

Kafka

```
WARNING: Some networks were defined but are not used by any service: emqx-  
net  
grafana is up-to-date  
prometheus is up-to-date  
Creating zookeeper ... error
```

```
ERROR: for zookeeper Cannot create container for service zookeeper: Conflict.  
The container name "/zookeeper" is already in use by container  
"f67adc3a840d55db5cedf6770f3cc435b481421918a97c46de272acf31d8ac92".  
You have to remove (or rename) that container to be able to reuse that name.
```

```
ERROR: for zookeeper Cannot create container for service zookeeper: Conflict.  
The container name "/zookeeper" is already in use by container  
"f67adc3a840d55db5cedf6770f3cc435b481421918a97c46de272acf31d8ac92".  
You have to remove (or rename) that container to be able to reuse that name.  
ERROR: Encountered errors while bringing up the project.
```

ChatGPT

O erro que você está encontrando ocorre porque já existe um contêiner em execução com o nome `/zookeeper`. Para resolver esse problema, você pode seguir um dos passos abaixo:

1. Parar e remover o contêiner existente:

1. Parar o contêiner existente:

```
docker stop zookeeper
```

2. Remover o contêiner existente:

```
shCopiar código  
docker rm zookeeper
```

Depois de parar e remover o contêiner existente, tente subir o projeto novamente:

```
shCopiar código  
docker-compose up
```

2. Remover o contêiner existente forçadamente:

Você pode forçar a remoção do contêiner existente diretamente:

```
shCopiar código  
docker rm -f zookeeper
```

Depois disso, execute novamente o `docker-compose up`.

3. Renomear o contêiner existente:

Se você preferir não parar o contêiner existente, você pode renomeá-lo:

```
shCopiar código  
docker rename zookeeper zookeeper_old
```

Em seguida, tente novamente o `docker-compose up`.

Verificação do contêiner existente:

Antes de tomar qualquer uma das ações acima, você pode verificar o estado do contêiner existente com:

```
shCopiar código  
docker ps -a | grep zookeeper
```

Isso lhe dará uma ideia do estado do contêiner e se ele está em execução ou parado.

Solução com Docker Compose:

Em casos onde o Docker Compose pode estar tentando criar contêineres com nomes em conflito, você também pode tentar usar a opção de remover contêineres órfãos:

```
shCopiar código  
docker-compose up --remove-orphans
```

Escolha a opção que melhor se adequa à sua situação e tente novamente. Essas etapas devem resolver o conflito de nomes de contêineres e permitir que você inicie seu ambiente com sucesso.

4o

como paro todos os containers

ChatGPT

Para parar todos os contêineres Docker em execução, você pode usar o comando `docker stop` com uma subcomando que seleciona todos os contêineres. Aqui está como você pode fazer isso:

Parar todos os contêineres em execução:

```
shCopiar código  
docker stop $(docker ps -q)
```

Parar e remover todos os contêineres:

Se você também quiser remover todos os contêineres parados, pode fazer o seguinte:

1. Parar todos os contêineres em execução:

```
shCopiar código  
docker stop $(docker ps -q)
```

2. Remover todos os contêineres:

```
shCopiar código  
docker rm $(docker ps -a -q)
```

Parar, remover e limpar o sistema Docker:

Para uma limpeza mais completa, incluindo a remoção de imagens, volumes e redes não utilizados, você pode usar o comando `docker system prune`:

```
shCopiar código  
docker system prune -a
```

Nota: O comando `docker system prune -a` removerá:

- Todos os contêineres parados
- Todas as redes não usadas
- Todas as imagens não referenciadas por contêineres
- Todos os volumes não utilizados (com a opção `-volumes`)

Se você quiser uma limpeza completa incluindo volumes, adicione a opção `--volumes`:

```
shCopiar código  
docker system prune -a --volumes
```

Resumo dos comandos úteis:

- Parar todos os contêineres:

```
shCopiar código
docker stop $(docker ps -q)
```

- Parar e remover todos os contêineres:

```
shCopiar código
docker stop $(docker ps -q) && docker rm $(docker ps -a -q)
```

- Limpar o sistema Docker (imagens, volumes, redes não utilizadas):

```
shCopiar código
docker system prune -a --volumes
```

Escolha o método que melhor se adapta às suas necessidades. Esses comandos ajudarão a liberar recursos e evitar conflitos de nomes de contêineres.

4o

WARNING: Some networks were defined but are not used by any service: emqx-net

Creating zookeeper ... done

Creating grafana ... done

Creating prometheus ... done

Creating kafka ... done

Creating emqx ...

Creating emqx ... error

ERROR: for emqx Cannot start service emqx-enterprise: driver failed programming external connectivity on endpoint emqx

(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):
Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in use

ERROR: for emqx-enterprise Cannot start service emqx-enterprise: driver failed
programming external connectivity on endpoint emqx
(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):
Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in
WARNING: Some networks were defined but are not used by any service: emqx-
net

Creating zookeeper ... done
Creating grafana ... done
Creating prometheus ... done
Creating kafka ... done
Creating emqx ...
Creating emqx ... error

ERROR: for emqx Cannot start service emqx-enterprise: driver failed
programming external connectivity on endpoint emqx
(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):
Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in use

ERROR: for emqx-enterprise Cannot start service emqx-enterprise: driver failed
programming external connectivity on endpoint emqx
(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):
Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in use
ERROR: Encountered errors while bringing up the project.use
ERROR: Encountered errors while bringing up the project.

ChatGPT

O erro que você está enfrentando ocorre porque a porta **1883** já está em uso por outro processo. Isso pode acontecer se outro contêiner ou serviço já estiver usando essa porta.

Passos para resolver o problema:

1. Identificar o processo que está usando a porta **1883** :

Você pode usar o comando `lsof` para listar todos os processos que estão usando a porta `1883`.

```
shCopiar código
sudo lsof -i :1883
```

Esse comando mostrará qual processo está usando a porta. Se for um contêiner Docker, você verá o nome ou ID do contêiner.

2. Parar o processo ou contêiner que está usando a porta:

Se for um contêiner Docker, você pode parar e remover o contêiner que está usando a porta.

```
shCopiar código
docker stop <container_id>
docker rm <container_id>
```

Substitua `<container_id>` pelo ID do contêiner que você encontrou no passo anterior.

3. Reiniciar os serviços Docker:

Após liberar a porta `1883`, tente reiniciar os serviços Docker.

```
shCopiar código
docker-compose up
```

Alternativa: Alterar a porta do serviço EMQX

Se você não pode parar o processo que está usando a porta `1883`, você pode alterar a configuração do seu `docker-compose.yml` para usar uma porta diferente para o serviço EMQX.

Edite o arquivo `docker-compose.yml` e altere a porta:

```
ymlCopiar código
services:
  emqx:
    image: emqx/emqx:latest
    ports:
      - "1884:1883" # Altere a porta host de 1883 para 1884
```

Isso mapeia a porta `1883` do contêiner para a porta `1884` no host.

Verificar e ajustar as redes Docker

Você mencionou um aviso sobre redes definidas, mas não usadas. Certifique-se de que todas as redes definidas no seu `docker-compose.yml` estão sendo referenciadas corretamente pelos serviços.

Comandos úteis:

- **Listar todos os contêineres Docker (em execução e parados):**

```
shCopiar código
docker ps -a
```

- **Parar todos os contêineres Docker em execução:**

```
shCopiar código
docker stop $(docker ps -q)
```

- **Remover todos os contêineres Docker (parados):**

```
shCopiar código
docker rm $(docker ps -a -q)
```


- **Remover todos os volumes, redes e imagens não utilizados:**

```
shCopiar código  
docker system prune -a --volumes
```

Após essas etapas, você deve conseguir levantar seus serviços Docker sem conflitos de porta.

4o

WARNING: Some networks were defined but are not used by any service: emqx-net

Creating zookeeper ... done

Creating grafana ... done

Creating prometheus ... done

Creating kafka ... done

Creating emqx ...

Creating emqx ... error

ERROR: for emqx Cannot start service emqx-enterprise: driver failed

programming external connectivity on endpoint emqx

(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):

Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in use

ERROR: for emqx-enterprise Cannot start service emqx-enterprise: driver failed

programming external connectivity on endpoint emqx

(7a73e17f109ae5dbb69e99805b2c0ad0eac99ee88d1c316dd7e254b00e0314a5):

Error starting userland proxy: listen tcp4 0.0.0.0:1883: bind: address already in use

ERROR: Encountered errors while bringing up the project.

ChatGPT

O erro indica que a porta **1883** já está em uso por outro processo, o que impede o Docker de iniciar o serviço **emqx**. Aqui estão os passos detalhados para resolver o problema:

Passo 1: Identificar o Processo que está Usando a Porta **1883**

1. Verifique qual processo está usando a porta 1883 :

```
shCopiar código
sudo lsof -i :1883
```

O comando exibirá informações sobre o processo que está usando a porta 1883 .

Passo 2: Parar o Processo ou Contêiner

1. Pare e remova o contêiner ou processo que está usando a porta 1883 :

Se for um contêiner Docker, você pode parar e remover o contêiner:

```
shCopiar código
docker stop <container_id>
docker rm <container_id>
```

Substitua <container_id> pelo ID do contêiner que você encontrou no passo anterior.

Passo 3: Alterar a Porta do Serviço EMQX

Se não for possível parar o processo que está usando a porta 1883 , você pode alterar a porta no arquivo `docker-compose.yml` :

1. Edite o `docker-compose.yml` para alterar a porta:

```
yamlCopiar código
services:
  emqx:
    image: emqx/emqx:latest
    ports:
      - "1884:1883" # Altere a porta do host de 1883 para
1884
```

Passo 4: Verificar e Ajustar Redes Docker

1. **Certifique-se de que todas as redes definidas no `docker-compose.yml` estão sendo usadas corretamente.**

Passo 5: Reiniciar os Serviços Docker

1. **Reinicie os serviços Docker:**

```
shCopiar código  
docker-compose up
```

Comandos Úteis

- **Listar todos os contêineres Docker (em execução e parados):**

```
shCopiar código  
docker ps -a
```

- **Parar todos os contêineres Docker em execução:**

```
shCopiar código  
docker stop $(docker ps -q)
```

- **Remover todos os contêineres Docker (parados):**

```
shCopiar código  
docker rm $(docker ps -a -q)
```

- **Remover todos os volumes, redes e imagens não utilizados:**

```
shCopiar código  
docker system prune -a --volumes
```

Exemplos de `docker-compose.yml` com Redes

Aqui está um exemplo básico de um arquivo `docker-compose.yml` que define e usa redes corretamente:

```
ymlCopiar código
version: '3.7'

networks:
  emqx-net:
    driver: bridge

services:
  zookeeper:
    image: zookeeper
    networks:
      - emqx-net

  grafana:
    image: grafana/grafana
    networks:
      - emqx-net

  prometheus:
    image: prom/prometheus
    networks:
      - emqx-net

  kafka:
    image: wurstmeister/kafka
    networks:
      - emqx-net

  emqx:
```

```
image: emqx/emqx:latest
ports:
  - "1884:1883"
networks:
  - emqx-net
```

Certifique-se de que o `docker-compose.yml` está correto e que as redes definidas estão sendo usadas pelos serviços apropriados.

1/2

4o

➔ `mqttx sub -t mqttx/simulate/tesla/+`

zsh: command not found: mqttx

ChatGPT

O erro "command not found: mqttx" indica que o comando `mqttx` não está disponível no seu sistema. Isso pode ocorrer por diversos motivos, como o `MQTTX` não estar instalado ou não estar no seu `PATH`.

Aqui estão os passos para instalar e usar o `MQTTX`:

Passo 1: Instalar o `MQTTX`

1. Instalar via Homebrew (macOS ou Linux)

Se você está usando macOS ou Linux e tem o Homebrew instalado, você pode instalar o `MQTTX` com o seguinte comando:

```
shCopiar código
brew install mqttx
```

2. Instalar via npm

`MQTTX` também pode ser instalado globalmente usando o npm:

```
shCopiar código  
npm install -g mqtttx
```

3. Baixar e Instalar Manualmente

Você pode baixar os binários diretamente do site oficial do MQTTX e seguir as instruções de instalação para o seu sistema operacional.

Passo 2: Verificar a Instalação

Após a instalação, verifique se o `MQTTX` está instalado corretamente e disponível no seu `PATH`:

```
shCopiar código  
mqtttx --version
```

Isso deve retornar a versão do `MQTTX` instalada.

Passo 3: Usar o Comando `mqtttx`

Agora você deve ser capaz de executar o comando que deseja:

```
shCopiar código  
mqtttx sub -t mqtttx/simulate/tesla/+
```

Passo 4: Adicionar `MQTTX` ao PATH (se necessário)

Se você instalou o `MQTTX` manualmente, pode ser necessário adicionar o diretório de instalação ao seu `PATH`. Isso pode ser feito editando o seu arquivo de configuração de shell (`~/.bashrc`, `~/.zshrc`, etc.).

Por exemplo, se você instalou o `MQTTX` em `/usr/local/bin`, adicione a seguinte linha ao seu arquivo de configuração do shell:

```
shCopiar código
export PATH=$PATH:/usr/local/bin
```

Depois, recarregue o arquivo de configuração:

```
shCopiar código
source ~/.zshrc # ou ~/.bashrc, dependendo do seu shell
```

Resumo

1. **Instalar o** `MQTTX` usando Homebrew, npm, ou manualmente.
2. **Verificar a instalação** com `mqttx --version`.
3. **Executar o comando desejado** (`mqttx sub -t mqttx/simulate/tesla/+`).

Seguindo esses passos, você deve ser capaz de usar o `MQTTX` no seu sistema sem problemas.

4o

anamarcacini in CarregadorBateria2 on ÷ main [!] took 2,4s

➔ brew install mqttx

zsh: command not found: brew

anamarcacini in CarregadorBateria2 on ÷ main [!]

➔ npm install -g mqttx

npm ERR! code EACCES

npm ERR! syscall mkdir

npm ERR! path /usr/lib/node_modules/mqttx

npm ERR! errno -13

npm ERR! Error: EACCES: permission denied, mkdir '/usr/lib/node_modules/mqttx'

npm ERR! [Error: EACCES: permission denied, mkdir

'/usr/lib/node_modules/mqttx'] {

npm ERR! errno: -13,

```
npm ERR! code: 'EACCES',
npm ERR! syscall: 'mkdir',
npm ERR! path: '/usr/lib/node_modules/mqtttx'
npm ERR! }
npm ERR!
npm ERR! The operation was rejected by your operating system.
npm ERR! It is likely you do not have the permissions to access this file as the
current user
npm ERR!
npm ERR! If you believe this might be a permissions issue, please double-check
the
npm ERR! permissions of the file and its containing directories, or try running
npm ERR! the command again as root/Administrator.
```

```
npm ERR! A complete log of this run can be found in:
/home/anamarcacini/.npm/_logs/2024-06-06T14_17_00_535Z-debug-0.log
anamarcacini in CarregadorBateria2 on ÷ main [!] took 3,0s
➔ mqtttx --version
```

```
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ sudo npm install -g mqtttx
```

```
npm WARN deprecated inflight@1.0.6: This module is not supported, and leaks
memory. Do not use it. Check out lru-cache if you want a good and tested way to
coalesce async requests by a key value, which is much more comprehensive and
powerful.
```

```
npm WARN deprecated glob@7.2.3: Glob versions prior to v9 are no longer
supported
```

```
added 78 packages in 3s
```

```
4 packages are looking for funding
```

```
run `npm fund` for details
```

```
npm notice
```

```
npm notice New minor version of npm available! 10.5.0 → 10.8.1
```

```
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.1
```



```
npm notice Run npm install -g npm@10.8.1 to update!
npm notice
anamarcacini in CarregadorBateria2 on ÷ main [!] took 2,8s
➔ npm install -g mqtttx
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ mqtttx --version
```

```
zsh: command not found: mqtttx
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ export PATH=$PATH:/usr/local/bin
```

```
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ mqtttx --version
```

```
zsh: command not found: mqtttx
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ source ~/.zshrc # ou ~/.bashrc, dependendo do seu shell
```

```
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ source ~/.zshrc
```

```
anamarcacini in CarregadorBateria2 on ÷ main [!]
➔ mqtttx --version
```

```
zsh: command not found: mqtttx
```

ChatGPT

Parece que o comando `mqtttx` não foi adicionado corretamente ao seu PATH ou foi instalado em um local diferente. Vamos verificar e corrigir isso.

Verifique o Local de Instalação do `mqtttx`

Primeiro, vamos encontrar onde o `mqtttx` foi instalado:

```
shCopiar código
npm root -g
```

Isso mostrará o caminho global onde os pacotes npm são instalados. Deve ser algo como `/usr/local/lib/node_modules`.

Encontre o Binário do `mqttx`

Dentro do diretório retornado pelo comando acima, verifique se o binário `mqttx` está presente:

```
shCopiar código
ls /usr/local/lib/node_modules/mqttx/bin
```

Substitua `/usr/local/lib/node_modules` pelo caminho retornado se for diferente.

Adicione o Diretório de Binários ao PATH

Se o binário `mqttx` estiver lá, adicione esse diretório ao PATH. Supondo que o binário esteja em `/usr/local/lib/node_modules/mqttx/bin`, adicione o seguinte ao seu `~/.zshrc`:

```
shCopiar código
export PATH=$PATH:/usr/local/lib/node_modules/mqttx/bin
```

Depois, recarregue o arquivo de configuração:

```
shCopiar código
source ~/.zshrc
```

Verifique a Instalação

Tente verificar a versão do `mqttx` novamente:

```
shCopiar código
mqttx --version
```

Alternativa: Instalar Usando Yarn

Como alternativa, você pode tentar instalar `mqttx` usando `yarn` (que pode lidar com permissões de forma diferente):

1. Instale o Yarn se ainda não o tiver:

```
shCopiar código
npm install -g yarn
```

2. Instale o `mqttx` usando o Yarn:

```
shCopiar código
yarn global add mqttx
```

3. Verifique se o `mqttx` está disponível:

```
shCopiar código
mqttx --version
```

Resumo dos Comandos

```
shCopiar código
# Encontrar o diretório global de instalação do npm
npm root -g

# Verificar se o binário do mqttx está presente
ls /usr/local/lib/node_modules/mqttx/bin
```

```
# Adicionar o diretório de binários ao PATH (ajuste o caminho conforme necessário)
echo 'export PATH=$PATH:/usr/local/lib/node_modules/mqtttx/bin' >> ~/.zshrc
source ~/.zshrc

# Verificar a versão do mqtttx
mqtttx --version
```

Se ainda assim não funcionar, me avise e exploraremos outras alternativas.

4o

anamarcacini in CarregadorBateria2/example on ÷ main [!]

→ ls /usr/lib/node_modules/mqtttx

index.js node_modules package.json readme.md

ChatGPT

Ótimo, parece que o diretório do `mqtttx` está presente em

`/usr/lib/node_modules/mqtttx/bin`. Vamos adicionar esse diretório ao seu PATH.

```
shCopiar código
echo 'export PATH=$PATH:/usr/lib/node_modules/mqtttx/bin' >>
~/.zshrc
source ~/.zshrc
```

Depois de executar esses comandos, tente verificar a versão do `mqtttx` novamente:

```
shCopiar código
mqtttx --version
```

Isso deve funcionar agora.

3.5

não tem o bin

anamarcacini in CarregadorBateria2/example on ÷ main [!]

→ ls /usr/lib/node_modules/mqttx -lsa

total 28

4 drwxr-xr-x 3 root root 4096 jun 6 11:17 .

4 drwxr-xr-x 7 root root 4096 jun 6 11:17 ..

8 -rw-r--r-- 1 root root 6852 jun 6 11:17 index.js

4 drwxr-xr-x 76 root root 4096 jun 6 11:17 node_modules

4 -rw-r--r-- 1 root root 1024 jun 6 11:17 package.json

4 -rw-r--r-- 1 root root 4013 jun 6 11:17 readme.md

ChatGPT

Entendi. Parece que o executável não está localizado no diretório `bin`. Vamos tentar encontrar o caminho correto para o executável.

Você pode tentar encontrar o caminho do executável `mqttx` usando o seguinte comando:

shCopiar código

```
find /usr/lib/node_modules/mqttx -type f -executable -name 'mqttx'
```

Isso deve imprimir o caminho para o executável `mqttx`, caso ele exista no seu sistema.

3.5

anamarcacini in CarregadorBateria2/example on ÷ main [!]

→ find /usr/lib/node_modules/mqttx -type f -executable -name 'mqttx'

anamarcacini in CarregadorBateria2/example on ÷ main [!]

→

ChatGPT

Parece que o comando não retornou nada, o que indica que o arquivo executável `mqttx` não está presente no diretório `/usr/lib/node_modules/mqttx`. Isso pode explicar por que o comando `mqttx` não está sendo encontrado.

Você pode tentar verificar se o `mqttx` foi instalado corretamente ou se está disponível em outro local. Talvez a instalação não tenha incluído o arquivo binário ou ele tenha sido instalado em um local diferente.

Se possível, tente verificar a documentação da instalação do `mqttx` ou entre em contato com o suporte do projeto para obter mais informações sobre como encontrar ou corrigir o problema de instalação.

3.5

```
anamarcacini in CarregadorBateria2/example on ÷ main [!]
```

```
→ yarn global add mqttx
```

```
yarn global v1.22.22
```

```
[1/4] Resolving packages...
```

```
warning mqttx > mqtt > help-me > glob-stream > glob@7.2.3: Glob versions prior to v9 are no longer supported
```

```
warning mqttx > mqtt > help-me > glob-stream > glob > inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out Iru-cache if you want a good and tested way to coalesce async requests by a key value, which is much more comprehensive and powerful.
```

```
warning mqttx > mqtt > help-me > glob-stream > unique-stream > through2-filter@3.1.0: this package has been deprecated for breaking semver major
```

```
[2/4] Fetching packages...
```

```
[3/4] Linking dependencies...
```

```
[4/4] Building fresh packages...
```

```
warning "mqttx@1.0.1" has no binaries
```

```
Done in 3.14s.
```

```
anamarcacini in CarregadorBateria2/example on ÷ main [!] took 3,3s
```

```
→ mqttx --version
```

```
zsh: command not found: mqttx
```

```
anamarcacini in CarregadorBateria2/example on ÷ main [!]
```

```
→
```

ChatGPT

Parece que o pacote `mqttx` foi instalado globalmente usando o Yarn, mas não possui binários associados, o que é necessário para usar o comando `mqttx` diretamente no terminal.

Você pode verificar se o `mqttx` foi instalado no local correto verificando os diretórios de instalação global do Yarn. Normalmente, os binários de pacotes globais do Yarn são instalados em um diretório específico, que geralmente está incluído no `PATH` do sistema.

Tente encontrar o diretório de instalação global do Yarn e verifique se o binário `mqttx` está presente lá. Se estiver presente, você pode adicionar esse diretório ao seu `PATH` para que o comando `mqttx` seja reconhecido pelo terminal. Se o binário não estiver presente, você pode precisar verificar a instalação do pacote ou tentar instalar novamente.

3.5

anamarcacini in CarregadorBateria2/example on ÷ main [!]

→ `curl -LO https://www.emqx.com/zh/downloads/MQTTX/${version}/mqttx-cli-linux-x64`

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
		Dload	Upload	Total	Spent	Left	Speed
100	146	0	146	0	0	96	0 --:--:-- 0:00:01 --:--:-- 96

anamarcacini in CarregadorBateria2/example on ÷ main [!?

→ `sudo install ./mqttx-cli-linux-x64 /usr/local/bin/mqttx`

anamarcacini in CarregadorBateria2/example on ÷ main [!?

→ `mqttx --version`

`/usr/local/bin/mqttx: 1: {url:/zh/downloads/MQTTX//mqttx-cli-linux-x64,statusCode:404,statusMessage:Page Not Found,message:Page Not Found,stack:}: not found`

anamarcacini in CarregadorBateria2/example on ÷ main [!?

ChatGPT

Parece que houve um erro na instalação do `mqttx-cli-linux-x64` usando o `curl` e ao tentar executar o comando `mqttx --version`.

Ao utilizar o comando `curl`, o URL que você está tentando acessar parece estar incorreto ou indisponível, pois resultou em um código de status 404 (Página não encontrada). Isso indica que o arquivo não foi baixado corretamente.

Para corrigir isso, primeiro, verifique se o URL está correto e se o arquivo que você deseja baixar está disponível. Depois, tente baixar novamente o arquivo `mqttx-cli-linux-x64` com o `curl`. Certifique-se de usar o URL correto e válido.

Após baixar o arquivo corretamente, você pode tentar instalar o `mqttx` novamente usando o comando `sudo install`. Certifique-se de que o caminho do arquivo baixado esteja correto.

Se a instalação for concluída com êxito, tente executar `mqttx --version` novamente para verificar se o `mqttx` está instalado corretamente e pode ser executado no terminal.

3.5