# **Configuração do Orion em VPS para Aplicações Dockerizadas**

## **1. Introdução ao Orion Setup e Docker em VPS**

A instalação e configuração de aplicações complexas em um Servidor Privado Virtual (VPS) pode ser um desafio. O script de configuração Orion surge como uma solução para simplificar este processo, especialmente para ferramentas de automação e marketing.1 O Orion Setup é uma solução gratuita que visa auxiliar a comunidade na instalação de ferramentas essenciais do mercado em seus servidores, tornando o processo acessível mesmo para usuários com menos experiência técnica.1 Este relatório detalha o processo de utilização do Orion Setup em conjunto com o Docker para implementar uma aplicação composta por um backend e frontend (rodando com Vite e FastAPI) juntamente com N8N, EvolutionAPI e Redis em um VPS, como o oferecido pela Hostinger ou outros provedores. O Docker, por sua vez, é uma plataforma poderosa para automatizar a implantação de aplicações usando contêineres leves e portáteis.2 A combinação dessas tecnologias permite um ambiente de desenvolvimento e produção consistente e eficiente.

## **2. Pré-requisitos para um VPS**

Antes de iniciar a configuração, é crucial garantir que o VPS atenda aos requisitos necessários. O desenvolvedor do Orion Setup recomenda o uso de VPS de provedores como Hostinger, Hetzner, Digital Ocean ou AWS.1 Embora outros provedores possam funcionar, o suporte e a compatibilidade são mais garantidos com os recomendados. Em termos de especificações do servidor, sugere-se o uso do Ubuntu 20.04 com um mínimo de 4GB de RAM e 2 vCPUs.1 No entanto, é importante verificar os requisitos específicos de cada aplicação que será instalada, pois algumas podem demandar mais recursos.1 Além disso, é fundamental que o servidor esteja vazio antes de executar o script Orion, pois a instalação em um servidor com ferramentas existentes pode levar à perda de dados.1 A Hostinger, por exemplo, oferece templates de VPS com o Docker pré-instalado, o que pode facilitar o processo.4

## **3. Instalação do Docker e Docker Compose**

Para executar aplicações em contêineres, o Docker precisa estar instalado no VPS. A Hostinger oferece templates de VPS com Ubuntu e Docker já instalados, simplificando o processo.4 Caso o seu VPS não possua o Docker pré-instalado, é possível instalá-lo manualmente seguindo as instruções fornecidas pela Hostinger.2 O processo geralmente envolve a atualização dos pacotes do sistema, a adição da chave GPG oficial do Docker, a instalação dos pacotes necessários (como docker-ce, docker-ce-cli e containerd.io) e a verificação da instalação através do comando docker --version.2

Além do Docker, o Docker Compose é uma ferramenta essencial para definir e gerenciar aplicações multi-contêiner.4 Ele permite definir todos os serviços da sua aplicação em um único arquivo docker-compose.yml. A Hostinger também oferece o Docker Compose pré-instalado em seus templates de VPS com Docker.4 Caso precise instalar manualmente, as instruções podem ser encontradas na documentação oficial do Docker Compose. Geralmente, envolve o download do binário e a configuração das permissões de execução.3

## **4. Criação de Dockerfiles (Opcional)**

Se você possui um backend e frontend personalizados, construídos com Vite e FastAPI, respectivamente, precisará criar Dockerfiles para cada um deles. Um Dockerfile é um arquivo de texto que contém todas as instruções para construir uma imagem Docker.6 Para o backend FastAPI, o Dockerfile incluirá a imagem base do Python, a instalação das dependências (listadas em um arquivo requirements.txt) e o comando para executar a aplicação. Da mesma forma, para o frontend Vite, o Dockerfile usará uma imagem base do Node.js, copiará os arquivos do projeto, instalará as dependências e definirá o comando para iniciar o servidor de desenvolvimento ou construir a versão de produção. Se você já possui imagens Docker para seu backend e frontend hospedadas em um registro de contêineres como o Docker Hub, esta etapa pode ser pulada, utilizando as imagens diretamente no arquivo docker-compose.yml.

## **5. Elaboração do Arquivo docker-compose.yml**

O arquivo docker-compose.yml é o coração da sua configuração Docker, pois define todos os serviços que compõem sua aplicação multi-contêiner, incluindo o backend, frontend, N8N, EvolutionAPI e Redis.4 Abaixo, um exemplo de como este arquivo pode ser estruturado:

YAML

version: '3.8'  
services:  
 backend:  
 image: seu-dockerhub/seu-backend-image:latest # Substitua pela sua imagem ou Dockerfile  
 container\_name: backend  
 ports:  
 - "8000:8000" # Mapeia a porta 8000 do host para a porta 8000 do contêiner  
 environment:  
 REDIS\_HOST: redis  
 REDIS\_PORT: 6379  
 networks:  
 - my-app-network  
 depends\_on:  
 - redis  
  
 frontend:  
 image: seu-dockerhub/seu-frontend-image:latest # Substitua pela sua imagem ou Dockerfile  
 container\_name: frontend  
 ports:  
 - "3000:80" # Assumindo que o frontend serve na porta 80 dentro do contêiner  
 networks:  
 - my-app-network  
 # depends\_on:  
 # - backend  
  
 n8n:  
 image: n8nio/n8n  
 container\_name: n8n  
 ports:  
 - "5678:5678"  
 environment:  
 N8N\_HOST: "n8n"  
 N8N\_PORT: "5678"  
 # Outras variáveis de ambiente do N8N  
 volumes:  
 - n8n\_data:/home/node/.n8n  
 networks:  
 - my-app-network  
  
 evolutionapi:  
 image: atendai/evolution-api  
 container\_name: evolutionapi  
 ports:  
 - "8080:8080"  
 environment:  
 AUTHENTICATION\_API\_KEY: sua-chave-evolution-api  
 # Outras variáveis de ambiente do EvolutionAPI  
 volumes:  
 - evolutionapi\_data:/evolution/instances  
 networks:  
 - my-app-network  
  
 redis:  
 image: redis:latest  
 container\_name: redis  
 ports:  
 - "6379:6379"  
 volumes:  
 - redis\_data:/data  
 networks:  
 - my-app-network  
  
networks:  
 my-app-network:  
 driver: bridge  
  
volumes:  
 redis\_data:  
 n8n\_data:  
 evolutionapi\_data:

A utilização de um arquivo docker-compose.yml é a abordagem recomendada para gerenciar aplicações Docker multi-contêiner em um VPS.4 Ele simplifica a implantação, o escalonamento e o networking entre os contêineres, permitindo definir toda a pilha de aplicações de forma declarativa. Definir uma rede Docker dedicada é crucial para habilitar a comunicação contínua entre os contêineres usando seus nomes de serviço como hostnames, evitando a necessidade de expor todas as portas ao host e aumentando a segurança.9 Além disso, configurar corretamente os volumes é essencial para o armazenamento persistente de dados, especialmente para Redis, N8N e EvolutionAPI, garantindo que os dados não sejam perdidos quando os contêineres forem interrompidos ou reiniciados.18 A diretiva depends\_on no Docker Compose garante que os serviços sejam iniciados na ordem correta, o que é importante se um serviço depender de outro (por exemplo, o backend pode precisar que o Redis esteja em execução).9

**Exemplo de Serviços docker-compose.yml:**

| **Nome do Serviço** | **Imagem** | **Nome do Contêiner** | **Portas** | **Variáveis de Ambiente** | **Redes** | **Volumes** | **Depende De** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| backend | seu-dockerhub/seu-backend-image:latest | backend | 8000:8000 | REDIS\_HOST, REDIS\_PORT | my-app-network | - | redis |
| frontend | seu-dockerhub/seu-frontend-image:latest | frontend | 3000:80 | - | my-app-network | - | - |
| n8n | n8nio/n8n | n8n | 5678:5678 | N8N\_HOST, N8N\_PORT,... | my-app-network | n8n\_data:/home/node/.n8n | - |
| evolutionapi | atendai/evolution-api | evolutionapi | 8080:8080 | AUTHENTICATION\_API\_KEY,... | my-app-network | evolutionapi\_data:/evolution/instances | - |
| redis | redis:latest | redis | 6379:6379 | - | my-app-network | redis\_data:/data | - |

## **6. Implantação da Sua Pilha de Aplicações com Docker Compose**

Para implantar sua aplicação, navegue até o diretório no seu VPS onde você salvou o arquivo docker-compose.yml. Em seguida, execute o seguinte comando para iniciar todos os serviços definidos no arquivo 4:

Bash

docker-compose up -d

O flag -d executa os contêineres em modo detached (em segundo plano). Você pode monitorar o status dos contêineres em execução com o comando docker ps, que mostrará os nomes dos contêineres, as imagens, o status e os mapeamentos de portas. Para visualizar os logs de um contêiner específico, use o comando docker logs <nome\_do\_contêiner>. Isso é crucial para solucionar problemas de inicialização ou erros de tempo de execução. Problemas comuns, como conflitos de portas (se uma porta do host já estiver em uso), podem ser resolvidos alterando a porta do host no arquivo docker-compose.yml. O comando docker-compose up -d simplifica a implantação de toda a pilha de aplicações com um único comando, tornando-o muito mais eficiente do que iniciar cada contêiner individualmente, pois orquestra a criação e inicialização de todos os serviços definidos. Monitorar o status e os logs dos contêineres é essencial para verificar a implantação bem-sucedida e identificar quaisquer problemas potenciais. A detecção precoce de erros pode economizar um tempo significativo de depuração posteriormente, pois os logs fornecem informações valiosas sobre o comportamento da aplicação.

## **7. Acessando Suas Aplicações no VPS**

Após a implantação, você poderá acessar suas aplicações usando o endereço IP do VPS e as portas do host mapeadas no arquivo docker-compose.yml. Por exemplo:

* Frontend: http://<seu\_ip\_do\_vps>:3000 (se a porta 3000 foi mapeada para o host).
* Backend: http://<seu\_ip\_do\_vps>:8000 (se a porta 8000 foi mapeada para o host).
* N8N: http://<seu\_ip\_do\_vps>:5678.
* EvolutionAPI: http://<seu\_ip\_do\_vps>:8080.
* Redis: Geralmente não é acessado diretamente via interface web, a menos que se utilize uma ferramenta como o RedisInsight.

Acessar múltiplos serviços em diferentes números de porta pode ser limitante, especialmente para usuários finais. Uma solução para gerenciar o acesso a múltiplos serviços em portas HTTP (porta 80) e HTTPS (porta 443) padrão, utilizando subdomínios ou caminhos, é a implementação de um **proxy reverso** (como Nginx ou Traefik).20 O Orion Setup oferece a possibilidade de instalar o Traefik 1, o que pode simplificar a configuração de um proxy reverso, pois o Traefik é projetado para funcionar bem com o Docker e pode descobrir e configurar rotas para contêineres automaticamente. Para utilizar um proxy reverso com roteamento baseado em domínio, é necessário configurar os registros DNS para apontar os subdomínios para o endereço IP do VPS.

## **8. Utilizando o Orion Setup para Ferramentas Adicionais (N8N, EvolutionAPI)**

Se você ainda não construiu seus próprios contêineres Docker para N8N e EvolutionAPI ou prefere uma instalação mais automatizada, pode utilizar o script de configuração Orion para instalá-los. No entanto, é crucial lembrar que o Orion Setup é recomendado para ser executado em um servidor vazio.1 Se você já configurou o Docker e outros serviços, executar o Orion pode interferir na sua configuração existente.

Para executar o script Orion no seu VPS, utilize o seguinte comando 1:

Bash

bash <(curl -sSL setup.oriondesign.art.br)

O script provavelmente apresentará um menu onde você poderá escolher quais ferramentas instalar.1 Selecione N8N e EvolutionAPI da lista. Siga as instruções e forneça as informações necessárias durante o processo de instalação. Uma vez instaladas via Orion, essas ferramentas podem ser executadas em seus próprios contêineres Docker (dependendo da implementação do script Orion). Você precisará garantir que esses contêineres estejam na mesma rede Docker (my-app-network definida no seu arquivo docker-compose.yml) para se comunicarem com seu backend e frontend. Isso pode exigir ajustes na configuração de rede dos contêineres instalados pelo Orion ou a inclusão deles no arquivo docker-compose.yml. Misturar a configuração manual do Docker Compose com um instalador baseado em script pode apresentar complexidades.

**Ferramentas Disponíveis no Script Orion Setup:**

| **Ferramenta** | **Descrição** |
| --- | --- |
| Traefik | Um proxy reverso HTTP e balanceador de carga moderno. |
| Portainer | Uma interface web para gerenciar ambientes Docker. |
| Chatwoot | Uma plataforma de engajamento com o cliente de código aberto. |
| Evolution API | Uma API para automação do WhatsApp. |
| MinIO | Um serviço de armazenamento de objetos de alto desempenho. |
| Typebot | Uma ferramenta para criar chatbots interativos. |
| N8N | Uma plataforma de automação de fluxo de trabalho sem código. |
| Flowise | Uma ferramenta para construir fluxos de trabalho de IA visualmente. |
| PgAdmin 4 | Uma ferramenta de administração e desenvolvimento para PostgreSQL. |
| Nocobase | Uma plataforma de banco de dados sem código de código aberto. |
| Botpress | Uma plataforma de construção de chatbots de código aberto. |
| Wordpress | Um popular sistema de gerenciamento de conteúdo (CMS). |
| Baserow | Uma alternativa de código aberto ao Airtable. |
| MongoDB | Um banco de dados NoSQL orientado a documentos. |
| RabbitMQ | Um broker de mensagens de código aberto. |
| Uptime Kuma | Uma ferramenta de monitoramento de tempo de atividade de código aberto. |
| Cal.com | Uma alternativa de código aberto ao Calendly. |
| Mautic | Uma plataforma de automação de marketing de código aberto. |
| Appsmith | Uma plataforma de código baixo para construir aplicações web internas. |
| Qdrant | Um mecanismo de busca vetorial. |
| Woofed CRM | Um sistema de gerenciamento de relacionamento com o cliente (CRM). |
| Formbricks | Uma ferramenta para criar formulários e pesquisas. |
| NocoDB | Uma plataforma que transforma qualquer banco de dados MySQL, PostgreSQL, SQLite, Microsoft SQL Server ou ClickHouse em uma planilha inteligente. |
| Langfuse | Uma plataforma de observabilidade para aplicações de linguagem. |
| Metabase | Uma ferramenta de inteligência de negócios de código aberto. |
| Odoo | Um conjunto de aplicações de negócios de código aberto. |
| Chatwoot Nestor | Uma extensão para o Chatwoot para fornecer respostas automatizadas. |
| Uno API | Uma API para gerar documentos. |
| N8N + Nodes Quepasa | Uma extensão do N8N com nós adicionais. |
| Quepasa API | Uma API para funcionalidades específicas. |
| Docuseal | Uma plataforma para criar e gerenciar documentos eletrônicos. |
| Grafana | Uma plataforma de visualização e análise de dados de código aberto. |
| Prometheus | Um sistema de monitoramento e alerta de código aberto. |
| cAdvisor | Um analisador de recursos de contêiner. |
| Dify AI | Uma plataforma para construir aplicações de IA. |
| Ollama | Uma ferramenta para executar modelos de linguagem localmente. |
| Affine | Uma ferramenta de colaboração e gerenciamento de projetos. |
| Directus | Um sistema de gerenciamento de conteúdo (CMS) de código aberto e sem cabeça. |
| VaultWarden | Uma alternativa de código aberto ao Bitwarden para gerenciamento de senhas. |
| NextCloud | Uma plataforma de código aberto para armazenamento em nuvem e colaboração. |
| Strapi | Um sistema de gerenciamento de conteúdo (CMS) de código aberto e sem cabeça. |
| PhpMyAdmin | Uma ferramenta para administrar bancos de dados MySQL via web. |
| Supabase | Uma alternativa de código aberto ao Firebase. |
| Ntfy | Um serviço de notificação simples. |
| LowCoder | Uma plataforma de desenvolvimento de código baixo. |
| LangFlow | Uma interface visual para construir aplicações de linguagem. |
| OpenProject | Um software de gerenciamento de projetos de código aberto. |
| ZEP | Uma plataforma para construir aplicações de conhecimento. |
| HumHub | Uma plataforma de rede social de código aberto. |
| Yourls | Uma ferramenta de código aberto para encurtar URLs. |
| TwentyCRM | Um sistema de gerenciamento de relacionamento com o cliente (CRM) de código aberto. |
| Mattermost | Uma plataforma de mensagens de equipe de código aberto. |
| Outline | Uma ferramenta de documentação de código aberto. |
| Focalboard | Uma alternativa de código aberto ao Trello para gerenciamento de projetos. |
| GLPI | Um software de gerenciamento de ativos de TI de código aberto. |
| Anything LLM | Uma ferramenta para interagir com modelos de linguagem grandes. |
| Excalidraw | Uma ferramenta de desenho colaborativo virtual de código aberto. |
| Easy!Apointments | Um sistema de agendamento online. |
| Documeso | Uma plataforma para criar e gerenciar documentos. |
| Moodle | Um sistema de gerenciamento de aprendizado (LMS) de código aberto. |
| ToolJet | Uma plataforma de código baixo para construir aplicações internas. |
| Stirling PDF | Uma ferramenta para manipular arquivos PDF. |
| ClickHouse | Um sistema de gerenciamento de banco de dados de código aberto orientado a colunas. |
| RedisInsight | Uma GUI para gerenciar e monitorar bancos de dados Redis. |
| Traccar | Um sistema de rastreamento GPS de código aberto. |
| Firecrawl | Uma ferramenta para rastrear e monitorar websites. |
| Wuzapi | Uma API para funcionalidades específicas. |
| krayin CRM | Um sistema de gerenciamento de relacionamento com o cliente (CRM) de código aberto. |

## **9. Protegendo Seu Ambiente Docker no VPS**

A segurança do VPS e do ambiente Docker é de suma importância, pois estarão acessíveis publicamente. Recomenda-se configurar um firewall (por exemplo, ufw no Ubuntu) para permitir apenas as portas necessárias.7 Isso deve incluir portas para SSH, HTTP (80), HTTPS (443 se estiver usando um proxy reverso) e quaisquer outras portas explicitamente necessárias. É aconselhável usar senhas fortes e exclusivas ou, preferencialmente, autenticação baseada em chave SSH para acessar o VPS.7 Manter o sistema operacional e todos os softwares instalados (incluindo o Docker) atualizados com os patches de segurança mais recentes é fundamental. Se estiver expondo portas diretamente (sem um proxy reverso), esteja ciente das implicações de segurança de cada serviço exposto. Considere usar uma ferramenta como o Portainer (que também pode ser instalado via Orion ou Docker) para gerenciar contêineres Docker através de uma interface web, mas proteja o acesso a ele. Se estiver usando um proxy reverso, certifique-se de que ele esteja configurado de forma segura, incluindo a configuração de certificados SSL (por exemplo, usando o Let's Encrypt) para acesso HTTPS.21

## **10. Solução de Problemas Comuns de Implantação Docker**

Ao implantar aplicações Docker, alguns problemas comuns podem surgir. Falhas na inicialização do contêiner podem ser investigadas verificando os logs do contêiner com o comando docker logs <nome\_do\_contêiner>. Causas comuns incluem configuração incorreta, variáveis de ambiente ausentes ou conflitos de portas. Problemas de conectividade de rede podem ser resolvidos garantindo que todos os contêineres que precisam se comunicar estejam na mesma rede Docker. Use docker inspect <nome\_do\_contêiner> para verificar as configurações de rede e certifique-se de que firewalls no host ou dentro dos contêineres não estejam bloqueando o tráfego necessário. Conflitos de portas podem ser resolvidos alterando o mapeamento de portas do host no arquivo docker-compose.yml. Problemas com a montagem de volumes podem ser verificados garantindo que os caminhos para os volumes estejam especificados corretamente no arquivo docker-compose.yml e que os diretórios do host existam com as permissões corretas. Erros ao puxar imagens podem ser causados por nomes ou tags de imagem incorretos no arquivo docker-compose.yml ou por falta de acesso do daemon Docker ao registro. Limitações de recursos no VPS podem levar a falhas ou lentidão dos contêineres; monitore o uso de recursos com docker stats ou htop e considere fazer um upgrade do plano do VPS se necessário.

## **11. Conclusão e Próximos Passos**

A implantação de uma aplicação multi-contêiner em um VPS utilizando Docker Compose, possivelmente com o auxílio do script Orion Setup para N8N e EvolutionAPI, envolve diversas etapas cruciais. Desde a preparação do VPS e a criação dos Dockerfiles até a elaboração do arquivo docker-compose.yml, a implantação da pilha, o acesso às aplicações e as considerações de segurança, cada fase contribui para um ambiente operacional eficiente e seguro.

Como próximos passos, o usuário pode considerar registrar um nome de domínio e apontá-lo para o endereço IP do VPS, configurar certificados SSL para acesso HTTPS seguro utilizando Let's Encrypt com um proxy reverso, implementar um pipeline de CI/CD para automatizar a construção e implantação de futuras atualizações, configurar soluções de monitoramento e logging para melhor visibilidade da saúde e desempenho das aplicações, e explorar ferramentas de orquestração de contêineres mais avançadas como Docker Swarm ou Kubernetes para escalabilidade e gerenciamento mais robustos caso a aplicação cresça significativamente. A jornada de implantação de uma aplicação multi-contêiner em um VPS é contínua e a adoção de práticas recomendadas garante um ambiente de produção robusto e escalável.

#### Referências citadas

1. oriondesign2015/SetupOrion: Dito como o maior e melhor ... - GitHub, acessado em maio 5, 2025, <https://github.com/oriondesign2015/setuporion>
2. How to Install Docker on Ubuntu Using VPS Template and Manually - Hostinger, acessado em maio 5, 2025, <https://www.hostinger.com/tutorials/how-to-install-docker-on-ubuntu>
3. Install Docker On a VPS - Simple Step By Step Tutorial - SSD Nodes, acessado em maio 5, 2025, <https://www.ssdnodes.com/blog/getting-started-docker-vps/>
4. How to use the Docker VPS template - Hostinger Help Center, acessado em maio 5, 2025, <https://support.hostinger.com/en/articles/8306612-how-to-use-the-docker-vps-template>
5. Docker VPS Hosting: Deploy Complex Apps in a Simple Way - Hostinger, acessado em maio 5, 2025, <https://www.hostinger.com/vps/docker-hosting>
6. How to Create a Docker Container in 2025 - Hostinger, acessado em maio 5, 2025, <https://www.hostinger.com/tutorials/docker-start-a-container>
7. Best practices for deploying Docker on a VPS? - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/docker/comments/1ac9w16/best_practices_for_deploying_docker_on_a_vps/>
8. Docker Compose - n8n Docs, acessado em maio 5, 2025, <https://docs.n8n.io/hosting/installation/server-setups/docker-compose/>
9. Multi-container applications - Docker Docs, acessado em maio 5, 2025, <https://docs.docker.com/get-started/docker-concepts/running-containers/multi-container-applications/>
10. Docker - Prefect Docs, acessado em maio 5, 2025, <https://orion-docs.prefect.io/2.20.1/guides/docker/>
11. Prefect Orion Docker Integration Guide - Restack, acessado em maio 5, 2025, <https://www.restack.io/docs/prefect-knowledge-prefect-orion-docker-integration>
12. Networking in Compose - Docker Docs, acessado em maio 5, 2025, <https://docs.docker.com/compose/how-tos/networking/>
13. How to create a network of containers that can communicate with each other interchangably? - Docker Desktop, acessado em maio 5, 2025, <https://forums.docker.com/t/how-to-create-a-network-of-containers-that-can-communicate-with-each-other-interchangably/134292>
14. How do I communicate between two docker containers - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/docker/comments/1c8htvj/how_do_i_communicate_between_two_docker_containers/>
15. Communication Between Multiple Docker Compose Projects | Baeldung on Ops, acessado em maio 5, 2025, <https://www.baeldung.com/ops/docker-compose-communication>
16. docker-compose make requests between containers - Stack Overflow, acessado em maio 5, 2025, <https://stackoverflow.com/questions/52010778/docker-compose-make-requests-between-containers>
17. Selfhosting with docker on VPS with reverse proxy and docker networks. #7351 - GitHub, acessado em maio 5, 2025, <https://github.com/Budibase/budibase/discussions/7351>
18. How to Use the Redis Docker Official Image, acessado em maio 5, 2025, <https://www.docker.com/blog/how-to-use-the-redis-docker-official-image/>
19. Using Redis with docker and docker-compose for local development a step-by-step tutorial, acessado em maio 5, 2025, <https://geshan.com.np/blog/2022/01/redis-docker/>
20. New to Docker - what's the absolute easiest way to host my Docker on a VPS? - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/docker/comments/1cvj82k/new_to_docker_whats_the_absolute_easiest_way_to/>
21. How to Set up Nginx as Reverse Proxy for Docker Container - UltaHost, acessado em maio 5, 2025, <https://ultahost.com/knowledge-base/setup-nginx-reverse-proxy-docker/>
22. Advice for Reverse Proxy/VPN on a VPS : r/selfhosted - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/selfhosted/comments/1htwif5/advice_for_reverse_proxyvpn_on_a_vps/>
23. Setting Up a VPS Server with Docker, Nginx Proxy Manager, and Portainer, acessado em maio 5, 2025, <https://dev.to/imzihad21/setting-up-a-vps-server-with-docker-nginx-proxy-manager-and-portainer-3hfk>
24. naskio/auto-reverse-proxy - GitHub, acessado em maio 5, 2025, <https://github.com/naskio/auto-reverse-proxy>
25. Need advice for docker VPS infrastructure with reverse proxy (new to this) - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/selfhosted/comments/16gklon/need_advice_for_docker_vps_infrastructure_with/>
26. How to Nginx Reverse Proxy with Docker Compose - Gcore, acessado em maio 5, 2025, <https://gcore.com/learning/reverse-proxy-with-docker-compose/>
27. Install Traefik for Docker Reverse Proxy: Break Free from Nginx - YouTube, acessado em maio 5, 2025, <https://www.youtube.com/watch?v=vce3EEkvuZ4>
28. Deploy with Docker on a VPS with reverse proxy? - Need Help - Wappler Community, acessado em maio 5, 2025, <https://community.wappler.io/t/deploy-with-docker-on-a-vps-with-reverse-proxy/49485>
29. Help, Best way to set up docker and firewall on vps? - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/docker/comments/zg1ww0/help_best_way_to_set_up_docker_and_firewall_on_vps/>
30. Securing a VPS running docker : r/selfhosted - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/selfhosted/comments/1239wnj/securing_a_vps_running_docker/>
31. Security PSA for anyone using Docker on a publicly accessible host. You may be exposing ports you're not aware of… - Reddit, acessado em maio 5, 2025, <https://www.reddit.com/r/selfhosted/comments/1cv2l3q/security_psa_for_anyone_using_docker_on_a/>