Júlio A. P. de Melo

Clusterized Hygieia DevOps Dashboard
Relatório Técnico

Brasil
2018, v-1.0.0

Júlio A. P. de Melo

Clusterized Hygieia DevOps Dashboard
Relatório Técnico

Relatório técnico que visa mostrar os impactos gerados pela clusterização da ferramenta de monitoramento Hygieia no ambiente DevOps.

Universidade Federal de Pernambuco -- UFPE
Centro de Informática
IF1007 - Tópicos Avançados em Sistemas de Informação 4

Brasil
2018, v-1.0.0

Resumo

Os sistemas de monitoramento são uma parte essencial de qualquer sistema de alta requisição, pois são necessários para detectar falhas, relatar se os usuários são afetados e se o problema desapareceu. E se seus sistemas de monitoramento falharem? Dependendo do seu conjunto de ferramentas você não conseguirá monitorar nada, ou quase nada, além de seus monitores podem ser alimentados com dados inválidos e ficar fora de controle, e então desligarem. Como você sabe quando um sistema de monitoramento falhou? Você precisa monitorá-lo, então como você deve monitorar os sistemas de monitoramento? Como proposta para a resolução dos questionamentos a ferramenta Hygieia da organização Capitalone foi clusterizada na plataforma kubernetes. Trazendo outro paradigma de monitoramento, "monitoramento de monitores".

Palavras-chaves: Hygieia, DevOps, Kubernetes, Monitoring.

Abstract

Monitoring systems are an essential part of any high-demand system because they are required to detect failures, report whether users are affected, and whether the problem has disappeared. What if your monitoring systems fail? Depending on your toolkit you will not be able to monitor anything, or almost anything, and your monitors may be fed with invalid data and out of control, and then shut down. How do you know when a monitoring system has failed? You need to monitor it, so how should you monitor the monitoring systems? As a proposal for solving the questions, the Hygieia tool of the Capitalone organization was clustered on the kubernetes platform. Bringing another monitoring paradigm, "monitoring monitors".

Key-words: Hygieia, DevOps, Kubernetes, Monitoring.

Sumário

1 Introdução

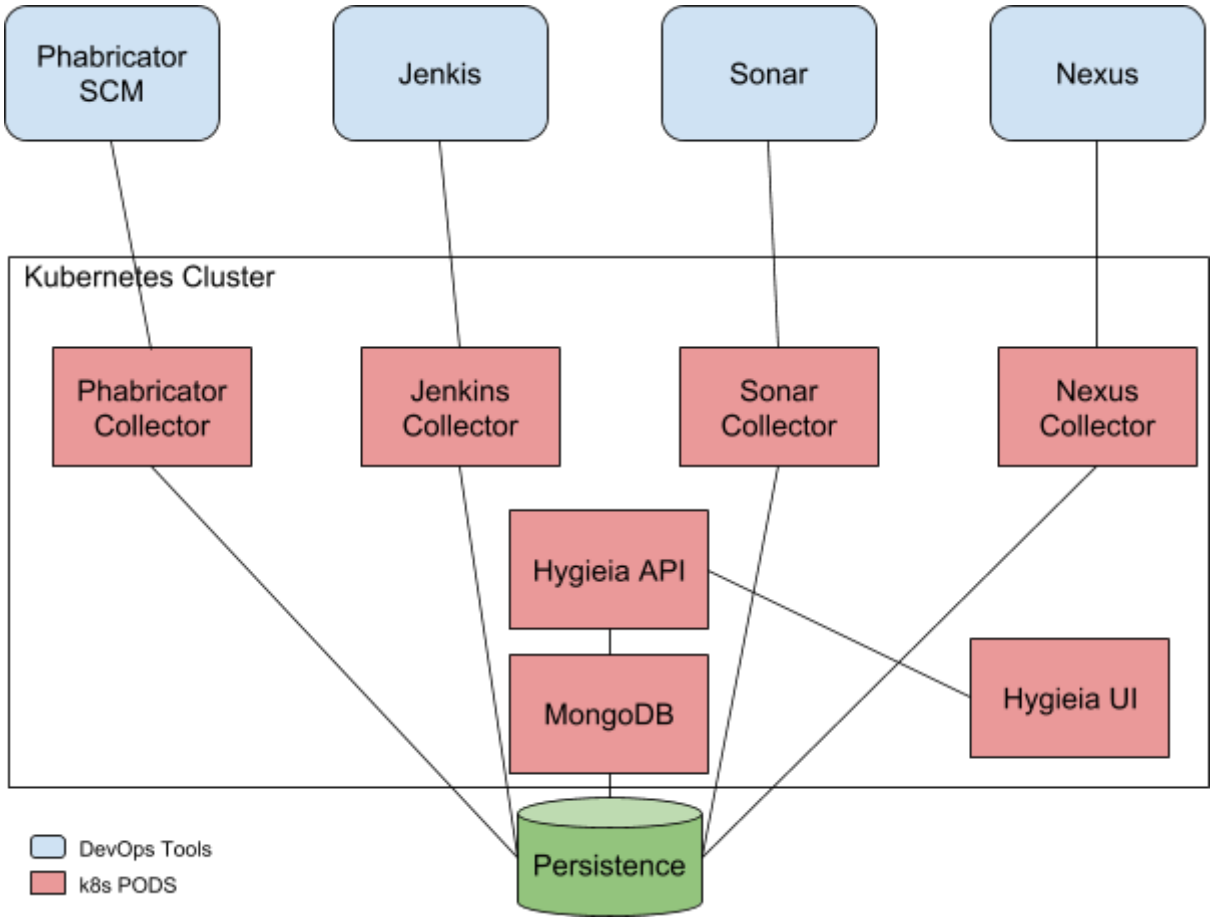
Os sistemas de monitoramento são uma parte essencial de qualquer sistema de alta requisição, pois são necessários para detectar falhas, relatar se os usuários são afetados e se o problema desapareceu.

Hygieia aparece em dois painéis independentes - um para engenheiros e outro para os demais stakeholders - que exibem visualmente os pipelines do CI/CD. Em essência, o próprio Hygieia é um agregador que extrai dados de várias ferramentas DevOps que as equipes usam em todo o ciclo-de-vida do projeto, tornando-o facilmente digerível nas exibições do painel. Ele simplificam a capacidade de visualizar pipelines quase em tempo real. Os painéis permitem que os engenheiros e executivos de DevOps monitorem a integridade do comprometimento de código com a implantação na produção final. Entre esses dois pontos - início (confirmação) até conclusão (prod) - os painéis também fornecem informações cruciais sobre a vitalidade geral e as métricas de desempenho de suas operações de software.

Mas e se sistema de monitoramento falhar? Dependendo do seu conjunto de ferramentas você não conseguirá monitorar nada, ou quase nada, além de que os seus monitores podem ser alimentados com dados inválidos e ficar fora de controle, chegando a desligarem. Como você sabe quando um sistema de monitoramento falhou? Você precisa monitorá-lo, então como você deve monitorar os sistemas de monitoramento?

Como proposta, o Hygieia foi clusterizado utilizando a plataforma Kubernetes. Por meio dele foi possível orquestrar os microserviços, escalonar rapidamente mediante a necessidade e disponibilidade de recursos, facilitando autorecuperação em caso de erro e sua investigação.

2 Hygieia Clusterizado



Layer	Description
Hygieia UI	A UI (Interface do usuário) é o front-end do Hygieia e contém todos os elementos da interface gráfica do usuário (GUI) para exibição pelos usuários. É aqui que os usuários também podem configurar o painel.
Hygieia API	A Hygieia API contêm todos os serviços típicos da API REST que funcionam com os dados do sistema de origem

	(coletados por tarefas de serviço) e a Internet. As APIs de auditoria da Hygieia são uma coleção de terminais da API que servem para auditar dados de CI / CD coletados pelos coletores da Hygieia.
k8s PODS	Collectors (Coletores e/ou plugins) e demais microserviços implantados em PODs
Persistence	Persistence Volume Claim, onde é feito mount para o armazenamento do MongoDB
DevOps Tools	Essa camada envolve a grande quantidade de ferramentas de DevOps em um pipeline de CI / CD. No diagrama, Phabricator, Jenkins, Sonar e Nexus são listados como exemplos.

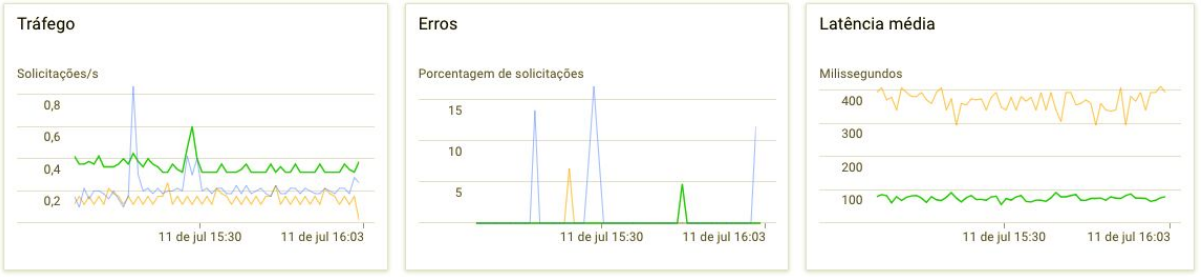
3 Projeto

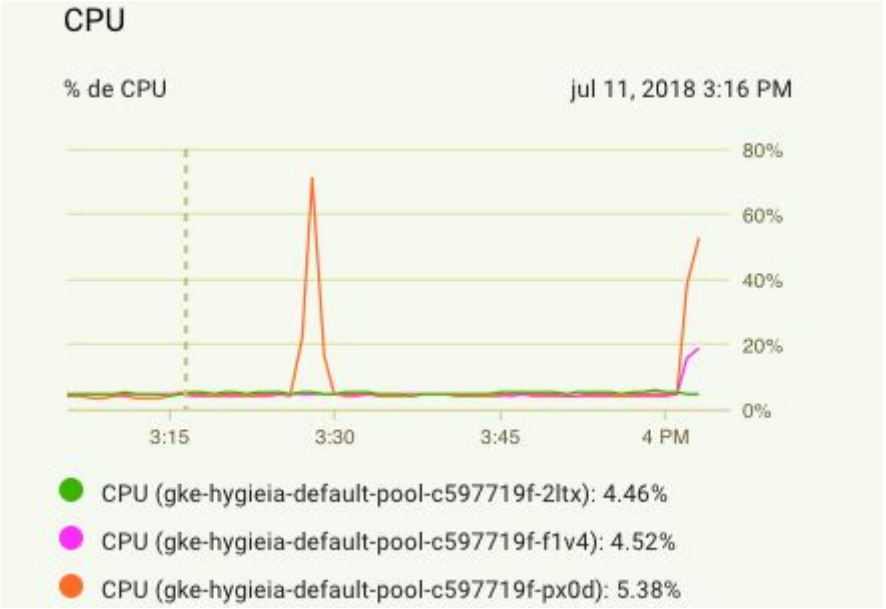
3.1 Desenvolvimento

O Hygieia foi construído em uma arquitetura microserviços sendo cada funcionalidade uma aplicação em Spring Boot, exceto pela UI que foi construída em frameworks node. Para a interface de usuário (UI) foi necessário construí-la novamente, visto que, não executava sem a dependência de outros microservices.Os demais serviços foram conteinerizados sem complicações, exceto por configurações específicas de cada serviço.

3.2 Resultados

Após a inicialização do cluster é possível verificar o status de cada serviço, consumo, desempenho - auxiliando no processo de coleta de métricas.





4 Conclusão

Dada à importância do assunto, os resultados obtidos foram satisfatórios. A clusterização no sistema de monitoramento Hygieia se mostrou fundamental na coleta de informações de seus microsserviços, tendo em vista que monitores cruciais e não podem exibir quaisquer mau funcionamento. A rapidez no seu provisionamento, manutenabilidade, escalabilidade e autorecuperação também se mostraram pontos chaves, mas acima de tudo reafirma a importância do monitoramento de monitores defendidas e explanadas por Adrian Cockrof.

5 Referências

- https://medium.com/@fidelissauro/kubernetes-criando-um-cluster-simples-com-o-kubeadm-e50a9eb4f4a6
- http://capitalone.github.io/Hygieia/getting_started.html
- http://nginx.org/en/docs/http/ngx_http_upstream_module.html
- https://kubernetes.io/docs/tutorials/stateless-application/expose-external-ip-address/
- https://kubernetes.io/docs/tasks/configure-pod-container/translate-compose-kubernetes/
- Who monitors the monitoring systems? - Adrian Cockcrof
- https://cloud.google.com/kubernetes-engine/docs/quickstart
- https://kubernetes.io/docs/tasks/access-application-cluster/connecting-frontend-backend/
- https://codelabs.developers.google.com/codelabs/gcp-infra-gke/index.html?index=..%2F..%2Fcloud#4