BookStack on Hostinger.Com KVM VPS

The Beginner’s Guide to BookStack on a Hostinger VPS is your step-by-step playbook for building a modern, self-hosted knowledge base from scratch.

The Beginner’s Guide

Welcome to BookStack at Hostinger.Com KVM VPS: The Beginner’s Guide.

The Beginner’s Guide

Table Of Contents

This book was written for newcomers who want to take a raw VPS and turn it into a **fully running knowledge base**. You don’t need to be a Docker guru or a Linux wizard — every command and config you’ll see here was tested on a **Hostinger KVM 2 VPS running Ubuntu 24**. The goal of this guide is not just to show you *what* to type, but also to explain *why* we type it, so you walk away with real confidence in managing your own stack.

You can read it straight through as a tutorial, or jump directly to the troubleshooting or update chapters when you need quick answers.
3. **Server Setup** – Updating Ubuntu, installing Docker, and securing your VPS.
Table Of Contents
Chapters
Tip: Treat this book like a reference manual.

Introduction

BookStack is an open-source wiki & documentation platform, designed to be simple and selfhosted.

Think of it like your personal **company knowledge base** or **digital bookshelf**.

Deployment

BookStack is an open-source wiki & documentation platform, designed to be simple and selfhosted. Think of it like your personal **company knowledge base** or **digital bookshelf**.
In this guide, we'll walk through deploying BookStack on:
- **Hostinger KVM 2 VPS** (upgradeable to KVM 4 if needed)
- **Ubuntu 24 LTS**
- **Docker & Docker Compose**
- **Traefik** reverse proxy for HTTPS
- **Domain name:** `books.aimatrix.one`
- **Email account:** `books@aimatrix.one`

You'll learn not just the "how" but also the "why" behind each step, so you can troubleshoot and grow your setup confidently.

Prerequisites

Before we begin, make sure you have

Prerequisites

Resourses

Before we begin, make sure you have:
A Hostinger VPS
- KVM 2 (2GB RAM, 1 CPU, 40GB storage) is enough for testing.
- KVM 4+ is recommended for production use.
Operating System
- Ubuntu Server 24.04 LTS installed.
Domain
- Example: `books.aimatrix.one` pointing to your VPS IP.
Email Account
- Example: `books@aimatrix.one`
- With SMTP access (username, password, host, port).
Basic Linux knowledge
- Comfort with running commands in terminal.

Server Setup

Server Setup

Install Essentials

```
sudo apt install curl git ufw -y
```

Server Setup

Install Docker & Compose

```
# Add Docker repo
sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | \
sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] \
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" \
| sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update
sudo apt install docker-ce docker-ce-cli containerd.io docker-compose-plugin -y
```

Server Setup

Allow your user to run Docker without sudo

```
sudo usermod -aG docker $USER
newgrp docker
```

Server Setup

Enable firewall (UFW)

```
sudo ufw allow OpenSSH
sudo ufw allow 80,443/tcp
sudo ufw enable
```

Traefik & Let’s Encrypt

Traefik & Let’s Encrypt

Traefik Reverse Proxy

Traefik is a reverse proxy that:

Routes requests from your domain to the right container.
Terminates SSL using **Let’s Encrypt**.
Handles automatic certificate renewals.

This means you don’t have to touch nginx config files or manually manage certs.

Traefik docker-compose.yml

Create a folder:

```
mkdir -p ~/traefik
cd ~/traefik
```

Create Traefik File

docker-compose.yml

```
networks:
proxy:
external: false
volumes:
traefik_letsencrypt: {}
services:
traefik:
image: traefik:v3.1
container_name: traefik
command:
- --providers.docker=true
- --providers.docker.exposedbydefault=false
- --entrypoints.web.address=:80
- --entrypoints.websecure.address=:443
- --entrypoints.web.http.redirections.entryPoint.to=websecure
- --entrypoints.web.http.redirections.entryPoint.scheme=https
- --certificatesresolvers.letsencrypt.acme.tlsChallenge=true
- --certificatesresolvers.letsencrypt.acme.email=books@aimatrix.one
- --certificatesresolvers.letsencrypt.acme.storage=/letsencrypt/acme.json
ports:
- "80:80"
- "443:443"
volumes:
- /var/run/docker.sock:/var/run/docker.sock:ro
- traefik_letsencrypt:/letsencrypt
networks:
- proxy
restart: unless-stopped
```