

# Manual básico para o gerenciamento de contêineres Docker utilizando o **Portainer** e o proxy reverso **Nginx Proxy Manager**

---

Adicionar um contêiner .....	2
Adicionar aplicação ao proxy reverso .....	4
Adicionar contêiner via Stack ( <b>Docker Compose</b> ) .....	7
Remover contêiner .....	9
Remover Stack .....	10
Remover apontamento no proxy reverso .....	10
Remover certificado SSL .....	11
Documentação de apoio .....	11

## **Observações:**

- *Este manual faz referência ao uso das ferramentas no ambiente de TI da Universidade Franciscana. Os endereços para acesso são: <https://app.ufn.edu.br> (Portainer) e <https://proxy.app.ufn.edu.br> (Nginx Proxy Manager).*
- *O endereço IP privado do servidor Docker só pode ser acessado a partir da rede interna da instituição.*
- *Os aliases disponibilizados para uso no proxy reverso pertencem ao subdomínio [\\*.app.ufn.edu.br](https://*.app.ufn.edu.br).*

## Adicionar um contêiner

Após realizar o login no Portainer, você será direcionado ao painel principal.

Em **Environments**, selecione o ambiente Docker que deseja gerenciar. Em seguida, no menu lateral, clique na opção **Containers**.

Na página de listagem, você verá todos os contêineres atualmente em execução ou parados.

Para criar um novo contêiner, clique no botão **+ Add container**, localizado no canto superior direito da tela.

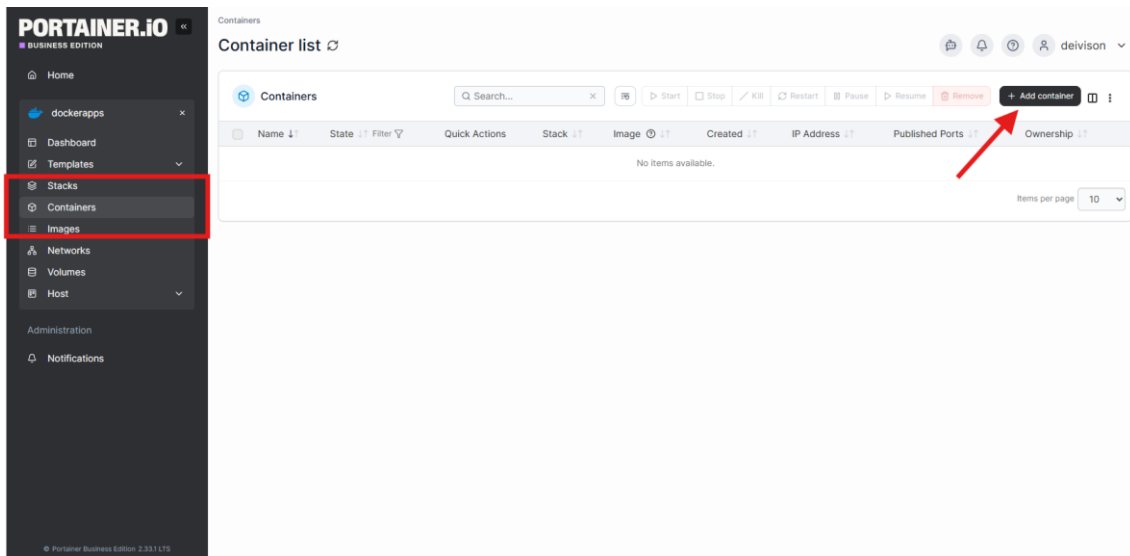


Figura 1

A página **Create container** será exibida, solicitando as informações necessárias para a criação do contêiner:

- **Name:** atribua um nome descritivo. Exemplo: `meu-servidor-web`.
- **Image:** especifique a imagem Docker que será utilizada. Exemplo: `nginx:latest`.

Para tornar o contêiner acessível pela rede, é necessário mapear as portas do contêiner para as portas do servidor:

1. Role a página até a seção **Network ports configuration**.
2. Clique no botão **+ Map additional port**.
3. Em **host**, insira a porta que deseja expor no servidor.
4. Em **container**, informe a porta na qual a aplicação está exposta dentro do contêiner.

Por exemplo: para um contêiner *Nginx*, você pode mapear a porta **8081** do host para a porta **80** do contêiner.

Assim, ao acessar `http://<seu-endereco-de-ip>:8081` no navegador, será estabelecida a conexão com o *Nginx* em execução no contêiner.

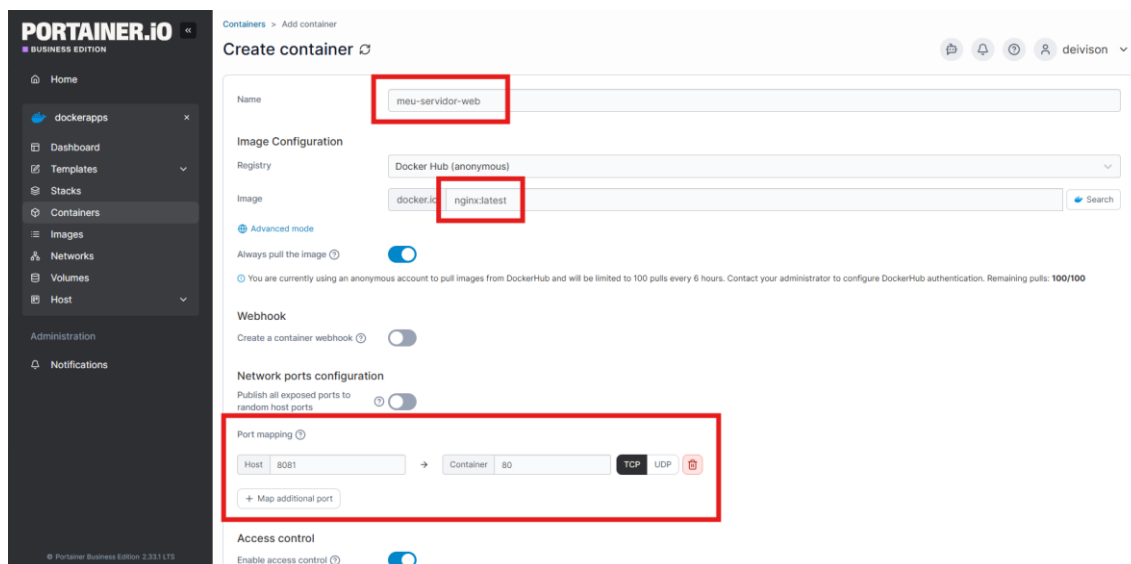


Figura 2

Após preencher as configurações necessárias, vá até a seção **Access control**, selecione a opção **Restricted** (para que todos os membros do time possam gerenciar o contêiner) e clique em **Deploy the container**.

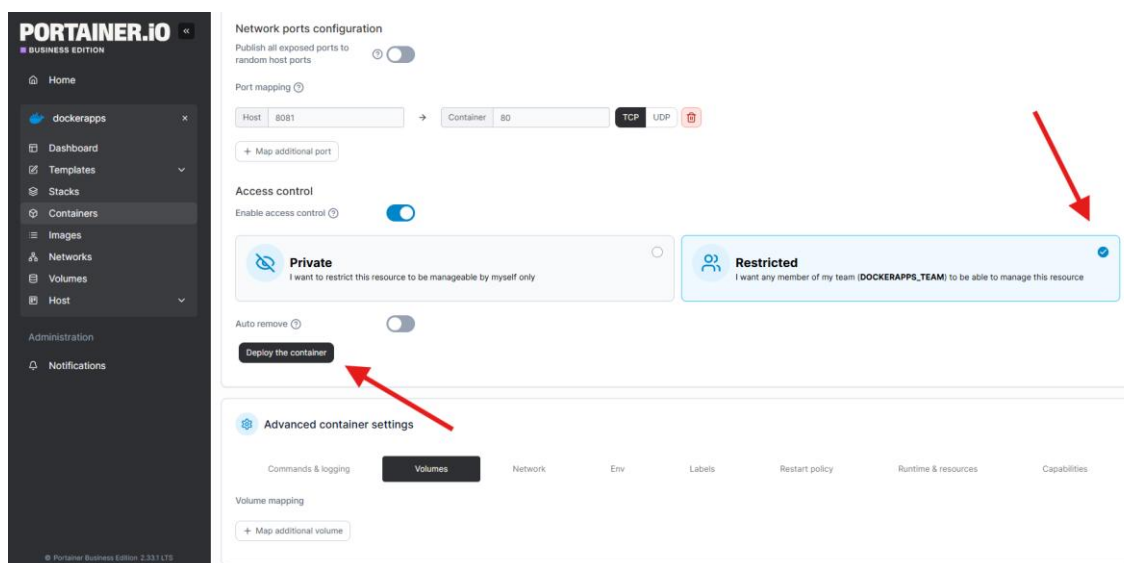


Figura 3

O Portainer fará o download da imagem (caso ainda não esteja presente localmente) e iniciará o novo contêiner com as configurações especificadas.

Após a implantação, você será redirecionado para a lista de contêineres, onde o novo contêiner aparecerá com o status **running**.

É possível clicar sobre o nome do contêiner para visualizar detalhes, acessar logs, estatísticas de uso de recursos e até abrir um console interativo para executar comandos dentro do contêiner.

No exemplo, a aplicação pode ser aberta clicando no link exibido na coluna **Published Ports**.

O servidor web Nginx será acessado pelo endereço privado <http://10.21.19.45:8081> (onde **10.21.19.45** é o IP do servidor e **8081** a porta configurada no mapeamento).

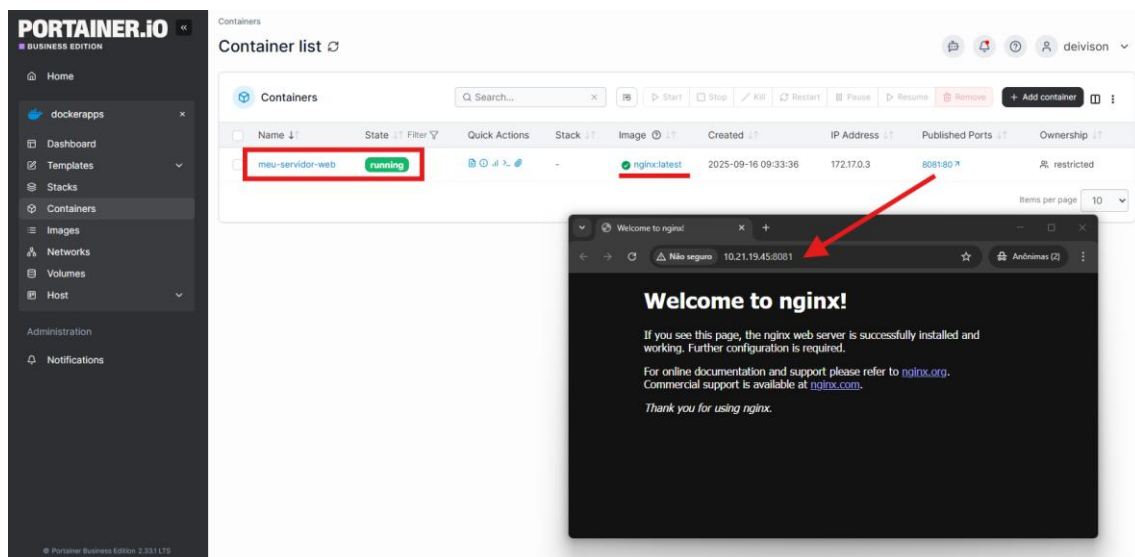


Figura 4

## Adicionar aplicação ao proxy reverso

Com o **Nginx Proxy Manager**, é possível configurar sua aplicação para receber acessos externos, redirecionando um endereço público para um IP interno e protegendo a conexão com um certificado SSL.

1. Na tela inicial do Nginx Proxy Manager, após realizar login:
  - No painel principal, clique em **Hosts** e depois em **Proxy Hosts**.
  - Clique em **Add Proxy Host**. Um formulário com várias seções será exibido.
2. No formulário, preencha as informações conforme necessário:
  - **Domain Names:** Insira o nome de domínio completo que você deseja usar para o seu site (no tutorial utilizamos [exemplo.app.ufn.edu.br](http://exemplo.app.ufn.edu.br)).
  - **Scheme:** Selecione o protocolo utilizado pelo seu serviço interno. Na maioria dos casos será **http**.
  - **Forward Hostname / IP:** Informe o endereço IP do servidor para o qual o tráfego será encaminhado (no tutorial [10.21.19.45](http://10.21.19.45)).
  - **Forward Port:** Insira a porta na qual o serviço está rodando (no tutorial utilizamos [8081](http://8081)).
  - **Cache Assets:** Ative para que o *Nginx* armazene em cache arquivos estáticos como imagens, CSS e JavaScript, melhorando o desempenho do site.
  - **Block Common Exploits:** Recomenda-se ativar para adicionar uma camada básica de proteção contra vulnerabilidades conhecidas.
  - **Websockets Support:** Ative se a aplicação utilizar *WebSockets* (por exemplo, em comunicação em tempo real).

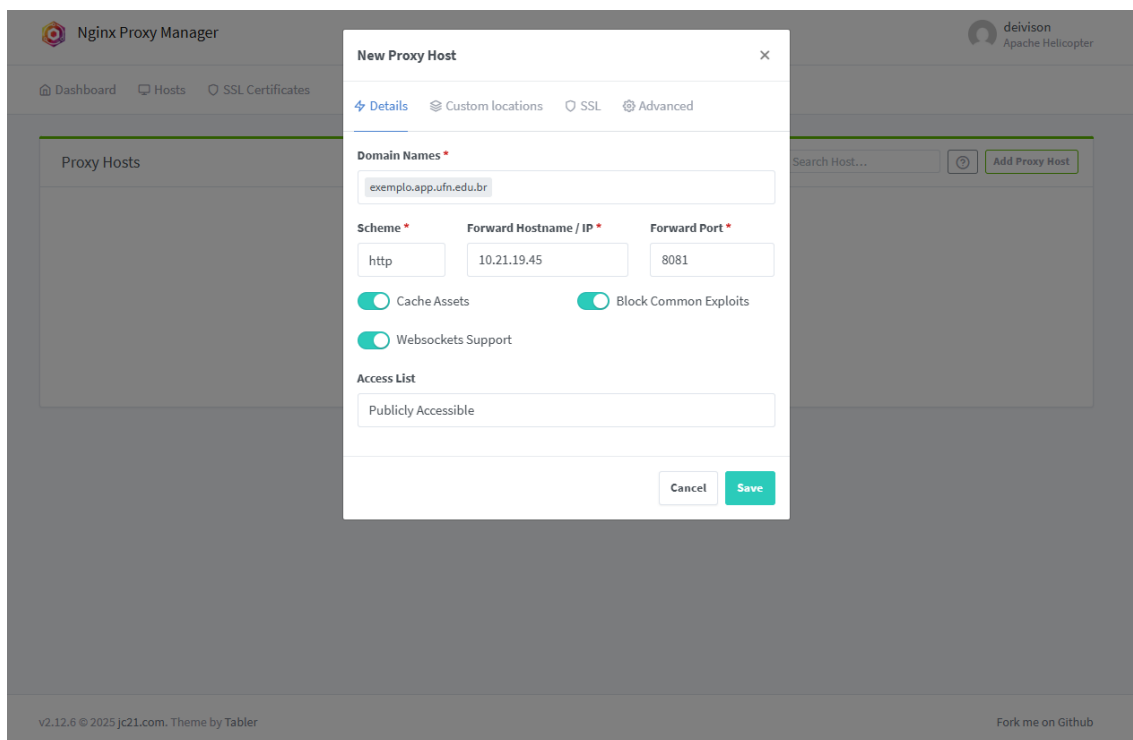


Figura 5

3. Ainda no formulário de criação do host, clique na aba **SSL** para adicionar um certificado via *Let's Encrypt*:
  - **SSL Certificate:** No menu suspenso, escolha **Request a new SSL Certificate**.
  - **Force SSL:** Altamente recomendado. Redireciona automaticamente todo o tráfego HTTP para HTTPS.
  - **HTTP/2 Support:** Ative para melhorias de desempenho em navegadores modernos.
  - **HSTS Enabled (HTTP Strict Transport Security):** Aumenta a segurança ao instruir navegadores a utilizarem apenas HTTPS. Ative somente se tiver certeza de que o site não precisará ser acessado via HTTP.
  - **Email Address for Let's Encrypt:** Informe um e-mail válido para receber notificações sobre os certificados.
  - **I Agree to the Let's Encrypt Terms of Service:** Marque a caixa para aceitar os termos de uso.

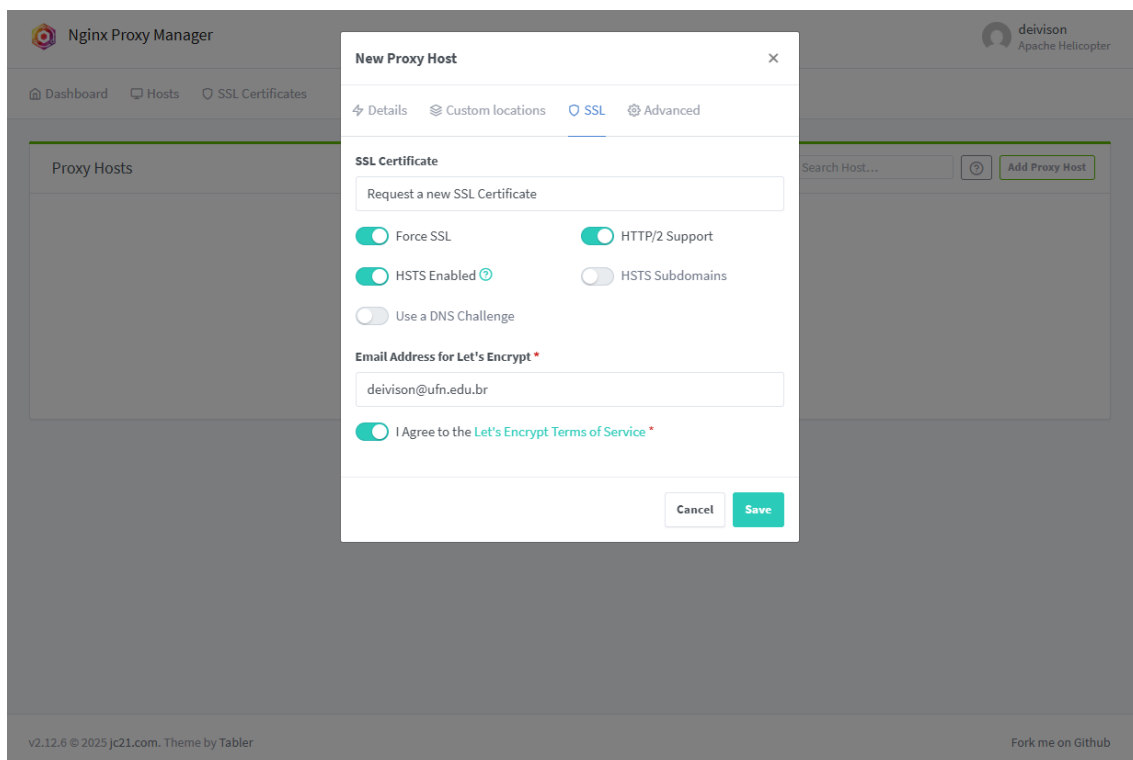


Figura 6

#### 4. Clique em **Save**

O **Nginx Proxy Manager** irá se comunicar com a *Let's Encrypt* para validar a propriedade do domínio e emitir o certificado. Esse processo pode levar de alguns segundos a alguns minutos.

Após a conclusão, seu site estará acessível via **HTTPS** de qualquer lugar, utilizando o endereço publicado e apontando para o IP interno do serviço configurado no **Portainer**, rodando em **Docker** na porta mapeada.

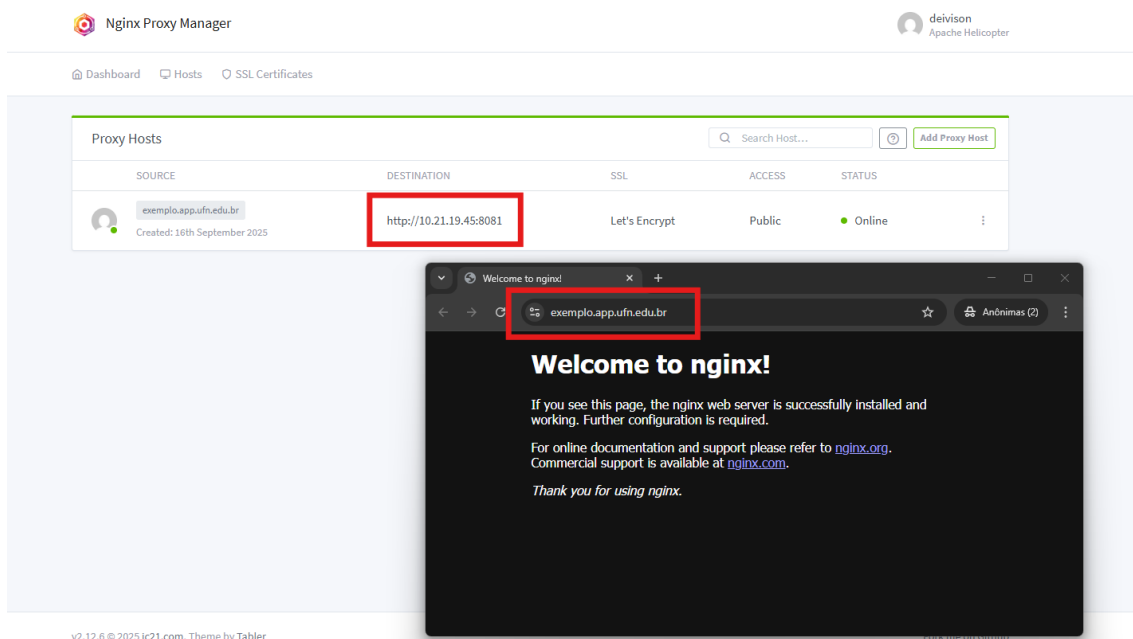


Figura 7

## Adicionar contêiner via Stack (Docker Compose)

Uma *stack* no Portainer é um conjunto de serviços, redes e volumes que compõem uma aplicação definida por um arquivo no formato **Docker Compose**. Ao implantar uma *stack*, o Portainer lê este arquivo e cria (ou atualiza) todos os contêineres e recursos necessários de uma só vez.

Na página **Stacks**, você verá a lista de todas as stacks já implantadas. Para criar uma nova, clique no botão **+ Add stack** no canto superior direito.

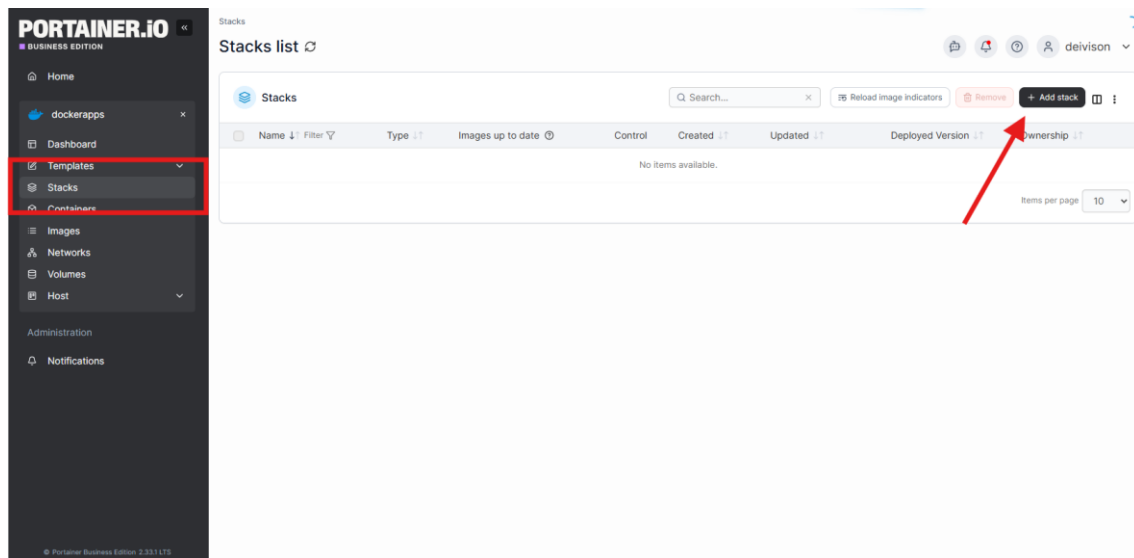


Figura 8

Você será levado à página **Create stack**. Existem várias maneiras de definir o conteúdo da sua stack. Neste tutorial, será utilizado o **Web editor** para criar um exemplo de aplicação WordPress, composta por dois serviços: o próprio **WordPress** e um banco de dados **MySQL**.

Na tela de criação da Stack:

1. **Name:** Defina um nome para a sua stack em letras minúsculas e sem espaços.

Exemplo: `meu-blog-wordpress`

2. **Web editor:** Crie ou copie e cole no editor web o conteúdo do seu arquivo **Docker Compose** em formato YAML. Este código corresponde a um `docker-compose.yml` padrão.

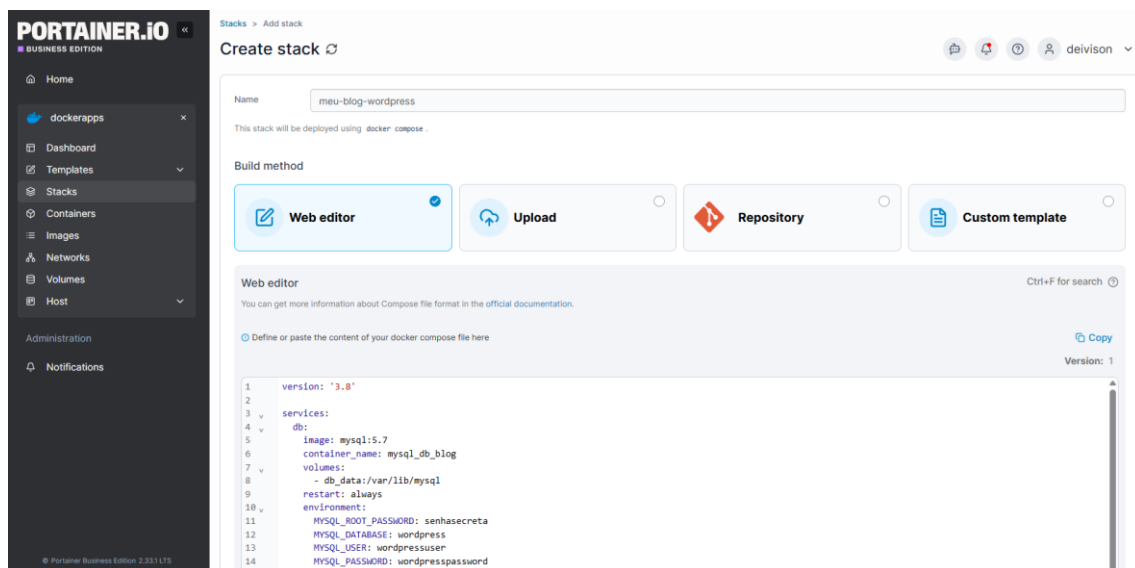


Figura 9

**Opcional:** Na seção **Environment variables**, você pode definir variáveis de ambiente diretamente pela interface do Portainer ou enviar um arquivo **.env** contendo suas variáveis.

3. **Access control:** Selecione a opção **Restricted** para que todos os membros do seu time possam gerenciar a stack. Em seguida, clique em **Deploy the stack**.

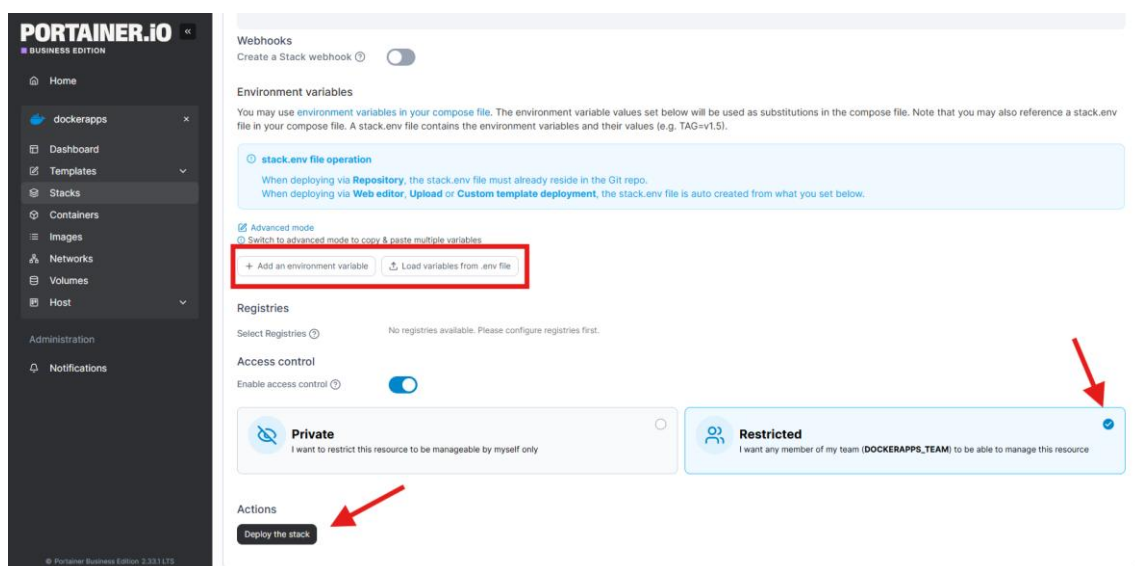


Figura 10

O Portainer irá ler o arquivo **Compose**, baixar as imagens necessárias (o que pode levar alguns minutos na primeira execução) e criar os volumes, a rede e os contêineres definidos.

Após a implantação, você será redirecionado de volta à lista de Stacks, onde sua nova stack (**meu-blog-wordpress**) aparecerá.

Clique no nome da stack para visualizar os detalhes. Você verá os dois serviços (**db** e **wordpress**) listados como contêineres individuais, funcionando em conjunto conforme definido no arquivo **Compose**.



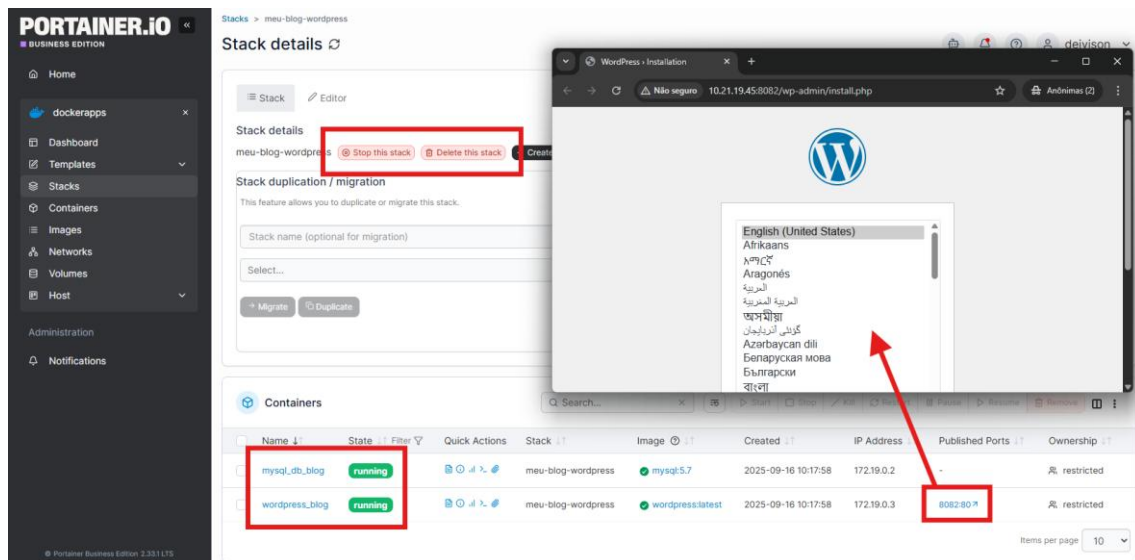


Figura 11

## Remover contêiner

Na tela **Container list**, selecione o contêiner que deseja remover e clique no botão **Remove**.

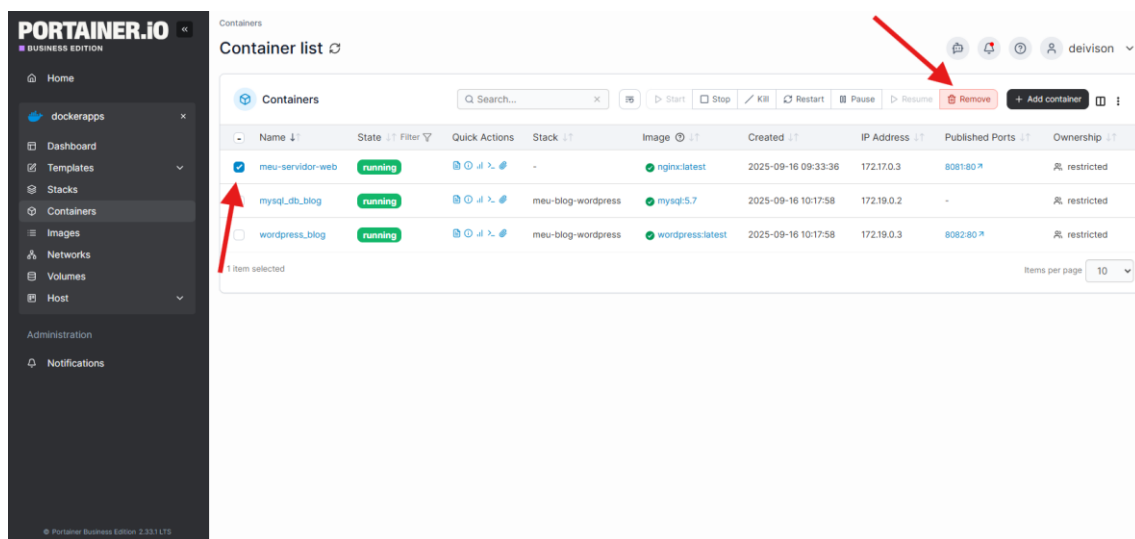


Figura 12

## Remover Stack

Na tela **Stacks list**, selecione a stack que deseja excluir e clique em **Remove**.

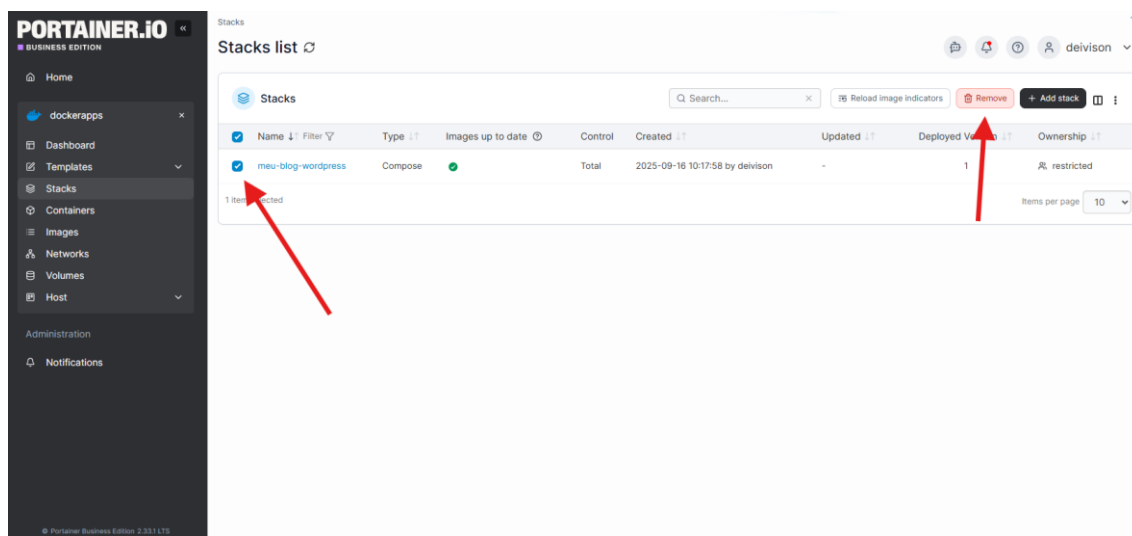


Figura 13

### ATENÇÃO

A remoção de contêineres ou stacks não exclui automaticamente os volumes persistentes, caso tenha. Se desejar remover também esses volumes, repita o processo na tela **Volume list**, dentro do menu **Volumes**.

## Remover apontamento no proxy reverso

Na tela **Proxy Hosts** do Nginx Proxy Manager, clique nos três pontos ao lado do host que deseja remover e, em seguida, clique em **Delete**.

### ATENÇÃO

O certificado SSL associado a esse host não será removido automaticamente. Caso não seja mais necessário, será preciso excluí-lo manualmente seguindo os passos do próximo tópico.

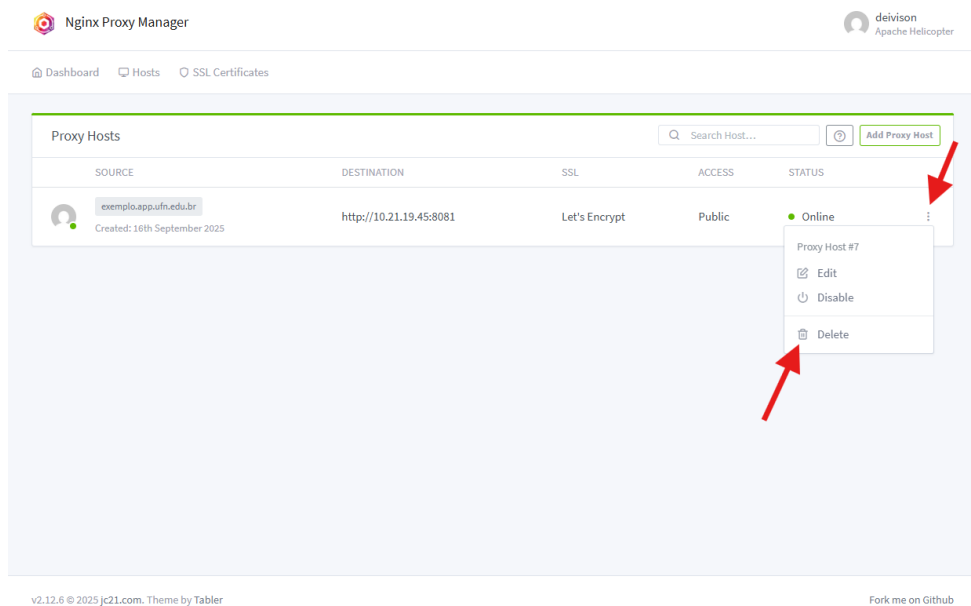


Figura 14

## Remover certificado SSL

No Nginx Proxy Manager, acesse a tela **SSL Certificates**, clique nos três pontos ao lado do certificado que deseja remover e selecione **Delete**.

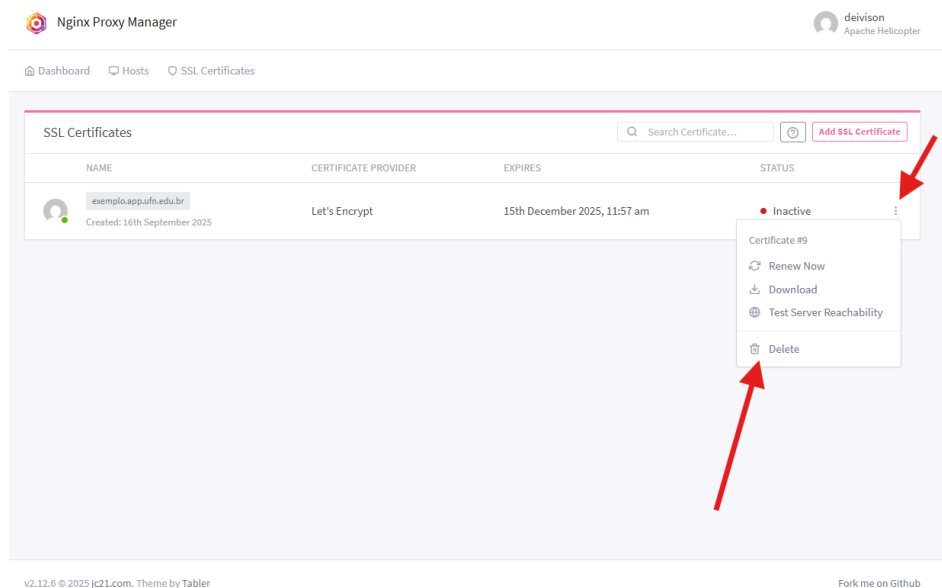


Figura 15

## Documentação de apoio

Para configurações adicionais e funcionalidades avançadas, consulte as documentações oficiais, elas podem auxiliar no processo:

- **Portainer** - <https://docs.portainer.io/>
- **Docker** - <https://docs.docker.com/manuals/>