als gebruiker onbedoeld onjuiste beheerhandelingen uitvoeren.

Testvormen en de beschrijving van de testen:

/01 gescheiden beheerinterface

SSD-17 Beheerinterface							
	indicatoren	I	O	B	S	A	P
/01	gescheiden beheerinterface						
/01.01	Het ontwerp en de applicatie waarborgen dat de beheerinterface via autorisaties en vooraf gedefinieerde functionaliteit volgens de gespecificeerde functiescheiding wordt gerealiseerd.		R	R	R	R	R
/01.02	Het ontwerp en de applicatie waarborgen dat alle beheerfunctionaliteit is ondergebracht in een als beheerinterface herkenbare interface.		R	H	H	H	H
/01.03	Het ontwerp en de applicatie waarborgen dat de beheerinterface niet beschikbaar is voor reguliere gebruikers.		R	H	H	H	H A
/01.04	De beheerinterface maakt alleen gebruik van veilige protocollen.			R	A	A	A
/01.05	De hostingpartij maakt de externe toegang tot de beheerinterface alleen mogelijk via een geautoriseerde en beveiligde verbinding.					H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat in ontwerp gescheiden beheerfunctionaliteit is opgenomen met eigen autorisatie.
Bouw	Stel vast dat in de applicatie een autorisatie matrix is geïmplementeerd waarbij alleen de beheerdersrol toegang heeft tot die functionaliteit die bedoeld is voor beheer van de applicatie. Meer info: OWASP ASVS 4.1
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat er een beheerrol is waarbij de functionaliteit via een gescheiden beheerinterface plaatsvindt.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*

Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Stel vast dat in de applicatie een autorisatie matrix is geïmplementeerd waarbij alleen de beheerdersrol toegang heeft tot de functionaliteit die bedoeld is voor beheer van de applicatie.
Systeemtest	*Zie vorige fase*
Accept. Test	1. Verzamel de links/pagina's die toegankelijk zijn voor een beheerders test account dan wel een account met de hoogste rechten. 2. Start een sessie met een normaal testaccount en Stel vast dat de onder 1) gevonden pagina's onder dit laatste testaccount niet benaderbaar zijn. Zie voor nadere informatie: OWASP ASVS 2.32, 4.1, 4.4 OWASP Testing Guide 4.0, Authorization https://www.owasp.org/index.php/Testing_for_Authorization
Productie	*Zie vorige fase*

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	Stel vast dat de beheerinterface voldoet aan SSD-4 Zie voor nadere informatie: OWASP ASVS, gehele paragraaf V10 en de links daaronder.
Systeemtest	*Zie vorige fase*
Accept. Test	Maak gebruik van een network vulnerability scanner om de SSL/TLS instellingen te analyseren, let daarbij op de laatste stand van de techniek. Stel vast dat de instellingen aan de vereisten voldoen.
Productie	*Zie vorige fase*

fase	Toelichting /01.05
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat toegang tot het netwerk alleen na authenticatie en autorisatie mogelijk is. Indien het toegangspad door middel van een VPN via een gedeeld netwerk loopt, dan moet worden vastgesteld dat het bron IP adres deel uitmaakt van een vertrouwd netwerk.
Productie	*Zie vorige fase*

3.17. SSD-18: HTTP validatie

SSD-18 (HTTP) validatie	
Criterium (wie en wat)	De (web)applicatie controleert de inhoud van een (HTTP-)request door deze te valideren alvorens die te gebruiken.
Doelstelling (waarom)	Compliance met dit criterium zorgt dat alleen gecontroleerde invoer wordt verwerkt.
Risico	Inzage, wijziging, verlies, of misbruik van gegevens door bijvoorbeeld manipulatie van de webapplicatielogica.

Testvormen en de beschrijving van de testen:

/01 valideren

SSD-18 (HTTP) validatie							
	indicatoren	I	O	B	S	A	P
/01	valideren						
/01.01	De webapplicatie controleert de correcte authenticatie en autorisatie van de initiator van een HTTP-request.			H	H	H	H
/01.02	De webapplicatie controleert een HTTP-request op type, lengte, formaat, karakters en patronen.		R	R H	R H	R H	R H
/01.03	De webapplicatie controleert de binnenkomende: • URL's; • Query parameters (variabelen die de client via een GET doorgeeft); • Form parameters (variabelen die de client via een POST doorgeeft); • Cookies; • HTTP-headers; • XML (hieronder vallen ook protocollen zoals SOAP, JSON en REST); • Bestanden.			R A H	R A H	R A H	R A H
/01.04	Voor andere protocollen dan HTTP worden met HTTP vergelijkbare controles uitgevoerd.		R	R A H	R A H	R A H	R A H
/01.05	De webapplicatie verwijdert ongewenste invoer en maakt risicovolle karakters uit de invoer onschadelijk.		R	A H	A H	A H	A H
/01.06	In de configuratiehandleiding van de softwareleverancier is beschreven welke controles moeten worden uitgevoerd en welke ongewenste invoer door de webserver moeten worden verwijderd.					R	R
/01.07	Door de hostingpartij is de configuratie uitgevoerd conform de configuratiehandleiding.						H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat request zonder de juiste authenticatie en autorisatie niet mogelijk is. Stel daartoe vast dat het niet mogelijk is om transacties voor gevalideerde gebruikers uit te voeren vanaf een andere website dan de website waarop de

	gebruiker is gevalideerd (bijvoorbeeld middels een unit test). Stel vast dat hiervoor niet vanuit een bepaalde flow naar een andere flow kan worden overgesprongen. *D*oorloop hiertoe per rol de flow die de applicatie heeft: • Bij welke pagina's kan ik? EN mag ik dit ook? • Maak een spider van alle mogelijke pagina's met parameters (met die rechten kan ik hierbij) en kijk waar verschillen zitten tussen de rollen; maak testgevallen voor delta (verschillen in rechten) en voer deze uit. Stel vast dat dat ook niet op een parallel spoor (horizontaal) mogelijk is, waarbij dezelfde actie uitgevoerd kan worden (bijv. deeplink) in een nieuw scherm.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat het voorgestelde framework of de gekozen programmeertaal ondersteuning heeft voor HTTP validatie. Stel vast of de limieten, formatteringen (karakters, typen, patronen) zijn beschreven?
Bouw	Stel vast dat is vastgelegd welke invoermogelijkheden er zijn. Kijk hierbij of naar alle aanvalsvlakten is gekeken: - Code cliënt side - Code server side - applicatie lagen - database Stel vast dat gedefinieerd is welke typen, lengten, formaten, karakters en patronen per invoermoglijkheid noodzakelijk zijn en daarom toegestaan zijn. Stel vast middels Code Review of alleen de toegestane HTTP methoden worden gebruikt en of alle onderdelen van een HTTP request worden gevalideerd op lengten, formaten, karakters en patronen.
Systeemtest	Stel middels Code Review vast dat alleen de toegestane HTTP methoden worden gebruikt en dat alle onderdelen van een HTTP request worden gevalideerd op lengten, formaten, karakters en patronen. Dit kan ook door middel van handmatig testen. Ga daarbij na dat niet valide requests worden afgehandeld en niet verder wordt verwerkt,
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Stel vast dat is vastgelegd welke invoermogelijkheden er zijn. Kijk hierbij of naar alle aanvalsvlakten is gekeken: - Code cliënt side - Code server side - applicatie lagen

	- database Stel vast dat gedefinieerd is welke dynamische invoer noodzakelijk is en daarom is toegestaan. Stel vast dat voor alle onderdelen van een http request validatie plaatsvindt. Valideer middels een code review of dynamische invoer afdoende/juist gecodeerd wordt. Zie voor voorbeelden ASVS 5.1, 5.3, 5.5, 5.6 en 5.10 tot en met 5.26 Als er alleen gebruik wordt gemaakt van standaardbibliotheken voor het controleren van de requests, stel dan vast dat er bronnen zijn verzameld die aangeven dat de voorgestelde bibliotheken als effectief worden bestempeld. Stel vast dat deze bronnen toonaangevend en recent zijn.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.04
Initiatie	
Ontwerp	Stel vast dat is vastgelegd welke protocollen anders dan HTTP gebruikt worden.
Bouw	Stel vast dat middels het raadplegen van bronnen is vastgesteld welke controles uitgevoerd moeten worden. Stel vast dat voor al de invoermogelijkheden waar deze protocollen gebruiken worden door middel van een codereview of de, volgens de bronnen, noodzakelijke controles zijn ingezet.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.05
Initiatie	
Ontwerp	
Bouw	Stel vast dat in aanvulling op /01.03 en /01.04 dit ook geldt voor het verwijderen van ongewenste invoer en het onschadelijk maken van risicovolle karakters.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.06
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat in de configuratiehandleiding is beschreven welke invoermogelijkheden er zijn, welke protocollen daar gebruikt worden, welke controles er uitgevoerd moeten worden en welke karakters verwijderd of onschadelijke gemaakt moeten worden. Als er (standaard)bibliotheken worden gebruikt, stel dan vast dat deze zijn vastgelegd in de configuratiehandleiding.
Productie	*Zie vorige fase*

fase	Toelichting /01.07
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de systemen bij de hosting partij (die ingezet worden voor de webapplicatie) de configuratie en standaardbibliotheken bevatten, zoals vast-gelegd in de configuratiehandleiding.

3.18. SSD-19: Invoer normalisatie

SSD-19 Invoer normalisatie	
Criterium (wie en wat)	De (web)applicatie voorkomt manipulatie door alle ontvangen invoer te normaliseren alvorens die te valideren.
Doelstelling (waarom)	Inzage, wijziging, verlies, of misbruik van gegevens door bijvoorbeeld manipulatie van de applicatielogica.
Risico	Zonder normalisatie van de ontvangen inhoud kunnen kwaadwillenden kwetsbaarheden benutten om gegevens te bemachtigen, te muteren of toe te voegen.

Testvormen en de beschrijving van de testen:

/01 normaliseren

SSD-19 Invoer normalisatie							
	indicatoren	I	O	B	S	A	P
/01	normaliseren						
/01.01	De (web)applicatie controleert de invoer op verdachte karakters of commando's.		R	R H A	R H A	R H A	R H A
/01.02	De (web)applicatie verzorgt onder andere: • Het omzetten van NULL karakters naar spaties; • Het coderen van bijzondere karakters naar een uniforme codering, zoals UTF-8; • Het normaliseren van pad verwijzingen zoals '/./' en '/../'; • Het verwijderen van overbodige spaties en regeleinden; • Het verwijderen van onnodige witruimtes; • Het omzetten van backslashes '\' naar forward slashes '/'.; • Het omzetten van mixed case strings naar lower case strings.			R H A	R H A	R H A	R H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het voorgestelde framework of de gekozen programeertaal ondersteuning geeft aan de normalisatie van de invoer. Stel vast dat de handleiding het framework of gekozen programmeertaal een (verwijzing naar een) uiteenzetting heeft over de normalisatie.
Bouw	Stel vast dat is vastgelegd welke invoermogelijkheden er zijn. Kijk hierbij of naar alle aanvalsvlakten is gekeken: - Code server side - applicatie lagen - database Stel vast dat gedefinieerd is welke karkaters en commando's als verdacht worden gezien. Stel vast dat middels het raadplegen van bronnen is vastgesteld welke controles uitgevoerd moeten worden. Controleer voor al de invoermogelijkheden door middel van een codereview of

<table>
<tr><td></td><td>de, volgens de bronnen, noodzakelijke controles zijn ingezet. Of open de code in de debug mode en plaats setpoints op de verschillende invoermogelijkheden. Tijdens het plaatsen van de verschillende breakpoints kan de code gecontroleerd worden op het naleven van het vastgelegde ontwerp. In de debug mode kan handmatig gecontroleerd worden wat de applicatie doet met de inhoud van variabelen en hoe de code reageert op een eventuele afwijking.

Hanteer bij de keuze tussen handmatig testen en unittests de volgende tabel:
<table>
<tr><th>Test</th><th>Clientside</th><th colspan="3">Serverside</th></tr>
<tr><th></th><th></th><th>Businesslaag</th><th>Service(s)</th><th>Database</th></tr>
<tr><td>NULL</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
<tr><td>UTF8</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
<tr><td>Overige tekens</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
</table></td></tr>
<tr><td>Systeemtest</td><td>Zie vorige fase</td></tr>
<tr><td>Accept. Test</td><td>Zie vorige fase</td></tr>
<tr><td>Productie</td><td>Zie vorige fase</td></tr>
</table>

fase	Toelichting /01.02
Initiatie	•
Ontwerp	
Bouw	Stel vast dat bij /01.01 minimaal is meegenomen: • Het omzetten van NULL karakters naar spaties; • Het coderen van bijzondere karakters naar een uniforme codering, zoals UTF-8; • Het normaliseren van pad verwijzingen zoals '/./' en '/../'; • Het verwijderen van overbodige spaties en regeleinden; • Het verwijderen van onnodige witruimtes; • Het omzetten van backslashes '\' naar forward slashes '/'; • Het omzetten van mixed case strings naar lower case strings.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.19. SSD-20: Codering van dynamische onderdelen

SSD-20 Codering van dynamische onderdelen in de uitvoer	
Criterium (wie en wat)	De (web)applicatie beperkt de uitvoer tot waarden die (veilig) verwerkt kunnen worden door deze te normaliseren.
Doelstelling (waarom)	Voorkom manipulatie in de context van de browser op het systeem van andere gebruikers.
Risico	Via uitvoer van de (web)applicatie, de manipulatie van de werking van - of informatie - in de browser van derden, doordat de (web)applicatie ongewenste opdrachten geeft aan de client, bijvoorbeeld in het geval van XSS.

Testvormen en de beschrijving van de testen:

/01 normaliseren

SSD-20 Codering van dynamische onderdelen in de uitvoer							
	indicatoren	I	O	B	S	A	P
/01	normaliseren						
/01.01	Alle dynamische uitvoer wordt naar een veilig formaat geconverteerd.		R	R H A	R H A	R H A	R H A
/01.02	In de configuratiehandleiding van de softwareleverancier is beschreven welke controles op het verwijderen van ongewenste invoer door de web-/applicatieserver moeten worden uitgevoerd.					R	R
/01.03	Door de hostingpartij is de configuratie uitgevoerd conform de configuratiehandleiding.						R

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het voorgestelde framework of de gekozen programmeertaal het coderen van de uitvoer uberhaupt ondersteund. Stel vast dat de documentatie van het framework of gekozen programmeertaal een uiteenzetting bevat m.b.t. het coderen van de uitvoer, of dat er referenties bestaan naar andere bronnen waarin dit is gedocumenteerd. Wordt coderen niet ondersteund, dan kan conversie plaatsvinden in SSD-22
Bouw	Stel vast dat is vastgelegd welke uitvoermogelijkheden er zijn, c.q. welke uitvoerslots er op een HTML pagina bestaan waarin 'onveilige gebruikers input' terecht kan komen en uit welke applicatielagen/bronnen deze kan komen. Kijk hierbij of naar alle slots en alle applicatielagen is gekeken: - code server side - applicatie lagen - database Stel vast dat voor elk slot gedefinieerd is welk type van codering hier noodzakelijk is, voor welke karakters codering noodzakelijk is, zodat het onmogelijk wordt gemaakt dat 'onveilige gebruikers input' uit de context van een slot kan uitbreken naar een context die code executie in de browser toestaat. Stel vast dat voor elk relevant slot dat middels het raadplegen van documentatie is vastgesteld welke type codering uitgevoerd (= welke escape syntax gebruikt moeten worden) Controleer voor al de relevante slots door middel van een codereview, of die

<table>
<tr><td></td><td>noodzakelijke coderingen zijn uitgevoerd.
Zie voor de verschillende typen van slots en hun coderingen:
https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet</td></tr>
<tr><td>Systeemtest</td><td>Stel vast dat als er op enig moment invoer mogelijk is en die invoer in een bepaald slot van een pagina landt, dat executie van javascript e.d. niet mogelijk is in de context van de browser.
Bedenkt hierbij dat gerichte codereviews in de regel meer zekerheid geven.

Hanteer bij de keuze tussen handmatig testen en de unittest de volgende tabel:
<table>
<tr><th>Test</th><th>Clientside</th><th></th><th>Serverside</th><th></th></tr>
<tr><th></th><th></th><th>Businesslaag</th><th>Service(s)</th><th>Database</th></tr>
<tr><td>NULL</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
<tr><td>UTF8</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
<tr><td>Overige te-kens</td><td>Handmatig</td><td>Unitttest</td><td>Unittest</td><td>Unittest</td></tr>
</table></td></tr>
<tr><td>Accept. Test</td><td>Zie vorige fase</td></tr>
<tr><td>Productie</td><td>Zie vorige fase</td></tr>
</table>

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Indien het voorgestelde framework of de gekozen programmeertaal het coderen van de uitvoer ondersteund, stel dan vast dat in de documentatie is beschreven welke relevante uitvoermogelijkheden/slots er bestaan en welke karakters binnen dat slot gecodeerd moeten worden. Als er (standaard)bibliotheken worden gebruikt, stel dan vast dat deze zijn vastgelegd in de documentatie van de software.
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de systemen bij de hosting partij de juiste codering bibliotheken bevatten(die ingezet worden voor de webapplicatie), zoals vastgelegd in de documentatie van de software.

3.20. SSD-21: Geparametriseerde queries

SSD-21 Geparametriseerde queries	
Criterium (wie en wat)	De webapplicatie communiceert alleen met onder- en achterliggende systemen op basis van statisch geconfigureerde (geparametriseerde) queries en commando's en uitsluitend ten behoeve van de noodzakelijke functionaliteit.
Doelstelling (waarom)	Kwetsbaarheid van de onder- en achterliggende systemen voorkomen en de integriteit en vertrouwelijkheid garanderen.
Risico	Via manipulatie (bijvoorbeeld commando- of SQL-injectie) kennis nemen van de inhoud van de onder- en achterliggende systemen of deze kunnen manipuleren.

Testvormen en de beschrijving van de testen:

/01 queries en commando's

SSD-21 Geparametriseerde queries							
	indicatoren	I	O	B	S	A	P
/01	queries en commando's						
/01.01	De broncode van de webapplicatie bevat uitsluitend commando- en queryteksten die opgebouwd worden uit in de code reeds aanwezige vaste tekstfragmenten.		R	H A	H A	H A	H A
/01.02	Gebruikersinvoer die gebruikt moet worden in commando's en queries wordt op een zodanige manier doorgegeven, dat de beoogde werking niet wordt gewijzigd.			H	H	H	

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het voorgestelde framework, ORM-laag of de gekozen programeertaal ondersteuning heeft voor geparameteriseerde queries.
Bouw	Stel vast dat database-vraagstelling nooit gebeurd door het aan elkaar plakken van gebruikersinvoer (data) en daadwerkelijke vraagstelling (logica). Stel vast dat het framework, de ORM laag of de programmeertaal zo worden worden aangesproken dat geparameteriseerde databasevragen genereert. Stel vast dat statische code analyse wordt gebruikt als hulpmiddel om onjuist opgestelde queries te detecteren. Stel vast dat libraries e.d. van derden en van de eigen ontwikkelaars geen SQL injectie hebben. Stel vast dat het ontwikkelproces van een component van derde partijen in staat is om eventuele SQL injecties te herkennen en op te lossen (er wordt actief aan ontwikkeld, het is mogelijk om bugs in te schieten of met ontwikkelaars te praten). Stel vast dat geen enkele bron van data wordt vertrouwd als zijnde risicoloos. Zie voor nadere informatie: https://www.owasp.org/index.php/Reviewing_Code_for_SQL_Injection
Systeemtest	*Zie vorige fase*. Zie voor nadere informatie: https://www.owasp.org/index.php/OWASP_Testing_Project https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-

	005)
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat de werking aan het vereiste van /01.02 voldoet.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	

/02 noodzakelijke functionaliteit

SSD-21 Geparametriseerde queries							
	indicatoren	I	O	B	S	A	P
/02	noodzakelijke functionaliteit						
/02.01	Van elke webapplicatie is bekend welke functionaliteit van backend-systemen nodig is.		R	R	R	H	H
/02.02	Directe data-toegang tot backend-systemen is ongewenst en alleen toegestaan indien andere opties niet voor handen zijn.		R	H	H	H	H

fase	Toelichting /02.01
Initiatie	
Ontwerp	Controleer dat de koppelingen met de backend-systemen en de noodzaak ervan zijn beschreven. Verifieer of dit voor alle lagen in het systeem geldt.
Bouw	Controleer dat van de koppelingen met de backend-systemen het protocol en de benodigde rechten zijn beschreven.
Systeemtest	*Zie vorige fase*
Accept. Test	Controleer dat van de koppelingen met de backend-systemen beperkt zijn tot de beschreven koppelingen, protocollen en rechten.
Productie	*Zie vorige fase*

fase	Toelichting /02.02
Initiatie	
Ontwerp	Stel vast dat in het ontwerp zich geen directe koppelingen met de back-end systemen zijn opgenomen, indien wel directe koppelingen zijn opgenomen is de noodzaak vastgelegd en vastgesteld en zijn de risico's en maatregelen ge-definieerd.
Bouw	Stel vast dat de in het ontwerp gedefinieerde maatregelen, zoals versleutelen van de database-bestanden of –velden en logging, zijn geïmplementeerd. Stel vast dat de maatregelen afdoende als worden gezien in de markt ofwel dat ze marktconform zijn.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.21. SSD-22: Invoervalidatie

SSD-22: Invoervalidatie	
Criterium (wie en wat)	De (web)applicatie beperkt de mogelijkheid tot manipulatie door de invoer te valideren, voordat deze invoer wordt verwerkt.
Doelstelling (waarom)	Voorkom (on)opzettelijke manipulatie van de (web)applicatie, waardoor de vertrouwelijkheid, integriteit en/of beschikbaarheid van de (web)applicatie aangetast worden.
Risico	Inzage, wijziging, verlies, of misbruik van gegevens door bijvoorbeeld manipulatie van de (web)applicatielogica.

Testvormen en de beschrijving van de testen:

/01 valideren

SSD-22: Invoervalidatie							
	indicatoren	I	O	B	S	A	P
/01	valideren						
/01.01	Foute, ongeldige of verboden invoer wordt geweigerd. De (web)applicatie voert deze controle van de invoer uit aan de serverzijde en vertrouwt niet op de maatregelen aan de client-zijde.		R	H	H A	H A	H A
/01.02	De (web)applicatie valideert alle invoer die de gebruiker aan de (web)applicatie verstrekt.			R A H	R A H	H A	H A
/01.03	Voor elke controle die de (web)applicatie uitvoert aan de client -zijde, is er een equivalent aanwezig aan de server-zijde.			R H	H	H	H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het ontwerp een matrix bevat met daarin voor alle invoermogelijkheden met per invoermogelijkheid en per gegeven: • het geldige bereik (grenswaarden), • ongeldige tekens, • wanneer de invoer onvolledig is • de juistheid van het formaat
Bouw	Stel vast dat validatie plaatsvindt van: • het geldige bereik (grenswaarden), • ongeldige tekens, • wanneer de invoer onvolledig is • de juistheid van het formaat Stel aanvullend vast dat de gegevens consistent zijn ten opzichte van andere gegevens binnen de invoer dan wel in andere gegevensbestanden.
Systeemtest	*Zie vorige fase* Afhankelijk van de in de matrix opgenomen validaties kan dit getest worden met behulp van de testgevallen OTG-INPVAL-001 t/m OTG-INPVAL-016 van de OWASP Testing Guide v4.0 (alternatief: een onderbouwde keuze daaruit).
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Zie /01.01, in de testbeschrijving is reeds de controle op compleetheid mee-genomen (controle of alle invoer is meegenomen).
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Stel vast dat de validatie, zoals beschreven is bij /01.01 ook plaatsvindt op het cliëntdeel van de applicatie.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

3.22. SSD-23: File includes

SSD-23: File includes					
Criterium (wie en wat)	De (web)applicatie voorkomt de mogelijkheid van dynamische file includes.				
Doelstelling (waarom)	Voorkom (on)opzettelijke manipulatie van de (web)applicatie, waardoor de vertrouwelijkheid, integriteit en/of beschikbaarheid van de (web)applicatie aangetast worden.				
Risico	Inzage, wijziging, verlies, of misbruik van gegevens door bijvoorbeeld manipulatie van de (web)applicatielogica.				
Referentie	NCSC	NIST	ISO27002		
	B3-7				

Testvormen en de beschrijving van de testen:

/01 dynamische file includes

SSD-23: File includes							
	indicatoren	I	O	B	S	A	P
/01	dynamische file includes						
/01.01	De (web)applicatie maakt geen gebruik van dynamische file includes, tenzij dit vanaf een vertrouwde locatie plaatsvindt.		R	H A	H A	H A	H A
/01.02	De (web)applicatie beperkt (op de server) de keuzemogelijkheid voor het uploaden van bestanden, bijvoorbeeld via whitelisting[8].		R	H	H A	H A	H A
/01.03	In de configuratiehandleiding van de softwareleverancier zijn eventueel benodigde maatregelen beschreven die includes moeten voorkomen.				R	R	
/01.04	De server is zo geconfigureerd door de hostingpartij dat file includes niet mogelijk zijn.						R H

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat het ontwerp geen dynamische file includes toepast, tenzij dit vanaf een vertrouwde locatie plaatsvindt.
Bouw	Stel vast dat de applicatie geen gebruik maakt van dynamische file includes. Controleer dat de server zo is geconfigureerd dat geen dynamische file includes mogelijk zijn.
Systeemtest	Test met behulp van testgeval OTG-INPAL-012 van de OWASP Testing Guide v4.0 (voor zover niet bij SSD 22 al uitgevoerd): • Testing for Code Injection (OTG-INPVAL-012), • Testing for Local File Inclusion - Testing for Remote File Inclusion Stel vast dat de server zo is geconfigureerd dat geen dynamische file includes mogelijk zijn.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

[8] http://en.wikipedia.org/wiki/Whitelist

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat als het ontwerp een dynamische file includes toegepast vanaf een vertrouwde locatie, de vertrouwde locatie is omschreven en als vertrouwd wordt gezien.
Bouw	*Zie vorige fase*
Systeemtest	Stel vast dat de op de server aanwezige mogelijkheden tot het uploaden van bestanden beperkt wordt tot alleen de vertrouwde locaties.
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat in de configuratiehandleiding van de softwareleverancier alle benodigde instellingen zijn beschreven die includes moeten voorkomen.
Accept. Test	*Zie vorige fase*
Productie	

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de server geconfigureerd is conform de in de configuratiehandleiding opgenomen maatregelen ter voorkoming van ongewenste file includes.

3.23. SSD-24: Beperking van te versturen HTTP-headers

SSD-24 Beperking van te versturen HTTP-headers	
Criterium (wie en wat)	De webserver stuurt bij een antwoord aan een gebruiker alleen die informatie in de HTTP-headers mee die van belang is voor het functioneren van HTTP.
Doelstelling (waarom)	Voorkom inzage, wijziging of verlies van gegevens door manipulatie van de logica van de webserver of webapplicatie, door zo min mogelijk overbodige informatie op te nemen in een antwoord aan gebruikers.
Risico	De werking van de webserver of webapplicatie wordt gemanipuleerd, waardoor deze onder controle komt van een aanvaller.

Testvormen en de beschrijving van de testen:

/01 HTTP-headers

SSD-24 Beperking van te versturen HTTP-headers							
	indicatoren	I	O	B	S	A	P
/01	HTTP-headers						
/01.01	De softwareleverancier legt in de ontwerpdocumentatie vast welke HTTP-headers worden gebruikt door de webapplicatie en hoe HTTP-headers uit de antwoorden worden verwijderd.		R				
/01.02	De softwareleverancier legt in de configuratiedocumentatie vast welke HTTP-headers worden gebruikt door de webserver en hoe HTTP-headers uit de antwoorden worden verwijderd.			R	R	R	R
/01.03	De softwareleverancier onderbouwt en beschrijft eventuele noodzakelijke afwijkingen van de standaard configuratie op de webserver, die nodig zijn voor het goed functioneren van de webapplicatie.			R			
/01.04	De webserver verstuurt alleen HTTP-headers die voor het functioneren van HTTP van belang zijn.			H A	H A	H A	H A
/01.05	De webserver verwijdert alle niet noodzakelijke informatie uit de HTTP-header, alvorens deze voor het antwoord te gebruiken.			H A	H A	H A	H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	Stel vast dat er een lijst is gespecifieerd van HTTP-headers die door de applicatie benodigd zijn.
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	

Bouw	Stel vast dat in de configuratiehandleiding is gespecificeerd welke HTTP-headers door de server moeten worden verwijderd welke configuratie-instellingen nodig zijn.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Stel vast dat is beschreven wat de eventueel noodzakelijke afwijkingen van de standaard configuratie op de webserver zijn. Stel vast dat de noodzaak van de afwijking is vastgelegd en vastgesteld.
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	Stel vast dat alleen de toegestane HTTP headers worden gebruikt: 1. Voer een configuratie- of code-review uit om na te gaan of de gespecificeerde lijst van toegestane HTTP-headers is geïmplementeerd. 2. Voer enkele steekproeven uit door aanvragen uit te voeren met niet toegestane headers (cliënt) 3. Voer enkele steekproeven uit door een aantal pagina's te bezoeken en de geretourneerde headers (server) te vergelijken met de lijst van toegestane headers.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

fase	Toelichting /01.05
Initiatie	Zie toelichting /01.01
Ontwerp	
Bouw	Stel vast dat gevoelige informatie uit de HTTP-headers wordt verwijderd (server) of geblokkeerd (cliënt). Stel daartoe door middel van een configuratie- of code-review vast gevoelige informatie uit de HTTP-headers wordt verwijderd (server) of geblokkeerd (cliënt).
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

3.24. SSD-25: Beperken van te tonen HTTP-header informatie

SSD-25 Beperken van te tonen HTTP-header informatie	
Criterium (wie en wat)	De webapplicatie toont alleen noodzakelijke informatie in HTTP-headers die van belang is voor het functioneren van HTTP.
Doelstelling (waarom)	Voorkom dat overbodige technische informatie in een antwoord leidt tot informatie die gebruikt kan worden voor manipulatie van de logica van de webserver of webapplicatie.
Risico	Gedetailleerde informatie over de technische inrichting van de webserver wordt gebruikt, waardoor deze onder controle komt van een aanvaller.

Testvormen en de beschrijving van de testen:

/01 informatie in HTTP-headers

SSD-25: Beperken van te tonen HTTP-header informatie							
	indicatoren	I	O	B	S	A	P
/01	informatie in HTTP-headers						
/01.01	De softwareleverancier legt in de ontwerpdocumentatie vast welke overbodige informatie uit antwoorden wordt verwijderd of tot een minimum wordt beperkt door de webserver.			R			
/01.02	De softwareleverancier onderbouwt en beschrijft eventuele noodzakelijke afwijkingen van de standaard configuratie op de webserver, die nodig zijn voor het goed functioneren van de webapplicatie.			R			
/01.03	De webserver verwijdert de 'Server'-header uit het antwoord of vult hiervoor een nietszeggende inhoud in.			H A	H A	H A	H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat duidelijk is gespecificeerd welke informatie binnen HTTP headers mag worden prijsgegeven en welke niet. Stel vast dat duidelijk is gespecificeerd welke componenten er verantwoordelijk zijn voor het beperken of filteren van identificerende informatie.
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat is beschreven wat de eventueel noodzakelijke afwijkingen van de standaard configuratie op de webserver zijn. Stel vast dat de noodzaak van de afwijking is vastgelegd en vastgesteld.
Systeemtest	
Accept. Test	

Productie	

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	Stel vast dat alleen de toegestane informatie in de HTTP headers wordt gebruikt: 1. Voer een configuratie- of code-review uit om na te gaan of de gespecificeerde lijst van toegestane informatie is geïmplementeerd. 2. Voer enkele steekproeven uit door een aantal pagina's te bezoeken en de geretourneerde headers (server) te vergelijken met de lijst van toegestane headers.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

3.25. SSD-26: HTTP-methoden

SSD-26: HTTP-methoden	
Criterium (wie en wat)	De webserver gebruikt alleen de HTTP-functionaliteiten die nodig zijn voor het functioneren van de webapplicatie.
Doelstelling (waarom)	Voorkom het gebruik van niet noodzakelijke methoden, die gebruikt kunnen worden voor de manipulatie van de logica van de webserver of webapplicatie.
Risico	De werking van de webserver of webapplicatie wordt gemanipuleerd, waardoor deze onder controle komt van een aanvaller.

Testvormen en de beschrijving van de testen:

/01 HTTP-functionaliteiten

SSD-26: HTTP-methoden							
	indicatoren	I	O	B	S	A	P
/01	HTTP-functionaliteiten						
/01.01	Op de webserver worden, indien mogelijk, alleen de GET en POST geactiveerd. De softwareleverancier onderbouwt en beschrijft eventuele noodzakelijke methoden anders dan GET en POST en legt dit vast in de ontwerpdocumentatie.			R			
/01.02	De softwareleverancier legt in de configuratiedocumentatie vast welke HTTP-methoden worden gebruikt door de webserver.			R	R	R	R
/01.03	De hostingpartij waarborgt dat alleen de door de webapplicaties benodigde HTTP-requests methoden op de webserver zijn toegestaan en de overige niet noodzakelijke HTTP-requests methoden zijn gedeactiveerd.					H A	H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat is beschreven welke http methoden noodzakelijk zijn. Wanneer het andere methoden zijn dan GET en POST is de noodzaak vastgelegd en vastgesteld. Voor meer info wordt verwezen naar OWASP_Testing_Guide_v4.pdf (Test HTTP Methods (OTG-CONFIG-006) en W3C Consortium.
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Stel vast dat de configuratiedocumentatie specificeert welke http methoden door de webserver doorgegeven moeten worden aan de applicatie.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*

Productie	*Zie vorige fase.*

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel met OTG-INPVAL-003 en OTG-CONFIG-006 vast dat alleen de gespecificeerde http methoden uit de configuratiedocumentatie werken. Voor het testen kan gebruik gemaakt worden van netcat (linux) of telnet (http) of nmap nse. Voor het gebruiken van nmap als vulnerability scanner moet eerst gecontroleerd worden of de NSE scripts[9] aanwezig zijn. (Een gebruiker kan oefenen op scanme.nmap.org.)
Productie	*Zie vorige fase.*

[9] https://nmap.org/nsedoc/

3.26. SSD-27: Error handling

SSD-27: Error handling	
Criterium (wie en wat)	De webserver neemt in een foutmelding aan de cliënt geen inhoudelijke foutinformatie op die misbruikt kan worden.
Doelstelling (waarom)	Voorkom dat technische informatie over de webapplicatie naar buiten lekt, waardoor manipulatie van de logica van de webserver of webapplicatie mogelijk wordt.
Risico	De foutinformatie wordt gebruikt om de werking van de webserver of webapplicatie te manipuleren, waardoor deze onder controle komt van een aanvaller.

Testvormen en de beschrijving van de testen:

/01 inhoudelijke foutinformatie

SSD-27: Error handling							
	indicatoren	I	O	B	S	A	P
/01	inhoudelijke foutinformatie						
/01.01	Bij het optreden van een fout wordt de informatie in een HTTP-response tot een minimum beperkt. Een eventuele foutmelding zegt wel *dat* er iets is fout gegaan, maar niet *hoe* het is fout gegaan.			H A	H A	H A	H A
/01.02	De softwareleverancier legt in de configuratiedocumentatie vast hoe de webserver en/of de application-level firewall moet worden geconfigureerd om alleen standaard foutmeldingen te tonen en versturen.				R	R	
/01.03	De webserver toont bij een fout geen gedetailleerde informatie aan de cliënt, maar alleen een standaard foutmelding.					H A	H A
/01.04	De webserver en/of application firewall verwijdert alle niet noodzakelijke foutinformatie uit de HTTP-response.					H A	H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Voedt de applicatie met onverwachte, misvormde invoer en analyseer hoe deze wordt verwerkt. Stel vast dat er geen detailgegevens worden gelekt naar eindgebruikers indien er een onverwachte foutsituatie (exceptie) optreedt. Voedt hiervoor de applicatie met onverwachte, misvormde invoer en analyseer hoe deze wordt verwerkt aan de hand van de respons. Stel vast dat in alle gevallen geen of een algemene foutmelding getoond wordt zonder enige (interne) technische details, zoals een exception met een stacktrace e.d.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

fase	Toelichting /01.02

Initiatie	
Ontwerp	
Bouw	
Systeemtest	Stel vast dat de configuratiedocumentatie specificeert hoe de webserver en/of de application-level firewall moet worden geconfigureerd om alleen standaard foutmeldingen te tonen en versturen.
Accept. Test	*Zie vorige fase.*
Productie	

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Voedt de webserver met onverwachte, misvormde invoer en analyseer hoe deze wordt verwerkt. Stel vast dat de webserver zo is ingesteld dat bij een fout geen gedetailleerde informatie aan de cliënt, maar alleen een standaard foutmelding, wordt getoond.
Productie	*Zie vorige fase.*

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Voedt de webserver en/of application firewall met onverwachte, misvormde invoer en analyseer hoe deze wordt verwerkt. Stel vast dat de webserver zo is ingesteld dat bij een fout alle niet noodzakelijke foutinformatie uit de HTTP-response verwijdert.
Productie	*Zie vorige fase.*

3.27. SSD-28: Commentaar(regels)

SSD-28: Commentaar(regels)	
Criterium (wie en wat)	De aan de gebruiker aangeboden scripts / code bevat geen commentaar, zodat die niet kunnen worden misbruikt.
Doelstelling (waarom)	Voorkom dat technische informatie over de webapplicatie naar buiten lekt, waardoor manipulatie van de logica van de webserver of webapplicatie mogelijk wordt.
Risico	Kennis nemen van de technologieën van de webapplicatie, om deze vervolgens te gebruiken om de webapplicatie aan te vallen.

Testvormen en de beschrijving van de testen:

/01 commentaar

SSD-28: Commentaar(regels)							
	indicatoren	I	O	B	S	A	P
/01	commentaar						
/01.01	De applicatieontwikkelaar heeft alle commentaarregels uit de scripts (code) verwijderd.			H A	H A	H A	H A
/01.02	Ook andere elementen dan commentaarregels, zoals META, PARAM, OBJECT en INPUT, bevatten commentaar die onnodig informatie vrijgegeven, zoals gebruikte software, versies of infrastructurele componenten, dienen uit de voor de gebruiker zichtbare code te worden verwijderd.			H A	H A	H A	H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat er geen comentaar, ToDo'tjes e.d aanwezig zijn in de code.
Systeemtest	Stel vast dat de code en scripts die zichtbaar gemaakt kunnen worden op de cliënt (CSS, JS e.d.) geen commentaar bevatten. Let daarbij ook op ontwikkelaar gegevens (naam, email telefoonnumers e.d.) Geautomatiseerd testen is mogelijk door te geautomatiseerd scannen op commentaarregels. Zie ook OWASP Testing Guide; Review webpage comments and metadata for information leakage (OTG-INFO-005)
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

fase	Toelichting /01.02
Initiatie	
Ontwerp	
Bouw	Neem in de testen van /01.01 alle elementen mee, zoals META, PARAM, OBJECT en INPUT.
Systeemtest	*Zie vorige fase.*
Accept. Test	*Zie vorige fase.*
Productie	*Zie vorige fase.*

3.28. SSD-29 Directory Listing

SSD-29 Directory listing	
Criterium (wie en wat)	De aan de gebruiker getoonde informatie bevat geen directory listings, zodat die niet kunnen worden misbruikt.
Doelstelling (waarom)	Compliance met dit criterium zorgt dat een buitenstaander ongeautoriseerd kennis kan nemen van wat in de directories van de (web)server is opgeslagen.
Risico	Kennis nemen van de inhoud van directories biedt de mogelijkheid van het compromitteren van vertrouwelijke informatie of van het kennis nemen van de technologieën van de webapplicatie. Deze kennis kan vervolgens gebruikt worden om de webapplicatie aan te vallen.

Testvormen en de beschrijving van de testen:

/01 directory listings

SSD-29 Directory Listing							
	indicatoren	I	O	B	S	A	P
/01	directory listings						
/01.01	De toegang tot de inhoud van bestanden in directories moet altijd verlopen via de webapplicatie.			H A	H A	H A	H A
/01.02	De softwareleverancier maakt geen gebruik van directory listings, tenzij er bewust voor deze functionaliteit is gekozen. De keuze is in de ontwerpdocumentatie onderbouwd en vastgelegd.		R	R			
/01.03	De softwareleverancier legt in de configuratiehandleiding vast welke directory listings niet uitgeschakeld moeten worden.					R	R
/01.04	De hostingpartij schakelt, met uitzondering van de in de configuratiehandleiding vastgelegde uitzonderingen, de directory listings uit.						R H A

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel handmatig of met behulp van geautomatiseerde tools, zoals URL fuzzer, vast dat bestanden niet rechtstreeks zijn te benaderen.
Systeemtest	Zie vorige fase
Accept. Test	Zie vorige fase
Productie	Zie vorige fase

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat in de ontwerpdocumentatie is vastgelegd tot welke directories (dus alleen rechtstreeks en dus niet via listing) en directory-extensies de applicatie toegang heeft en hoe. Stel vast dat de toegang tot bestanden in directories buiten de webapplicatie alleen gevraagd wordt als deze keuze is onderbouwd.
Bouw	Zie vorige fase

Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.03
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat er een configuratiehandleiding aanwezig is en of de daarin opgenomen directory listings overeenkomen met de onderbouwde keuzen in de ontwerpdocumentatie.
Productie	Zie vorige fase

fase	Toelichting /01.04
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel handmatig of met behulp van geautomatiseerde tools, zoals URL fuzzer, vast dat bestanden niet rechtstreeks zijn te benaderen. Stel vast dat de keuze is onderbouwd, als bestanden wel zijn te benaderen,.

3.29. SSD-30: Applicatie logging

SSD-30: Applicatie logging	
Criterium (wie en wat)	In de (web)applicatieomgeving zijn signaleringsfuncties (registratie en detectie) actief en efficiënt, effectief en beveiligd ingericht.
Doelstelling (waarom)	Het maakt mogelijk eventuele schendingen van functionele en beveiligingseisen te kunnen detecteren en achteraf de juistheid van de uitgevoerde acties, zowel op strategisch als operationeel niveau te kunnen vaststellen.
Risico	Tekortkomingen en zwakheden in de geleverde producten/diensten kunnen niet gesignaleerd worden en herstelacties kunnen niet tijdig worden genomen.

Testvormen en de beschrijving van de testen:

/01 registratie en detectie

SSD-30: Applicatie logging							
	indicatoren	I	O	B	S	A	P
/01	registratie en detectie						
/01.01	De te registreren acties zijn gecentraliseerd.			A H	A H	A H	A H
/01.02	Er is bepaald welke gebeurtenissen en/of beheeractiviteiten aan de webapplicatie vastgelegd moeten worden.		R	R	R	R	
/01.03	In de webapplicatie-infrastructuur zijn detectiesystemen actief voor het detecteren van aanvallen.						A H

fase	Toelichting /01.01
Initiatie	
Ontwerp	
Bouw	Stel vast dat de applicatielogging centraal opgeslagen gaat worden. Deze centrale locatie is vastgelegd en in het ontwerp is omschreven hoe met logging, logresultaten en bestanden om te gaan, specifiek voor meldingen van de applicatie. Stel daarbij vast dat er een lijst is van acties die worden uitgevoerd wanneer bepaalde scenario's zich voordoen. Voorbeelden van deze scenario's zijn o.a. denial of service aanvallen, fraude, tekortkomingen in de logs e.d. Het plan moet inzicht geven in de acties zoals onderzoek, aangifte en bijbehorende drempelwaarden. Stel vast dat de applicatie gebruikmaakt van logvoorzieningen die loginformatie op een gestructureerde manier genereert.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	*Zie vorige fase*

fase	Toelichting /01.02
Initiatie	
Ontwerp	Stel vast dat beschreven is dat minimaal de volgende zaken worden gelogd: • Wijzigingen in de configuratie en beveiligingsinstellingen. • Succesvolle en gefaalde login pogingen. • Alle handelingen rondom gevoelige data, waaronder bekijken.

	• Opvraag en mutatie van bestanden. Stel vast dat de logs antwoord geven op kernvragen als: wie, wat, wanneer, hoe, waarvandaan. Stel vast dat er inzicht wordt gegeven in wat wordt gezien als "verdacht gedrag", situaties die niet voor zouden mogen komen, situaties die altijd nader onderzoek vereisen en dergelijke. Stel vast dat er een overzicht bestaat waarin is vastgelegd en uitgelegd welke onderdelen van de programmatuur logging nodig hebben. Stel vast dat gevoelige onderdelen van de applicatie het soort logging oplevert dat kan functioneren als een audit-trail. Stel vast dat is aangegeven op welke plekken en hoe IDS / SIEM systemen gebruik kunnen maken van de applicatielogging.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	
Productie	

fase	Toelichting /01.03 *In de webapplicatie-infrastructuur zijn detectiesystemen actief voor het detecteren van aanvallen.*
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de applicatie log-informatie kan publiceren naar standaard voorzieningen, zoals het Windows Event Log of Syslog.

/02 efficiënt en effectief

SSD-30: Applicatie logging								
	indicatoren	I	O	B	S	A	P
/02	efficiënt en effectief							
/02.01	De systemen zijn zodanig geconfigureerd dat interne systeemklokken automatisch gesynchroniseerd worden.							H

fase	Toelichting /02.01 *De systemen zijn zodanig geconfigureerd dat interne systeemklokken automatisch gesynchroniseerd worden.*
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	Stel vast dat de logbronnen gesynchroniseerd zijn met een centrale klok zodat er door een SIEM of andere analyse beter dwarsverbanden kunnen worden gelegd bij acties die resulteren in logging door meerdere systemen tegelijkertijd.

/03 beveiligd

SSD-30: Applicatie logging							
	indicatoren	I	O	B	S	A	P
/03	beveiligd						
/03.01	Er is vooraf bepaald wat te doen bij het uitvallen van loggingsmechanismen (alternatieve paden).		R			H	
/03.02	De (online of offline) bewaartermijn voor logging is vastgesteld en komt tot uitdrukking in de configuratie-instellingen van de systemen.		R			H	H
/03.03	De loggegevens zijn beveiligd tegen achteraf wijzigen/verwijderen.						

fase	Toelichting /03.01
Initiatie	
Ontwerp	Stel vast dat is vastgelegd welke onbeschikbaarheid van de loggingsmechanismen acceptabel is. Stel vast dat de vervolgactie is vastgelegd. (De actie kan variëren van geen actie tot het stoppen van de applicatie totdat de logfunctionaliteit weer beschikbaar is.) Stel vast dat als onbeschikbaarheid voorkomen moet worden er een failover is gedefinieerd. (Failovers zijn bijvoorbeeld het schrijven naar een bestand of naar een alternatieve bron.)
Bouw	Stel vast dat de applicatie in staat is om te voorspellen wanneer er problemen onstaan door gebrek aan intern geheugen, hoeveelheid aanvragen per tijdseenheid en dergelijke.
Systeemtest	
Accept. Test	Controleer de werking van de in het ontwerp vastgelegde mechanismen
Productie	

fase	Toelichting /03.02
Initiatie	
Ontwerp	Stel vast dat de bewaartermijnen van de gelogde gegevens is vastgelegd.
Bouw	
Systeemtest	
Accept. Test	Stel vast dat het niet mogelijk is om de applicatie onder de minimum bewaartermijnen is te configureren. Door de configuratie-instellingen aan te passen. Dit blijkt meestal uit de configuratie.
Productie	*Zie vorige fase*

fase	Toelichting /03.03 *De loggegevens zijn beveiligd tegen achteraf wijzigen/verwijderen.*
Initiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat op de logbestanden alleen rechten zijn gegeven om data aan de file toe te voegen. Onder linux is dit de a-vlag voor een bestand, maar dat is niet in ieder filesysteem / kernel mogelijk. (chattr a). Dit is te controleren

	door lsattr op de file uit te voeren. Onder Windows is dit te controleren bij de advanced permissions van een object, waar staat "create folders / append data". Natuurlijk moet de systeemgebruiker dan niet een gebruiker zijn die deze permissies dan mag wijzigen. Stel vast dat bij overige oplossingen er informatie beschikbaar is over hoe append-only operaties mogelijk zijn.
Productie	*Zie vorige fase*

3.30. SSD-31: Standaard stack

SSD-31: Standaard stack	
Criterium (wie en wat)	De (web-)applicatie(-omgeving) maakt gebruik van systeemcomponenten en services die onderdeel zijn van een formeel gespecificeerde stack.
Doelstelling (waarom)	Compliance met dit criterium zorgt ervoor dat integrale beveiliging mogelijk is om adequate bescherming te bieden tegen de actuele bedreigingen.
Risico	Het gebruik van onbekende componenten of services kan leiden tot nieuwe en onbekende zwakheden en risico's.

Testvormen en de beschrijving van de testen:

/01 systeemcomponenten en services

SSD-31 Standaard stack							
	indicatoren	I	O	B	S	A	P
/01	systeemcomponenten en services						
/01.01	Voor het vastleggen en het beheren van de standaard stack is één partij aangewezen.	R					
/01.02	De softwareleverancier maakt uitsluitend gebruik van systeemcomponenten en services die zijn opgenomen in de standaard stack.		R	R	R	R	
/01.03	De softwareleverancier onderbouwt en documenteert afwijkingen op de stack en laat dergelijke afwijkingen goedkeuren door de hostingpartij.			R	R	R	
/01.04	De hostingpartij maakt gebruik van de laatste beveiligingsmaatregelen, beleid en procedures voor de systeemcomponenten en services die in gebruik zijn.					R	R

fase	Toelichting /01.01
Initatiatie	Stel vast dat er een standaard stack is gekozen.
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	
Productie	

fase	Toelichting /01.02
Initatiatie	
Ontwerp	Stel vast dat er een standaard stack is gehanteerd.
Bouw	*Zie vorige fase*
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	

fase	Toelichting /01.03
Initatiatie	
Ontwerp	
Bouw	Stel vast dat als er geen standaard stack is gekozen de afwijking is onder-

	bouwd, is afgestemd en goedgekeurd door de hostingpartij. Stel vast dat is onderbouwd dat de beveiliging door de keuze nog steeds op niveau is en gehouden kan worden.
Systeemtest	*Zie vorige fase*
Accept. Test	*Zie vorige fase*
Productie	

fase	Toelichting /01.04
Initatiatie	
Ontwerp	
Bouw	
Systeemtest	
Accept. Test	Stel vast dat laatste beveiligingsmaatregelen, beleid en procedures voor de systeemcomponenten en services die in gebruik zijn, zijn toegepast.
Productie	*Zie vorige fase*