

云原生 k8s 一面

- 1.自我介绍
- 2.读研还是直接工作?工作
- 3.开局做算法题,两数之和,哈希表轻松解决
- 4.Arrays.sort 使用了什么排序算法
- 5.解释下快排的过程
- 6.快排的空间复杂度+推导
- 7.除了快排还有哪些排序算法
- 8.排序算法的稳定性(如果本身有序,是否会打乱原来的相对位置,这个我答错了,然后这里面面试官告诉了我增设个定义)
- 9.基于第八点问我冒泡排序是否稳定
- 10.问我常用的数据结构,答了个 hashmap
- 11.如果让我实现一个 hasmap,会怎么做
- 12.进程和线程的关系和区别
- 13.进程和线程共享了资源,线程也自己独有一些资源,那么具体哪些资源是共享的,哪些资源是线程独有的
- 14.传输层的协议常用的有哪些
- 15.TCP 和 UDP 的区别
- 16.TCP 怎么做到可靠传输
- 17.16 点提到了滑动窗口,所以追问滑动窗口除了用在可靠传输的保证,还问了有什么作用,我答的是流量控制
- 18.追问滑动窗口流量控制怎么实现的,这里除了课本上的那种一开始两倍增长然后再线性增长,然后再减到一半,然后我追答了一个 bbr,所以 19 问追问 bbr
- 19.bbr 的过程
- 20.追问 bbr 相比于传统流量控制算法的优点
- 21.问 mysql 使用的时候有几种可用的索引方式(业务层面)
- 22.聚簇索引,非聚簇索引的区别
- 23.普通索引在实现的时候可以用哪些技术来实现
- 24.除了 innodb 还用过其他引擎吗(提了 myisam 和内存索引)
- 25.myisam 索引有没有什么地方用到的
- 26.应用在线上运行比较长时间后,sql 查询比较慢,怎么排查(这里我把 explain 里面的字段含义都答了一遍,因为提到了因为扫描函数估算错误而导致的索引错误选择而导致的效率低下,所以有了 27 的追问)
- 27.如果因为索引选择错误而导致了选择索引错误,那有什么办法修正呢,我回答了 analyse xxx 和在 sql 语句里面强制指定索引
- 28.问项目
- 29.为什么很多项目都选择 Spring?
- 30.为什么在项目中使用 ioc?(这里我答了交给 spring 管理之后,可以通过干预 Spring Bean 的生命周期来达到一个可扩展 解耦合的地步,由于答了 Spring 管理 bean 的生命周期,所以有了 31 问的追问)
- 31.Spring 的生命周期,(反正大概把源码那一套东西说了一下,大概答了 4 分钟,由于提到了 Spring 在 postBeforeInstiation 提供了给 bean aop 的机会,所以有了 32 问的追问)
- 32.Spring 自身有什么地方使用了 aop 么?这我答的是 @Transactional
- 33.@Transactional 在使用的时候有没有遇见过因为 aop 的缺陷而导致的不生效的问题,比如

一个类里,没加 `Transactional` 的方法,调用加了 `@Transactional` 注解的方法,可能生效也可能不生效,不生效的可能原因是什么,要求围绕 `aop` 的实现来答,这个我答的不好,面试官要我下去之后看看源码再思考一下

34.netty 有实际使用过吗

35.问 `epoll` 原理

36.问我 `gc` 调优经验,说了一个内存泄漏排查,还有上线容量估算

37.问我 `K8S` 了解什么

38.排查系统性能问题的时候,用什么思路 and 工具,`uptime top pidstat netstat vmstat dstat sar -n DEV` 能答的工具还有一些字段的含义都丢出去了,然后说了软中断 硬中断 `iowait` 进程之间 `wait` 过高的时候,排查的思路

39.`K8S` 了解多少,项目中用过吗

40.最近在学什么

41.有什么要问的

得出一个经验,面试的内容和难度可能会随着你的回答而变化,阿里的算法两数之和是真的舒服。

云原生 `k8s` 二面(2月19早上9点半面的,电话面)

1.你的学习经历?(这边说了看过的一些中间件的源码,然后最近看的源码)

2.你看过 `K8S` 源码吗?(这个答了没有)

3.你说说你懂的 `K8S` 原理,尽可能详细的?(这里答了 `K8S` 里面各种组件和大致原理,稍微漏了东西,不过漏掉东西是因为这个问题有点大,一时间漏了,后面在追问项目细节的时候他追问回来了)

4.`K8S` 调度 `pod` 到节点的原理(这里答了和 `nodeAffinity`、污点、资源占用和 `pod` 的资源需求相关的字段有关的一些东西,这里多答了一个 `label` 和 `selector` 有关的,后来想想好像不该答)

5.问项目

6.问我项目里的自定义集群怎么做的?(这里我答了 `openresty`,我那个项目是要对接另外一个开源产品,出于某些原因要把一些业务逻辑写在网关那)

7.你项目有用到 `K8S` 吗(没有)?然后追问了为什么不用?我答的因为我对接的那个开源产品是要修改他源码的认证流程的,我当时还不具备去看那个产品和 `K8S` 相关的源码,第一反应是不要去接 `K8S` 的

8.追问其他项目细节(细节我大概都对上了)

9.如果现在让你把那个项目用 `K8S` 来做的话你会怎么做?(因为一时间没 `get` 到面试官要问的点,就给了个不完美方案,然后就有了 10 11 的追问)

10.针对我项目问:接入 `K8S` 之后,网关的转发逻辑怎么处理(在每个 `pod` 启动的时候,在他的生命周期里对网关进行一个配置路由的过程)

11.然后问了如果每个用户都需要一个硬盘空间的话,应该用什么(这个就是面试官在追问我第三点没答出来的有状态服务和 `volume` 那套了,然后我才反应过来,然后就回答了在 `pod` 的 `template` 里面用 `volumeClaim` 申请存储,然后挂载到某个目录就行)

12.用户态和内核态的区别

13.容器了解吗(答的大概了解,不过我只会 `docker` 容器)

14.`docker` 容器实现资源隔离原理(答了 `Linux` 的 `cgroup` 和 `namespace` 技术)

15.`TCP` 和 `UDP` 的区别

16.`java` 进程和 `go` 进程应该怎么通信(这个我回答的是,如果能统一通信协议和序列化协议的

话,可以使用 `grpc`,然后就是 `http`,然后答了个 `socket`,还有同一台机器上 `ipc`)

17.做了道回溯题,比如下面这个矩阵,找出目标的字符串,ALBABA,返回 `true` 或 `false`,下面这个的话从 0 0 开始往下走然后一直往右走就是了

A B C D E

L B A B A

W D A D L

18.问我为什么学了 K8S(我回答的是,一开始我被他提供的一些特性所吸引就学了一下,然后后来看 Dubbo 源码的时候就很好奇,这些东西直接在 K8S 上做就好了,当时还迷惑了一段时间,后来慢慢学习才知道,dubbo3.0 就在干这事,所以最近就看起了 dubbo 和 k8s)

19.因为 18 问的说法,所以这一问追问了 dubbo 的什么东西可以直接在 k8s 做(一开始先说了服务自省的机制,这一点是和 SpringCloud 云原生统一的基础,服务注册和 pod 的生命周期结合,还有 API Client,或者是 DNS Client,还有和 istio 结合)

20.反问环节,你们部门是做什么的,这里面试官回答的是弄 K8S 的。(怪不得问了那么多 K8S 的东西)

云原生 k8s 三面(2 月 19 下午 4 点 22 打电话约我下午 5 点面试,稍微有点匆忙,视频面)

1.自我介绍(我就说了三句,他觉得有点简洁)

2.是怎么学习一些技术的

3.问项目?

4.问我项目里面为什么网关选择 `openresty(nginx+lua)`?(对接着开源产品的官方文档直接选了 `nginx`,然后后来业务逻辑复杂之后发现要在网关层加逻辑,就用了 `openresty`)

5.`nginx` 的缺点?(该配置需要 `reload`)

6.追问那如果现在让你技术选型你会怎么选?(我会选择开源产品本身支持的一个可配置的网络代理)

7.详细的追问了项目里的认证流程的细节

8.问了项目里其他人怎么分工

9.还有其他落地的项目吗?

10.问我的秒杀方案(`rocketmq` 事务+`redis` 自增自减+秒杀逻辑和 `openresty` 网关降级请求的处理)

11.追问那怎么保证 `redis` 的数据和数据库一致呢?(我说我的方案如果解耦的话是不需要保证强一致吧)

12.那像这样的话,秒杀和正常购物流程怎么共存呢,你如果是用这种方案去保证的话,是没有办法保证秒杀不影响正常购物吧(讲到这我就认输了,我自己就说可能设计的不周到,说实话如果不是没写过什么真正的项目,我是真的不想写秒杀的,我一直觉得秒杀是个雷区,不过面试官还是很好人的,没为难我,说我思路大概没错,尽可能保证了安全,但是真正的电商系统没有你们想象的那么简单,可能还会涉及跨地区调度,比如广州的人在广州这发货,北京那边的人从北京那边发货)。

13.为什么你觉得 `redis` 原子性的自增+`RocketMQ` 会比直接操作 `mysql` 的效率 high 呢?(这里把 `redis` 原子性自增自减都答了,还有把 `RocketMQ` 的架构答上去了,还有 `mysql` 的日志和 `rocketmq` 的日志,因为他感觉我一直没答到点上,在他一步一步诱导下,`mysql` 可重复读下的 `next-key lock` 也说出来了,不过这里他还有点惊讶我居然知道 `next-key lock`,可是似乎还不是他想要的答案,最后在进一步引导终于答出来了,`rocketmq` 投递消息到 `commitlog` 的时候,是单线程顺序写,而 `mysql` 要靠锁,其实我一开始回答了 `rocketmq` 利用了磁盘的顺序写的,我感觉我应该答到点上了)

14.之后打算工作还是读研,还有之后的学习计划(直接工作,研究 dubbo3.0 和 k8s 相关的一些东西)

15. 为什么学 k8s?和我二面答的差不多(先学了个大概,然后看了 dubbo 源码对为什么有些事不直接在 k8s 上做感到好奇,然后知道 dubbo3.0 是这么干之后就开始学)

16.你还投了阿里的其他部门吗?

17.kubectI 去更新一个 pod 的配置的时候,kubenetes 发生了什么(这个答的巨差,因为之前面试一直没问过,所以 k8s 源码我没怎么了解,虽然立过要看 k8s 源码的 flag,但是综合考虑还是没去看,不过如果有机会能进到这个部门,我肯定第一时间去看的)

18.然后就跟我说,因为是提前批嘛,系统不流通,到时候录入系统的时候只能录一个的,到时如果不注意容易进到自己不想进的部门。然后我就说我会和内推的师兄交流好?