Plan van Aanpak
Afstuderen

Michiel Graat

27-09-2005

Inhoudsopgave

Hoofdstuk 1

Inleiding

In dit document wordt beschreven hoe ik mijn afstudeeropdracht, die loopt in de periode september 2005 - januari 2006, ga aanpakken. Dit plan van aanpak dient als richtlijn en als planning bij mijn afstuderen; ik leg hierin zowel de opdracht als het proces vast.

1.1 Terminologie

Voordat ik dieper op mijn opdracht in ga, zal ik eerst de terminologie uitleggen.

Code guidelines/standards
In de industrie worden vaak *coding guidelines/standards* gebruikt, waar programmeurs zich aan dienen te houden. Het doel hiervan is om de kwaliteit van de software te verbeteren.

Code reviews
Voor software waar kwaliteit van groot belang is, worden *code reviews* gedaan. Bij zo'n code review wordt de source code door een expert bekeken, om na te gaan of de coding standards gerespecteerd zijn en om eventuele bugs te ontdekken.

Source code analyzers
Source code analyzers zijn tools die nagaan of software voldoet aan bepaalde vooraf opgestelde criteria. Dit betekent dat source code analyzers een gedeelte van de code reviews voor hun rekening kunnen nemen. Overigens werken sommige source code analyzers op object code (zoals bv. Java bytecode) in plaats van op source code. Daarom wordt in plaats van source code analyse vaak over *statische analyse (static analysis)* gesproken.

1.2 Opdracht

Ik zal mijn afstudeeropdracht uitvoeren in de vorm van een onderzoek. Doel van het onderzoek is om na te gaan tot in hoe verre de code analyse van Java Card applicaties geautomatiseerd kan worden en dus of deze code analyse goedkoper kan. Daarvoor zal ik eerst een onderzoek doen naar code analyzers voor gewone Java applicaties, daar de meest geschikte uitkiezen en proberen deze uit te breiden met functionaliteit voor Java Card applicaties. Uiteindelijk zal ik een rapport (in de vorm van een scriptie) over dit onderzoek opleveren.

1.3 JavaCard

Java Card is een technologie waarbij kleine Java-based applicaties (applets) veilig gedraaid kunnen worden op smart cards. Het wordt onder andere veel gebruikt in SIM cards in mobiele telefoons. De belangrijkste eigenschappen van deze technologie zijn portability en security. De Java Card specification werd ontwikkeld door Sun Microsystems.

1.4 Non Disclosure Agreement

Sommige gedeeltes van mijn afstudeerscriptie zullen onder een zogenaamde Non Disclosure Agreement (NDA) vallen. Dit komt omdat ik tijdens mijn scriptie gebruik zal maken van JavaCard coding standards uit de industrie waarvan de inhoud niet openbaar gemaakt mag worden. Concreet houdt dit in dat sommige gedeeltes van mijn afstudeerscriptie niet voor iedereen beschikbaar zullen zijn. Ik zal in dit Plan van Aanpak zoveel mogelijk aangeven voor welke gedeeltes dit geldt.

1.5 Begeleiding

Ik word begeleid door dr.ir. Erik Poll, werkzaam op de Security of Systems afdeling. Hij zal de academische kant van mijn afstuderen in de gaten houden en zal ondersteuning bieden bij het schrijven van mijn scriptie.

Om de voortgang in de gaten te houden zal ik eenmaal per twee weken contact met Erik Poll hebben.

1.6 Over dit document

Dit document is, naast deze inleiding, opgedeeld in nog vier hoofdstukken. Eerst zal ik de onderzoeksvraag geven. Daarna zal ik de opdracht omschrijven en

vervolgens precies vastleggen. Hierna zal ik een korte beschrijving geven van wat het proces dat hierbij hoort inhoudt. Tenslotte hoort bij dat proces een planning en ook die zal ik geven.

Hoofdstuk 2

Onderzoeksvraag

In mijn onderzoek staan Java Card en code analyzers centraal. Daarom wordt mijn onderzoeksvraag de volgende:

Tot in hoe verre is het mogelijk om de code analyse voor Java Card applicaties te automatiseren?

Hoewel ik naar Java Card kijk, zal mijn onderzoek ook antwoord geven op algemenere vragen, namelijk de volgende:

- Hoe goed zijn moderne source code analyzers voor Java?
- Welke soorten van eigenschappen kunnen zulke tools herkennen?
- Hoe makkelijk is het zulke tools uit te breiden voor specifieke coding guidelines?

Hoofdstuk 3

Opdracht

In dit hoofdstuk zal ik de opdracht die ik ga uitvoeren beschrijven. Doel van dit hoofdstuk is precies af te bakenen wat wel en wat niet bij mijn opdracht hoort. Doel van mijn afstuderen als geheel is alles dat in dit hoofdstuk beschreven staat uit te voeren en te beschrijven in een scriptie.

3.1 Huidige situatie

Java code kan voor een deel automatisch geanalyseerd worden door tools als FindBugs, PMD en Jtest. Deze zijn echter niet op de Java Card gericht. Op dit moment wordt Java Card code dan ook met behulp van JavaCard coding standards uit de industrie met de hand geanalyseerd door bedrijven als TNO-ITSEF en Riscure. Dat document beschrijft veel gemaakte fouten. Dit is uiteraard een verre van ideale situatie.

3.2 Opdrachtomschrijving

Mijn opdracht is om een onderzoek te doen naar de mogelijkheden tot automatisering van code analyse van Java Card applicaties.

Concreet houdt dit het volgende in:

- Het doen van onderzoek naar (en het maken van een evaluatie van) bestaande code analyzers voor Java code.
- Het vergelijken van deze analyzers om de meest geschikte eruit te kiezen.
- Deze code analyzer uitbreiden zodat hij Java Card code kan analyseren.

- Nagaan welke soorten fouten die in de coding standards beschreven staan wel en welke niet met automatische code analyse kunnen worden gevonden.
- Het beschrijven van bovenstaande onderzoek in een scriptie.

Na het uitvoeren van mijn opdracht zullen dus de volgende zaken opgeleverd worden:

- Een scriptie met een uitgebreid verslag van het onderzoek.
- Enkele uitgewerkte voorbeelden over hoe een code analyzer Java Card code analyseert (als onderdeel van mijn scriptie).

3.3 Belang

Automatische code analyse van Java Card applicaties is van groot belang. Wanneer de analyse versneld kan worden door het automatiseren van deze analyse, dan scheelt dit uiteraard in de kosten. Java Card applicaties worden veel gebruikt en de code dient vanwege security overwegingen geanalyseerd te worden, dus automatisering zou een enorme kostenbesparing kunnen opleveren.

Maar het belang van dit onderzoek beperkt zich niet tot Java Card. Ook in andere applicatiedomeinen bestaan specifieke guidelines, bv. voor Java MIDP applets (zoals spellen op een mobiele telefoon) of realtime Java applicaties. Verder hebben veel bedrijven hun eigen coding guidelines. Omdat dit onderzoek een indruk zal geven van de mogelijkheden van source code analyzers, is het interessant voor een veel breder publiek dan bedrijven die zich op Java Card richten alleen.

Hoofdstuk 4

Proces

Mijn afstuderen is een proces dat ik in dit hoofdstuk zal beschrijven. Niet alleen de verschillende fases binnen mijn afstuderen komen hier aan bod, maar ook de verschillende *milestones* die deze fases moeten opleveren. Al deze milestones zullen onderdeel worden van mijn scriptie. In hoofdstuk 5 zal de planning van deze fases besproken worden.

4.1 Plan van Aanpak

Het doel van deze fase is het schrijven van het document dat u nu leest: het Plan van Aanpak. Zie hoofdstuk 1 voor een beschrijving van dit document.

Milestone het opleveren van dit document

4.2 Inventarisatie & vergelijk van source code analyzers

In deze fase zal ik onderzoek doen naar de verschillende code analyzers voor Java applicaties. Deze analyzers zal ik beschrijven, met elkaar vergelijken en redenen opnoemen waarom de ene analyzer meer geschikt is voor uitbreiding met functionaliteit voor Java Card code dan de andere. Het uiteindelijke doel is het vinden van de voor Java Card code meest geschikte analyzer. Het kan gebeuren dat er meer dan één analyzer geschikt lijkt. In dat geval zullen de betreffende analyzers eerst uitgeprobeerd moeten worden (zie ook hoofdstuk 4.3), om de meest geschikte te bepalen.

Milestone het opleveren van een onderzoeksrapport (als onderdeel van mijn scriptie) over bovenstaande fase

4.3 Experimenten met source code analyse voor Java Card

In deze fase zal ik met de analyzer, die ik in de fase beschreven in hoofdstuk 4.2 gevonden heb, experimenteren. Deze experimenten houden in dat ik zal proberen om de analyzer uit te breiden met eenvoudige checks. Het doel van deze experimenten is om er achter te komen met welke soort analyses de analyzer uitgebreid kan worden, hoeveel werk dit uitbreiden is en hoe goed de analyzer met die uitbreidingen is (dit kan bv. gemeten worden aan de hoeveelheid false positives).

4.4 Automatische code analyse voor Java Card

In deze fase zal ik proberen om de verschillende fouten die in de coding standards beschreven staan in te delen in categoriën. Deze categoriën moeten aangeven welke soorten fouten eenvoudig en welke moeilijker of niet met behulp van de uitbreidingen op de code analyzer te vinden zijn. Van de fouten in de categoriën die wel met de code analyzer te vinden zijn, zal ik enkele voorbeelden implementeren.

Milestone het opleveren (als onderdeel van mijn scriptie) van uitgewerkte voorbeelden over hoe een code analyzer Java Card code analyseert.

N.B.: dit gedeelte van mijn onderzoek zal onder een NDA vallen.

4.5 Afronding

Het afsluiten van mijn stage gebeurt door het schrijven van een scriptie (en later, het houden van een presentatie). In deze scriptie beschrijf ik precies wat ik tijdens mijn onderzoek allemaal gedaan heb, welke keuzes ik gemaakt heb en waarom, welke problemen ik ben tegengekomen en hoe ik deze opgelost heb, welke zaken ik de volgende keer anders zou aanpakken en waarom.

Ik zal mijn scriptie voor een deel al tijdens de hiervoor genoemde fases schrijven, om zoveel mogelijk informatie op te schrijven op het moment dat het voor mij nog het meest duidelijk is.

Milestone het opleveren van mijn scriptie

Hoofdstuk 5

Planning

In dit hoofdstuk zal ik een globale planning geven van mijn afstuderen, gebaseerd op de opdeling in fases zoals die in hoofdstuk 4 is besproken. Ik zal proberen mij zo strikt mogelijk aan deze planning te houden en dus zal mijn voortgang aan deze planning getoetst kunnen worden.

- **Week 38 - Week 39 (19 september - 30 september): Plan van Aanpak**
- **Week 40 - Week 45 (3 oktober - 11 november): Onderzoek**
- **Week 46 - Week 2 (14 november - 13 januari): Implementatie**
- **Week 3 - Week 4 (16 januari - 27 januari): Afronding**