Plan van Aanpak

Ontwikkeling van software om een trigger box aan te sturen

Studie: Technische Informatica, Delft
School: Haagse Hogeschool
Startdatum: 10 februari 2014
Einddatum: 6 juni 2014
Begeleidende docenten: Anthony van Geest & Tony Andrioli
Begeleiders Brightsight: Rob Bekkers & Remko Foekema
Stagiair: Tom Conijn, 10010017

Inhoudsopgave

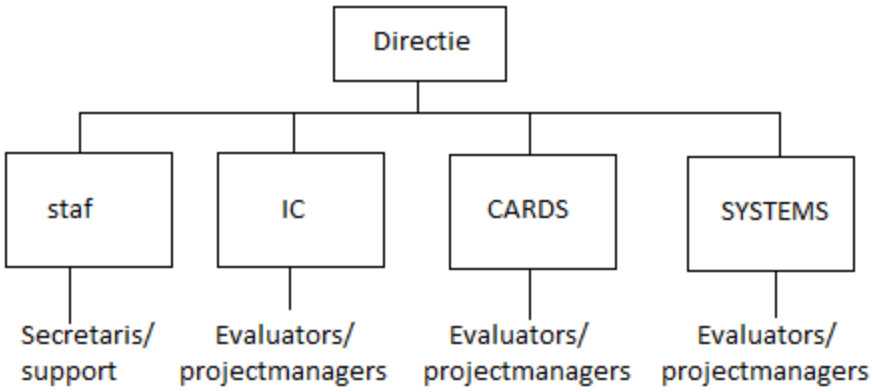
1. Achtergrond

1.1 Bedrijfsomschrijving

Brightsight is een bedrijf gespecialiseerd in het uitvoeren van beveiligingsevaluaties op producten zoals smart card software, betalingsautomaten en geïntegreerde schakelingen. Brightsight is het grootste onafhankelijk evaluatie laboratorium in de wereld en heeft de naam Brightsight sinds 2006. Voordat het bedrijf de naam Brightsight heeft gekregen is het onderdeel geweest van TNO. Het bedrijf is gevestigd in het Delftechpark in Delft.

Brightsight staat in de top vijf van internationale bedrijven die beveiligingsevaluaties uitvoeren. Hierdoor komen opdrachten voor evaluaties uit verschillende delen van de wereld. Om ervoor te zorgen dat er goed met deze buitenlandse klanten gecommuniceerd kan worden is er bij Brightsight personeel uit verschillende landen aangenomen. Dit heeft als nadeel dat niet iedereen gemakkelijk Nederlands spreekt en daarom wordt er vooral in het Engels gecommuniceerd. Hierdoor moet documentatie in het Engels geschreven worden.

1.2 Plaats in de organisatie



Afbeelding 1 Organogram

Brightsight is als organisatie een plat bedrijf. Dit is ook te zien in Afbeelding 1. Brightsight bestaat uit een directie van twee personen. Vervolgens zitten er vier groepen onder de directie. De staf groep zorgt voor de ondersteuning en bestaat uit het secretariaat en de ICT afdeling. De andere drie groepen richten zich op het testen en evalueren van verschillende producten. In het organogram is geen aparte groep beschikbaar voor softwareontwikkeling. Dit project wordt uitgevoerd onder de Integrated Circuits(IC) groep, omdat de begeleider deel uitmaakt van deze groep. Doordat het een informele organisatie is en de begeleider op dezelfde kamer zit als de student is het gemakkelijk om vragen te stellen. Dit zorgt ervoor dat er afspraken gemaakt kunnen worden, zonder dat er interviews gehouden worden.

1.3 Probleemstelling

Brightsight gebruikt verschillende testopstellingen voor het uitvoeren van penetratietesten op producten die beveiligd zijn met cryptografie. Al deze testopstellingen worden aangestuurd met hetzelfde softwarepakket dat door Brightsight is ontwikkeld. Dit softwarepakket wordt de Matrix genoemd.

In een eenvoudige opstelling voor het testen van een secure microcontroller wordt gebruik gemaakt van drie verschillende apparaten, namelijk een oscilloscoop, de te testen microcontroller en een laser. Hierbij wordt het stroomverbruik van de microcontroller gemeten met de oscilloscoop. Op het moment dat de microcontroller een cryptografische berekening uitvoert heeft dit een verandering in het stroomverbruik tot gevolg.

Op het moment dat het stroomverbruik een vooraf bepaalde drempelwaarde overschrijdt, wordt een signaal naar de laser gestuurd om een fout te injecteren in de microcontroller. Deze overschrijding kan echter ook het gevolg zijn van verstoringen in het meetsignaal, waardoor de laser niet op het juiste moment wordt aangestuurd.

Om nauwkeuriger het moment te kunnen bepalen waarop de laser aangestuurd moet worden, is een apparaat aangeschaft waarmee een serie van opgeslagen meetwaarde herkend kan worden. Dit apparaat is de icWaves trigger box, die speciaal voor dit doel ontwikkeld is door de firma Riscure.

2 Projectopdracht

In dit hoofdstuk wordt het doel beschreven van de afstudeeropdracht, het verwachte resultaat en de eisen van de opdrachtgever.

2.1 Doelstelling

Het doel van dit project is het ontwikkelen van een module waarmee de icWaves trigger box, vanuit het intern ontwikkelde softwarepakket de Matrix, aangestuurd kan worden.

Dit betekent dat het mogelijk moet zijn om de volgende acties uit te voeren vanuit de Matrix:

- Er moet een signaal opgenomen worden met de icWaves. Dit kan bijvoorbeeld het stroomverbruik zijn van een microcontroller die getest wordt.

- Er moet een patroon herkend worden, dat bestaat uit een aantal opeenvolgende meetpunten die geselecteerd worden uit het opgenomen signaal.

- Het opgeslagen signaal moet herkend worden, waarna een signaal gegenereerd wordt. Het signaal dat gegenereerd wordt kan bijvoorbeeld een laser laten weten dat een fout geïnjecteerd moet worden.

2.2 Resultaat

Aan het eind van de afstudeeropdracht kan de icWaves trigger box vanuit de Matrix aangestuurd worden. Hierdoor is het mogelijk om met een hogere nauwkeurigheid een fout te injecteren, waardoor er met meer zekerheid een advies gegeven kan worden over het beveiligingsniveau van de secure microcontroller.

2.3 Eisen

Aan de hand van een gesprek met de opdrachtgever en de begeleider is aan het begin van het project duidelijk geworden dat de opdrachtgever zo optimaal mogelijk gebruik wil maken van de icWaves. Dit betekent dat het mogelijk moet zijn om alle functionaliteiten van de icWaves te gebruiken vanuit de module voor de Matrix.

Om een beter beeld te krijgen wat alle functionaliteiten zijn die de icWaves aanbied is de meegeleverde documentatie onderzocht. Hierbij zit een Excelbestand met een overzicht van functies, die ondersteund worden door de icWaves. Dit overzicht is te zien in Afbeelding 5. Door gebruik te maken van dit bestand en overleg met de opdrachtgever zijn de functionele eisen tot stand gekomen. De niet-functionele eisen zijn vervolgens opgesteld door overleg met de opdrachtgever en informatie dat gevonden is op het interne netwerk.

Related parameters	Target function block	Parameter Setter call	Parameter Getter call
Maximum sampling rate	**N/A**	N/A	icwaves_get_max_sampling_rate()
Time base	**Time Base**	icwaves_set_timebase()	icwaves_get_timebase()
AC/DC coupling	**AD/DC Coupling**	icwaves_set_acdc_switch()	icwaves_get_acdc_switch()
Signal input voltage range	**Input Range Select**	icwaves_set_range()	icwaves_get_range()
Maximum trace length	**N/A**	N/A	icwaves_get_max_trace_length
Acquiring trace	**Acquisition Control**	icwaves_start_acquisition()	icwaves_get_acq_data()
Maximum SAD tree size	**N/A**	N/A	icwaves_get_max_sad_size()
Number of pattern	**Pattern Matching**	icwaves_set_number_of_patterns()	icwaves_get_number_of_patterns()
Threshold	**Pattern Matching**	icwaves_set_threshold()	icwaves_get_threshold()
Hold off time	**Pattern Matching**	icwaves_set_holdoff()	icwaves_get_holdoff()
Pattern count	**Pattern Matching**	icwaves_set_pattern_count()	icwaves_get_pattern_count()
Pattern count timeout	**Pattern Matching**	icwaves_set_pattern_count_timeout()	icwaves_get_pattern_count_timeout()
Arming trigger	**Trigger Module**	icwaves_set_arm()	icwaves_get_arm()
Trigger counter	**Trigger Module**	N/A	icwaves_get_trigger_counter()
Trigger delay	**Trigger Module**	icwaves_set_trigger_delay()	icwaves_get_trigger_delay()
Input signal range	**Filter**	icwaves_set_filter_input_range()	icwaves_get_filter_input_range()
Filter center frequency	**Filter**	icwaves_set_filter_frequency()	icwaves_get_filter_frequency()

Afbeelding 2 Overzicht functies van de icWaves

Hieronder is een lijst te zien van de opgestelde functionele en niet-functionele eisen.

Functionele eisen:

1. De icWaves moet zichtbaar zijn in het instrumentenoverzicht van de Matrix.
2. De module moet alle functies uit Afbeelding 5 kunnen aanroepen op de icWaves.
3. De module moet een patroon kunnen opslaan op de icWaves welke herkend moet worden.
4. Het moet mogelijk zijn om de icWaves aan te sturen met behulp van de Script Engine binnen de Matrix.
5. De module moet de volgende oscilloscoopfuncties ondersteunen.
    a. De icWaves laten wachten op een triggersignaal om een meting te starten.
    b. De opgenomen data vanuit de icWaves als Waveform[1] aanbieden aan de Matrix.
6. De instellingen van de icWaves moeten aangepast kunnen worden met behulp van een eigenschappenscherm.
7. Een patroon moet geselecteerd kunnen worden met behulp van het plotscherm in de Matrix.

Niet-functionele eisen:

1. De module moet werken binnen de Matrix.
2. De module moet ontwikkeld worden in de taal Java.
3. De module moet ontwikkeld worden in de Netbeans ontwikkelomgeving.
4. De module moet werken op Windows.
5. Code documentatie moet gemaakt worden met Javadoc.
6. Code commentaar moet in het Engels geschreven zijn.
7. Code standaarden van Java Code Conventions moeten gebruikt worden (Oracle).
8. Tests moeten geschreven worden met behulp van JUnit.

[1] Een Waveform is een klasse binnen de Matrix waarin metingen opgeslagen kunnen worden.

3 Projectgrenzen

In dit hoofdstuk wordt aangegeven wat de grenzen van het project zijn. Het project duurt 17 weken met 5 werkdagen per week. Het project begint 10-02-2014 en eindigt op 6-06-2014.

Het eindproduct zal voldoen aan de eisen die besproken zijn in hoofdstuk 2 Projectopdracht. Dit gebeurt door de planning te gebruiken uit hoofdstuk 8 en rekening te houden met de risico's uit hoofdstuk 10.

Bij dit project is het niet de bedoeling dat er extra modules worden geschreven om analyse te doen op de opgenomen signalen. Ook is het niet nodig om nieuwe modules te maken voor het weergeven van grafieken, er kan gebruik gemaakt worden van bestaande functionaliteit in de Matrix. Het kan wel zo zijn dat deze bestaande modules moeten worden aangepast zodat deze werken met de icWaves module.

4 Aanpak

In dit hoofdstuk wordt de keuze van de ontwikkelmethode besproken.

Bij de keuze van een goede ontwikkelmethode moet rekening gehouden worden met één randvoorwaarde van de opdrachtgever. De software moet gemaakt worden in onderdelen, waarbij elk onderdeel als afgerond product, dus getest en gedocumenteerd, opgeleverd kan worden.

Voor het kiezen van de ontwikkelmethode zijn de volgende iteratieve ontwikkelmethodes onderzocht: Rational Unified Process (RUP), Scrum, Feature-Driven Development (FDD) en Extreme Programming (XP). De methodes zijn in tabel 1 gezet en worden vergeleken met eisen die hieraan gesteld zijn. Deze eisen zijn opgesteld aan de hand van de opdracht en de randvoorwaarde van de opdrachtgever.

In de tabel worden onderstaande tekens gebruikt:

+ De methode voldoet aan de eis.
- De methode voldoet niet aan de eis.
~ De methode ondersteunt de eis niet optimaal.

Eigenschappen/eisen	**Rup**	**Scrum**	**FDD**	**XP**
Iteratieve aanpak: maakt gebruik van verschillende iteraties.	+	+	+	+
Werkend product per iteratie: levert een werkend product af per iteratie.	-	+	-	-
Testen per iteratie: voert testen uit per iteratie.	+	~	+	+
Voortgang bewaking: heeft genoeg momenten gepland waarop de voortgang gecontroleerd wordt.	~	+	~	+
Gedetailleerde planning: heeft een duidelijk overzicht van de taken die uitgevoerd moeten worden tijdens een iteratie.	+	+	+	-
Omgang bij verandering eisen: kan omgaan met veranderingen van eisen tijdens het project.	~	+	+	+

Individueel gebruik: is mogelijk om individueel uit te voeren.	~	-	+	-
Documentatie: Heeft genoeg momenten om documentatie te maken.	+	+	~	~

Tabel 1 Vergelijking ontwikkelmethodes

Wanneer er gekeken wordt naar de eisen die gesteld zijn aan de ontwikkelmethodes is te zien dat alle gekozen methodes gebruik maken van iteraties. Bij het vergelijken van de overige eisen komt Scrum het beste in de buurt van een bruikbare methode. Dat komt voornamelijk doordat deze voldoet aan de voorwaarde van de opdrachtgever, terwijl RUP, FDD en XP dit niet doen.

Bij gebruik van Scrum is er een probleem, omdat het project maar door één persoon wordt uitgevoerd. Dit betekent dat de dagelijkse gesprekken die voorgeschreven worden bij Scrum niet uitgevoerd kunnen worden. Dit wordt tijdens het project opgevangen door korte lijnen met zowel de opdrachtgever als de begeleider. De problemen die ontstaan bij de ontwikkeling of het proces kunnen gelijk besproken worden met de begeleider. Door wekelijks overleg is er controle op de voortgang van het project en worden problemen omtrent de planning snel gedetecteerd.

Bij Scrum wordt er gebruik gemaakt van sprints oftewel iteraties met een lengte tussen de één en vier weken. Aan het begin van een sprint worden de taken gekozen die tijdens deze sprint uitgevoerd worden. Hiervoor wordt gebruik gemaakt van een tabel met taken, waarbij per taak een aantal uren zijn gepland. Doordat aan het begin van een sprint taken worden gekozen is het mogelijk om, afhankelijk van de prioriteiten van de opdrachtgever, de volgorde van werken aan te passen.

De keuze voor Scrum is gemaakt tijdens de oriëntatiefase die in het volgende hoofdstuk besproken wordt. In deze fase is een globale planning gemaakt waarin staat wat er per sprint uitgevoerd moet worden. Het verslag is vervolgens ingedeeld naar aanleiding van deze planning.

5 Producten

In dit hoofdstuk staan alle producten opgesomd. Belangrijke evenementen staan ook genoteerd als product.

- Plan van aanpak
- Broncode van de module
- Test code
- Code documentatie
- Afstudeerverslag
- Bedrijfsbezoek begeleider op ongeveer 25% van de doorlooptijd
- Opleveren van voortgangsverslag aan de begeleidende examinator op ongeveer 45% van de doorlooptijd
- Bespreken van het concept-afstudeerverslag op ongeveer 60%-70 van de doorlooptijd
- Tussentijds assessment uiterlijk 3 werkweken voor de inleverdatum van het afstudeerverslag (6 juni)

6 Kwaliteit

In dit hoofdstuk wordt besproken hoe de kwaliteit van de opgeleverde producten wordt gewaarborgd.

Om de kwaliteit van zowel de tussenproducten als het eindproduct te garanderen, wordt er gebruik gemaakt van code standaarden. Deze code standaarden zijn te vinden op de site van Oracle [2]. Door gebruik te maken van code standaarden wordt er gezorgd dat de code leesbaar blijft en dat andere ontwikkelaars gemakkelijk de code kunnen begrijpen.

Behalve het gebruik van code standaarden worden er ook tests uitgevoerd op de producten. De beschrijving van de tests en de resultaten worden later gemaakt en opgeleverd in een apart document. Aan de hand van deze tests kan gecontroleerd worden of het product naar behoren werkt.

Om de kwaliteit en de voortgang van het project in de gaten te houden is er elke week een overleg met de opdrachtgever en de begeleider. In dit overleg kan worden aangegeven als er problemen zijn of verwacht worden, zodat dit op tijd opgelost kan worden.

[2] http://www.oracle.com/technetwork/java/codeconventions-150003.pdf

7 Projectorganisatie

Dit hoofdstuk gaat over hoe de organisatie is geregeld van dit project.

De student Tom Conijn voert dit project uit.

7.1 Begeleiders

Tijdens dit project wordt de student begeleid door een aantal verschillende personen. Deze personen zijn te verdelen in twee groepen. Als eerste de begeleiders vanuit het bedrijf en als tweede de begeleiders vanuit school.

Bedrijf

De begeleiders vanuit het bedrijf bestaat uit twee personen. Als eerste is er Rob Bekkers, hij is werknemer bij Brightsight en is tevens de opdrachtgever. Als tweede is er Remko Foekema, hij is werknemer bij Brightsight en begeleid de student verder bij het uitvoeren van dit project.

School

De tweede groep bestaat uit begeleiders vanuit school. Als eerste persoon is dit de heer A. van Geest. De heer Van Geest is het aanspreekpunt voor de student en komt ook op bezoek bij Brightsight. De tweede begeleider is de heer A. Andrioli. Hij is de tweede examinator van dit project.

7.2 Communicatie

Bij dit project wordt er met het bedrijf op verschillende manieren gecommuniceerd. Dit gebeurt via de telefoon, e-mail en mondelinge gesprekken. Ook is er wekelijks een vast moment waarop de voorgang besproken wordt met de twee begeleiders binnen het bedrijf.

Communicatie met de begeleider van school gebeurt voornamelijk via e-mail.

Op de volgende pagina zijn de contactgegevens te vinden van de betrokken personen bij dit project.

7.3 Contactgegevens

Opdrachtgever

Naam:	Rob Bekkers
Organisatie:	Brightsight
Functie:	Senior Security Evaluator
E-mailadres:	bekkers@brightsight.com

Bedrijfsbegeleider

Naam:	Remko Foekema
Organisatie:	Brightsight
Functie:	Senior Security Evaluator
E-mailadres:	foekema@brightsight.com

Begeleider/examinator 1

Naam:	Anthony van Geest
Organisatie:	De Haagse Hogeschool
Functie:	Business Architect
E-mailadres:	a.vangeest@intermedio.eu

Begeleider/examinator 2

Naam:	Tony Andrioli
Organisatie:	De Haagse Hogeschool
Functie:	Docent
E-mailadres:	a.andrioli@hhs.nl

Opdrachtnemer

Naam:	Tom Conijn
Organisatie:	Brightsight
Functie:	Stagiair
E-mailadres:	tom.conijn@hotmail.com

8 Planning

Bij de planning wordt elke week gewerkt aan het eindverslag, dit wordt dus niet genoemd in de planning. De planning is te zien in onderstaande afbeelding.

Planning	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16	Week 17
Oriëntatiefase																	
Plan van Aanpak maken		x															
Eisen opstellen		x															
Ontwikkelmethode kiezen		x	x														
Onderzoek ontwikkel mogelijkheden icWaves	x	x	x														
UML diagrammen maken			x														
Implementatie eerste versie																	
Basis module opzetten				x													
Signaal opnemen toevoegen				x	x												
Patroon herkenning toevoegen					x	x											
Documentatie maken							x										
Testen uitvoeren						x	x										
Implementatie tweede versie																	
Trigger generatie toevoegen								x	x								
Filter toevoegen									x	x							
Docmumentatie maken											x						
Testen uitvoeren											x						
Ontwikkeling gebruikersinterface																	
Eisen gebruikersinterface opstellen												x	x				
Gebruikersinterface maken												x	x	x			
Gebruikersinterface testen														x	x		
Afronding project																	
Handleiding schrijven																x	
Eindproduct testen																x	
Verslag afmaken																	x
Belangrijke evenementen																	
Bedrijfsbezoek begeleider				x													
Opleveren voortgangsverslag							x										
Bespreken concept/afstudeerverslag											x						
Tussentijds assessment														x			

Afbeelding 3 Globale planning

In de planning is te zien dat er drie weken gereserveerd zijn voor de oriëntatiefase. De werkzaamheden van deze fase worden besproken in dit hoofdstuk. Na deze drie weken zijn er drie sprints gepland, waarin gewerkt wordt aan de eerste versie van de module, de tweede versie van de module en de ontwikkeling van de gebruikersinterface. De laatste twee weken zijn geen onderdeel van een sprint, omdat deze bedoeld zijn om het project af te ronden en een handleiding te schrijven over het gebruik van de icWaves trigger box.

De planning die gemaakt is gaat uit van drie sprints van elk vier weken. Er is gekozen voor sprints van vier weken, omdat nog niet alle werkzaamheden bekend zijn en er genoeg tijd uitgetrokken moet worden om zowel de implementatie als het testen uit te voeren. Wanneer er gebruik wordt gemaakt van sprints van vier weken is er meer tijd om eisen helder te krijgen en de implementatie uit te voeren. Bij dit project is het van belang dat er een langere tijd per sprint beschikbaar is. Vooral bij de implementatie is de kans groot dat er problemen optreden die meer tijd kosten dan vooraf verwacht. Bij een tweewekelijkse sprint kunnen problemen bij de implementatie het eindproduct in gevaar brengen, omdat er te weinig tijd over is om te testen of documentatie te schrijven.

In de planning is gekozen om in de eerste sprint te werken aan het opzetten van de basis voor de module, het opnemen van een signaal en het herkennen van een patroon. Het opslaan van een signaal wordt als eerste uitgewerkt, zodat de overige onderdelen dit opgeslagen signaal kunnen gebruiken. Bij de eerste sprint wordt extra tijd uitgetrokken voor het schrijven van documentatie en het kiezen van een teststrategie.

In de tweede sprint kan de overige functionaliteit worden toegevoegd. De belangrijkste onderdelen hiervan zijn het genereren van een triggersignaal en het instellen van de filter. Dit kan in dezelfde sprint toegevoegd worden, omdat al een basis is opgezet voor de module en het schrijven van schrijven van documentatie en testen kost minder tijd.

De derde sprint bestaat uit het maken van de gebruikersinterface. Hiervoor is gekozen, omdat het een belangrijk onderdeel is voor het gebruik van de module. Met de huidige planning is hiervoor alle benodigde functionaliteit toegevoegd en moet dit alleen nog beschikbaar worden voor de gebruiker.

9 Kosten en baten

In dit hoofdstuk worden de kosten en baten besproken die dit project met zich mee brengt

9.1 Student

De student besteed 17 werkweken aan de afstudeeropdracht, wat overeenkomt met 680 uur. In ruil voor deze tijd krijgt de student een stagevergoeding en 30 studiepunten die nodig zijn om de opleiding te halen.

9.2 Bedrijf

Het bedrijf betaalt de student een stagevergoeding en investeert tijd in de begeleiding van de student. Het bedrijf krijgt hiervoor een module binnen de Matrix waarmee de trigger module bestuurd kan worden. Door de module kan er met een betere nauwkeurigheid een fout worden geïnjecteerd. Hierdoor kan er met meer zekerheid een advies gegeven kan worden over het beveiligingsniveau van de secure microcontroller.

10 Risico’s

Bij het organiseren en uitvoeren van een project zijn er allerlei risico’s die het succes van het project kunnen beïnvloeden. In dit hoofdstuk worden daarom de verschillende risico’s beschreven en de maatregelen om deze te voorkomen.

De risico’s die op dit moment onderkend zijn hieronder te vinden.

Risico	Kans	Effect	Maatregel
Het project loopt uit	2	Te weinig tijd om het project af te ronden	Een planning maken en per weer kijken of deze nog klopt
Er is niet genoeg kennis	1	Te weinig tijd om het project af te ronden	Vragen voor uitleg bij een van de begeleiders of werknemers bij Brightsight
De eisen veranderen	2	De opdrachtgever krijgt niet het gewenste product	Eisen een prioriteit geven en met de belangrijkste eisen beginnen en goed communiceren met de opdrachtgever
Verlies van data, zowel code als documentatie	1	Te weinig tijd om het project af te ronden	De code kan veilig opgeslagen worden door gebruik te maken van versiebeheer. Voor de documentatie kan gebruik worden gemaakt van google docs.