**Productbeschrijving**

Project SecMon

Opdrachtgever: Arnim Eijkhoudt, Hogeschool van Amsterdam

Project: Project SecMon

Originele Auteurs: Dave van Ast, Stanley Numan, Jasper Groot, Sujen Kandasamy en Vinesh Koenjbiharie

Verbeterd door: Christiaan Druif

Datum: 30 juni 2014

Versie: 0.3

Status: Final

# Versiebeheer

|  |  |  |  |
| --- | --- | --- | --- |
| Versie | Datum | Omschrijving | Auteur |
| 0.1 | 28-06-2013 | Final versie | Dave, Stanley, Jasper, Sujen en Vinesh |
| 0.2 | 27-06-2014 | Concept van vernieuwde versie | Christiaan Druif |
| 0.3 | 30-06-2014 | Final versie | Christiaan Druif |

Table of Contents

[Versiebeheer 2](#_Toc391642147)

[1. Inleiding 4](#_Toc391642148)

[2. De belangrijke onderdelen 5](#_Toc391642149)

[3. Activity Diagram 6](#_Toc391642150)

[4. Sequence Diagram 7](#_Toc391642151)

[5. Class Diagram 8](#_Toc391642152)

# Inleiding

Dit product dient als productbeschrijving van het project SecMon. In dit document worden de hoofdzakelijke modules besproken van de applicatie. Tevens wordt vermeld welke modules worden aangeroepen. Dit document bevat ook een klasse diagram, stroomdiagram en een activiteit diagram.

De applicatie SecMon is een product wat is voortgekomen uit het project Security Monitor, onderdeel van de minor Forensic Intelligence and Security aan de Hogeschool van Amsterdam.

Het project is voor het eerst in 2013 gemaakt door Dave van Ast, Stanley Numan, Jasper Groot, Sujen Kandasamy en Vinesh Koenjbiharie. In 2014 is het project door Christiaan Druif aangepast en verbeterd.

Het doel van het project was het opleveren van een applicatie die de VMWare omgeving van de Hogeschool van Amsterdam kan beschermen tegen cyber-aanvallen van zowel binnen uit als buitenaf. De applicatie moet in staat zijn verschillende aanvallen te herkennen en daarop volgend een actie uit te voeren. Dit moet gerealiseerd worden door middel van het parsen van de firewall log(s). Daarna is de applicatie uitgebreid waardoor het nu ook mogelijk is om andere logfiles, zoals Apache logfiles te parsen.

# De belangrijke onderdelen

Onderstaand de belangrijke onderdelen van de applicatie beschreven. Het doel hiervan is om de belangrijke elementen extra te benoemen om de structuur van de applicatie te verduidelijken.

**Main**

In de main wordt de Monitor() functie in de klasse Monitor aangeroepen en gestart. Dit start het monitoren van de applicatie.

**Rules – Rules()**

Hier worden de rules opgehaald uit de rule directory en in een dictionary gezet. Dit gebeurd ook met de RuleDefinitionsTable.txt. De monitor gebruikt deze rules om threads te starten voor elke rule

**Monitor – Monitor()**

Dit is het hart van de applicatie, vanuit hier worden alle modules aangeroepen en doorgegeven. Allereest worden de rule bestanden uitgelezen en voor elke rule wordt er een aparte thread aangemaakt. Daarna stelt de monitor de verschillende klassen in verbinding met elkaar.

**MatchManager – Matching()**

Hier worden de matches uit de rule dictionary gelezen en verwerkt. Tevens worden de values in de matchlist gekoppeld aan de inhoud van RuleDefinitionTable.txt.

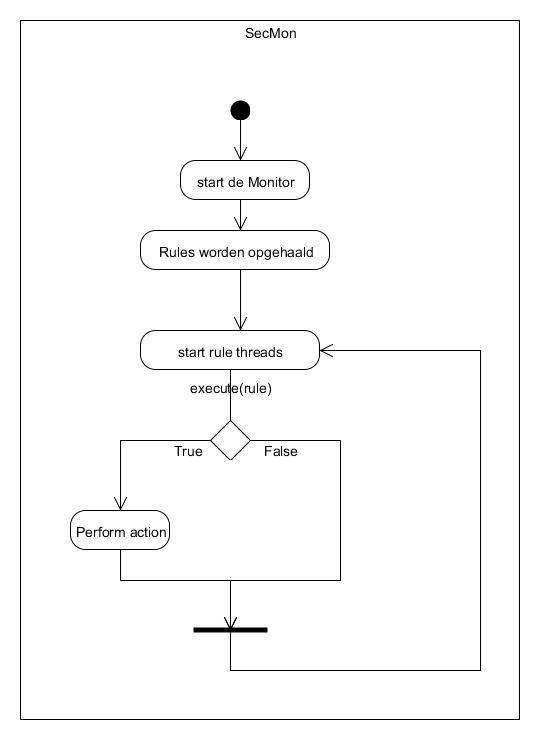
**SearchManager – SearchManager()**

Hier wordt matchlist gebruikt om een regex te bouwen. Met deze regex wordt de logfile uitgelezen en match counts bijgehouden. Is er sprake van een positieve match op de inhoud van de rule, dan zal er een True worden gestuurd naar de monitor, die het op zijn beurt weer doorgeeft aan de trigger.

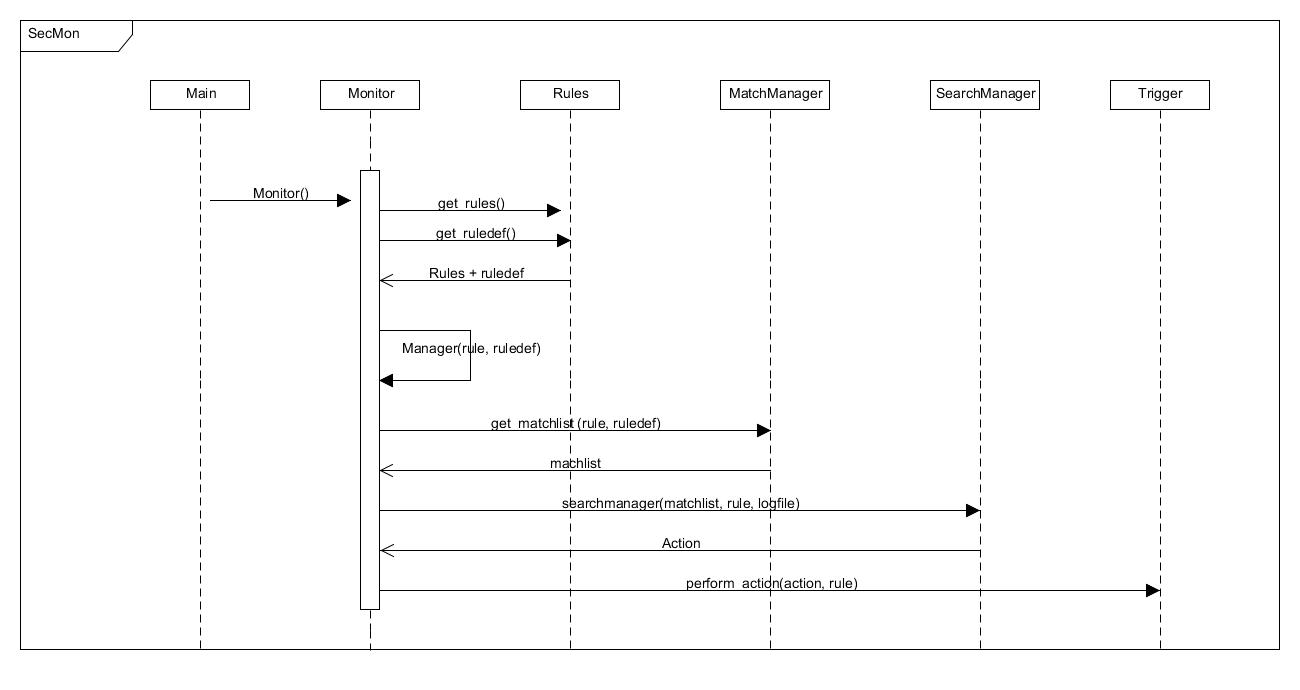
**Trigger – Perform action()**

Hier wordt een actie uitgevoerd indien er een True wordt doorgegeven. De ACTION keyword wordt uitgelezen uit de rule dictionary en een actie wordt gestart. Tevens is het mogelijk om meerdere acties uit te voeren.

# Activity Diagram



# Sequence Diagram



# G:\Dropbox\HVA\Afstudeerfase\MINFIS\SecMon\Documentatie\Handleiding&Productbeschrijving\new\diagrammen\class_diagram.jpgClass Diagram