



GESTÃO DE STACKS NO PORTAINER

Responsável: João Pedro Toledo Gonçalves.

Data: 26/01/2026

MANUAL TÉCNICO - GESTÃO DE STACKS NO PORTAINER

Código: ITGSUP 0009/26 | **Classificação:** RESTRITO

Responsável: João Pedro Toledo Gonçalves | **Data:** 26/01/2026

1. HISTÓRICO DE REVISÃO

NOTA

REGRA DE OURO:

1. **Autor:** João Pedro Toledo Gonçalves.
2. **Descrição:** Criação do documento.

Data	Versão	Descrição	Autor
26/01/2026	1.0	Criação Inicial	João Pedro Toledo Gonçalves

2. OBJETIVO

Padronizar o deploy de aplicações (Stacks) no Portainer utilizando três métodos: Upload Manual, Templates de App e Repositório Git (GitOps Básico).

3. PRÉ-REQUISITOS

- [] Portainer instalado e acessível.
- [] Um arquivo `docker-compose.yml` válido E/OU Uma URL de repositório Git (GitHub/GitLab).

4. PASSO A PASSO (EXECUÇÃO)

Etapa 1: Deploy via Web Editor (Rápido)

1. [x] No menu lateral, clique em **Stacks** -> **Add stack**.
2. [x] Selecione a opção **Web editor**.
3. [x] **Name:** Dê um nome único (ex: `meu-nginx`).
4. [x] Cole o conteúdo do seu `docker-compose.yml` na área de texto.
5. [x] Clique em **Deploy the stack**.

Etapa 2: Deploy via App Templates (Iniciantes)

1. [x] No menu lateral, clique em **App Templates**.
2. [x] Selecione uma aplicação pronta (ex: "Wordpress" ou "Nginx").
3. [x] Preencha os campos (Nome, Senha do Banco).
4. [x] Clique em **Deploy the stack**.

Etapa 3: Deploy via Repositório Git (Recomendado/GitOps)

NOTA

MELHOR PRÁTICA: Permite versionamento e "Single Source of Truth".

1. [x] Vá em **Stacks** -> **Add stack**.
2. [x] Selecione **Git Repository**.
3. [x] **Configuração:**
 - **Repository URL:** <https://github.com/usuario/repo.git>
 - **Repository Reference:** `main` (ou `refs/heads/main`).
 - **Compose path:** `docker-compose.yml` (se estiver na raiz).
4. [x] **Authentication (Se privado):**
 - Ative "Authentication".
 - Use seu Username e, preferencialmente, um **Personal Access Token** (não sua senha).
5. [x] Ative **Automatic updates** (Opcional) para polling.
6. [x] Clique em **Deploy the stack**.

Etapa 4: Edição e Updates

- **Web Editor:** Clique na stack -> Editor -> Update the stack.
- **Git:** Faça o commit no Git -> No Portainer, clique em **"Pull and redeploy"** (ou aguarde o Auto-update).

5. SOLUÇÃO DE PROBLEMAS (TROUBLESHOOTING)

Problema 1: "Deployment error: Stack already exists"

- **Causa:** Já existe uma stack com esse nome (mesmo que parada).
- **Solução:** Delete a stack antiga ou use outro nome.

Problema 2: "Authentication failed" no Git

- **Causa:** Senha incorreta ou uso de senha ao invés de Token (GitHub exige Token).
- **Solução:** Gere um PAT (Personal Access Token) no GitHub (Settings -> Developer Settings) e use como senha.

6. DADOS TÉCNICOS

Campo	Valor	Descrição
Padrão	docker-compose.yml	Formato obrigatório
Limit	Stack	Equivalente a um projeto Docker Compose

7. VALIDAÇÃO FINAL

[] A stack aparece com status "Active"?

- ☐ Os containeres subiram corretamente?
- ☐ Alterar o Git reflete no Portainer (se testado pull)?