



IBM Software Services



Workshop Blockchain: Hyperledger Fabric 1.4

Laboratório 06

Versão: 1.0

Author: Richard S Marques – rseberin@br.ibm.com

Summary

Summary	2
1. Introdução	3
2. Proposta do laboratório	3
3. Execução.....	3
3.1. Criar Instancia Prometheus e Grafana	3
3.2. Configurar Grafana	4
3.3. Importar Dashboard.....	6
3.4. Submeter transações para testar o dashboard.....	7

Data	Versão	Autor	Descrição
10/10/2019	1.0	Richard Marques	Criação do documento

1. Introdução

Este documento tem o objetivo de guiar o participante do workshop nas tarefas propostas no laboratório de número 06 do workshop de blockchain: Hyperledger Fabric 1.4.

Este e os demais laboratórios foram testados em um servidor Ubuntu 18.04 c/ 4 Cores de CPU e 8 GB de memória RAM.

Este laboratório tem como pré-requisito que o laboratório anterior (número 05) tenha sido concluído com sucesso.

2. Proposta do laboratório

A proposta deste laboratório será:

- Criar uma instancia de Prometheus para monitorar os peers e Orderers
- Criar uma instancia de Grafana conectada ao Prometheus
- Importar um dashboard no grafana
- Submeter transações e ver como o ambiente se comporta pelo dashboard

3. Execução

Siga atentamente todos os passos para concluir o laboratório:

3.1. Criar Instancia Prometheus e Grafana

Para esta atividade navegue até a pasta “lab06” e execute o script “./criarRecursos.sh”

```
blockchain@ubuntu:~/workshop/lab06$ ./criarRecursos.sh
service/prometheus-svc created
statefulset.apps/prometheus created
service/grafana-svc created
statefulset.apps/grafana created
```

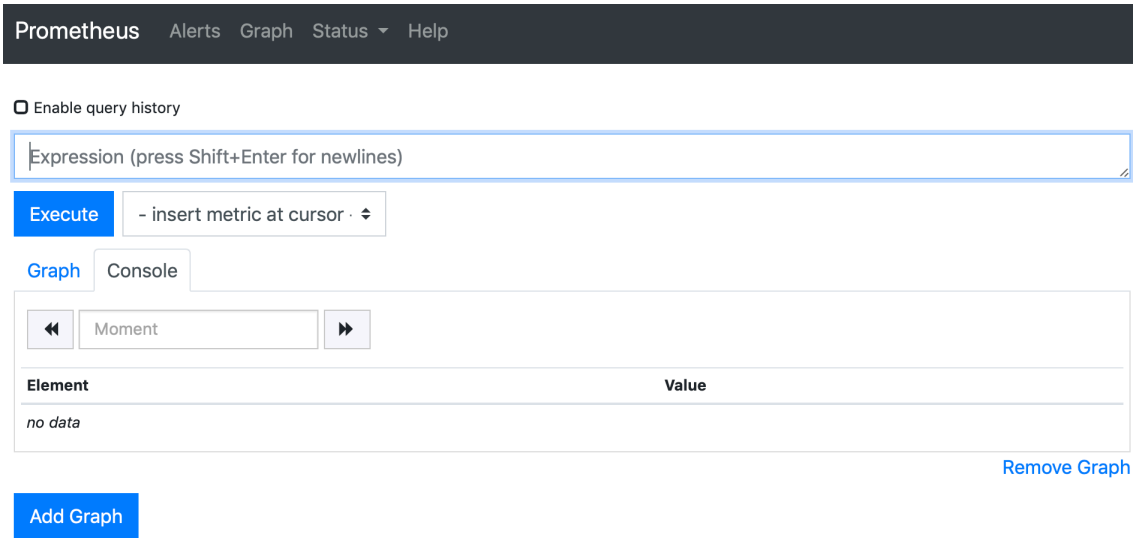
Para verificar se os componentes foram criados com sucesso, execute o comando “kubectl get pods” e verifique a existência dos Pods do prometheus-0 e grafana-0, como na imagem abaixo:

```
blockchain@ubuntu:~/workshop/lab06$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
ca-0                 1/1     Running   0           47m
cli-peer0-pod        1/1     Running   0           12m
cli-peer1-pod        1/1     Running   0           12m
cli-peer2-pod        1/1     Running   0           12m
couchdb-peer0-pod-0  1/1     Running   0           12m
couchdb-peer1-pod-0  1/1     Running   0           12m
couchdb-peer2-pod-0  1/1     Running   0           12m
grafana-0            1/1     Running   0           2m22s
kafka-pod1-0         1/1     Running   0           47m
kafka-pod2-0         1/1     Running   0           47m
kafka-pod3-0         1/1     Running   0           47m
orderer-pod1-0       1/1     Running   0           46m
orderer-pod2-0       1/1     Running   0           46m
orderer-pod3-0       1/1     Running   0           46m
peer-pod1-0          1/1     Running   0           12m
peer-pod2-0          1/1     Running   0           12m
peer-redhat-0        1/1     Running   0           12m
postgres-0           1/1     Running   0           47m
prometheus-0         1/1     Running   0           2m22s
zookeeper-pod1-0     1/1     Running   0           47m
zookeeper-pod2-0     1/1     Running   0           47m
zookeeper-pod3-0     1/1     Running   0           47m
```

Verifique se o prometheus já esta monitorando os peers e orderers da nossa solução, abra um navegador e digite o endereço abaixo:

<http://localhost:31990/graph>

Uma imagem como a seguir deve aparecer no seu navegador:



Clique no menu, no item “Status -> targets”, e verifique se os itens aparecem na lista e se estão com o status “UP”.

Prometheus Alerts Graph Status Help

Targets

All Unhealthy

orderer1 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://orderer1:9443/metrics	UP	instance="orderer1:9443" job="orderer1"	1.476s ago	3.025ms	

orderer2 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://orderer2:9443/metrics	UP	instance="orderer2:9443" job="orderer2"	1.236s ago	2.41ms	

orderer3 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://orderer3:9443/metrics	UP	instance="orderer3:9443" job="orderer3"	1.404s ago	3.486ms	

peer1 (1/1 up) show less

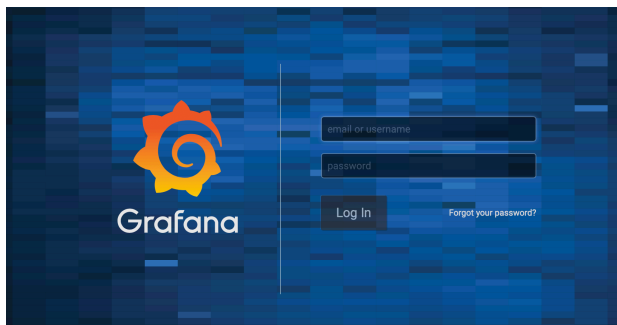
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://peer1:9443/metrics	UP	instance="peer1:9443" job="peer1"	2.048s ago	5.919ms	

3.2. Configurar Grafana

Abra o navegador no endereço:

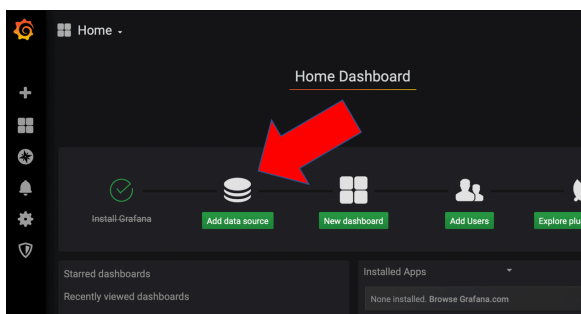
<http://localhost:31300>

Uma página como a da imagem abaixo será exibida:

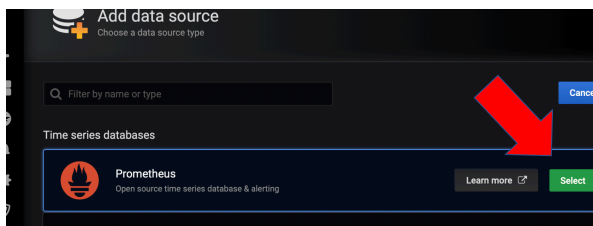


Entre com o usuário admin e senha admin, será solicitado para trocar a primeira vez, redefina a senha novamente para admin.

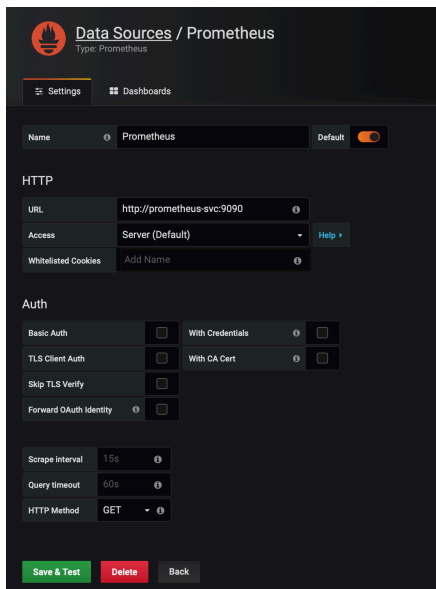
Na tela que abrir clique no item “Add data source”, para podermos registrar o Prometheus.



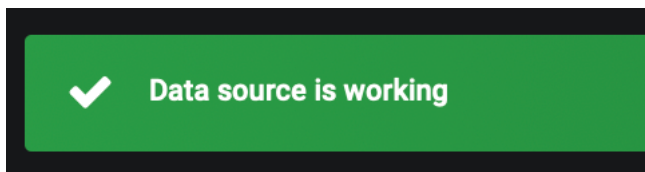
Na tela que abrir clique no botão “select” ao lado do nome prometheus na lista de datasource.



Na tela que abrir informe como endereço do prometheus a url <http://prometheus-svc:9090> e clique no botão “Save & Test”

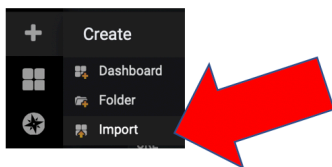


Se tudo estiver correto, você receberá a mensagem de sucesso como na imagem abaixo:

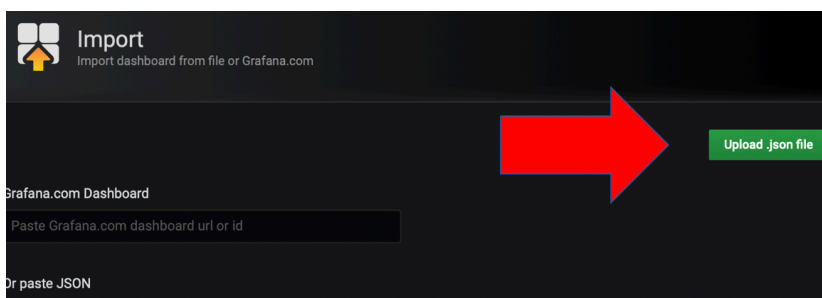


3.3. Importar Dashboard

Ainda no navegador na tela do Grafana, clique no menu lateral esquerdo no ícone “+” e escolha a opção “Import”.



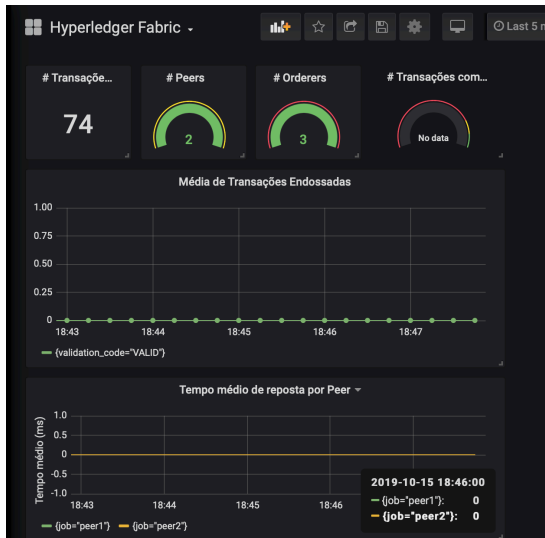
Na tela que abrir clique no botão “Upload .json file” e escolha o arquivo “dashboard.json” que esta na pasta “lab06”.



Confirme o nome do dashboard e clique no botão “Import”

Import dialog for Hyperledger Fabric dashboard. The 'Name' field is set to 'Hyperledger Fabric'. The 'Unique Identifier (uid)' field is empty, and a red arrow points to it. The 'Import' button is highlighted in green.

O dashboard será exibido na sequencia.



3.4. Submeter transações para testar o dashboard

No terminal, retorne a pasta do laboratório anterior (lab05) e execute o script “./submeteTransacoes.sh” e acompanhe a evolução do dashboard.

