Proof of Concept: Logisland

Realisatie

Bachelor in de toegepaste informatica

Academiejaar 2016-2017

Campus Geel, Kleinhoefstraat 4, BE-2440 Geel

Hanot Lukas

LID VAN ASSOCIATIE KU LEUVEN

THOMAS MORE

INHOUDSTAFEL

INLEIDING

Wat is Logisland was de vraag die mij overviel bij de start van mijn stage. Een nieuwe tool, een nieuwe manier van werken, en dit in een onbekende omgeving waren enorm overweldigend aan het begin van de stage. 3 maanden later kan ik met een opgeheven hoofd mijn verworven kennis delen door dit realisatie document.

Om de lezer een beter inzicht te geven in de basis van mijn kennis is het eerste hoofdstuk **De basis** gericht op mijn voorgaande ervaring met Linux en Open Source software en hoe dit opweegt tegenover de verreisten van Inuits. Het hoofdstuk erna zal zich verdiepen in de werkwijze van Inuits: de technologieën, de software en de samenwerking. Dit wordt natuurlijk opgevolgt door de Proof of Concept en wordt afgesloten met mijn bevindingen en conclusie over de al dan niet bruikbaarheid van Logisland binnen Inuits.

Logisland is een event minent schaalbaar platform dat gericht is op een grote doorvloed van events. Deze zin zal sommigen een idee geven van wat Logisland is en andere eens aan hun hoofd doen krabben. Daarom staat in het hoofdstuk **Logisland** een duidelijkere/ uitgebreidere uitleg waarmee u, de lezer, hopelijk een inzicht krijgt in deze tool.

Logisland opzetten was de grote uitdaging voor mij maar het eigenlijke doel was niet zomaar een tool opzetten. Mijn opdracht was namelijk uittesten of deze tool een aanwist zou zijn voor Inuits dit kan echter niet alleen door de tool op te zetten, hiervoor moet een analyse gedaan worden. Dit document eindigt dan ook met een analyse van de tool en de redenen waarom het wel of niet een aanwinst is voor Inuits. De conclusie is een samenvatting van zowel mijn ervaring als de mening van Kris mijn stagementor.

Ik wil hier graag een aantal mensen bedanken die mij de kans hebben gegeven tot deze fantastische ervaring. Mijn ouders die mij alle jaren van mijn studie hebben gesteunt door al mijn goede en leerrijke ervaringen. De scholen waar ik mijn opleiding heb gevolgt zowel Karel de Grote te Antwerpen als Thomas More Geel dat mij met open armen heeft ontvangen om mij een aangenaam laatste jaar te geven. Mijn stage bedrijf Inuits dat mij een hoop ervaring heeft gegeven en mijn interesse in Open Source zowel software als community extra hard heeft aangewakkerd. Specifiek mijn stagementor Kris Buytaert die mij veel heeft bijgeleerd en die ik hoop nog vaak tegen te komen tijdens Linux evenementen. Bart Portier mijn stagebegeleider en docent aan Thomas More die mij streng maar rechtvaardig het juiste pad heeft geleid door deze stage en de documentatie hiervan. En als laatste maar zeker niet minste mijn vriendin Melissa die mij door deze soms stressvolle periode heeft blijven verdragen en steunen.

1 De basis

Een stage doen in een bedrijf dat zich focust op Open Source kan niet anders dan met de nodige voorkennis van het Linux besturingssysteem. Ik ben voor het eerst Linux beginnen gebruiken eerder uit noodzaak. Tijdens mijn stage in het middelbaar heb ik een pc gemaakt uit reserve onderdelen die nog in het bedrijf lagen en hier miste nog een besturingsssyteem op. Aangezien windows op een systeem zetten dat ik misschien 1 maand ging gebruiken een verspilling zou zijn koos ik om hierop Ubuntu te zetten. Ondertussen is Linux hierdoor al ongeveer 6 jaar mijn daily driver. Linux is mijn go to oplossing voor al mijn besturingssystemen van raspberry pi tot de server die ik thuis draai. Maar ik gebruik Linux niet enkel thuis ik ga al jaren steevast elk jaar naar Fosdem en daar is dit jaar ConfMgmtCamp en Loadays bijgekomen. Daarbovenop ben ik ook deel van een hackerspace waar zo goed als iedereen Linux gebruikt op hun machines en in projecten.

Zoals hierboven vermeld werk ik zelf al 6 jaar met Linux als mijn standaard daily driver dit was natuurlijk een enorme invloed voor de keuze van mijn stagebedrijf. Ik heb reeds meerdere jaren niet de keuze gehad om enkel met Linux te werken dus wou ik niet liever dan een stage doen waar Linux de standaard was. Door deze eerdere ervaring had ik natuurlijk al een grote stap in de goed richting zowel een basis kennis van de tools binnen het bedrijf als de wereld er rond waren een noodzaak tijdens deze stage.

Echter niet alle tools die Inuits gebruikt waren mij bekent dit op zich was al een heel leerrijke ervaring, van sommige tools had ik nog nooit gehoord of een idee van het bestaan. Het werken met pipelines die automatie mogelijk maken zoals Jenkins en Puppet was een volledig nieuwe ervaring voor mij die mij pas echt een inzicht gaf op de werkwijze binnen bedrijven. De verschillende communicatie platformen zoals Rocket.chat en Zimbra die volwaardige alternatieven zijn op gesloten commerciele tools zoals exchange en skype. En de ethos om met en aan open source te werken en deze te verbeteren door actief deel te nemen aan deze cultuur waren een ware oog opener voor mij en een doel om naar te streven.

De grootste uitdaging was natuurlijk om tijdens deze stage de werkwijze te leren van Inuits. Hoe doe ik juist aanpassingen aan de infrastructuur, hoe zorg ik dat dit ge-automatiseerd wordt, hoe monitor ik deze wijzigingen en waarom vragen mensen mij om problemen te ack'en(verwijzing 1)?

2 INUITS

Inuits is opgericht in 2007 dit betekent ondertussen toch al 12 jaar van IT evolutie waardoor ondertussen een groot netwerk van virtuele server, verschillende environments en tools is ontstaan. Dit maakt het voor nieuwe werknemers en stagairs moeilijk om snel een grip te krijgen op de omvang en complexiteit van het netwerk. Om dit nog moeilijker te maken bezit Inuits zelf geen fysieke infrastructuur deze bevind zich allemaal op cloud server bij andere providers zoals ovh, * Vraag meer voorbeelden *. Hieronder een voorbeeld van een klein deeltje van de infrastructuur.

*** Insert infrastructure.jpg ***

Dit netwerk onderhouden door op elke afzonderlijke machine settings te gaan aan en uit zetten is dan ook geen optie. De logische oplossing is dan ook automatisatie hiervoor gebruikt Inuits een samenwerking tussen redmine, jenkins en puppet.

- Redmine: een web-gebaseerde project manager en issue tracker applicatie. Te vergelijken met bijvoorbeeld Jira, Sourceforge en Microsoft Project.

- Jenkins: Een automatisatie server die de non-human delen van seftware development op zich neemt en continuous delivery mogelijk maakt. Vergelijkbare voorbeelden zijn Jetbrains Teamcity, Azure DevOps en Travis CI

- Puppet: een software configuration management tool gebaseerd op een client-server model. De clients halen hun configuratie op bij de server waarna de puppet client de machine configureerd zoals aangegeven op de server. Vergelijkbare voorbeelden zijn Ansible, Chef en System Center Configuration Manager

^ * Zet dit in een tabel * ^

Wanneer een administrator een aanpassing wil doen aan de configuratie van een node zoals bijvoorbeeld kibana.playground.inuits.eu cloned de gebruiker de puppet configuratie vanaf Redmine. Voert de nodige aanpassingen uit in zijn locale repository en commit deze dan terug naar Redmine. Jenkins merkt automatisch op dat er een wijziging is gebeurt aan het bestand en voegt deze toe aan de queue. Als de wijziging alle Jenkins tests doorstaat word de wijzing toegepast en de actieve puppet configuratie aangepast. Om de 30 minuten vind er een puppet run plaats op de client waardoor de puppet client bij de server gaat controleren of er wijzigingen moeten worden doorgevoerd en dus de laatste configuratie wordt toegepast.

*** Insert automation.jpg ***

3 POC – LOGISLAND

3.1 Inleiding

Logisland is een applicatie ontwikkeld om events te verzamelen en verwerken. Onder events verstaan we bijvoorbeeld logs, kliks op een webpagina, alerts van andere diensten, … met andere woorden alle gebeurtenissen waar we een tijdwaarde aan kunnen bevestigen. Inuits wil weten of deze tool een meerwaarde kan leveren aan de logs die ze momenteel al verzamelen en dan voornamelijk of ze op deze events kunnen zoeken naar sequentiële patronen, frequente patronen en de correlaties tussen tijd series en gebeurtenissen.

3.2 Fasering

Mijn Fasering kan u terugvinden in mijn Plan van Aanpak maar zit ook toegevoegt bij de bronnen * **Voegt bron Fasering.tabel toe *.** Tijdens mijn stage heb ik mij heel goed kunnen houden aan mijn vooropgestelde planning en ik ben mezelf heel dankbaar voor de extra tijd die ik had ingeplant want deze was nodig. Elke stap vooruit in dit project bracht een nieuwe klif mee die eerst bestudeert moest worden voor deze aangepakt kon worden. Desondanks ben ik erin geslaagd om mijn doelen te bereiken binnen de planning.

3.3 Hoe werkt Logisland

Logisland is eigenlijk een uitbreiding op de ELK * verwijzing Elasticsearch Logstash Kibana * stack wat dus zorgt voor een grote complexiteit. Ik zal proberen om de uitleg zo simpel mogelijk te houden maar let voornamelijk op de illustratie hieronder om het overzicht te bewaren. Laten we van links naar rechts gaan en zo de datastroom volgen. Aan de linkerkant beginnen we met de Logstash server deze ontvangt logs en andere records en filtert deze. De gefilterde records worden dan doorgestuurd naar het Kafka topic waar deze worden opgevangen en het “Logisland systeem” betreden. De records wachten in de kafka server tot deze door spark verwerkt kunnen worden naar events. Wanneer één spark verwerking klaar is kan deze of terug naar kafka gestuurd worden of naar elasticsearch/redis weggeschreven worden. Hierdoor kan data verwerkt worden en dan op de verwerkte data verdere verwerkingen verricht worden. Als alle spark operaties verricht zijn wordt de data weggeschreven naar Elasticsearch en/of redis. In Elasticsearch wordt data bijgehouden in een actieve staat deze is dus snel doorzoekbaar maar minder stabiel terwijl redis dient voor stabiele opslag. De laatste stap is Kibana deze leest de events die zich in Elasticsearch bevinden en daardoor kan de gebruiker de events bekijken en in grafieken gieten om deze te bestuderen.

*** Insert Logisland_structure.jpg ***

3.4 Lokale testomgeving

Elk project moet ergens beginnen en dat van mij begon nadat ik de werkwijze van Inuits een beetje geleerd had. Hiermee kende ik de infrastructuur nog niet maar dat was voor stap 1 nog geen probleem want dit was namelijk de tool opzetten op mijn eigen laptop. Logisland is terug te vinden op de Hurence Github * verwijzing github pagina * pagina en bezit online documentatie * verwijzing online documentatie *.

Het was al snel duidelijk dat de docker setup veel makkelijker zou verlopen dan zelf een hele ELK * footnote Elasticsearch Logstash Kibana * stack opzetten. Docker opzetten in een Centos omgeving is simpel en zeker omdat deze gebruik maakt van docker-compose.

*** Insert screenshot_docker-compose.jpg ***

```

**yum install docker**

**systemctl start docker**

**systemctl enable docker**

**sudo curl -L "https://github.com/docker/compose/releases/download/1.24.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose**

**sudo chmod +x /usr/local/bin/docker-compose**

**docker-compose -version**

```

De docker setup opstarten leverde helaas ook de eerste problemen op maar na een korte google sessie vond ik snel een oplossing. Elasticsearch gebruikt een hybrid mmapfs direcotry om zijn indices op te slaan dus deze wilt niet opstarten als de map counts te laag zijn. 2 korte commando's later en de logisland containers willen zonder problemen opstarten.

```

sudo sysctl -w vm.max_map_count=262144

sudo sh -c 'echo "vm.max_map_count=262144" >> /etc/sysctl.conf

```

Nu dat de containers waren opgestart kon ik een aantal van de tutorials uitproberen. De docker images komen standaard al met de configuratie files van de tutorial dus was dit redelijk simpel om te testen.

```

sudo docker exec -i -t conf_logisland_1 bin/logisland.sh --conf conf/logs-to-events.yml

```

Dit commando zegt tegen docker "voer voor mij het commando in de bin folder logisland.sh met de configuratie logs-to-events.yml in de conf folder uit op de conf_logisland_1 container". Waardoor de geselecteerde configuratie wordt uitgevoerd en het kafka topic klaar staat om data te ontvangen.

* Vanaf hier is het nog TODO *

3.5 Rebootstrappen van Lisbon.hypprod.inuits.eu

3.6 Installatie van logisland in Logisland.playground.inuits.eu

3.7 Verwerking van events

4 CONCLUSSIE

5 BRONNEN