ATIVIDADE TÉCNICA - ANALISTA DE INFRAESTRUTURA

Candidato: José Bezerra
Repositório: https://github.com/JoJoseB/projeto-esig

TAREFAS

Criar um contêiner para uma aplicação simples e expor as métricas via Jolokia

Execute a atividade da Etapa 1 em um cluster kubernetes.

Configurar o Prometheus para coletar métricas via Jolokia e do Node Exporter.

TAREFA 1

OBJETIVOS

01.

Baixe o arquivo WAR do Jenkins mais recente do site oficial ou de outra fonte confiável.

02.

Configure um servidor de aplicação local, como JBoss ou Tomcat e Implante o arquivo WAR do Jenkins no servidor configurado.

03.

Inicie o servidor de aplicação e verifique se o Jenkins está sendo executado corretamente e Acesse a interface web do Jenkins e verifique se é possível fazer login e acessar as configurações básicas

04.

Construa e execute o contêiner, garantindo que as métricas estejam acessíveis via jolokia.

DESENVOLVIMENTO

O repositório inclui um Dockerfile projetado para construir um container Tomcat, que durante o processo de build, o container realiza o download dos arquivos .war do Jenkins e do Jolokia diretamente de seus repositórios oficiais, armazenando-os em /usr/local/tomcat/webapps/ para implantação automática no servidor.

Paralelamente, para viabilizar a coleta de métricas compatíveis com Prometheus, o container configura o JMX Exporter, responsável por coletar dados de MBeans do JMX e expô-los em formato legível através de um endpoint HTTP na porta 9400.

Dockerfile > ...

```
1  FROM tomcat:11.0.6-jdk21
2
3  # Baixa JMX EXPORTER
4  ADD https://repo1.maven.org/maven2/io/prometheus/jmx/jmx_prometheus_javaagent/1.0.1/jmx_prometheus_javaagent-1.0.1.jar \
5      /opt/jmx_exporter/jmx_exporter.jar
6
7  # Criando arquivo de configuração do JMX EXPORTER
8  RUN mkdir -p /opt/jmx_exporter/config
9  COPY <<EOF /opt/jmx_exporter/config/config.yml
10 rules:
11 - pattern: ".*"
12 EOF
13
14 # Baixa jolokia WAR agent
15 ADD https://repo1.maven.org/maven2/org/jolokia/jolokia-agent-war-unsecured/2.2.9/jolokia-agent-war-unsecured-2.2.9.war \
16     /usr/local/tomcat/webapps/jolokia.war
17
18 # Baixa jenkins WAR agent
19 ADD https://get.jenkins.io/war-stable/2.504.1/jenkins.war /usr/local/tomcat/webapps/jenkins.war
20
21 # Configurando o Catalina para o JMX exporter
22 ENV CATALINA_OPTS="-javaagent:/opt/jmx_exporter/jmx_exporter.jar=9400:/opt/jmx_exporter/config/config.yml"
23
24 # Expose Tomcat e JMX EXPORTER porta
25 EXPOSE 8080 9400
26
27 CMD ["catalina.sh", "run"]
```

TAREFA 2

OBJETIVOS

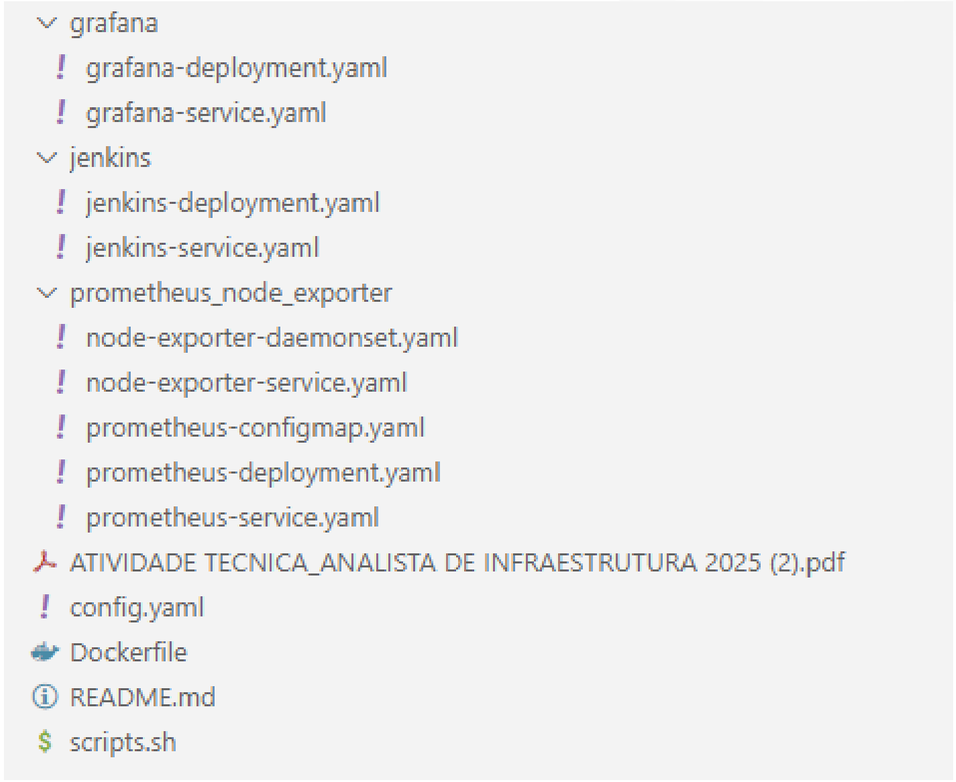
01.

Crie os manifestos YAML necessários para implantar a aplicação no kubernetes, incluindo Deployment e Service.

DESENVOLVIMENTO

Foram desenvolvidos todos os arquivos necessários para a implantação do cluster Kubernetes, incluindo os seguintes componentes:

- Deployments
  - Prometheus, Grafana e Jenkins
- Service
  - Prometheus, Grafana, Jenkins, Node Exporter
- Configmap
  - Prometheus
- Daemonset
  - Node Exporter



TAREFA 3

OBJETIVOS

01.

Implante o Prometheus no cluster Kubernetes.

02.

Configure o Prometheus para coletar métricas da aplicação utilizando o endpoint Jolokia.

03.

Implante o Node Exporter para coletar métricas dos nós do cluster.

04.

Garanta que as métricas coletadas estejam acessíveis no Prometheus.

DESENVOLVIMENTO

Como solicitado, foi implantado e configurado o prometheus colentado métricas do container com Jolokia e o do Node explorer.

Nesse exemplo, estão sendo coletados o IO do disco dos nós e o números de threads sendo utilizadas pelo java nas aplicações no container do Jolokia