

Linux后台工程师专栏

1. 算法与设计专题

排序与查找

- 插入排序
- 快速排序
- 希尔排序
- 桶排序
- 基数排序
- 归并排序

常用算法

- 布隆过滤器
- 字符串匹配 KMP算法
- 回溯算法
- 贪心算法
- 推荐算法
- 深度优先, 广度优先

常用的数据结构

- 平衡二叉树
- 红黑树
- B-树
- KMP算法
- 栈/队列
- 布隆过滤器

常用的设计模式

- 单例模式
- 责任链模式
- 过滤器模式
- 发布订阅模式
- 代理模式
- 工厂模式

2. 后台组件编程专题

持久化 MySQL

- MySQL安装配置与远程连接
- 项目: 数据操作源于SQL语句
- 项目: 存储过程与事务处理
- 项目: SQL函数, 运算, 临时表
- 项目: 防数据丢失 备份与恢复
- 项目: MySQL建库建表建索引

消息队列 ZeroMQ

- ZMQ编译安装与开发环境搭建
- 项目: publisher-subscriber模式实现
- 项目: request-response模式实现
- 项目: Router-Dealer模式实现
- 项目: 史上最快的消息队列—性能分析

缓存 Redis

- Redis编译安装配置
- 项目: 客户端全局唯一ID保存机制
- 项目: Redis消息队列机制 发布订阅
- 项目: Redis事务实战
- 项目: Redis安全性能, 数据备份与恢复
- 项目: Redis分布式锁详解

反向代理 Nginx

- Nginx开发介绍
- 项目: 反向代理负载均衡配置详解
- 项目: 自定义协议Upstream开发

Linux高级互联网架构师专栏

1. 源码分析专题

Nginx源码

- Nginx基础架构
- HTTP架构
- 进程间的通信机制
- Nginx高级数据结构
- slab共享内存
- upstream机制设计

Redis源码

- Redis存储系统原理
- 数据模型与键值映射
- 内存操作与磁盘同步
- 主从同步, 原子操作
- 底层IO实现解析

skynet源码

- 多核并发编程
- 消息队列, 线程池
- actor消息调度
- 网络模块实现
- 时间轮定时器实现
- lua/c接口编程
- skynet编程精要

ZeroMQ源码

- 消息模型 发布订阅/推拉模型
- 通信协议 inproc/ipc/tcp/pgm
- 性能分析与经典MQ对比
- 底层网络通信实现机制
- zmq系统架构与实现原理

2. 中间件开发专题

高性能组件

- 如何设计内存池
- 高并发场景下的消息队列
- 协程框架的实现
- 手写线程池
- 异步请求池
- 连接池

高并发网络IO

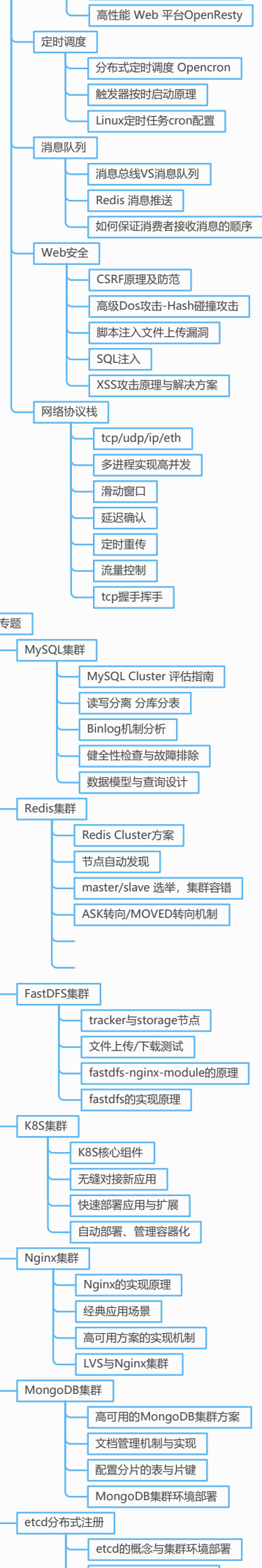
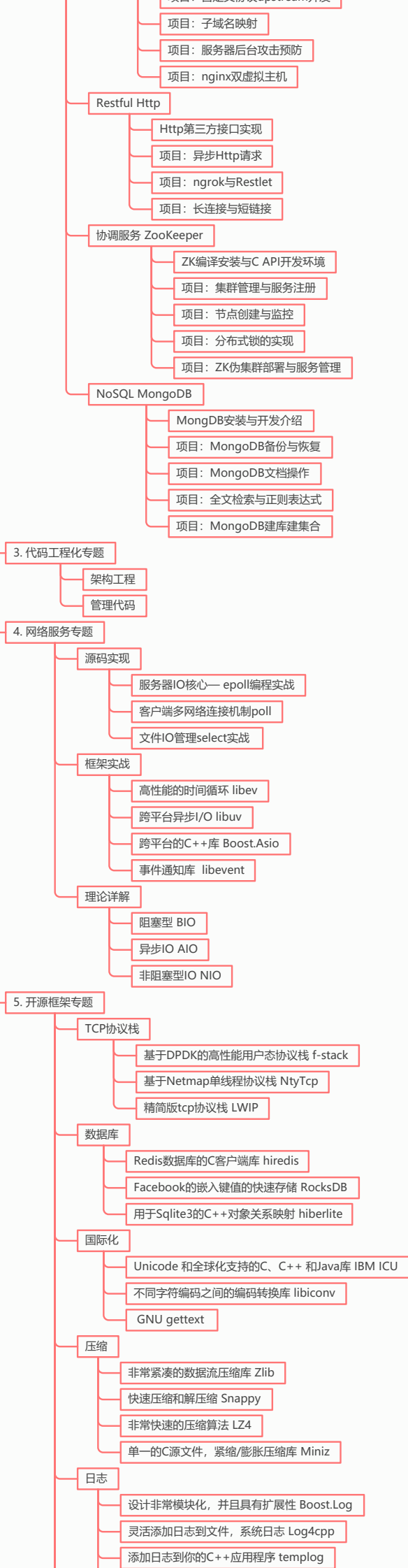
- select,poll和epoll模型的区别
- Linux系统IO模型
- 理解Linux 服务器高并发编程
- 网络接入层架构设计
- 你必须懂得Epoll玩法

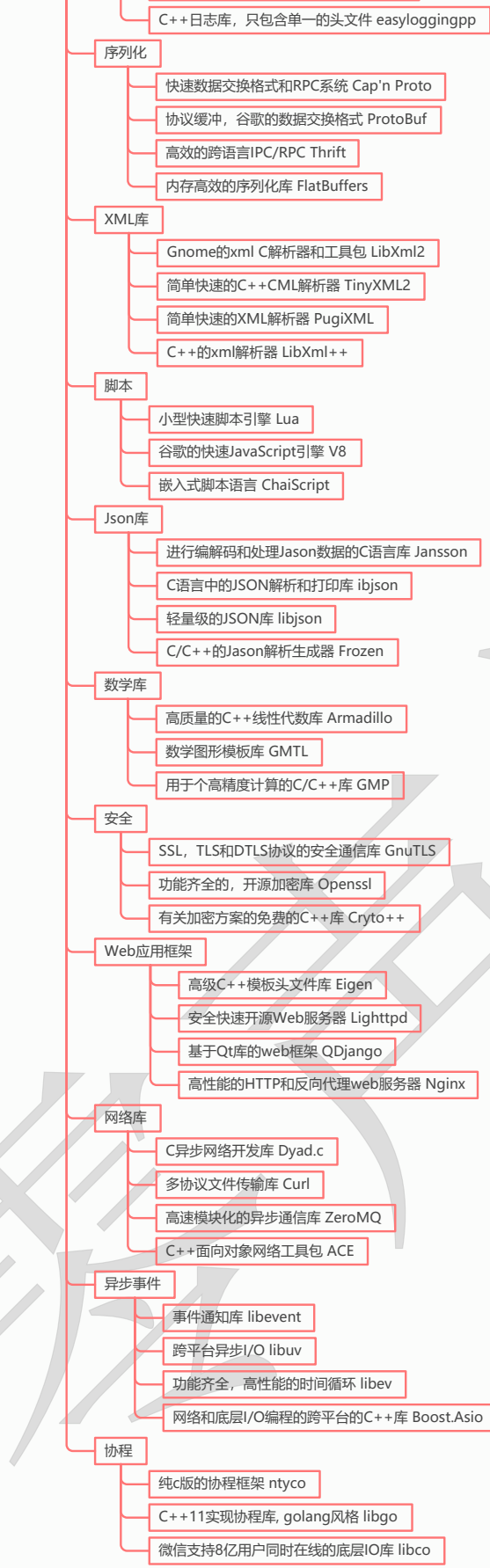
并发性

- 多线程与线程安全
- 事务的隔离级别
- 公平锁&非公平锁
- CopyOnWrite容器
- 悲观锁&乐观锁&CAS&ABA问题
- 事务ACID特性

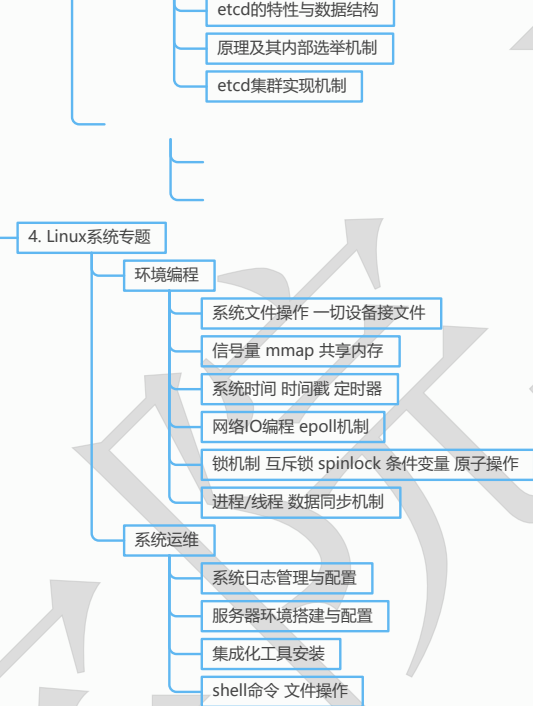
Web服务

- 反向代理Nginx
- 最广泛的web服务器 Httpd

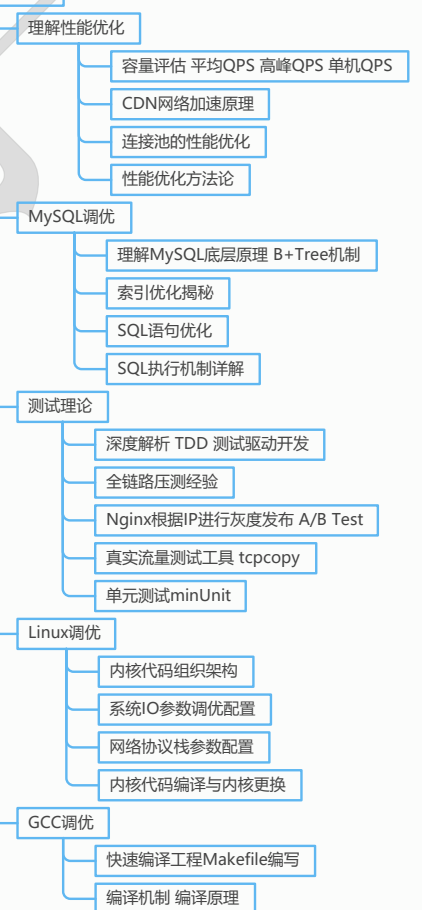




6. 性能测试专题



5. 性能与测试专题



6. 运维统计专题

