Productbeschrijving

In dit document worden de hoofdzakelijke modules besproken met wat daarin gedaan wordt, en wat er wordt aangeroepen, plus de motivatie van het maken van de klasse/functie. Dit document bevat ook een klassedigram, stroomdiagram en activiteitendiagram. Deze zijn aan het einde te vinden van dit document.

**Main**

In de main wordt een instantie gemaakt van Configuration. De init functie van de klasse wordt aangeroepen, hier wordt de definities gedaan voor de variables van Configuration. Dit wordt ook gedaan wanneer er een object van monitor wordt aangemaakt. Vanuit main wordt wordt een daemon gestart. Deze daemon roept de functie startMonitoring() aan van de klasse Monitor. Hier wordt dus de main thread gestart.

Wij hebben gekozen om de main klasse te maken, zodat we een duidelijke initialisatie klasse hebben voor de applicatie.

**Monitor – startMonitor()**

De functie startMonitoring() leest alle rule files in van de gegeven directory. Hiervoor wordt gebruik gemaakt van de functie readRules van de klasse Rule. Daarna wordt voor elk rule een nieuwe monitoring thread gestart.

Wij hebben gekozen om de monitor klasse zo in elkaar te zetten, zodat er tegelijkertijd per rule gemonitored kan worden zonder afhankelijkheid.

**Monitor –monitor() / QueryManager – execute()**

De monitoring thread voert voor elke rule een check uit. De check geeft aan of er sprake is van een attack zoals beschreven staat in de rule. De check wordt gedaan door middel van functie execute(rule) van QueryManager. Hier worden alle reguliere expressies opgehaald dat bij de keywords horen en de filtering wordt gedaan met executeRegex(). De resultaat hiervan wordt in de mainResult opgeslagen als global variable, zodat het overal gebruikt kan worden. Als de mainResult niet leeg is wordt er een True afgevuurt, want dat wilt zeggen dat er een resultaat is gevonden voor de rule. Er is dan dus sprake van een attack. Als er False wordt terug gegeven gaat de monitor verder met checken anders wordt er een execuetTrigger van Trigger afgevuurd.

Wij hebben gekozen om de QueryManager klasse zo in elkaar te zetten, zodat we met zo min mogelijk resources een effectieve filterering uit kunnen voeren.

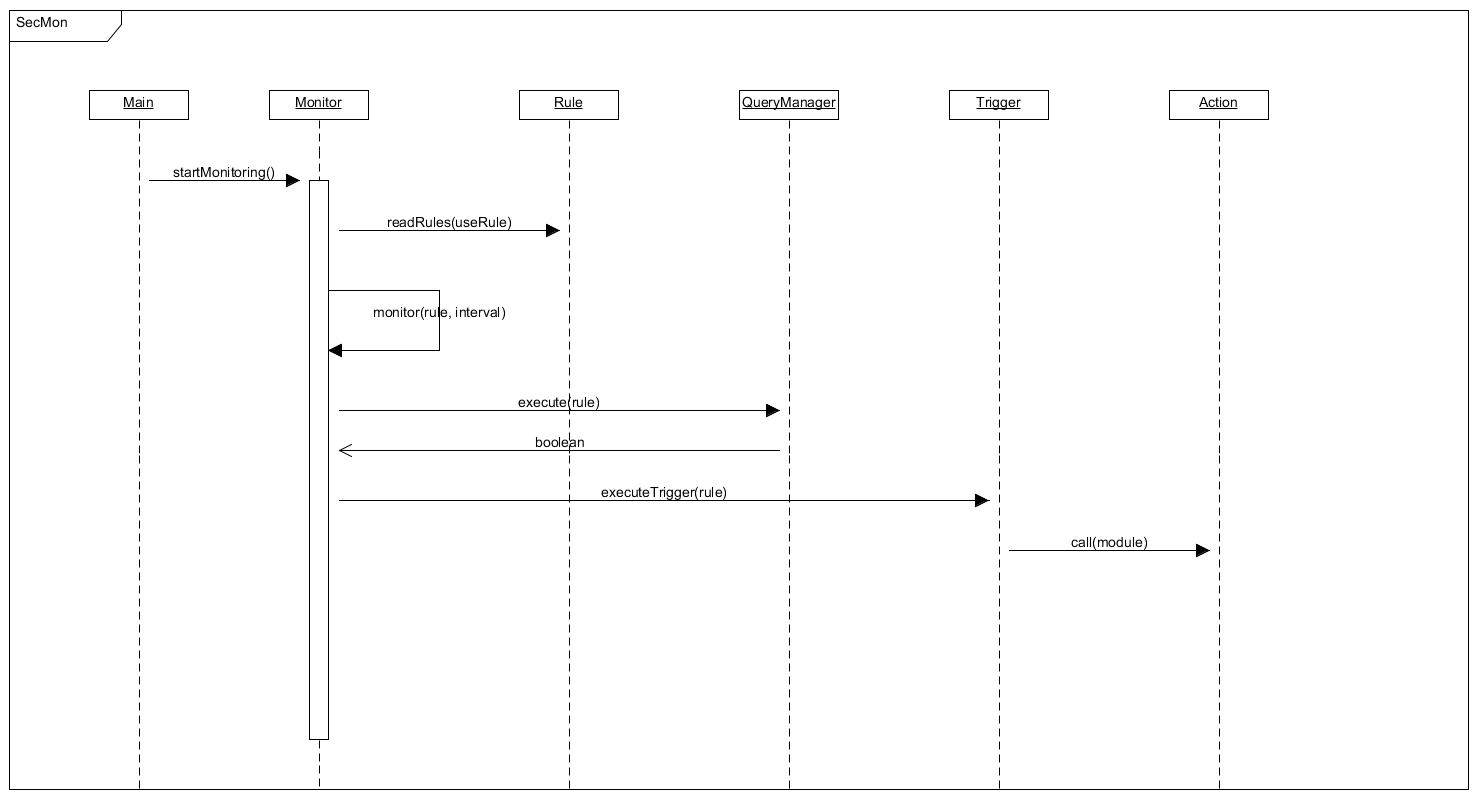
**Trigger – executeTrigger()**

De functie executeTrigger() kijkt welke actions er allemaal worden triggered bij de bijbehorende rule en voert deze vervolgens uit met de gegeven parameters.

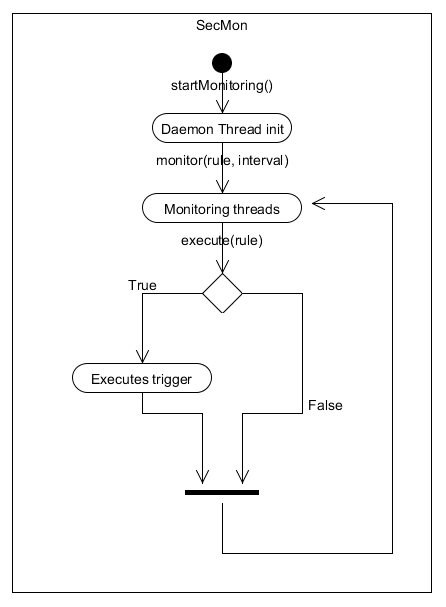
Wij hebben gekozen om de Trigger klasse zo in elkaar te zetten, zodat we modulair een of meerdere acties kunnen uitvoeren met modules van derde partij.

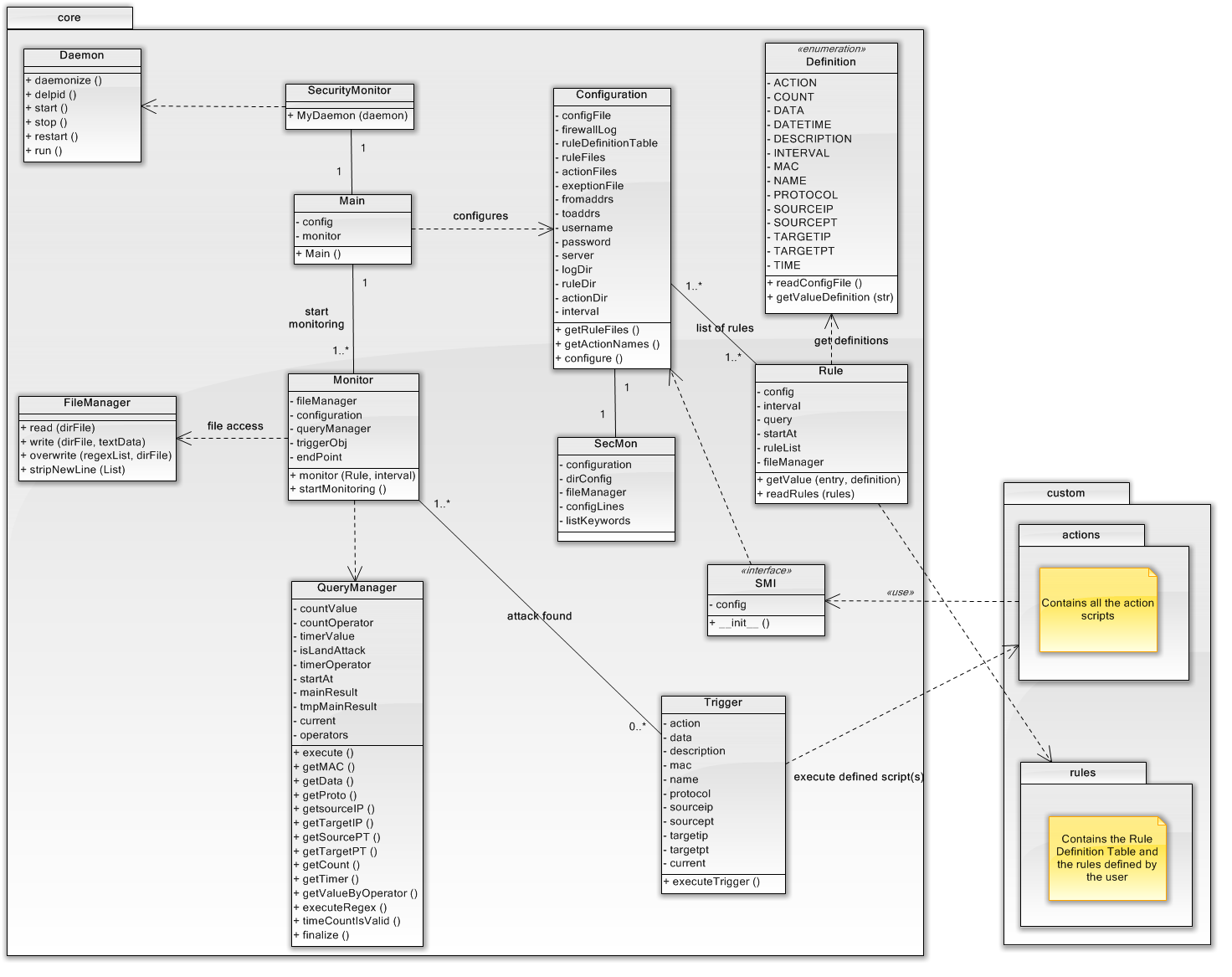
**SMI**

SMI wordt gebruikt als een interface voor alle externe modules om op en modulaire manier de waarde van de bijbehorende values door te parsen.

Sequence diagram Security monitor:

Activity diagram Security monitor:



Class diagram Security Monitor