

Monitoringtools voor Docker en Kubernetes

Troch Olivier, Bert Van Vreckem

Hogeschool Gent, Valentin Vaerwyckweg 1, 9000 Gent

olivier.troch.w2257@student.hogent.be

HO
GENT

Abstract

Kosten besparen en winst verhogen zijn legitieme zakelijke doelstellingen. Preventie door monitoring betekent dat er minder geld wordt uitgegeven in vergelijking met de kosten van oplossingen voor problemen die zich plotseling kunnen voordoen. Helaas wordt monitoring van een infrastructuur vaak onderschat en steeds eerder als een kost gezien dan als een opbrengst. IT-afdelingen besteden vaker tijd aan het reageren op de onverwachte problemen of fouten dan aan het voorkomen ervan. Hierdoor lijkt het van goed belang dat de basisconcepten van monitoring in het lessenpakket verwerkt worden, zodat het nut van monitoring wordt meegegeven en zich gemakkelijk in de bedrijfswereld kan mengen.

Introductie

HOGENT kan helaas niet beschikken over alle mogelijke technologieën die bestaan in de informatica wereld. Toch zijn er bepaalde onderwerpen die onderdeel van het lessenpakket zouden moeten zijn. Monitoring van de infrastructuur is hier één van en meer specifiek de monitoring van een containerorkestratie. Containerisatie vindt in een recordtempo zijn weg naar de data-omgeving van de ondernemingen. Het gemak waarmee container-platformen zoals Docker kunnen worden ingezet, suggereert dat ze de dominantste architectuur zijn en zullen blijven. De uitdaging is om goede monitoring voor deze containers op te stellen. Containers zijn zeer vluchtig in ontstaan en net zo snel in het verdwijnen. Zoals te verwachten, zullen traditionele monitoringsplatforms die vooral gebaseerd zijn op virtualisatie, niet voldoende zijn. Om het monitoringproces te vergemakkelijken kan er beroep gedaan worden op monitoringtools die gecreëerd of aangepast zijn om een containerorkestratie te kunnen monitoren.

Conclusie

Door het uitvoeren van de requirements-analyse zijn 3 tools overgebleven die als geschikt verklaard waren. Op basis van de analyse waren deze 3 tools praktisch hetzelfde en om de finale beslissing te maken werd de monitoringcommunity geraadpleegd. Hieruit bleek dat Prometheus de tool was die met voorsprong uitblonk. Achteraf werden, aan de hand van een stappenplan, alle opties uitgelegd die nodig waren om een antwoord op de onderzoeksvraag en deelvragen te formuleren. Door het gebruik van de Helm-tool was de installatie van de tool relatief eenvoudig waardoor de toekomstige studenten de opstelling makkelijk zullen kunnen nabootsen om snel aan de slag te kunnen.



minikube

Minikube¹



COLLECT METRIC



Prometheus

Prometheus²



DISPLAY METRIC



Grafana

Grafana³

Onderzoek

Via de state-of-the-art werd gezocht naar een aantal tools die geschikt zijn voor het monitoren van een containerorkestratie waarna aan de hand van een requirements-analyse die gevonden tools vergeleken werden met elkaar. Nadien werd een Proof-of-Concept opgesteld waarin de werking van de, in dit onderzoek, beste tool toegepast werd. Door het onderzoek werd beslist dat een combinatie van Prometheus en Grafana geschikt was. In deze proof-of-concept werd gebruik gemaakt van standaard dashboards, maar ook van het importeren en creëren van eigen dashboards om de metriekeken van de containers te weergeven en analyseren.

Bijkomend Onderzoek

Bijkomend onderzoek kan nodig zijn om de hoofdonderzoeksvraag of deelvragen uit te breiden, om meer functies aan bod te brengen dan degene die werden meegegeven in het huidige onderzoek. Indien het de bedoeling is om dit onderzoek te gebruiken als basis voor het introduceren van andere, nieuwe technologieën kan dit betekenen dat dit onderzoek het einde nog niet bereikt heeft.

42mate⁴



Referenties

(1) <https://minikube.sigs.k8s.io/docs/>

(2) <https://prometheus.io/>

(3) <https://grafana.com/>

(4) <https://blog.42mate.com/icinga2-monitoring-mysql/>