## Documentação do Dockerfile para MySQL com Zabbix Agent

### Introdução

Este documento fornece uma visão detalhada da construção de uma imagem Docker que combina MySQL 8.0.37 com o Zabbix Agent. Este Dockerfile foi desenvolvido levando em consideração as melhores práticas de DevOps e segurança. Abaixo, abordaremos cada seção do Dockerfile e explicaremos o propósito e a importância de cada comando.

### Conteúdo do Dockerfile

```dockerfile

# Use a imagem base do MySQL 8.0.37-Debian

FROM mysql:8.0.37-bookworm

# Instale as dependências necessárias para o Zabbix Agent e o wget

RUN apt-get update \

&& apt-get install -y wget gnupg2 \

&& wget https://repo.zabbix.com/zabbix-official-repo.key \

&& apt-key add zabbix-official-repo.key \

&& echo "deb https://repo.zabbix.com/zabbix/6.5/debian bookworm main" > /etc/apt/sources.list.d/zabbix.list \

&& apt-get update \

&& apt-get install -y zabbix-agent \

&& rm -rf /var/lib/apt/lists/\*

# Crie o diretório necessário para o PID do Zabbix Agent

RUN mkdir -p /var/run/zabbix && chown zabbix:zabbix /var/run/zabbix

# Copie o arquivo de configuração do Zabbix Agent para dentro do contêiner

COPY zabbix\_agentd.conf /etc/zabbix/zabbix\_agentd.conf

# Copie o script de entrada para o contêiner e defina permissões

COPY entrypoint.sh /usr/local/bin/entrypoint.sh

RUN chmod +x /usr/local/bin/entrypoint.sh

# Exponha a porta padrão do Zabbix Agent

EXPOSE 10050

# Defina o script de entrada como o comando inicial

ENTRYPOINT ["/usr/local/bin/entrypoint.sh"]

```

### Explicação Detalhada

#### 1. Escolha da Imagem Base

```dockerfile

FROM mysql:8.0.37-bookworm

```

A imagem base escolhida é o MySQL 8.0.37, baseada no Debian Bookworm. Esta escolha oferece uma combinação de um banco de dados robusto e um sistema operacional estável.

#### 2. Instalação de Dependências

```dockerfile

RUN apt-get update \

&& apt-get install -y wget gnupg2 \

&& wget https://repo.zabbix.com/zabbix-official-repo.key \

&& apt-key add zabbix-official-repo.key \

&& echo "deb https://repo.zabbix.com/zabbix/6.5/debian bookworm main" > /etc/apt/sources.list.d/zabbix.list \

&& apt-get update \

&& apt-get install -y zabbix-agent \

&& rm -rf /var/lib/apt/lists/\*

```

Este bloco de comandos realiza as seguintes ações:

- Atualiza a lista de pacotes (`apt-get update`).

- Instala o `wget` e o `gnupg2` necessários para baixar e adicionar a chave GPG do repositório Zabbix.

- Baixa e adiciona a chave GPG do repositório oficial do Zabbix.

- Adiciona o repositório Zabbix à lista de fontes (`sources.list.d`).

- Atualiza novamente a lista de pacotes para incluir o repositório Zabbix.

- Instala o Zabbix Agent.

- Remove a lista de pacotes (`/var/lib/apt/lists/\*`) para reduzir o tamanho da imagem e melhorar a segurança.

#### 3. Criação de Diretórios

```dockerfile

RUN mkdir -p /var/run/zabbix && chown zabbix:zabbix /var/run/zabbix

```

Este comando cria o diretório `/var/run/zabbix` e define as permissões apropriadas para o usuário `zabbix`, garantindo que o Zabbix Agent tenha acesso adequado para criar seu arquivo PID.

#### 4. Configuração do Zabbix Agent

```dockerfile

COPY zabbix\_agentd.conf /etc/zabbix/zabbix\_agentd.conf

```

Copia o arquivo de configuração do Zabbix Agent (`zabbix\_agentd.conf`) do sistema host para dentro do contêiner no diretório `/etc/zabbix/`.

#### 5. Script de Entrada

```dockerfile

COPY entrypoint.sh /usr/local/bin/entrypoint.sh

RUN chmod +x /usr/local/bin/entrypoint.sh

```

- Copia o script de entrada (`entrypoint.sh`) para o diretório `/usr/local/bin/` dentro do contêiner.

- Define permissões executáveis para o script (`chmod +x`).

#### 6. Exposição da Porta

```dockerfile

EXPOSE 10050

```

Exponha a porta 10050, que é a porta padrão para comunicação do Zabbix Agent.

#### 7. Definição do Entrypoint

```dockerfile

ENTRYPOINT ["/usr/local/bin/entrypoint.sh"]

```

Define o script de entrada (`entrypoint.sh`) como o comando inicial a ser executado quando o contêiner é iniciado.

### Melhorias e Considerações de Segurança

1. \*\*Minimize o uso de RUN:\*\* Sempre que possível, combine múltiplos comandos `RUN` em uma única instrução para reduzir o número de camadas na imagem final.

2. \*\*Reduza o Tamanho da Imagem:\*\* Utilize o comando `rm -rf /var/lib/apt/lists/\*` após a instalação de pacotes para remover os caches de pacotes e reduzir o tamanho da imagem.

3. \*\*Gerenciamento de Permissões:\*\* Certifique-se de que os arquivos e diretórios importantes tenham as permissões corretas para evitar problemas de segurança.

4. \*\*Arquivo de Configuração:\*\* Mantenha o arquivo de configuração (`zabbix\_agentd.conf`) fora do repositório de código-fonte para evitar a exposição de informações sensíveis.

5. \*\*Script de Entrada:\*\* Garanta que o script de entrada (`entrypoint.sh`) inclua verificações de erro e práticas seguras de manuseio de dados.

### Conclusão

Este Dockerfile configura uma imagem combinada de MySQL com o Zabbix Agent, seguindo as melhores práticas de DevOps e segurança. A documentação detalhada de cada comando ajuda a entender o propósito e a importância de cada etapa no processo de construção da imagem. Esta abordagem garante uma configuração robusta, segura e eficiente para monitoramento com o Zabbix Agent em um ambiente Dockerizado.