Zelf-lerende netwerkbeveiliging, de opkomst van Skynet?

Axel Faes

June, 2016

Is het mogelijk om een beveiligingssysteem te maken dat zelf kan leren wanneer een systeem aangevallen wordt? Een systeem dat zelf kan leren, klinkt als iets dat vanuit een film zoals Terminator komt. Hierin kan een programma, Skynet, aanvallen detecteren en uiteindelijk begint het een aanval tegen de mensheid. Skynet is uiteraard fictie, maar kan een systeem zoals Skynet gemaakt worden om met dodelijke precisie aanvallen te detecteren? Uiteraard wel zonder het gehele "einde van de mensheid" erbij.

Afgelegde weg

De weg naar het bouwen van een intelligent systeem brengt ons voorbij verschillende onderwerpen. Er moet geweten zijn welke soorten aanvallen er bestaan. Dit gaat van simpele virussen tot zogenaamde botnets die gehele netwerken kunnen overnemen. Technieken om een systeem intelligent te maken, wordt machine learning genoemd. Deze technieken moeten uitgetest en vergeleken worden. Er moet dus ook een manier gevonden worden om het zelf-lerende systeem goed te kunnen evalueren. Dit gebeurt op verschillende manieren. Uiteindelijk moet ook gezien worden hoe het systeem zich gedraagt in de echte wereld. Gaat het systeem aanvallen goed kunnen detecteren wanneer het uitgetest wordt in de echte wereld?

Data

We kunnen wel een intelligent systeem hebben, maar uiteraard moet er eerst data zijn. Zonder deze data kan het intelligente systeem namelijk niet leren. Binnen een netwerk is er enorm veel data. Om deze data voor te stellen kunnen we een netwerk vergelijken met een postbedrijf. Hierin worden postbrieven verwerkt. Er zijn zo veel brieven dat elke brief nakijken op gevaarlijke inhoud heel moeilijk wordt. Om dit op te lossen, kan er enkel gekeken worden naar de envelop. Een envelop bevat niet alle informatie die in de brief staat, maar het bevat wel informatie zoals de afzender. Binnen een netwerk wordt zo een groep enveloppen die tussen dezelfde mensen gestuurd worden *IP Flows* genoemd. Vanuit deze enveloppen kan ook extra informatie verzameld worden. Er kan

bijvoorbeeld afgeleid worden uit welk land de envelop komt. *IP Flows* bevatten dus niet álle informatie die er is binnen een netwerk. Dat is een grote limitatie. Een vraag die hier optreedt is ofdat het systeem aanvallen kan detecteren zonder dat het álle informatie binnen het netwerk kent.

Intelligente systemen
Er bestaan verschillende soorten 'magische' technieken om een systeem zelf-lerend te maken. Natuurlijk is er niets écht magisch aan deze technieken. Deze technieken spelen enkel op een bijna magische manier met wiskunde om een zelf-lerend systeem te maken. Al deze technieken hebben een gemeenschappelijk aspect. Om een systeem intelligent te maken, moet het systeem getrained worden. Deze training gebeurt door het systeem data te laten zien. Hieruit kan het systeem patronen leren herkennen en kan het deze patronen leren. Bij dit trainen kunnen enkele problemen optreden. Een probleem dat kan optreden is overfitting. Bij dit probleem gaat het systeem patronen herkennen die enkel voorkomen binnen de data die gebruikt wordt om te trainen. Stel je wilt een intelligent systeem hebben om stoelen te herkennen. Om het systeem te trainen laat je het bureaustoelen zien. Bij overfitting gaat het systeem niet leren hoe een stoel eruit ziet, maar wel hoe een bureaustoel eruit ziet. Als je niet genoeg varierende data geeft gaat het systeem overfitten.

Er zijn veel verschillende soorten technieken. In totaal zijn er 12 verschillende technieken getest. Zoals eerder uitgelegd kunnen we uit *IP Flows* meer informatie halen. De technieken zijn eerst getest zonder deze extra informatie. Hierna zijn de technieken opnieuw getest en is de extra informatie wel gebruikt. Door deze twee testen kan bekeken worden hoe nuttig deze extra informatie is.

Classificatie
Nu er data is en verschillende technieken gekend zijn, kunnen we een zelf-lerend systeem bouwen. Het is echter belangrijk om een goede classificatie te maken van de data. De *IP Flows* in de data moeten eerst geclassificeerd worden. Classificeren betekent dat de data in verschillende klasses moet worden ingedeeld. Er kan gekozen worden om *IP Flows* enkel binair in te delen. Dit betekent dat er enkel een classificatie "aanval" en "geen aanval" is. Een andere optie is om een exacte classificatie te maken. Bij deze classificatie wordt er niet enkel gezegd ofdat een IP Flow een aanval is, maar ook wat voor soort aanval. Aangezien deze tweede optie veel meer details geeft over de soort aanval is hiervoor gekozen.

Evaluatie van het systeem
De verschillende algoritmes moeten ook geëvalueerd worden. Een eerste gedachte is om gewoon te kijken naar hoeveel procent van de aanvallen gedetecteerd kan worden. Dit wordt de accuraatheid genoemd, maar blijkt echter een slechte maatstaaf. Dit percentage houdt geen rekening met het feit dat aanvallen amper voorkomen in een netwerk. Allereerst moet er gekeken worden naar wat er

gebeurt met het algoritme als er meer en meer data gebruikt wordt om het te trainen. Dit wordt een *learning curve* genoemd. Deze curve laat zien ofdat er problemen optreden bij de training. Overfitting is heel duidelijk te zien in een *learning curve*. Een volgende meetstaaf is de *F-score*. Een *F-score* is gelijkaardig aan de accuraatheid, echter houdt deze score rekening met de imbalans tussen aanvallen en normaal gedrag. De *F-score* is een gemiddelde van de *precision* en *recall*. *Precision* geeft aan hoeveel voorspelde aanvallen, feitelijk aanvallen waren. *Recall* geeft aan hoeveel procent van de aanvallen gedetecteerd zijn.

Loslaten in de echte wereld
Nu kan het systeem geëvalueerd worden. Maar kan het systeem ook werken in een echte omgeving? Om dit te testen is informatie verzameld van een groot bedrijf, *Cegeka*. Deze informatie is vervolgens gebruikt om het systeem eens te testen in een realistische omgeving. Het blijkt dat sommige technieken inderdaad goed werken voor netwerkbeveiliging. Andere technieken werken echter helemaal niet goed.

Conclusie
Deze thesis zal concreet bekijken welke aanvallen wel of niet gedetecteerd kunnen worden. Het bekijkt ook wat *IP Flows* nu exact zijn. Een grote vraag hierbij is of de extra informatie van de *IP Flows* nuttig is of niet?

Een zelf-lerend systeem voor netwerkbeveiliging zoals hier beschreven is, zal nog niet onmiddelijk de wereld overnemen. Maar hoe effectief is het systeem om aanvallen te detecteren? Wat is de 'magie' van de machine learning technieken en welke technieken zijn nu precies gebruikt? Één van de belangrijkste vragen is hoe effectief werkt een zelf-lerend systeem in de echte wereld? Zelflerende technieken worden meer en meer gebruikt, dus er is een goede hoop dat een zelf-lerend netwerkbeveiligingsysteem ook goed zal werken.