开篇词 | 网络排查是工程师的必备能力

2022-01-12 杨胜辉

《网络排查案例课》 课程介绍 >

讲述：杨胜辉
时长 17:38 大小 16.16M

你好，我是杨胜辉，欢迎和我一起开启这场网络排查之旅。

为什么在这个时代，网络排查能力变得越来越重要了？

先和你简单介绍一下我自己，我目前是 eBay 中国卓越技术中心基础架构部门的运维经理，主要负责 eBay 全球的流量管理业务，推动 Kubernetes 在 eBay 流量管理场景中的落地。

我算是一个基础架构的老兵了，在 18 年的工作经历中，我实实在在地遇到过太多技术的疑难杂症。尤其是最近几年，随着微服务和云计算的不断普及，越来越多的系统从本地的单体服务，变成跨网络的分布式的微服务。那么随之而来，就是数不清的跟网络相关的问题。比如：

为什么我的应用在单体应用的时候很正常，拆分成微服务以后却时常超时、报错呢？

为什么我的带宽是足够的，但数据传输速度却很慢？

为什么我的应用偶尔会卡住，但又不是每次都这样？

为什么……

面对这么多问题，我们经常束手无策。有些同学自嘲是优秀的“SRE”（Server Restart Engineer），遇到问题先上“重启大法”，也许也能搞定不少问题。但是呢，根因依然是未知，即使问题暂时消失了，但不知道什么时候，它又会再次到来，然后再次重启……

所有这一切，都让我深深地感到：**我们的工程师，太需要网络排查方面的能力了。**

无论你是运维还是开发，无论是产品还是测试，都没法彻底回避网络问题。但是，因为大部分同学并不是网络出身，对于跟网络相关的问题，经常无从下手，或者事倍功半。在这方面，常见的困难就有：

1. 对于网络知识点好像是懂的，但一遇到实际的网络故障，就又不懂了，更不知道排查工作如何做起；
2. 网络知识是懂的，也能做一些排查，但是遇到艰深一点的问题时，就“卡壳”了，无法把排查工作进一步推动下去；
3. 对于网络排查有一些经验，比如前面说的“重启大法”，比如知道修改某个配置可能有用，但是因为不知道其原理，在三板斧不见效的时候，就别无他法；
4. 本身负责应用开发，也有兴趣自己来排查网络问题，但苦于没有这方面的背景和积累，排查工作很容易“熄火”，令人沮丧。

如果你符合 1，那么请不要失去信心，只要你踏实打好网络知识的基础，网络排查并没有那么艰难和神秘，通过学习，你也一样能掌握网络排查的能力。

如果你符合 2，那么请多一些坚持，因为你还需要继续打磨对网络关键知识点的深刻理解。同时，也需要强化自己在一些高级技巧，比如抓包分析上的能力。因为只有这样，你才能突破瓶颈，真正把问题搞透。

如果你符合 3，那么请多一些耐心，先不要过于依赖你过往的经验，而是沉下心来，剖析你的经验之所以有效的底层原理。这样，你既有经验也有理论，你的排查能力将又提升一个层级。

如果你符合 4，那么请不要气馁，你能成为应用方面的高手，那么一样有潜能成为网络排查方面的高手。事实上，要成为应用开发方面的大牛，网络也是极为重要的一关，搞不定网络，你的应用依然是跛脚的，无法达到最优的状态。

我的网络排查能力是如何成长起来的？

不过，虽然我现在能给很多人讲解网络排查的技巧，其实我以前也是一枚小白。

刚工作时，我在一家央企中国远洋集团工作，主要维护公司的几百台 Windows 服务器、邮件系统等，不太接触网络。但是有一件跟网络相关的事情，给我留下了很深的印象。

当时欧洲分公司的 Exchange 邮件系统遇到故障，我们从 Exchange 系统本身来排查，却怎么也找不到问题所在。团队奋战数天后，最终在网络组的协助下，才发现是网络层问题导致的。假如时光可以倒流，我带着现在的网络排查能力回到过去，是不是可以把几天变成几小时甚至几分钟呢？我忍不住会这样遐想。

也是从那次开始，我深刻地体会到：**无论是不是专职的网络工程师，我们都应该掌握好网络，只有这样才能把本职工作做到更好。**

后来我来到了 eBay，开始负责起 eBay 的整体的 Web 流量。我们知道，网络和应用的关系是十分错综复杂的，而正是因为要处理 eBay 这种海量级的访问场景，多年来，我在这方面就积累了很多经验和理解，逐步形成了对于“网络排查”这个宏大主题的一些自己的实操经验和方法论。

在 eBay 做了几年后，因为被云计算的丰富的技术性所吸引，我加入了一家公有云初创企业 UCloud，负责起售后技术服务。

如我所愿，由于客户和云平台的多样性，大部分网络场景和协议我都能遇到，也帮助我积累了鲜活的案例，以及接地气的排查经验，我也很享受这个过程，不断地丰富和打磨我的网络排查体系。我有一个习惯是每处理一个问题就开一个文件夹，里面放上相关的日志、

抓包文件等，当时记录有 500 个文件夹之多。这几年有一个新流行的说法叫“**刻意练习**”，而那段日子，也是印证了这个说法，我也确实从中收获良多。

就比如说，我们负责的一个客户是做健康硬件的，就是一个硬件的体重计，配合他们的手机 App，帮助消费者做身材管理。有一段时间，他们的 Nginx 上有大量的 499 报错。起初他们怀疑问题在我们云平台网络这里。为了“自证清白”，同时也是为了寻求真相，我花了好几天研究这个案子，最终证明这并不是云平台的问题，并且找到了根本性的解决办法。

有意思的是，在合作过程中，我跟另一家大厂的工程师建立了网络上的友谊。这也让我体会到：**技术本身就是技术人之间的共同语言，也是连接我们的桥梁**。

也是从那时候开始，我会在博客上分享自己的排查经验，包括从云计算公司回 eBay 后也一直在坚持，并逐渐得到了一些朋友的关注。能给身边的人以帮助，同时还发挥了自己的长处，这让我非常开心。我也特别想把这份收获，通过极客时间这么好的平台，分享给更多在这个技术行业里的其他同学。

也许你也对这个领域有兴趣，但苦于找不到合适的老师；也许你在实际工作中面临着类似的技术问题，但找不到好的方法，那我的这门课程可能对你会比较有帮助。倘若你通过学习我的课程，把很多知识点搞明白了，或者在实际工作中，确实能把网络问题给解决了，那我就再高兴不过了。

这门课程能给你带来什么？

那么现在你可能会问了，跟着我学习这门实战案例课程，都会有怎么样的收获呢？具体来说，有这么几点。

更扎实的对网络各层知识的理解

我举个例子，技术面试的时候，一个常见的问题就是：请介绍一下 TCP 握手的过程。这也算是面试八股文了，很多人背一下也能过，不过是否真的理解就难说了。那么在这门课程里面，我会把握手、挥手等各种细节都带到，而且会结合真实案例来讲解这些知识点，使你不仅知其然，而且知其所以然。下次回答这个问题的时候，恐怕面试官都要对你刮目相看了。

更广阔的排查视野

一般来说，网络问题用网络方面的工具排查，系统的问题用系统方面的工具排查，应用层也是如此。不过现实情况是，很多问题在一开始，并不能明确地归属到网络，还是系统，还是应用代码。所以作为排查者，我们不应该在一个预设的立场下展开排查工作，因为那样很容易就偏离了真相。

而在这门课里，很多案例也都是“看起来是 A，查着是 B，最后定位出来是 C”的情况，这其实也是最真实的现实了。有了更广阔的排查视野，你就不会因为不熟悉另外一个领域，就不得不放弃，或者“硬扛”了；有了更广阔的排查视野，你就不会局限在原有的一亩三分地里面，而是真正地把问题搞清楚，你的价值也不用说，会得到更大的认可。

更熟练的排查技术

很多同学其实也用过 tcpdump 和 Wireshark，包括我做面试的时候，不少候选人也都能就 tcpdump 和 Wireshark 侃侃而谈，但是追问细节的时候就语焉不详了。以 Wireshark 为例，它确实提供了相当多的信息提示，比如丢包和重传都会用不同的颜色跟正常的数据包区分开。

不过，为什么重传、为什么丢包，这些问题的答案，Wireshark 会告诉你吗？不会。是 Wireshark 不够强吗？也不是。

因为每个组织、每个应用的情况都相差极大，只有靠你自己平时积累的经验，以及结合具体的网络和应用环境，才能获得最符合你这个特定案例的答案。而这门课里就有很多这样的例子，特别是我做公有云服务的时候的多样化的案例经验，都是我个人的独家经验，我也相信一定能给你很多启发。

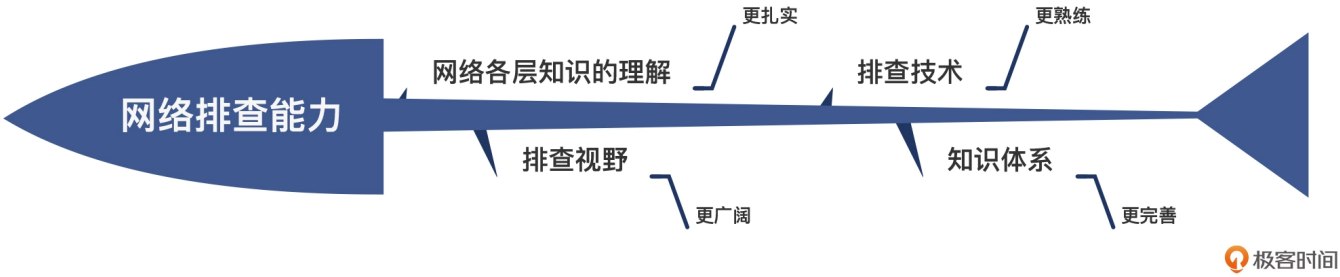
更完善的知识体系

如果你是做开发的，现在各种软件库和框架都极大地提升了我们的开发效率，但同时也屏蔽了很多底层细节。举个例子，应用层的“connection reset by peer”的报错，又如何跟底层网络的实际情况结合起来呢？应用层本身并不能回答这个问题。

那么通过这门课程，我们将解开很多的技术点的层层包裹，端详它们本来的模样，真正地理解这些技术设计者的初心。这也会帮助我们更好地理解这些技术的来龙去脉，反过来也能帮助我们更好地完成上层业务。

如果你是做运维的，那么这个作用会更加明显。对底层原理的掌握，将会极大地提升运维工作的质量，无论是平时工作时候的“气定神闲”，还是故障危机时候的“英勇相救”，都将是你通过这门课程获得的能力。

所以呢，这次我的网络排查课，形式会跟其他课程有所不同。不是单纯地讲理论或者讲工具，而是围绕**案例**这个核心，展开我们的排查过程，并会聚焦到工具的使用，以及深入到关键技术点的分析上。



这门课程是怎么安排的？

那么，课程具体是怎么设置的呢？我把这门课分成了五大模块，分别是：

预习篇

在这个部分，我会从网络分层模型出发，来带你了解、学习并掌握整个网络世界的大体层次，和每层的相关工具。然后带你进入抓包分析这个技术殿堂，了解它的历史和现在，以及初步的使用方法。通过对分层模型和每层工具的理解，以及对抓包分析技术的认识，你就能打下网络排查的底层基础，为后续的学习铺平道路。

实战一：TCP 真实案例揭秘篇

接下来，我们就要进入真正的实战了。

在这个模块里，我会从各种跟 TCP 相关的实际案例出发，来带你了解、学习并掌握 TCP 这个精密仪器的核心技术，包括传输性能的关键点、TCP 重传的原因和对策、拥塞的优化策略、TCP 保活机制等。我会通过一个个真实的案例，帮助你达成对这些核心知识点的真正理解，最后能够融会贯通，再也不怵 TCP 相关的难题。

实战二：应用层真实案例揭秘篇

在理解了 TCP 这部重要篇章之后，网络排查的核心知识，你就掌握了快一半了。不过，还有另外一个同等重量级的篇章等待你去学习，它就是应用层网络排查。其中，我们还需要补齐一个重大的短板：应用和网络之间的桥梁。

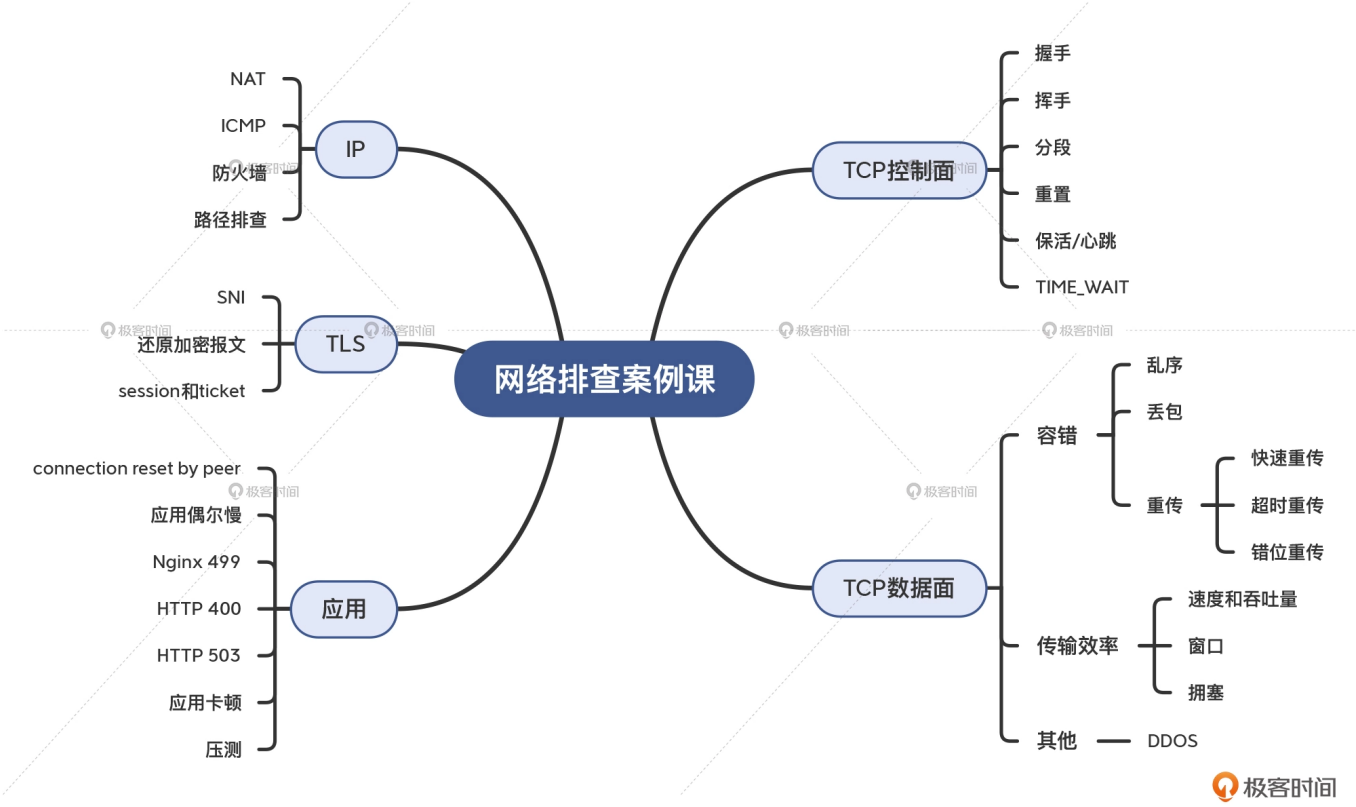
那么在这个模块里，我会从一个个典型的应用层网络排查案例出发，来带你了解、学习并掌握如何排查应用层的网络问题，让你通过对抓包分析这个核心技术在应用层的运用，搭建起应用和网络之间的“桥梁”。学完这个部分后，你在应对应用层的网络问题时就会成竹在胸了。

实战三：不用抓包就能做的网络排查篇

最后，我们还需要学习抓包分析之外的其他网络排查方法，因为掌握抓包分析相当于掌握了网络排查的主干，但还需要补充枝叶，这样我们的网络排查技能树才足够完整。所以在这个模块里，依然是从实际案例出发，来带你了解、学习并掌握这些工具的背后原理、使用场景、个人总结，让你能够通过对原理和实践经验的理解，达成融会贯通的目的。

总结篇

在学完前四个模块的具体技术之后，现在我们应该来一次总结了。所以在最后，我想再带你整体沉淀升华一下，一起把前面学习过的网络知识、抓包分析技术、所有其他的网络工具的技巧复习一遍，把它们打碎后，再次拼接在一起，形成你自己的技术体系。这样，你不仅可以学习到我的经验，还能够转化为你自己的理解，从而实现真正突破网络排查瓶颈的最终目标。



最后，我还想说的是，网络问题的排查过程，其实就像读一本侦探小说一样，充满了神秘感和吸引力。当你掌握了网络排查技术之后，你在遇到问题的那一刻，就不会再像过去那样想要逃避，反而会像猎人遇到猎物一样兴奋，很想一试身手，最终把案件调查彻底，水落石出。对于这样一个美妙的状态，你是不是很期待呢？

从这个角度上说，你甚至可以把课程当故事来看，不过你一样可以收获到知识，经历我曾经历过的起伏，从一个胡同到一个旮旯，兜兜转转，找到灯火阑珊处。我也许做不到让你“衣带渐宽”，但我愿你“终不悔”。共勉。

在这里，我想跟你一起立一个 flag：**你可以每节课后在留言区进行提问和交流，我也会及时回复你，哪怕只是打个卡，等两个月后你学完全部课程，一定会有很大的收获。**

最后的最后，我想说：

二十多讲的课程，就是我们有二十多次的相遇，而随着课程的推进，我们或许还会有更多的交集。我要感谢你决定花这么多时间，来学习这门课程。同时我也建议你，感谢下你自己，因为你愿意花时间来这个课程学习，因为你想做更好的自己。只要坚持下去，你一定可以做到！

下一篇 01 | 网络模型和工具：网络为什么要分层？

精选留言 (17)

写留言

01

2022-01-12

老师你好，我们今天线上遇到一个http 503 的issue，提示service unviable的问题，请求过程是这样的，内网机器通过proxy代理服务器去call另外的内网机器，大概有可能是哪里的问题呢？

proxy server和另外的内网机器我们都没权限查看。。。

展开

作者回复: 嗯，这也是典型问题场景：应用层有报错，但我们并不知道网络上具体发生了什么，更无法知道这些网络行为对应的技术原理和对策是什么。这也是我在刚结束的直播课里，给大家提到的网络排查的两大核心难点之一：应用症状和网络现象之间的鸿沟。另外一个核心难点，是wireshark等工具提示跟技术原理之间的鸿沟。只有跨过了这两道鸿沟，我们才真正能把网络排查这门技术给做起来。

所以如果要彻底查清楚，是需要抓包后做分析的，过程不是一两句能说清楚，我在课程里会重点讨论这方面的思路和方法，你可以期待一下：）

关于http报错代码，也可以从协议规范出发，初步判断问题在哪里。比如这个503，意思是service unavailable，也就是这个proxy想要找后端的机器，但是没有找到。这个原因也有多种，比如proxy到后端机器在网络上就不可达，或者健康检查发现没有可用的机器，等等。

你可以让有权限的同事帮你到proxy server上抓取网络报文，要想办法在抓取期间，重现503问题。然后你就可以借助我后面课程里的方法，进行分析了。

5

分清云淡
2022-01-13

https://plantegg.github.io/categories/network/ 看完这些网络排查案例相信对大家学习这门课程很有热情的 ，雷锋，不用谢

作者回复: 嗯 也是实际案例，做案例真的是成长最快的方式之一

4

x
2022-01-12

我一般是本机telnet 0.0.0.0 端口和telnet 127.0.0.1 端口.
然后外面 telnet ip 端口, 这样排查问题, 看通不通

作者回复: 嗯多数情况下可以这么做，不过有个前提：监听这个端口的程序，申请的socket地址是统配地址，也就是0.0.0.0。如果程序显式的声明其监听地址为某个特定ip，那么这个telnet 127.0.0.0 port的做法就会失败，但不表面这个端口没在监听。
其实你可以用netstat -ant | grep 端口号，看看这个端口的状态是不是LISTEN，以及监听的地址是0.0.0.0还是什么。 或者lsof -i:端口号，也可以看到。不过要注意下，netstat不需要用sudo权限，但lsof -i需要sudo。

共 2 条评论 >

2

GAC·DU
2022-01-12

一个线上服务出现网络故障，SRE需要一分钟，而排查需要时间未知，如何选择？

作者回复: 一般对于明确的故障，特别是影响到业务可用性的故障，首先是做恢复，而不是排查。先通过切换到健康集群或者其他数据中心的方式，把业务尽快恢复。之后再做排查，或者两者同时做。但不应该把恢复放在后面哦

共 2 条评论 >

2

Horizon_carry
2022-01-12

打卡学习，大哥，跨境限制抓包看他三次握手里面ttl时间与64的大小，跨境在这里限制了，可以这样理解吗

作者回复: 你是指great wall吗 简单回答你，是的

1

晴天

2022-01-12

工作中遇到connect timeout和read timeout，总有一种无助感，希望学完课程能突破

作者回复: 好的，这两个timeout也都是常见的timeout。一个是连接超时，一个是读取（等待数据）超时。我在TCP实战篇里都会讲到的：）

1

姜姜

2022-01-12

课程有没有对容器网络问题排查的讲解？

作者回复: 嗯 直播课的时候有同学提到了，我后续会做加餐，安排这方面的案例介绍

共 2 条评论 >

1

includestdio.h

2022-01-12

打卡，按时学习，按时吃饭~

作者回复: 我也打卡，按时回复留言：）

1

Howe

2022-01-14

看到connection reset by peer立马购买哈哈哈，市面上跟网络有关的资料都是理论的，没有提供实战，这是我一直找的资料，期待大佬的分享。
大佬在分享时能不能也教一下如何去制造问题，然后如何排查，再讲解决方法？

作者回复: 欢迎来学习：）你说的确实是非常典型的痛点：理论好像能看明白，遇到问题就糊涂了。我在直播课里也提到了两大鸿沟：

1. 应用现象和网络现象之间的鸿沟
2. 工具提示和协议理解之间的鸿沟

很多同学还没完全意识到这两个鸿沟的存在，以及如何跨过它们。所以这个课程的核心或者说精髓，就是教你如何把这两个鸿沟给填平，从而真正的突破网络排查能力的瓶颈。
只要坚持学习，有疑问可以这里留言，我一定会回复你。

关于你提到的知道如何去制造这种问题的前提，还是要有能力去解决这种问题。我的经验是，很多问题在解决之前都是谜，做的过程就很有趣味。
当然制造问题，在02课已经有了，我用iptables模拟了服务端静默丢弃和拒绝这两种场景，然后“假装不知道”，通过客户端侧的抓包展开分析。
在实战二的应用层真实案例解密篇，都是典型的应用层问题。学到每节课的后面，你知道了谜底，也就知道了如何制造这个问题，并不难：）

记事本
2022-01-14

排查问题与解决问题，也是体现价值的时候呀

作者回复: 嗯 怕的不是正常，怕的是不正常：）日常调试、问题处理，都离不开排查能力。

Frank
2022-01-14

期待解决我网络中遇到的问题

作者回复: 嗯 预习篇部分主要是复习和打基础，要照顾到不同阶段的同学。实战篇我们会上真实案例，这个都跟咱们日常工作密切相关，希望能给大家启发。

大飞守角
2022-01-13

打卡记录

作者回复: 加油：）

ican_只会0到9

2022-01-13

拜读下，多实战

作者回复: 一起坚持和加油！：）


2022-01-13

打卡学习

作者回复: 加油：）


Chao

2022-01-13

虽然是开发，每次都很享受排查网络问题的过程。

作者回复: 没错，网络排查是很有趣味的一件事，相信你越享受这个过程，提高也就越快：）


Pantheon

2022-01-12

牛逼哈拉少

作者回复: 谢谢支持！


kerry

2022-01-12

期待很久了，终于上线了，打卡。

作者回复: 谢谢支持！