

石磊

男 28 岁 | 6 年工作经验

15265311954 | 383821949@qq.com | 已婚

求职意向

DevOps 高级运维开发架构师 | 北京 | 不抵触高强度加班

教育经历

2009 - 2010 | 山东省人民武装学院 | 经济学管理 | 大专

2016 - 2019 | 吉林大学 | 计算机技术与科学 | 本科

Wiki 地址

维护个人 Wiki, 自身学习技术后, 记录文档并方便热爱技术的人进行交流推广, 目前**文章量**已经到达 **2300** 余份, 总编辑次数 1.8 万余次, 在圈内有一定知名度。地址:

<https://wiki.shileizcc.com>

Github 出众项目 <https://github.com/slzcc/crane> 并通过 **CNCF** 一致性检测。

工作经历

2019.08 - 至今 **掌阅科技股份有限公司 高级应用运维工程师**

工作方向: 优化安全, 解决架构难点痛点, 及时处理疑难杂症深入解决潜在因素, 降低项目资源成本, 增加整体可靠性

工作内容: 从事全方面的运维工作, 多方面优化业务架构提高整体性能, 着重解决服务器或业务的安全方面问题, 及自身所能做到做好问题发生后及时通知及时处理, 并在一定程度上做好高度冗余策略避免一个小故障造成的影响范围以及时间, 对 devops 有实质性的落地并根据不同的项目作出合理的解决方案, 对多个云供应商有长久的使用以及维护经验, 对服务器业务以及非业务问题有自己独特的理解方案, 可以带动团队客服各种难题

2016.07 - 2019.07 **北京慧博云通软件技术有限公司 Docker 工程师**

工作方向: 从事公司项目技术运维支持, 项目运维支持, 企业内部技术支持, Docker 项目化支持。DevOps 技术向导。

工作内容: 对企业内部相关项目做 DevOps 管理维护, 对相关项目 Docker 化进行孵化, 对现有资源进行自动化运维管理推进。辅助开发人员制定企业文化, 对每一位新老员工培训适合于企业自身的文档管理, 以及 DevOps 文化制定。合理的减少并有效的规划开发人员个人的工作时间, 高度松耦合开发人员在团队内的依赖程度。制定愉快而又活力的工作环境。

2016.03 - 2016.07 **北京思享聚合科技有限公司(轻课) 运维工程师**

工作方向: 从事公司内部网络环境维护, 服务器日常维护, 员工个人 PC 维修

工作内容： 对企业办公环境部署维护网络环境，物理服务器日常管理维护，线上云环境的日常管理维护，企业内部软硬设备技术支持，Docker 日常维护。

项目经历

2020.08 - 2021.09 **Kubernetes AWS 落地**

项目介绍： 新项目迭代时由于老项目的部署模式比较传统化,人员变更流失较多细节问题从而造成很严重的服务器阐述不明确业务混乱的问题进行改造

功能介绍： 对所有 170 余个海外线上项目基于 **Kubernetes** 进行改造, 集群规模 150 台云实例

功能实现： 对所有项目所在环境基于 docker 进行容器化改造，期间需要解决不同代码不同环境下的打包处理，并且对业务架构进行改造，因项目有很多定制化的东西所以做了很多非业务上的嵌入性改造以提高后续维护效率，并基于 **devops** 进行多个版本的深入迭代以便开发人员离开运维人员也可以知道自己所需的所有资源情况，期间由于云环境的特殊性也没有使用云提供商自带的 k8s 平台，中间发生过**特大 bug** 引发很多灾难性问题从中吸取经验做了多次深入改造，并配合 **aws nlb/clb** 配合 **istio** 做了很多 **server mesh** 治理方案，并且做到了多套 kubernetes 同时提供业务扩大承载能力。基于 devops 做了很多定制化的东西逐步满足开发人员日常需求，目前正在基于 istio 开发开放性平台使用 Go、Vue、Python 语言实现，其中使用 **istio/Kubernetes/metics** 的 **client-go** 整合为一套 Api 提供服务，并使用 **Vue** 把 **Istio HttpRoute** 通过 Form 表单实现注入，不在通过原始 Yaml 文件进行部署，迭代 devops 时由于完全基于 **Jenkins** 很多需求需要开发人员有一定学习经验正在评估舍弃或封装 **Jenkins** 到业务平台实现所以操作统一化。

2020.02 - 2020.09 **Kubernetes IDC 落地**

项目介绍： 由于 IDC 机房业务管理混乱以及 **Hadoop** 生态部署过于混乱，造成运维成本日益增加

功能介绍： 对所有 300 余台线上业务整合 Kubernetes 集群改造，涉及到双机房同时提供业务

功能实现： 对线上 Hdfs、Zookeeper、Hbase、Rabbitmq 等有状态业务通过亲和性策略部署到合适机器，其中 Rabbitmq 有 25 套集群、Hadoop 有 9 套主要提供 Hbase 相关服务，并且部署近百台 Spark 服务用于 hive-on-spark 和 sparkSQL 计算使用。对正式环境部分没有接入 SMP 系统业务整合后部署至 Kubernetes 双机房涉及 200 组非主线项目，通过 **PrometheusOperator** 接入多套 K8s 集群并开发 Alert 程序对多种高级策略进行整合统一录入飞书，可通过飞书系统获取所需监控指标进行日常运维。

2019.11 - 2020.05 **日志服务器升级改造**

项目介绍： idc 机房由于业务极其庞大 ELK 等日志系统无法满足承载能力也不需要做实时搜索能力特进行日志冷数据备份方案，但由于使用较为古老的 mfs 性能上无法得到满足，准备进行替换

功能介绍： 比较多个分部署存储后最终选择使用 ceph 作为替换方案

功能实现： Ceph 最大的难点是维护上，由于我们不需要过多的读取数据所以完全基于 cephfs 进行冷数据的存储，其中由于业务的扩张从开始单个集群的 **300T 扩大到了 500T**，并且部署了多套 Ceph 解决了很多之前因 mfs 存在的读写慢、有头的集群模式、删除海量数据时元数据同步效率慢等等问题。也引发过较大的事故，吸取很多教训做了深入优化，曾经因单台服务器挂掉后没有充足空间导致 Ceph 集群集群阻塞造成 15000+ 的 load, 但做了较多的优化措施服务器没有因此死机，**几十万个任务等待 Ceph 集群恢复后得到完整数据落盘。**

2020.01 - 2020.09 **版本控制系统升级改造**

项目介绍: 由于 Gitlab 使用 6.2 版本, 很多项目无法打开造成页面报错, 开发人员无法日常工作, 还需要每天定时重启 Gitlab 解决内部数据过大引起的服务不可以问题, 同时 Gitlab 还需要进行多个云或地域服务器进行同步底层代码问题已做到灾备等方案做了很多定制化脚本造成的 Gitlab 巨大的安全隐患

功能介绍: 对 Gitlab 进行改造升级至 13.x 最新版本, 同时解决 Gitlab 多个地域同步代码做到时时同步提供准确性, 并区别权限管控, 改造 6 套 Gitlab

功能实现: Gitlab 升级是最好时的, 需要经过大量的测试, 从原来的 6.2 二进制升级到了最新的 13.x 容器化版本, 其中需要做很多人干预, 并且需要对 Mysql 转为 PostgreSQL 数据库作为支撑, 还需要对各种同步策略做细致化的优化以便达到完全自动化处理, 由于老旧的使用习惯造成的很多 Git 使用上的陋习, 这样可以通过多套 gitlab 进行权限管控以便处理如 reset 等策略带来的灾难性问题的发生。并推动 git workflow 计划。

2019.09 - 2020.02 **多云堡垒机落地**

项目介绍: 由于多个云环境资产过于混乱, 没有统一且安全的登入平台进行审计而进行改造

功能介绍: 由于使用的云供应商较多地域环境过于复杂, 选择 Jumpserver 基于 OpenLDAP 集群方案进行服务器用户以及 sudo 权限的管控处理, 总 250 多台云实例

功能实现: 首先对所有资产注入 JumpServer, 由于存在境内外不同环境的下的资产设置了很多中间隧道以便达到中间跳板机的角色进行方案, 并且由于服务器用户过于混乱, 同时使用 OpenLDAP 集群方案对所有资产的登入以及 sudo 权限进行可控, 真正的做到了可追中可高度半自动运维, 降低运维成本。

2019.03-2019.07 **泰禾集团自动化运维技术支持**

项目介绍: 因公司需求外派泰禾集团进行自动化运维技术支持

功能介绍: 对现有的技术进行整合, 并统一使用 Kubernetes 进行调度管理, 集群规模 200+。

功能实现: 公司内部暂时使用 Ansible 进行统一发版回滚策略, 各个项目独立编写 Shell 以及 Ansible 脚本, 使用 Git 单一的分支进行发版, 并通过 Jenkins 创建对应的项目进行发布, 所以存在一定的局限性, 各个项目之间耦合度较高, 并使用 IP 地址等方式进行关联, 存在高风险的问题, 并且掺杂 Windows 服务各种版本的 Tomcat 以及各个项目都有独立的 iPhone、Android、IPD 移动端打包后的可用程度无法保证, 开发人员单一, 开发完项目后无法感知项目部署的过程与细节, 无法有效的调整服务自身配置文件的问题, 即各种 MySQL 版本以及无高可用问题。对所有项目实行 Git 工作流并让开发人员直接对接 Jenkins Pipeline 并且构件独立化, 使用 Kubernetes 集群方式部署实现现有开发生产资源的整合利用。

2018.05 - 2019.02 **公司内部招聘系统技术支持**

项目介绍: 因公司招聘工作日益累计的工作内容, 现有的工作环境存在漏洞, 以公司后续的招聘管理做铺垫, 提出面向整个公司层面的招聘系统开发。

功能介绍: 从第三方下载的简历, 可直接对简历进行录入系统, 并采集数据后返回用户, 后续支持项目部门提出需求, HR 申请任务后对任务自身进行周期工作, 并对每一份简历进行合理的资源利用以及避免 HR 部门内部人员竞争关系, 并推送报表等信息以供 KPI 考核。

功能实现: 调研简历解析, 目前是通过第三方, 后续会通过内部直接解决已经针对过技术做过评估和测试, 对源简历的录如通过 Nginx + Django 编写了可供存储及其展示的上传服务器并使用 uWSGI 做好了服务并发性能支撑, 通过一系列技术支持可针对 HTML/Word/Txt 等常规的简历, 进行的转 PDF 支持, 可对其中不同格式使用不同的转码技术以保证转码的效率, 可实现页面的源简历可视化功能, 并对其中的 51Job/智联等特殊格式的简历进行了优化处理, 使用 Solr 实现全文内容搜索功能。并部署招聘系统的 CI/CD 以及 Docker 化支持, 以及完整的通过测试环境, 正在对其进行生产环境的支持, 后续会接入 OpenResty 对其进行灰度发布的架构改进。使用 Prometheus 对所有虚拟环境进行了监控, 使用

Zabbix 对相应的服务进行了监控，并通过 **Grafana** 做统一的 Dashboard 展示。通过 **Jenkins** 实施统一的环境备份以及**自动化运维**实施的管控。

2018.04 - 2018.07 **Bissantz 中国区站点落地**

项目介绍: 因拓展中国区市场，Bissantz 站点在国内落地

功能介绍: Bissantz 自身站点添加中国元素落地阿里云，并做翻译支持。

功能实现: 采用**阿里云**内部产品架构对 Bissantz 使用 wordpress 技术做建站支持，通过数台服务器做负载均衡及其高可用实现，并使用阿里云监控对其进行告警支持。

2017.08 - 2018.02 **成都西门子 DevOps 实施落地**

项目介绍: 对新建立的 西门子成都区 内部职员统一进行 DevOps 实施培训方案

功能介绍: 项目分为三个步骤，AWS 学习，DevOps 培训，以及 CI/CD 项目实施落地

功能实现: 因项目因素 **驻场** 成都西门子办公区 1 个月有余，主要负责 CI/CD 项目实施，对现有的一个产品进行 CI 的前期孵化，以及线上应用 **Jenkins、Gitlab、Artifactory、Confluence、Jira** 等应用进行部署支撑完善后续 CD 的支持，并部署周边服务的底层服务支撑，提倡用户统一管理体系，独立完成整个项目落地的技术引导与支持。并提供后期的项目技术指导以及流程图说明。

2017.03 - 2017.07 **Docker 中国区站点落地**

项目介绍: 因拓展中国区市场，Docker 站点在国内落地

功能介绍: Docker 站点 Docs 站点中国区阿里云环境落地，以及对相应站点进行翻译支持。

功能实现: Docker 官方站点使用 Drupal 对内容进行了调整，并对 Docs 站点使用 **Jekyll** 进行了中文的实现，并架设了 **CI/CD** 部署环境支持，官方 Docs 使用 Google 内嵌搜索的支持，但是国内区域不支持 Google 的使用，所以使用 **Scrapy** 实现数据的爬取并通过 **Elasticsearch** 存储实现了整个搜索过程。并一定程度上编写了**自动化部署**脚本稳固部署流程的完成率。

2016.05 - 2016.07 **阿里云数据迁移青云**

项目介绍: 因阿里云不支持灵活的服务器资源利用，对现有的资源进行迁移至青云

功能模块: 由于业务是通过微信公众号来运营产品，所以后续服务通过 Docker 化来进行迁移

功能实现: 当时阿里云只支持按月付费的功能，对资源的使用存在不合理性，所以提出迁移青云的计划，业务迁移青云后，统一进行 Docker 化，并使用 consul 自动注册与发现的技术，来维护整合系统的服务支撑，并在业务推广时瞬间的大流量做好青云的服务器动态扩容，通过 Nginx Plus 的 DNS 获取资源机制来动态获取 Consul 资源池，实现整体业务的负载均衡及服务发现。

技能经验

- 1、 高质量的项目技术文档编写能力，可以指导新老员工企业文档编写能力以及规范，来完成企业内部的知识库体系。
- 2、 熟悉 **Docker**，即周边生态圈 **Mesos、Kubernetes、Rancher** 等系列产品，以及原生 **Swarm** 编排，可对 Linux 下所有应用定制容器化，并制定 CI/CD。熟悉 Docker 调优，熟悉远程 SSL/TLS 配置管理，熟悉 证书 管理，熟悉 Registry 凭证管理等。精通任何应用的容器化迁移部署方案，可配置编写任意维度的 Dockerfile 来辅助来辅助 CI/CD 流程。并通过一切皆容器的理念，方便快捷的使用容器部署、测试、备份等业务逻辑。
- 3、 熟悉 **Kubernetes** 生态，熟悉 RBAC 等策略的管理，熟悉 Kubernetes 部署迁移等回滚自动化部署，并开源 **Ansible** 方式部署 Kubernetes 最新版本部署（项目名: [slzcc/Crane](#) 通过 CNCF 一致性检测)项目支持: 版本热更新、CA 根证书更新、Etcd 根证书更新、添加 Node、添加

Master、添加 Etcd、添加 Add Ons 如 Istio、PrometheusOperator、IngressNginx、Helm 等服务，并且支持离线/在线部署，非二进制部署，与 Kubeadm 方式类似。熟悉 PrometheusOperator 维护即应用的 Exporter 设计。熟悉 Istio 生态的使用，有基于云或 IDC 落地方案。

- 4、熟悉 **MySQL** 日常维护，熟练部署及应用 **MySQL Proxy**、**MySQL Router**、**MySQL 主从**、**MySQL 集群**、**热备**等。对 MySQL 原理有一部分的了解。对 MySQL 性能及创建优化索引、创建分区，用户管理，事务等有一定的认知能力。对读取 Binlog、慢查询、通用日志有一定的管理能力。熟悉 PXC 部署，熟悉同步机制与管理。
- 5、熟悉 **Nginx** 及其动态获取上游池实现动态负载均衡策略(不通过容器编排)，熟悉 **OpenResty** 开发，熟练操作 **Lua** 语言编写定制化的网关应用，熟悉搭配单点或多点的 **PHP-FPM** 业务。熟悉 **Apache** 配合 **PHP** 处理逻辑业务。
- 6、熟悉 **Elasticsearch** 及其周边服务(Elastic Stack)的管理，可定制 mapping 及其动态添加字段，以及热备恢复 Pipeline 操作，可部署 **logspout/filbeat > kafka > logstash > es** 的日志管理架构，及其可视化操作。
- 7、熟悉 **Redis**，及其周边集群等应用的部署及维护，可实现自动化热备及回滚。
- 8、熟悉 **AD**，可以部署主从服务器，并管理服务器的日常管理及维护，可通过 **DNS** 策略管理员工个人上网行为。
- 9、熟悉 **Ansible**，可以通过高质量的 **Ansible Palybook** 编写对服务器日常或新增的业务管理，可编写部署 Kubernetes 等服务的集群部署逻辑代码，可实现通过 Ansible 控制集群生态。
- 10、熟悉 **OpenLDAP** 的日常管理维护，熟悉 OpenLDAP SSI/TLS 支持，熟悉 sudo 等协议的 shcema 规范，熟练操作主从部署以及备份恢复策略，熟悉索引创建规则。
- 11、熟悉 **Gitlab** 搭建，熟练操作常规配置以及接入 LDAP/SMTP/备份 等逻辑配置，可实现动态回滚测了。熟悉 Git workflow，可指导培训 DevOps Git 敏捷开发。熟悉 hook 策略，可以维护多个 Gitlab 共同协作使用，可做到大版本升级 Gitlab 实现用户无缝感知，指导 git workflow 或 Code Reviews 等相关经验。
- 12、熟悉 **Jenkins**，可熟悉编写 **DevOps** 流程管道，实现 CI/CD 的高质量业务，可熟练编写 **Pipeline** 模式的逻辑业务代码，通过高质量的模块 Job 组件，实现高容错低耦合度的管道逻辑。可通过接入 LDAP 实现角色级别的高维度的权限管理策略，可通过双备份安全的保证的 Jenkins 的配置安全，可灵活搭配 **Webhook** 处理所有业务。可结合 Kubernetes 最终做到开发人员不依赖运维人员而做到 DevOps 理念。
- 13、熟悉 **Gerrit** 部署维护，以及架设 Gerrit + Gitlab + Jenkins 的 Code Review 的通道。
- 14、熟悉 **Git**，熟练操作 Git 操作及其管理，可实现管理代码库历史文件清理等操作，对问题排查有一定的认知能力。
- 15、熟悉 **Python/Shell/Lua/Go/Vue** 语言，可通过 **Django** 实现 **WebSocket**、**Celery** 等策略编写高质量的业务，可编写 Scrapy 爬虫等分布式应用，可爬取知乎等大型门户网站资料，并结合 ES 实现高质量的搜索功能，可通过 Python 编写自动化运维脚本可配合 Ansible API 管理多节点服务等。使用 Go 整合多个 **Client-go** 实现 Api 统一调度, 使用 **Operator Framework** 想代替 Client-go 相关需求但发现架构整体有局限性，使用 **Vue2.0** 写过简历系统平台中间通过 Ptyhon 自学整理过自然语言处理做到简易处理语言的能力。
- 16、熟悉 **Etcd** 应用的多节点部署维护，支持 SSL/TLS。
- 17、熟悉 Java、**Tomcat** 应用部署，调优、集成 Docker、项目优化等方案有一定的经验。

- 18、 数据 Linux 系统的调优，系统网络、IO、内存排障，突发状况处理，高抗压能力。
- 19、 熟悉 Rabbitmq 多集群维护, 对优化以及使用上有一定的经验。
- 20、 熟悉 CDH5 生态，维护过 500+ 实例集群，维护多个 Hadoop 生态产品。

自我评价

热爱交流，懂得分享，拥有团队精神，帮助公司做好下一步的创新与推进。

DevOps 案例:

