

极客时间运维进阶训练营课程大纲与项目简介

课程大纲

模块一：容器和单机容器编排入门与实战

教学目标

1. 理解现代应用容器技术，掌握 Docker 容器技术的典型用法
2. 掌握 Docker 卷的工作机制和典型用法
3. 掌握 Docker 镜像的制作方法和分发机制
4. 掌握 Docker 容器网络的工作机制
5. 掌握 docker-compose 编排

通过学习掌握的核心能力

1. 为传统应用构建 Docker Image，并基于私有或公有 Registry 完成分发的能力
2. 熟练运行 Docker 容器化应用，能将主流传统应用迁移并运行于 Docker 容器中的能力
3. 熟练使用 docker-compose 编排运行小型应用环境的能力
4. 熟练使用 containerd 运行容器化应用的能力

实践练习

1. 使用 Docker 容器运行主流的传统应用，例如 Nginx、MySQL 和 WordPress 等
2. 测试并分析 Docker 原生的网络模型下的通信机制
3. 将某传统应用构建成为 Docker Image，并测试其优化机制
4. 基于 Harbor 构建企业级私有 Registry，提供内部私有镜像仓库服务
5. 使用 docker-compose 编排运行 Nginx、PHP-FPM 和 MySQL，并运行 Wordpress

详细内容

第一周

1. Linux 容器技术基础：Cgroups 及 Namespaces 简介
2. 低级容器运行时与高级容器运行时、OCI、CRI、cri-dockerd、CRI-O
3. 现代应用容器技术 Docker 的特性、架构与工作机制、存储引擎介绍
4. 部署 Docker，对 Docker 进行镜像加速、使用 systemd 进行资源限制等参数优化
5. Docker 的常用命令
6. Docker 的存储管理、数据卷和数据卷容器
7. Docker 网络模型与跨主机的容器间通信方式，以及可能面临的问题

第二周

1. Docker 网络模型与跨主机的容器间通信原理、过程展示与分析
2. Dockerfile 构建镜像参数详解

3. 案例：企业级 Docker 镜像的分层与构建
4. Docker 资源限制介绍、在容器上进行 CPU 与内存资源限制的方法
5. 容器运行时 Docker 与 containerd 对比、优缺点详解，containerd 安装、常用客户端及命令使用简介
6. Docker Registry 和 Harbor 镜像仓库简介、高可用机制，部署自签名的 Harbor 镜像仓库并实现镜像统一分发
7. docker-compose 使用简介，基于 docker-compose 实现 Nginx+Java+MySQL 服务部署

模块二：基于 Jenkins 的 CI/CD 及 DevOps 实战

教学目标

1. 理解 CI/CD 的工作流程，理解 CI/CD Pipeline 的工作流程
2. 掌握 Git 工具的常见用法，能使用 Git 管理代码版本、进行版本跟踪
3. 理解 GitLab 的功能、关键组件及工作机制
4. 理解 Jenkins 的架构体系与工作机制，能熟练使用 Jenkins Web UI 构建应用的 CI/CD Pipeline
5. 理解 Jenkinsfile 的工作机制，熟练使用 Jenkinsfile 定义 Pipeline

通常学习掌握的核心能力

1. 基于 Git 保存和跟踪代码的能力，熟练使用 Git branch 进行项目协作的能力
2. 基于 Jenkins 为应用构建 CI/CD Pipeline 的能力
3. 基于 Jenkins Pipeline 完成应用发布的能力

实践练习

1. 使用 Git 进行代码管理和版本跟踪
2. 部署 GitLab，并通过远程仓库完成项目协同
3. 构建自由风格的软件项目，以及使用声明式 Pipeline 构建软件项目
4. 在 Pipeline 中使用 Ansible 完成应用的自动化部署
5. 在 Pipeline 上使用 SonarQube 实现代码扫描
6. 在 Pipeline 上构建 Image 目标，并结合 Docker 完成应用的自动化部署

详细内容

第三周

1. CI/CD、敏捷开发、DevOps，以及 CI/CD 工具链介绍
2. 常见的代码部署方式，如蓝绿部署、灰度发布、金丝雀部署、滚动部署、A/B 测试等
3. 分布式版本控制工具的工作模型、Git 基础概念及基本应用
4. 分支管理、常见的分支模型及各自的协作流程
5. GitLab 简介、安装、账户与权限管理、项目管理
6. GitLab 分支管理、分支合并、代码克隆、数据备份、数据恢复
7. Jenkins 功能、特性、架构体系及工作机制
8. Jenkins 安装、插件管理、作业类型及各自的特性
9. Jenkins 凭据、常用类型的触发器，以及基于 GitLab 的自动触发机制
10. Jenkins 的参数化构建及案例

第四周

1. Jenkins 分布式架构概念与实现
2. 声明式 Pipeline 语法结构
3. 声明式 Pipeline 快速入门
4. 参数化 Pipeline 及高级应用
5. Jenkins 通知机制，在声明式 Pipeline 中使用 post
6. Jenkins 视图简介与应用场景
7. 代码质量测试工具 SonarQube 介绍
8. SonarQube 部署、插件管理、实现代码扫描
9. Jenkins 集成 SonarQube 实现代码自动扫描
10. 实战案例：基于 Docker 系统实现代码自动发布和回滚

模块三：分布式搜索服务与日志中心

教学目标

1. 了解全文检索引擎的基本理论和倒排索引原理
2. 了解 Lucene 的基本概念和底层存储结构
3. 理解 ELK 技术栈中的各关键组件的功能、特性及协同逻辑
4. 掌握 Elasticsearch 的架构组件及关键工作机制
5. 熟练使用 Elasticsearch 存储索引数据，并能通过 API 接口进行常规的数据检索
6. 掌握 Logstash 常见插件的使用，如 grok 和 geoip 等
7. 熟练使用 Kibana 完成数据分析、图形定义及数据趋势展示

通过学习掌握的能力

1. 熟练使用 Elastic Stack 技术栈规划和部署企业级分布式搜索服务的能力
2. 熟练使用 Elasticsearch 索引数据的能力
3. 熟练使用 Logstash 插件规范化数据的能力
4. 熟练使用 Kibana 定制图形和 Dashboard 的能力

实践练习

部署企业级生产可用的 Elasticsearch 集群，索引日志及任何需要由 Elasticsearch 纳管的数据，并提供大盘展示。

1. 部署 Elasticsearch，并通过 Filebeat 等收集、索引数据
2. 将收集的日志经由 Logstash 预处理后，再经由 Kafka 或 Redis 缓冲后存入 Elasticsearch
3. 使用 Kibana 为 Elasticsearch 中的数据分析和展示提供 Web UI
4. 自定义 Kibana 图形和 Dashboard 展示数据检索和分析的结果

详细内容

第五周：

1. 搜索引擎及功能组件介绍
2. 索引组件 Lucene，以及搜索组件 Elasticsearch 和 Solr 简介
3. Elastic Stack 及小、中、大型规模下的日志采集、存储、展示、搜索平台的架构体系详解
4. Elasticsearch 集群组件及工作机制、集群部署及 API 接口使用入门
5. Elasticsearch 第三方插件 head 管理 ES 集群、检索 ES 数据等
6. Elasticsearch 常用的 ETL 工具栈及 Logstash 和各 Beats 组件功能简介、使用及案例

7. Logstash 部署、配置文件语法简介及生产示例、性能优化、插件管理
8. Logstash 常用的 input、filter、output 等插件使用及案例
9. Kibana 基础、展示接口的配置与应用，以及认证功能

第六周：

1. Logstash 多类型服务访问日志及系统日志收集
2. Java 日志合并、Nginx 访问日志及 TCP 日志
3. syslog 日志收集、基于 Redis 收集 Nginx 访问日志
4. Kibana Dashboard 使用及自定义图形
5. 案例：Logstash 和 Filebeat 结合 Redis 实现企业级日志收集
6. 日志写入数据库、地图显示客户端城市
7. Kibana 日志分析
 - 垂直条形图(y|x)、水平条形图、折线图
 - 指标、数据表、目标图
 - 饼图、热力图
 - 区域地图、坐标地图
 - Kibana 分析 Nginx 日志（PV、UV、业务等）
8. 使用 docker-compose 单机快速启动可用的 ELK 环境

模块四：云原生时代的监控利器 Prometheus

教学目标

1. 理解 Prometheus 监控系统的工作模式及其优缺点
2. 掌握 PromQL 语句的用法，能熟练编写 PromQL 表达式获取时间序列数据
3. 理解 Prometheus 的服务发现机制，能根据需要配置所需要的服务发现功能
4. 掌握 Prometheus 发现并监控 Kubernetes 组件的方式
5. 熟练使用 Alertmanager 进行告警
6. 了解 Push Gateway 及 Prometheus 联邦集群

通过学习掌握的核心能力

1. 基于 Prometheus 和 Alertmanager 部署企业级生产可用的指标监控系统的能力
2. 熟练将主机、系统、中间件、数据存储、缓存、业务应用等纳入 Prometheus 监控的能力

实践练习

部署由 Prometheus、Alertmanager 组成的企业级指标监控系统，结合各类 Exporter 和应用自带的指标 Instrumentation，构建起完备的指标监控和告警体系。

1. 部署 Prometheus Operator，并创建 Prometheus 集群
2. 编写 PromQL 查询时间序列数据，并通过 Grafana 进行图形化展示
3. 基于 Consul、File 和 DNS 等发现要监控的服务
4. 使用 Prometheus 采集 Redis、MySQL 和 Nginx 指标
5. 使用 Alertmanager 对指标的异常值通过钉钉、邮件等发送告警

详细内容

第七周

1. 监控工具对比、黑盒监控与白盒监控、Kubernetes 监控简介
2. Prometheus 详解：概念与原理、优缺点对比、组件介绍、数据采集流程、TSDB
3. 部署 Prometheus Server 和各类 Exporter 完成目标监控
4. PromQL 语句：指标数据、数据类型、匹配器
5. PromQL 语句：时间范围、运算符、聚合运算及示例
6. 配置 Grafana 对接 Prometheus，以及通过 Web UI 聚合并展示监控数据
7. Prometheus 服务发现基础，Relabel 基础、语法介绍和使用案例

第八周

1. Prometheus 基于 Consul、File 和 DNS 实现服务发现
2. 监控案例：配置 Prometheus 监控 JVM 和 Tomcat
3. 监控案例：配置 Prometheus 监控 Redis、MySQL、Nginx
4. Prometheus 告警机制与 Alertmanager 介绍、部署与配置
5. 告警规则、告警模板与告警路由
6. 告警案例：通过邮件和钉钉构建告警体系
7. 告警案例：通过企业微信进行告警，并自定义告警信息模板
8. PushGateway 及 Prometheus 联邦集群介绍及配置

模块五：微服务的必备组件——分布式链路追踪系统

教学目标

1. 理解分布式链路跟踪的工作原理，掌握其核心工作机制
2. 掌握 SkyWalking 的特性，并对比其它跟踪系统的优缺点
3. 掌握如何使用 SkyWalking 对应用进行跟踪
4. 了解 SkyWalking 的指标采集、展示与告警

通过学习掌握的核心能力

1. 使用 SkyWalking 追踪分布式应用的能力
2. 使用 SkyWalking 采集指标数据并完成告警的能力

实践练习

部署企业级生产可用的 SkyWalking 系统，并以之跟踪支持跟踪的分布式应用。

1. 二进制及 docker-compose 部署 SkyWalking 系统
2. 使用 SkyWalking 跟踪 Java 应用和 Python 应用
3. 应用 SkyWalking 的指标数据采集、告警表达式及告警机制

详细内容

第九周

1. 分布式链路追踪需求分析及功能简介
2. SkyWalking 简介、与其它链路追踪系统的对比
3. SkyWalking 二进制及 docker-compose 安装、Web 界面介绍、Java 博客追踪案例
4. Java 服务案例：Tomcat 运行 Jenkins 实现链路跟踪案例
5. Java 服务案例：Dubbo 微服务实现链路跟踪案例
6. Python 服务案例：基于 Django 项目追踪案例

模块六：企业级分布式存储服务入门与实战

教学目标

1. 了解分布式存储中的块存储、文件存储、对象存储的区别与联系
2. 掌握 Ceph 存储系统的架构、组件及工作原理
3. 掌握 Ceph 的 CephFS、RBD 和 RadosGW 存储服务的工作原理及适用场景
4. 理解 CRUSH 算法，理解存储池工作原理
5. 掌握如何规划、部署生产可用的 Ceph 集群

通过学习掌握的核心能力

1. 使用 Ceph 构建企业级生产可用的分布式块存储、文件存储和对象存储的能力
2. 按需扩容 Ceph 存储的能力
3. 持续跟踪和维护 Ceph 存储服务的能力

实践练习

部署企业级生产可用的 Ceph 集群，并为业务提供可靠、高性能的块存储、文件存储及对象存储服务。

1. 部署生产可用的 Ceph 集群，并测试使用其 RBD、CephFS 和 RadosGW 存储服务接口
2. 添加节点来扩容 Ceph 集群
3. Ceph 的常见维护操作
4. 配置 Prometheus 监控 Ceph 集群

详细内容

第十周

1. 分布式存储介绍，块存储、文件存储及对象存储对比及使用场景
2. Ceph 分布式存储历史、版本、组件功能
3. Ceph 数据访问流程及 CRUSH 简介
4. Ceph 环境初始化及 mon 和 mgr 节点，以及横向扩容添加 node 节点
5. Ceph 块存储及对象存储使用
6. CephFS 简介及 MDS 服务部署
7. Ceph 集群管理与维护
8. Ceph 存储池介绍及操作与管理、PG 与 PGP 介绍及数量计算方式
9. Ceph 存储池管理

第十一周

1. Ceph 认证流程和多用户管理及授权
2. Ceph RBD 块存储使用详解及普通用户挂载
3. CephFS 使用详解、Kernel 挂载、ceph-fuse 挂载、普通用户挂载
4. 对象存储概述、存储方式、访问流程
5. Ceph RadosGW 简介及部署、Web 界面使用
6. Ceph 对象存储网关使用详解及 s3cmd 使用、OSS 客户端简介
7. 案例：Ceph 对象存储使用案例、基于 Nginx+RGW 的动静分离及短视频案例

模块七：云原生操作系统 Kubernetes 核心理念与落地实践

教学目标

1. 理解 Kubernetes 架构体系，掌握其基本运行原理
2. 掌握高可用 Kubernetes 集群的常见部署方式
3. 理解 Kubernetes 的声明式 API 及控制器模式
4. 掌握 Kubernetes 下核心资源的功能及其使用，例如 Deployment、Service、Pod 等
5. 掌握 Kubernetes 下容器数据持久化存储的方式
6. 理解并掌握如何在 Kubernetes 系统上配置容器化应用

通过学习掌握的核心能力

1. 部署高可用 Kubernetes 集群的能力
2. 使用 Service、Deployment、Ingress、ConfigMap 等将应用编排运行于 Kubernetes 集群的能力
3. 使用 PV 和 PVC 等为后端应用和业务应用提供持久化存储服务的能力

实践练习

部署企业级生产可用的 Kubernetes 集群，将传统应用环境分阶段迁移至 Kubernetes 平台。

1. 使用 kubeadm 及一款其他常用工具部署高可用的 Kubernetes 集群
2. 通过 re-platform 的方式将传统应用部署并运行于 Kubernetes 集群之上，例如 PHP-FPM 和 WordPress
3. 为 Kubernetes 集群部署 Dashboard，提供 Web 管理接口
4. 使用 livenessProbe 和 readinessProbe 监视容器的存活状态和就绪状态
5. 将 ZooKeeper、MySQL 等主流后端技术运行于 Kubernetes 集群
6. 测试使用 Ingress 发布基于 HTTP 和 HTTPS 协议的应用

详细内容

第十二周

1. 云原生历史、云原生生态、景观图及常用组件
2. Kubernetes 架构组件、工作原理及关键概念
3. Kubernetes 私有云生产环境高可用架构案例、公有云架构案例、性能瓶颈、服务器配置推荐
4. Kubernetes Master 组件及 Node 组件详解
5. Kubernetes 多种部署方式介绍、部署工具及高可用集群部署环境规划
6. kubectl 命令认证机制、命令自动补全、参数详解
7. Dashboard 简介、部署及 Web 界面使用
8. 创建和运行应用：创建和使用 Pod 资源对象

第十三周

1. Kubernetes API 功能介绍、API 版本、资源对象类型简介、声明式与命令式、API 设计原则
2. Service 资源：Service 类型、功用及其管理
3. Pod 探针介绍，容器存活探针及就绪探针案例
4. Kubernetes Controller 控制器简介、ReplicaSet 及 ReplicationController 控制器详解

5. Kubernetes Deployment、Job、CronJob、Service (ClusterIP、NodePort、LoadBalancer、ExternalName) 介绍、访问流程及生产案例
6. Kubernetes Volume 存储卷实现数据持久化介绍、emptyDir、hostPath 及 NFS 案例
7. Kubernetes ConfigMap、StatefulSet、DaemonSet
8. 案例：Kubernetes 运行动静分离的 Web 服务，并结合 NFS 实现数据共享和持久化

第十四周

1. PersistentVolume 与 PersistentVolumeClaim 资源出现的原因与工作原理
2. Kubernetes 基于 Ceph 块存储实现数据持久化
3. Kubernetes 基于 CephFS 实现数据共享与持久化
4. PV/PVC 概述
5. 案例：Kubernetes 运行 ZooKeeper 集群并基于 PVC 实现数据持久化
6. 案例：Kubernetes 基于 StatefulSet 运行 MySQL 一主多从并基于 PVC 实现数据持久化
7. 案例：在 Kubernetes 运行 Jenkins、以及在单 Pod 中以多容器模式运行基于 LNMP 的 WordPress 案例，该 WordPress 会使用前一案例中的 MySQL 数据库
8. Ingress-Nginx 简介及 Controller 部署
9. Ingress-Nginx 虚拟主机、URL 及 HTTPS 案例
10. Kubernetes Container、Pod、Namespace 内存及 CPU 限制、生产案例、生产案例及故障分享

模块八：构建企业级生产可用的 Kubernetes 集群

教学目标

1. 掌握将常规主流后端应用运行于 Kubernetes 的方法、要点和注意事项
2. 理解容器网络模型及其工作机制
3. 理解并掌握 Flannel 网络插件和 Calico 网络插件
4. 理解 NetworkPolicy 及其流量管控机制
5. 理解 Ingress，掌握如何基于 Ingress 发布应用
6. 理解 RBAC 授权插件的工作机制，能熟练使用 Role、ClusterRole、RoleBinding 和 ClusterRoleBinding 进行多级授权
7. 理解 HPA，能熟练使用 HPA 进行应用的自动扩容和缩容

通过学习掌握的核心能力

1. 使用主流的网络插件 (Flannel 或 Calico) 为运行在 Kubernetes 系统上的业务应用构建适用 Pod 网络的能力
2. 追踪、定位和解决常见容器网络问题的能力
3. 为容器编排平台构建出立体化监控系统的能力
4. 升级和扩容 Kubernetes 集群的能力
5. 备份和恢复 Kubernetes 集群的能力

实践练习

构建拥有立体化监控能力的企业级 Kubernetes 平台，选择合用的网络插件，并配置指标监控系统、日志采集系统，来全方位采集指标和日志数据。

1. 使用 etcdctl 命令备份和恢复 etcd 数据
2. Kubernetes 集群升级、扩容和缩容

3. 部署使用 Flannel 网络插件，测试其不同 Backend 模式下的报文转发机制
4. 部署使用 Calico 网络插件，测试其不同转发模式的报文转发机制
5. 使用 etcdctl 备份和恢复 Kubernetes 集群状态数据
6. 升级和扩容 Kubernetes 集群

详细内容

第十五周

1. 网络基础：二层网络基础与 VLAN、三层网络基础、Underlay 和 Overlay
2. Overlay 网络简介、Underlay 网络简介及演示
3. 网络组件 Flannel 介绍及通信流程分析
4. 网络组件 Calico 介绍及通信流程分析
5. NetworkPolicy Ingress 及 Egress 概述与案例
6. Kubernetes RBAC 概述及多账户授权
7. Helm 及其基本应用

第十六周

1. 集群及应用监控：部署 Prometheus Operator 管理 Prometheus 集群
2. 集群及应用监控：配置 Prometheus 监控 Kubernetes 组件：Node、Pod、Service 和 Ingress 等
3. 日志收集、存储与检索：Kubernetes 中的日志收集方式、基于 ELK 的日志收集案例
4. Kubernetes DNS 组件简介、部署、域名格式及域名解析流程
5. CoreDNS 工作机制、性能监控、生产环境瓶颈、常见问题分享
6. etcdctl 命令使用、数据备份及恢复案例
7. Kubernetes 集群维护、Master 节点扩容与缩容、Node 节点扩容与缩容
8. Kubernetes 版本升级案例

模块九：基于典型微服务电子商城项目 mall-swarm 的项目实战

教学目标

1. 综合运用前面课程中的知识，构建起应用的整体运维框架
2. 熟练将示例的微服务化应用及其依赖的后端服务，部署于 Kubernetes 之上
3. 为目标应用构建起立体监控体系，包括指标、日志和调用链跟踪数据的采集、展示和告警
4. 为目标应用构建起 CI/CD Pipeline

通过学习掌握的核心能力

1. 综合运用掌握的运维知识体系，构建出支撑微服务应用基础设施的能力
2. 将微服务应用及其依赖的后端服务迁移至 Kubernetes 系统的能力

实践练习

1. 部规划、部署生产级别的多 Master、多 etcd、多 Node 节点的 Kubernetes 高可用集群
2. 为 Kubernetes 集群部署可观测性系统的基础设施：Prometheus、ELK 和 SkyWalking
3. 配置 Ceph 集群为 Kubernetes 上运行的应用提供持久存储
4. 部署 mall-swarm 依赖到的各后端服务：MySQL、Redis、MongoDB、RabbitMQ 等
5. 为 mall-swarm 的各服务准备 Jenkins CI/CD Pipeline，并完成应用的自动化构建、交付及部署

6. 借助于 Prometheus 监控整个系统
7. 借助于 ELK 收集、检查和展示各应用的日志

详细内容

第十七周

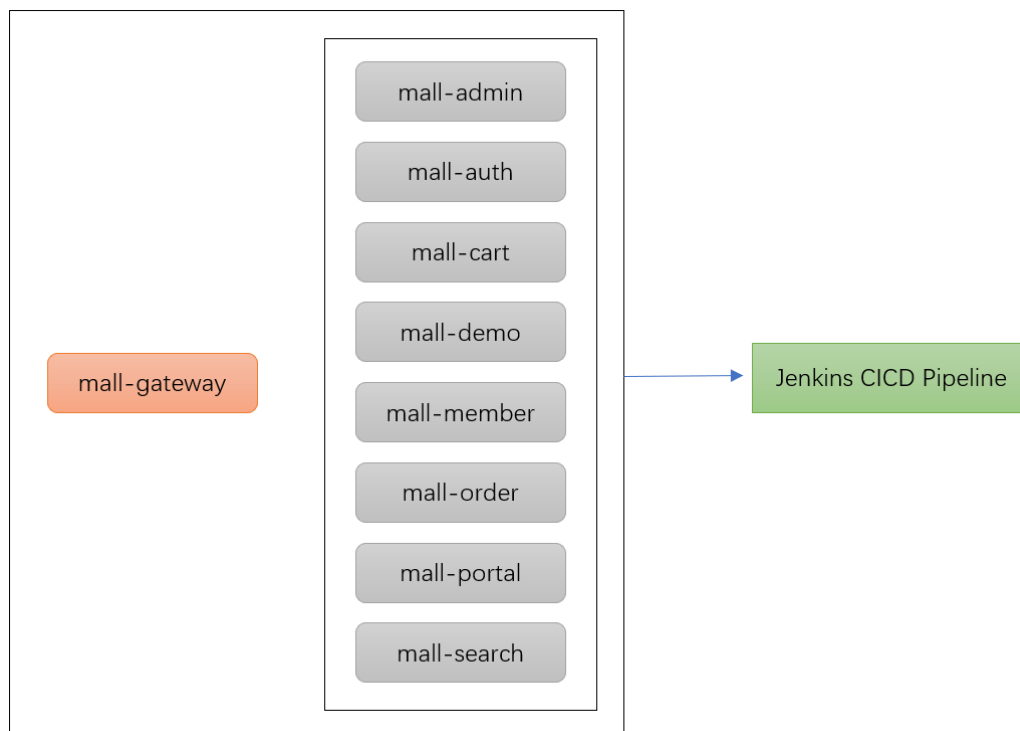
1. 将 mall-swarm 项目依赖的各后端服务编排运行于 Kubernetes，并基于 NFS 实现数据持久化
2. 将 mall-swarm 项目各业务应用服务以容器方式编排运行于 Kubernetes 集群，并验证服务可用性
3. 将 mall-swarm 项目业务模块实现基于 Jenkins 的自动化 CI/CD Pipeline
4. 基于 Prometheus 采集各应用的指标并完成监控配置，相关的监控展示由 Grafana 完成
5. 借助于 Filebeat（或 Fluent-bit）和 Logstash 将日志收集至 Elasticsearch，并通过 Kibana 展示
6. 基于 SkyWalking 实现请求链路跟踪和分析调用链

项目简介

案例一：为业务代码提供全自动化的 CI/CD Pipeline

内容

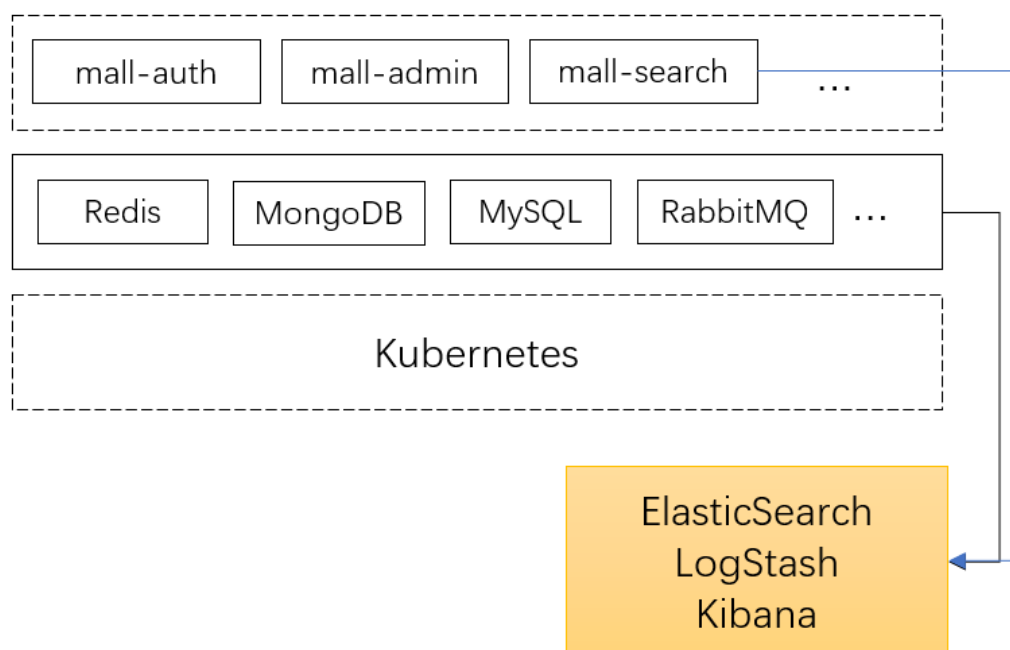
1. 部署生产可用的 GitLab 系统，并将项目代码托管于 GitLab 中
2. 部署生产可用的 Jenkins，为业务应用提供可由 Push 或 PR 事件触发的自动化 CI Pipeline
3. 为业务应用的 CI Pipeline 添加 Image 构建步骤，并自动推送至 Harbor 或其他镜像仓库中，实现为 CD（Delivery）Pipeline
4. 为业务应用的 CD Pipeline 添加部署步骤，实现为 CD（Deployment）Pipeline
5. 为 CI/CD Pipeline 添加 SonarQube 扫描步骤，进行代码质量评估



案例二：将各类应用的日志统一收集至 Elasticsearch 日志中心

内容

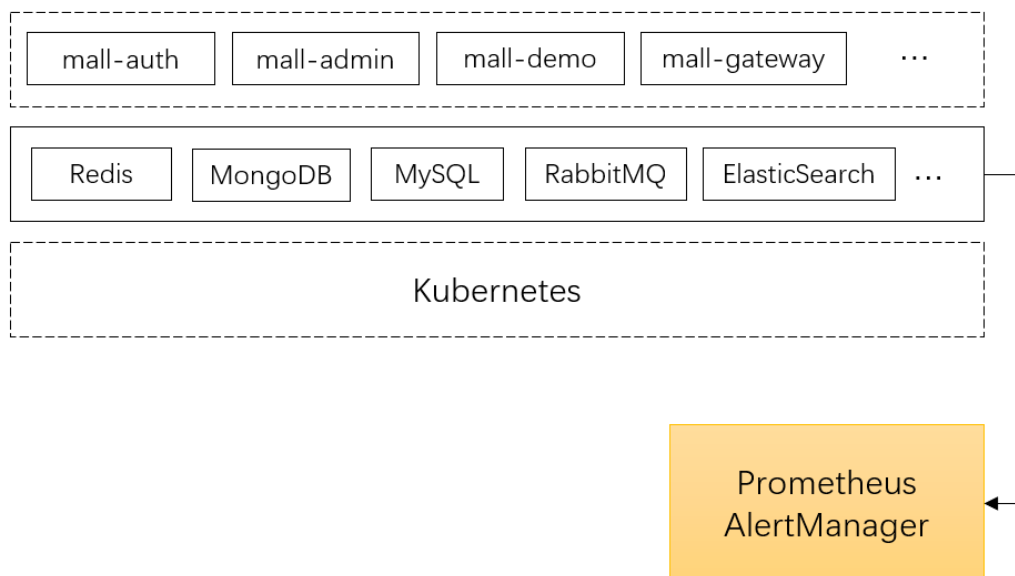
1. 部署生产可用的 Elasticsearch 集群，并为 mall-search 提供搜索服务的后端
2. 在各节点部署 Filebeat 收集后端服务及业务应用的日志，经由 Redis 发往 Logstash
3. Logstash 从 Redis 获取并规范日志格式，按需要存储在 Elasticsearch 上的索引中
4. 使用 Kibana UI 搜索并展示日志数据的聚合结果



案例三：配置 Prometheus 对系统进行全栈监控

内容

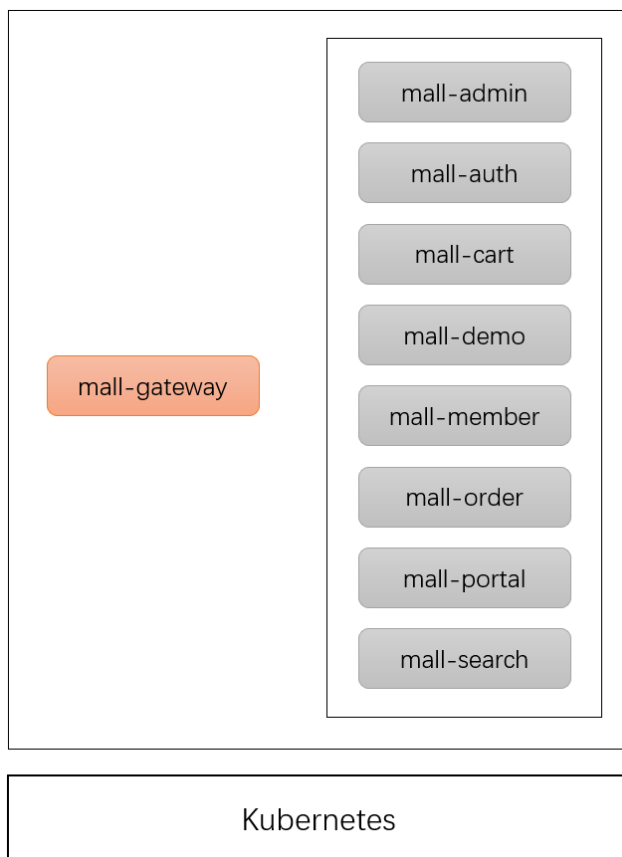
1. 准备生产可用的 Prometheus 集群，配置其自动发现并完成设备层监控、主机层监控、中间件层监控和业务层监控
2. 定制合理的 Alerting Rules，将异常信息经由钉钉或企业微信进行告警
3. 定制 Grafana Dashboard，完成监控数据和告警信息可视化
4. 制订合理的告警收敛和抑制机制，避免告警信息轰炸



案例四：为各业务应用编写 Dockerfile，制作出镜像并完成向 Kubernetes 的迁移部署

内容

1. 为各应用设计制作和优化 Dockerfile，完成其 Docker Image 的构建；测试完成后，构建过程将由 Jenkins CICD Pipeline 自动化完成
2. 将 mall-swarm 的各组件迁移至 Kubernetes 系统运行
 - a. 各服务间通过 Service 进行通信
 - b. 相关配置保存于 ConfigMap 和 Secret 资源中
 - c. mall-gateway 通过 Ingress Controller 发布到集群外部
 - d. 业务数据借助于 PV/PVC 进行持久化
3. 配置 Prometheus 借助于 Kubernetes API Server 发现并自动监控各应用



综合项目：以 mall-swarm 电子商城项目为中心，构建出一个完整的线上业务系统

内容

1. 部署生产可用的 Kubernetes 集群
2. 将业务应用依赖的各后端服务部署于 Kubernetes 集群，或以传统方式运行为集群外部
3. 将 mall-swarm 的各应用部署运行于 Kubernetes 集群，各应用支持基于 Jenkins CICD Pipeline 的自动化部署
4. 将整体系统纳入 Prometheus 监控之下，并定义合理的告警机制完成告警
5. 将整体系统的容器日志纳入 Elasticsearch 中，并能够由 Kibana 接口使用这些日志
6. 使用 SkyWalking 追踪各业务应用相关的调用链

