

朱琪琪

29岁 | 男 | 18782177403 | 2660996862@qq.com | GO、云原生开发

教育背景

成都信息工程大学	计算机科学与技术	本科	2015.09~2019.06
----------	----------	----	-----------------

工作经历

成都天空卫士网络安全技术有限公司	Go 开发工程师	2021.04~至今
北京华宇信息技术有限公司	JAVA 开发工程师	2019.06~2021.04

项目经历

GatorCloud 云平台	2021.08~2025.10
----------------	-----------------

- **技术栈:** K8S、Kratos、Gorm、Wire、gRPC、gRPC-Gateway、Client-go、Redis、Postgresql、Prometheus、Grafana、EFK。
- **项目简介:** GatorCloud 旨在将天空卫士的核心 DLP（数据防泄漏）产品从传统的物理设备模式，彻底重构为基于云的多租户 SaaS 服务。该项目解决了从大型企业到中小企业的客户对数据安全防护的迫切需求，通过利用云的弹性、可扩展性和按需付费模式，显著降低了客户使用顶尖 DLP 技术的门槛和成本。云平台负责统一管理租户信息、DLP 安全服务、K8S 集群等信息。
- **功能与技术实现:**
 1. 多集群管理：包括集群注册、使用 PopCode 实现跨集群认证、基于 EFK 完成集群的日志配置与收集、资源监控、Web 终端、K8S 常见 Workload 信息展示等功能。
 2. 多租户管理：包括租户信息同步与展示、安全服务的组件构成与运行状态等功能；基于 Client-go 获取数据并聚合展示。
 3. 安全服务管理：安全服务资源定义的维护、版本控制、接口测试、租户独立升降级安全产品等功能。
 4. 基于 zap 并结合业务需求定制化开发日志包，统一项目日志规范；定义统一错误处理 ErrorX，包含 HTTP/gRPC 状态码、业务错误码、错误信息等。
 5. 在多集群全局配置更新时，使用基于 ETCD 的分布式锁确保更新的原子性；使用 Client-go 中的 workqueue 的限速队列机制实现租户策略的稳定更新与同步。
 6. 利用泛型、反射提供工具包，利用 Redis 实现一个分布式锁，并且结合 singleflight 模式来减少分布式锁竞争。
 7. 利用 pprof 定位频繁内存分配、协程泄露问题，通过协程池技术和 sync.Pool 解决协程泄露与内存分配问题。

安全产品 Operator	2021.11~2024.10
---------------	-----------------

- **技术栈:** K8S、Client-go、KubeBuilder、Etcd、Controller-Runtime。
- **项目简介:** 为解决多个云化安全产品（DLP 系列）在复杂多租户场景下的统一交付、依赖治理与生命周期管理难题，参与开发了基于 Kubernetes Operator 框架的统一管理平台。该平台将传统的脚本化、手工部署转变为声明式、自动化的交付与管理体验，显著提升了交付效率与运维可靠性，并解决了产品间复杂的依赖与资源共享问题。
- **功能与技术实现:**
 1. 管理租户 DLP 产品订购、取消订购以及产品到期等功能。
 2. 管理不同 DLP 产品容器依赖、容器共享、容器优先级、Apisix 路由维护等功能。
 3. 针对多租户高频资源变更场景，利用 Workqueue 实现事件去重与异步协调，结合 Informer 缓存机制降低 APIServer 压力。
 4. 设计到期回收机制，结合 K8s Finalizer 实现资源自动释放，避免资源泄漏。
 5. 使用三路合并算法更新资源对象。

Secgator 私有化部署

2024.03~2025.04

- **技术栈：** K8S(Pod & Service 网络、ConfigMap、Namespace 资源隔离、PV & PVC 持久化存储、StorageClass、Init Container、节点亲和性)、Cobra、Viper、Pflag。
- **项目简介：** 响应某集团数字化转型战略，将传统硬件部署的 UCWI-11000 安全产品改造为 Kubernetes 云原生架构。实现从物理设备到容器化 K8S 部署的转型，满足客户对弹性伸缩、高可用和统一管理的核心需求。
- **功能与技术实现：**
 1. 设计并实现云原生迁移架构：基于 Kubernetes 重新设计了产品部署架构，将原有 Docker-Compose 组件拆分为多个微服务 Pod (如 Appliance, UCWI, DCRP/RDE, SPE/KVServer)，通过精细化的 Pod 亲和性、共享存储卷 (PV) 及 Init Container 机制，解决了组件间严格的启动顺序、网络通信 (127.0.0.1 调用) 和文件系统挂载传播等关键挑战。
 2. 开发自动化部署与运维 CLI 工具：使用 Cobra、Viper 和 Pflag 构建了功能强大的命令行管理工具 (CLI)，实现了产品的全生命周期管理，包括：一键部署/升级/回滚、状态与配置管理、证书与注册自动化。
 3. 解决关键技术与运维难题：采用 HostPort 模式在指定 Worker 节点固定端口暴露服务，以满足上级平台对访问端口的强要求，并通过 Namespace 实现多套 UCWI 实例的资源与网络隔离，支持水平扩容与负载均衡。将用户配置与生成的资源清单自动备份至集群内 ConfigMap，确保每次部署版本可追溯，支持快速回滚，保障了部署的可靠性与可维护性。

集约化智慧执行系统（华宇信息）

2019.06~2021.04

- **技术栈：** Springboot、MyBatis、MySQL、SpringCloud、Redis、K8S、Vue、MinIO
- **项目简介：** 采用 VUE+Artery6 与 springboot、springcloud 完成执行案件办理系统开发，后端通过 Swagger 与前端进行数据交互和接口联调。用户在智慧审判的立案系统进行立案，通过 xxl-job 定时推送数据到本系统的案件表；案件通过身份验证接收后在执行通知模块生成案件相关人的文书并签章，集约人员同时进行传统查询和对外委托，对外委托通过 RocketMQ 将委托信息推送到最高法院的委托平台；委托过程通过调用审批平台完成多级审批；此后法官将对案件进行繁简认定并分案，承办法官进入法官平台进行办理；法官平台主要有执行组织、财产查询、财产控制、财产变现、强制措施、案件结案、卷宗归档、审限管理、任务管理等。

专业技能

- 熟悉 Gin、GORM、Wire、Kratos、gRPC 等主流框架，熟练运用 Cobra、Viper、Pflag 等构建企业级 CLI 应用框架，实现配置的集中化与多源化管理。
- 熟悉 GO 并发编程，包括 sync 包中的 Mutex、RWMutex、WaitGroup、Cond、Channel、Context 等实现原理。对拥有丰富的系统性能调优实战经验，熟练运用 pprof 定位和解决协程泄漏、锁竞争等问题，显著提升系统吞吐与稳定性。
- 深入掌握 Kubernetes 核心组件和基本原理。
- 熟练使用 Client-go，熟悉循环控制模式、Informer 源码分析，开发 Operator 管理集群资源。
- 熟悉分布式存储服务 ETCD 的基本原理，熟悉基于 ETCD 的分布式锁，了解 ETCD 的 Watch 机制、租约机制。
- 熟悉容器技术，理解 Namespace、Cgroups，熟悉容器子进程管理，集群网络插件如 Flannel、Calico。
- 熟练运用 Prometheus + Grafana 构建监控体系、日志聚合 (EFK) 方案，实现分布式系统问题的快速定位与根源分析。
- 熟悉 CI/CD 流程设计，包括 GitLab CI、ArgoCD、Harbor、Helm 等。
- 熟悉 MySQL，掌握 InnoDB 事务、索引优化及日志机制，具备 SQL 性能调优能力。熟悉 Redis 核心数据结构与持久化机制，能够设计高效的缓存方案提升系统性能，熟悉 Redis 分布式锁的实现方案。