**Integração JMeter com Grafana**

Leandro Gonçalves Freitas

Especialista em teste de Performance na NTT Data

Novembro/2021

Sumário

[Introdução 3](#_Toc90498057)

[Ferramentas necessárias 3](#_Toc90498058)

[Pré-condições 3](#_Toc90498059)

[Ativar o WSL no Windows 4](#_Toc90498060)

[Instalação do WSL 2 5](#_Toc90498061)

[Instalação Docker 6](#_Toc90498062)

[Instalação InfluxDB via Docker 7](#_Toc90498063)

[Instalação Grafana via Docker 8](#_Toc90498064)

[Integrando o JMeter com o InfluxDB 9](#_Toc90498065)

[Configuração do InfluxDB no Grafana 10](#_Toc90498066)

[Importando Dashboard no Grafana 13](#_Toc90498067)

# Introdução

Os testes em JMeter necessitam de uma estrutura não suportada pela própria ferramenta para monitorar o comportamento das requisições durante a janela de teste e para auxiliar em uma melhor análise de resultados, e consequentemente enriquecer o relatório técnico. Devido essas necessidades, enxergamos no mercado uma ferramenta que nos possibilitará uma melhor visão de um teste de performance. O Grafana é uma ferramenta gráfica que auxilia no monitoramento de teste de desempenho possibilitando visualizar os resultados em tempo real ou pós teste.

# Ferramentas necessárias

Durante as etapas desse documento abordaremos as ferramentas listadas abaixo:

- WSL

- Docker Engine

- InfluxDB (versão 1.8.10)

- Grafana Desktop

# Pré-condições

* Necessário atualizar o Windows 10 para a versão 21H1 ou superior;
* Ter o **Apache JMeter** na versão 5.4 ou superior instalado.

# Ativar o WSL no Windows

Existem duas formas:

1. Abra o “PowerShell” como administrador e execute os comandos abaixo:

a. DISM /Online /Enable-Feature /Featurename:VirtualMachinePlatform /all /norestart

b. DISM /Online /Enable-Feature /All /FeatureName:Microsoft-Hyper-V

c. Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux

Reinicie o computador.

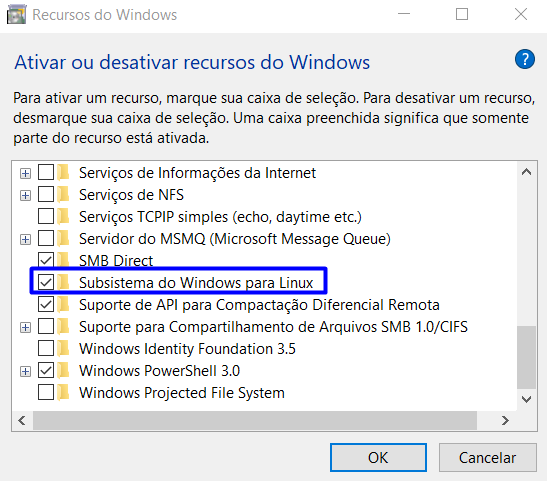
2. Acesse “Painel de Controle”, “Programas e Recursos” e clique no link “Ativar ou desativar recursos do Windows”

Marque os 3 componentes abaixo e reinicie o computador.

- “Hyper-V”

- “Plataforma de Máquina Virtual”

- “Subsistema do Windows para Linux”



# Instalação do WSL 2

1. Abra o “Prompt de Comando (cmd)”

2. Comando que define que vamos trabalhar por padrão com o WSL versão 2.

wsl --set-default-version 2

3. Vamos instalar um Linux Ubuntu (vai instalar a ultima versão estável)

wsl --install -d Ubuntu

Após executar o comando acima, irá exibir uma janela do terminal do Linux Ubuntu, conforme imagem abaixo:

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Caso exiba a mensagem abaixo, favor baixar (<https://aka.ms/wsl2kernel>) e instalar o Kernel do WSL2.

Interface gráfica do usuário

Descrição gerada automaticamente

A partir daqui os comandos serão no terminal do Linux.

4. Após a instalação, buscar a lista de atualizações para o SO.

sudo apt update

5. Fazer a atualização do SO.

sudo apt upgrade

Comando para entrar na pasta do Windows do seu usario local

Cd /mnt/c/Users/[seu\_user]

Ao final da atualização, fechar a janela do terminal Linux.

Para forçar o restart do kernel Linux, na janela de terminal do Windows (cmd) executar o comando:

wsl --shutdown

# Instalação Docker

1. No menu iniciar, procurar por "Ubuntu" e iniciar o Ubuntu que aparecerá.

Abrirá terminal do Linux.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

#### **Se já tiver instalado Docker antes executar esses comandos para limpar**

1. sudo apt-get remove docker docker-engine docker.io containerd runc

#### **Prerparar ambiente para instalar Docker**

1. sudo apt-get update

2. sudo apt-get install \

ca-certificates \

curl \

gnupg \

lsb-release

3. sudo mkdir -p /etc/apt/keyrings

4. curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

5. echo \

"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \

$(lsb\_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

#### **Instalar Docker Engine**

1. sudo apt-get update

2. sudo chmod a+r /etc/apt/keyrings/docker.gpg

3. sudo apt-get update

4. sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin

5. sudo apt-get install docker-compose

6. sudo groupadd Docker

7. sudo usermod -aG docker $USER

8. newgrp Docker

9.sudo dockerd

Abra uma nova janela ubuntu e comece a usar o Docker

10. docker run hello-world

#### **Verificar se Docker está funcionando**

sudo docker run hello-world

Dicas:

Para trabalhar com o Docker é necessário executar o seguinte comando:

sudo dockerd

Sempre usar Ctrl+C para finalizar o serviço do Docker.

**Troubleshoot**

Se tivere erro de que não está baixando as imagens , execute

1.sudo update-alternatives --set iptables /usr/sbin/iptables-legacy

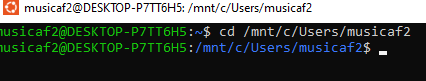
2.sudo update-alternatives --set ip6tables /usr/sbin/ip6tables-legacy

3.sudo dockerd &

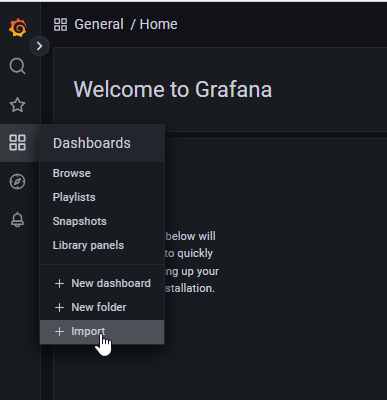
# Instalação InfluxDB e grafana via Docker Automatizado

Primeiramente copiar a pasta “App-Load-Test” para seu usuário local do Windows “C:/Users/[seu\_user]”

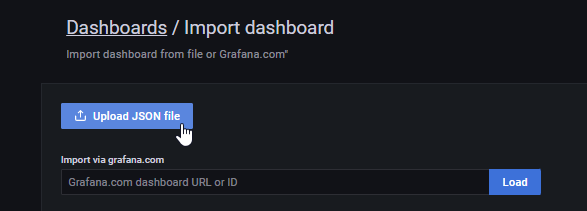
Depois abrir o terminal Ubuntu e executar o comando “cd /mnt/c/Users/[seu\_user]” para acessar dentro do Windows a pasta “App-Load-Test”.



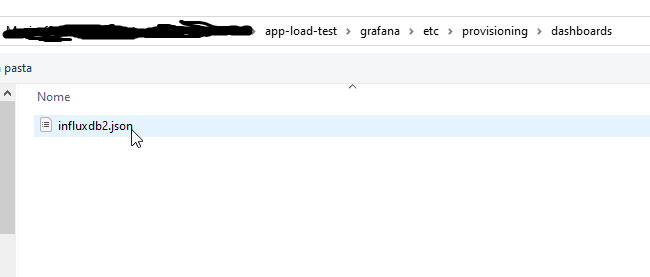
Feito isso digite “cd App-Load-Test” agora já dentro execute o comando “docker-compose up -d” aguarde enquanto é gerado e baixado os containers.

Agora abra um navegador EX. Chrome e digite na barra o endereço “localhost:3000” entre com usuário e senha = admin:admin e importe o dashboard no grafana

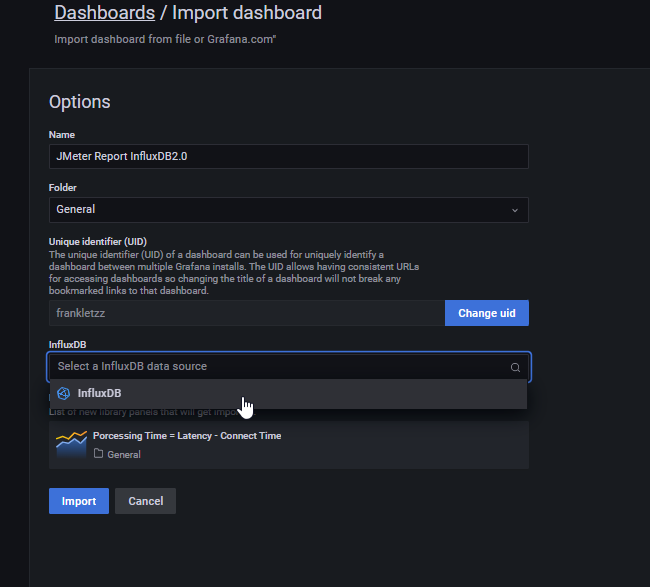
Depois clique em Upload Json file



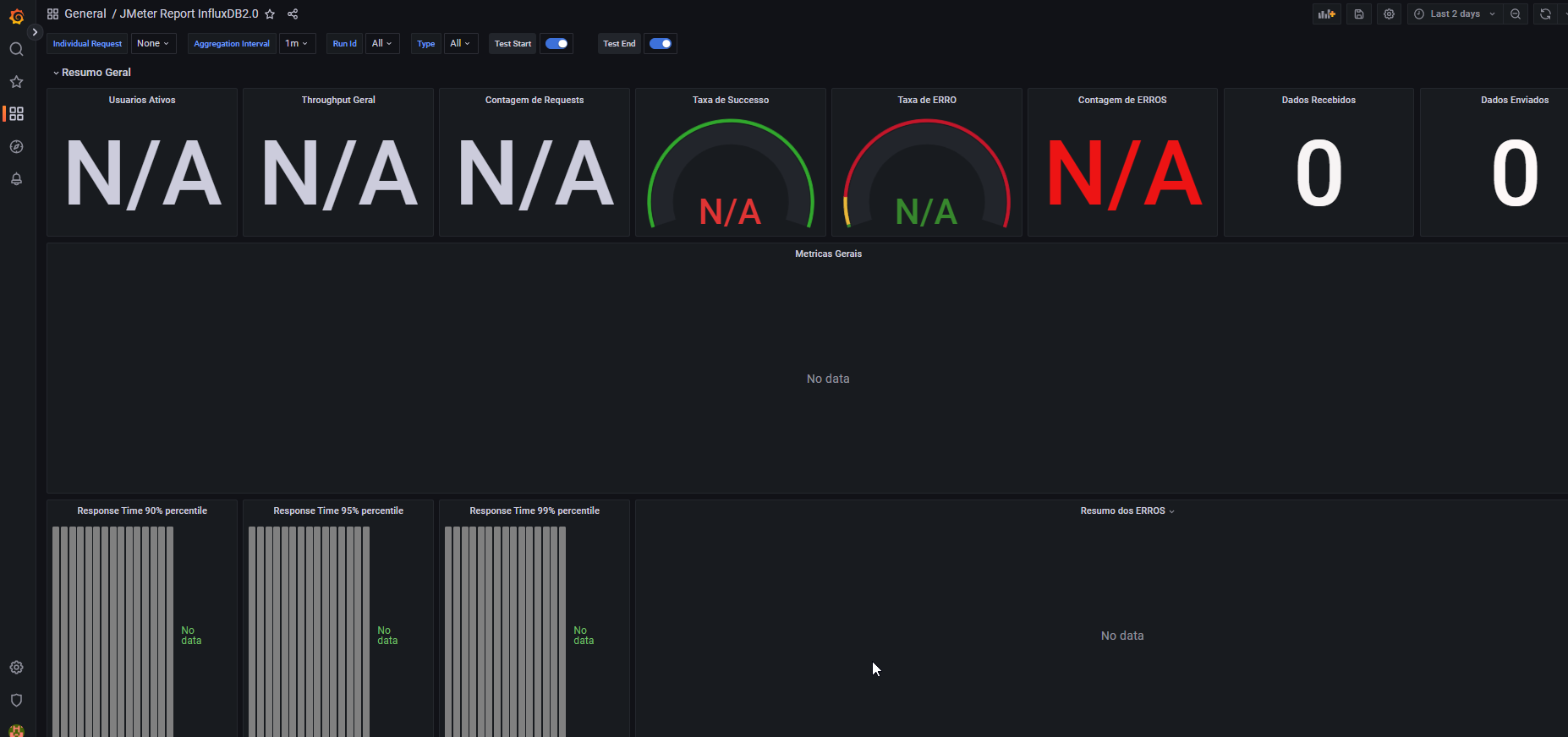
Procure pelo arquivo dentro do /app-load-test/grafana/etc/provisioning/dashboards



Selecione influxdb Data Sourcee clique em Import



Pronto! Agora é so colocar o listener no seu jmeter e rodar seu teste!



Observação: No repositório já tem a ultima versão do Jmeter com todos plugins necessários inclusive o Listener necessário para o influxdb2, e alguns scripts de exemplos, é so descompactar e colar no endereço C:\Users\Public\Documents\apache-jmeter-5.5

# Instalação InfluxDB via Docker(manualmente)

1. Iniciar o Ubuntu 20.04.

No terminal do Linux, executar o comando sudo dockerd para inicializar o Docker.

Texto

Descrição gerada automaticamente

2. Abra um novo terminal do Linux para executar os comandos abaixo.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

3. Baixa, criar container e executar o Influxdb.

sudo docker run -d -p 8086:8086 -v influxdb:/var/lib/influxdb --name influxdb influxdb:1.8.10

4. Iniciar o container do Influxdb.

sudo docker start influxdb

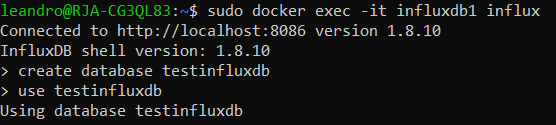
5. Entrar no container do Influxdb.

sudo docker exec -it influxdb influx

6. Criar database e conceder permissão.

create database testinfluxdb

use testinfluxdb



Obs.: Sempre ao inicializar o Docker, precisa iniciar o container do Influxdb.

sudo docker start influxdb

# Integrando o JMeter com o InfluxDB

1. Abra o JMeter (sugestão a partir da versão 5.4).

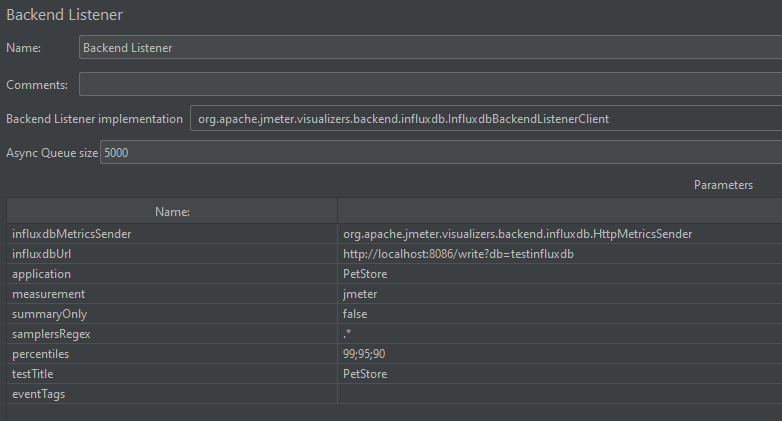
2. Em ***Thread Group*** procurar por ***Backend Listener.***

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

3. Configurar o ***Backend Listener*** para que o JMeter se comunique com o InfluxDB.

Realizar a configuração conforme a imagem abaixo:



3.1. No campo influxdbUrl o valor de host\_to\_change deverá ser informado o IP da máquina onde o InfluxDB está sendo executado, provavelmente informando “localhost” resolve.

3.2. Também para o campo influxdbUrl é necessário informar o nome da database criada anteriormente.

3.3. Em “application” informar o nome da aplicação.

# Configuração do InfluxDB no Grafana

1. Abra o Grafana acessando a URL: <http://localhost:3000>.

No primeiro acesso informe o usuário e senha padrão, admin/admin.

2. Criando o ***Data Source***.

2.1. Na home, clique no link ***Add data source.***

2.2. Localize em ***Time series databases*** o ***InfluxDB***.

Tela de celular com aplicativo aberto

Descrição gerada automaticamente

3. Configurando o ***Data Source***.

3.1. Na URL, precisará informar o IP do container do InfluxDB, então:

No terminal do Linux, informe o comando sudo docker exec -it influxdb bash e depois o comando ip address.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

3.2. No campo ***Database*** informe o database criado no terminal Linux.

Ao clicar no botão ***Save & test***, você será informado se a conexão com o influxDB foi realizada com sucesso.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

# Importando Dashboard no Grafana

1. Faça download de algum Dashboard compatível com o InfluxDB 1.8.

Link para download: <https://grafana.com/grafana/dashboards/5496>.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

2. Importando Dashboard.

2.1. Na página inicial do Grafana, clicar no menu  da barra a esquerda e em ***Import***.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

2.2. Clicar no botão ***Upload JSON file*** e abra o arquivo JSON do dashboard baixado acima.

Tela de celular com publicação numa rede social

Descrição gerada automaticamente

2.3. Selecione no campo ***DB name*** o database criado anteriormente e clique no botão ***Import***.

Tela de computador com texto preto sobre fundo escuro

Descrição gerada automaticamente

2.4. Será apresentado o painel do Dashboard, que será atualizado conforme o influxDB for recebendo as informações do teste executado no JMeter.

Tela de computador com jogo

Descrição gerada automaticamente