

杨栋森

项目作品集

面向 Linux 运维、服务部署、容器化联调、自动化运维与云原生方向的实践整理

手机：18683236881 邮箱：yangyus8@163.com GitHub：github.com/YangYuS8

在线简历：<https://yds.nesoriel.cn>

定位概述

本作品集用于补充一页简历，重点展示在个人 Homelab、Kubernetes 工具、自动化运维平台、实习项目和团队项目中的工程实践。内容围绕“能把项目跑起来、能定位问题、能整理部署步骤、能把联调过程复现给他人”展开。

表述保持克制：不把学习项目包装成大型商业系统，也不夸大为独立负责完整平台；重点说明真实参与的部署、联调、配置排障、接口验证、文档整理和工程交付工作。

关注点：服务能否稳定启动，链路能否验证，问题能否定位，部署步骤能否复现，交接材料是否可用。

能力关键词

Linux

PVE Homelab

Docker

Docker Compose

Nginx

Kubernetes

Helm

Prometheus

Go

FastAPI

Ansible Runner

MySQL

Redis

接口联调

日志排查

部署文档

统一可验证入口：在线简历 <https://yds.nesoriel.cn>；公开代码与补充材料以 GitHub / 博客 / 项目仓库为准。

01	个人 Homelab 运维与服务迁移实践	2
02	Kube-Sentinel：Kubernetes 夜间值班哨兵工具	3
03	Astralith：轻量级自动化运维平台	4
04	MedKnow：梦溪比特 Go 后端实习项目	5
05	Ocean：多组件业务系统部署联调实践	6
06	补充实践与可承担工作	7

01 | 个人 Homelab 运维与服务迁移实践

以个人 PVE Homelab 和多台 Linux 服务器为长期实践环境，围绕服务部署、远程访问、容器编排、反向代理和故障记录沉淀经验。

角色

项目负责人 / 运维实践

技术栈

PVE、Debian、Docker Compose、Dockge、Nginx / OpenResty、Tailscale、WireGuard / OpenWrt、1Panel、Git

重点能力

Linux 服务维护、容器化部署、远程访问配置、反向代理、故障排查、部署文档整理

可验证材料：在线简历 <https://yds.nesoriel.cn>；个人博客 blog.yangyus8.top 中的 Homelab、Docker Compose、服务器迁移与排障记录。

背景与目标

早期环境中，PVE 宿主机同时承载 Docker、面板、媒体、代理和反向代理服务，服务边界不够清晰，排查问题时容易混淆宿主机、容器、网关和业务服务责任。后续将环境逐步整理为“PVE 宿主机 / infra-docker-01 / net-gateway-01 / 业务 VM”的分层结构。

简化架构图



主要实践

- 维护 Linux 虚拟机、Docker 服务机、代理网关和多类自托管服务，作为 DevOps、SRE 与云原生技术学习的长期实验环境。
- 使用 Docker Compose / Dockge 管理多个自托管服务，整理 Stack、.env、配置目录、数据目录和 Git 版本化流程，提升服务迁移和复查便利性。
- 配置 Tailscale、WireGuard / OpenWrt 旁路网关和 Nginx / OpenResty 反向代理，处理远程访问、Docker Hub 拉取、代理链路、IP 冲突和端口暴露问题。
- 将部署、迁移和排障过程整理为博客文章与检查清单，形成可复用的服务器维护和 Compose 管理经验。

可展示的工程价值

问题类型	处理方式	沉淀结果
服务混跑导致边界混乱	按宿主机、Docker 服务机、网关和业务 VM 拆分职责	降低排障时的定位范围
环境迁移容易遗漏配置	整理 Compose、环境变量、数据目录和启动顺序	形成可迁移、可回滚、可复查的服务管理方式
远程访问链路复杂	结合 Tailscale、WireGuard、OpenWrt 与 Nginx 反代排查	积累网络链路、端口和代理配置经验

投递关联：适合支撑运维实习、DevOps 实习、部署联调实习中的 Linux 基础、容器服务管理、Nginx 反代和故障记录能力。

02 | Kube-Sentinel: Kubernetes 告警处置工具原型

面向 Kubernetes 夜间值班场景的告警处置工具原型，重点验证 Alertmanager 接入、CRD / 控制器思路、Prometheus 指标、Helm 部署和自动化风险边界；不表述为已成熟落地的生产级平台。

- 角色

开发者 / 云原生运维原型实践
- 技术栈

Go、Kubernetes、Controller Runtime、CRD、Helm、Alertmanager、Prometheus、Grafana、Telegram Bot
- 重点能力

Kubernetes 控制器原型、告警接入、Prometheus 指标、Helm 安装、最小化自动处置设计

可验证材料：GitHub: <https://github.com/YangYuS8/kube-sentinel>；在线简历: <https://yds.nesoriel.cn>

设计思路

当前定位为原型验证：工具设计为接收 Alertmanager Webhook 告警后映射为 HealingRequest，再围绕 Deployment L1 处置设计有限、可解释、可人工接管的流程。设计重点不是让脚本无限制地修改集群，而是在安全边界内验证从告警、分诊、最小处置到审计记录的闭环思路。

关键工作

- 原型验证 Alertmanager Webhook 到 Kubernetes 自定义资源的接入流程，形成统一告警入口的设计雏形。
- 围绕 Deployment 处置设计维护窗口、速率限制、爆炸半径控制、熔断和写前快照校验，用于说明自动化风险边界。
- 设计 Kubernetes Event、结构化审计、Prometheus 指标和 Telegram incident card，用于保留可解释的处置证据。
- 整理 Helm / YAML 安装清单、Grafana Dashboard、ServiceMonitor、发布与回滚文档，并通过本地测试、静态检查和最小安装脚本做原型验证。

运维视角总结

模块	实践内容	体现能力
告警接入	Alertmanager Webhook → HealingRequest	理解监控告警链路和事件建模
安全边界	维护窗口、限速、熔断、快照校验	避免自动化工具扩大事故影响
可观测与审计	Event、审计日志、Prometheus 指标、通知卡片	为人工接管和复盘保留证据
交付验证	Helm/YAML、Dashboard、ServiceMonitor、回滚文档（原型验证）	具备部署与交付意识

投递关联：可用于说明 Kubernetes 基础、控制器原型实践、监控告警理解，以及对自动化运维风险边界的认识。

03 | Astralith：轻量级自动化运维平台

面向个人 PVE / Homelab 和中小型 Linux 服务器环境的轻量级自动化运维平台。当前以毕业设计和原型开发为边界，强调模块拆分、主机管理、任务执行、日志留存与后续扩展规划。

- 项目题目**
- 基于 FastAPI 与 Ansible 的轻量级自动化运维平台设计与实现
- 技术栈**
- FastAPI、Ansible Runner、SQLAlchemy、SQLite、Celery、Redis、APScheduler、JWT、Vue 3、Docker Compose、AI Agent 辅助分析
- 重点能力**
- 自动化运维平台设计、异步任务、执行日志、Docker Compose 部署规划、故障分析材料整理

可验证材料：GitHub：<https://github.com/Nesoriel/astralith>；在线简历：<https://yds.nesoriel.cn>

进度边界

已完成

项目定位、模块拆分、技术路线梳理、部分后端原型设计。

开发中

主机管理、任务执行、执行日志留存、前端日志展示。

规划中

GitOps diff、Apply Plan、Docker Compose 管理、AI Evidence Pack。

功能范围

- 已完成：明确登录认证、主机管理、内置运维模块、任务创建、定时巡检和日志展示等模块边界。
- 开发中：远程操作计划统一通过 Ansible Runner 执行，Celery + Redis 处理异步任务，APScheduler 触发定时任务。
- 开发中：设计 SQLite 保存任务状态、每台主机 stdout / stderr 和原始事件，便于后续排障、复盘和演示。
- 规划中：GitOps desired-state sync、Desired / Actual diff、Apply Plan、Docker Compose Apply 记录和回滚元数据。

工程取舍

平台侧强调“受控执行”而不是把 Agent 或脚本直接放到高风险路径。AI Evidence Pack 等能力目前作为规划方向，用于整理任务输出、错误日志和执行事件，再辅助生成故障分析、Runbook 或变更建议；真正执行仍保留人工确认、策略校验、审计日志和回退边界。

能力点	项目体现
自动化执行	开发中：以 Ansible Runner 统一远程操作入口和输出结构。
异步与定时	开发中：Celery + Redis 处理长任务，APScheduler 触发周期巡检。
日志与审计	开发中：保存 stdout、stderr、任务状态和原始事件，便于排查。
部署联调	规划中：使用 Docker Compose 整理运行环境和组件依赖。
风险控制	设计中：强调人工审批、策略校验和可回退，不夸大无人值守。

投递关联：适合展示自动化运维、任务平台、日志留存、Docker Compose 部署和 AI 辅助运维的工程理解。

04 | MedKnow：梦溪比特 Go 后端实习项目

梦溪比特（鞍山）软件信息技术有限公司实习项目。项目为养生知识科普短视频平台，包含用户、专家、视频、评论、收藏、浏览历史等业务模块。

实习时间 2025.09 - 2026.03
职位 Go 后端开发实习生 / 后端开发实习生
技术栈 Go、Gin、Gorm、MySQL、Docker、Git、接口联调、服务部署、问题排查、文档整理

可验证材料：项目仓库：<https://github.com/NotSleeply/MedKnow>；在线简历：<https://yds.nesoriel.cn>

参与内容

- 参与后端项目结构梳理，理解用户、专家、视频、评论、收藏、浏览历史等模块的数据关系与接口调用流程。
- 配合前端进行接口联调，验证接口返回、参数传递和核心业务流程，协助定位配置、参数、数据库连接和接口返回问题。
- 参与本地运行环境搭建，协助处理数据库连接、端口占用、配置文件、依赖安装等常见运行问题。
- 使用 Docker 整理并验证项目运行环境，降低团队成员因本地环境差异导致的部署与联调成本。
- 整理项目运行说明、配置注意事项和问题记录，为后续演示、部署和维护提供文档支持。

项目定位说明

该经历定位为真实的后端参与和部署联调实习经历：重点体现对 Go Gin + Gorm + MySQL 项目结构的理解、接口联调、Docker 环境验证、配置排障和文档整理，不夸大为独立负责完整商业系统。

工作场景	具体动作	体现能力
后端结构理解	梳理用户、专家、视频、评论、收藏、浏览历史模块关系	能阅读业务代码和数据关系
接口联调	与前端验证参数、返回结构和业务流程	具备 HTTP API 联调意识
环境排障	处理数据库连接、端口占用、配置文件、依赖安装问题	能参与项目本地启动和运行问题定位
部署验证	使用 Docker 整理运行环境并记录问题	降低团队环境差异带来的联调成本

05 | Ocean：多组件业务系统部署联调实践

海洋生态样本与设备巡检管理平台，覆盖巡检任务、样本登记、检测结果、异常处理、分析任务和统计看板等流程。作品集重点展示本地运行、部署联调和文档整理能力。

技术栈 Docker Compose、Nginx、Laravel、MariaDB、Redis、Python Worker、React、TypeScript
参与重点 多组件启动、服务依赖整理、数据库初始化、缓存队列、异步 Worker、HTTP 返回验证、交付说明

可验证材料：GitHub：<https://github.com/YangYuS8/ocean>；在线简历：<https://yds.nesoriel.cn>

主要工作

- 参与前后端分离业务系统的本地运行、部署联调和文档整理，围绕后端接口、数据库、缓存队列和异步 Worker 验证链路。
- 根据项目文档与配置文件梳理服务启动流程，使用 Docker Compose 编排 Nginx、Laravel、MariaDB、Redis 和 Python Worker 等组件。
- 整理端口、环境变量、数据库初始化、缓存队列和服务依赖关系，通过容器状态、服务日志、HTTP 返回和页面访问结果定位配置或依赖问题。
- 维护项目概览、部署说明、接口说明、分工记录和交付检查清单，形成可复现的运行步骤和交接材料。

部署联调链路

组件	联调关注点	验证方式
Nginx	反向代理、静态资源、入口路由	页面访问、HTTP 状态码、服务日志
Laravel	后端接口、环境变量、数据库连接	接口返回、容器日志、配置文件检查
MariaDB	数据库初始化、账号权限、连接参数	连接测试、迁移/初始化结果、错误日志
Redis / Worker	缓存队列、异步任务、分析流程	队列状态、Worker 日志、业务流程验证

项目收获

该项目强化了对“多组件系统如何跑起来”的理解：不仅要看代码功能，还需要关注端口、环境变量、服务依赖、数据库初始化、队列 Worker、接口返回和文档交接。对于运维 / DevOps 实习岗位，这类经验能直接对应部署、联调、排障和交付支持工作。

06 | 补充实践与可承担工作

除核心项目外，也通过开源维护、论文项目材料支持、博客和开发环境维护积累工程协作与文档化能力。

开源与工程实践

- **lwe Linux 桌面工具**：个人维护的 Linux 桌面开源项目，用于浏览、管理和应用 Wallpaper Engine 内容；维护 GitHub Releases、AUR 包、中英文在线文档和用户反馈记录。可验证材料：GitHub <https://github.com/YangYuS8/lwe>。
- **CubeSandbox 开源贡献**：参与 TencentCloud/CubeSandbox 小范围代码阅读、修改验证和协作练习；因本作品集不展开具体 PR 编号，仅作为开源协作流程补充材料保守呈现。
- **目标检测项目材料支持**：参与实验环境部署、依赖配置、训练环境排障、部分数据处理和文档归档，不将算法方向作为主求职线索。

可承担的实习工作

- Linux 服务器基础巡检、服务状态检查、日志初步定位和故障记录整理。
- Docker / Docker Compose 项目的本地部署、环境变量整理、端口和数据目录检查。
- Nginx 反向代理、HTTP 接口连通性验证、前后端联调支持。
- Kubernetes 基础资源、Helm 安装、Prometheus / Grafana 监控组件的学习型协助。
- README、部署文档、接口说明、检查清单和项目交接材料整理。

综合能力映射

能力方向	支撑经历	投递表达
Linux / 服务运维	Homelab、服务器维护、博客复盘	能从服务状态、日志、端口、资源占用检查切入
容器与部署	Homelab、Ocean、MedKnow、Astralith	能整理 Compose、环境变量、服务依赖和部署说明
云原生学习	Kube-Sentinel、Kubernetes / Helm / Prometheus 实践	了解 K8s 资源、告警链路和受控自动化思路
接口联调与排障	MedKnow、Ocean	能配合前后端验证参数、返回、数据库连接和业务流程
文档与交接	所有项目	重视可复现、可验证、可交接，而不是只停留在“能跑一次”

可验证材料：在线简历：<https://yds.nesoriel.cn>；GitHub：github.com/YangYuS8。

求职方向：DevOps 实习 / SRE 实习 / 运维开发实习 / 云原生运维实习 / 部署联调实习。