

褚霸、德哥、张亮、吴晟等近10位来自阿里、京东等知名企业的专家
高度评价并推荐



HikariCP In Action

HikariCP

数据库连接池实战

朱政科◎著



机械工业出版社
China Machine Press

HikariCP In Action

HikariCP

数据库连接池实战

朱政科◎著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

HikariCP 数据库连接池实战 / 朱政科著. —北京: 机械工业出版社, 2019.6

ISBN 978-7-111-63321-1

I.H… II.朱… III.数据库系统 IV.TP311.13

中国版本图书馆 CIP 数据核字 (2019) 第 160892 号

HikariCP 数据库连接池实战

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 孙海亮

责任校对: 李秋荣

印 刷: 三河市宏图印务有限公司

版 次: 2019 年 8 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 22.5

书 号: ISBN 978-7-111-63321-1

定 价: 99.00 元

客服电话: (010) 88361066 88379833 68326294

投稿热线: (010) 88379604

华章网站: www.hzbook.com

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

Praise 赞誉

数据库连接池是程序员接触最多但又最容易忽略的重要技术组件，并不是随意找个数据库连接池就可以让应用发挥最大功效，我们需要从性能、扩展性、连接池管理、监控及社区的活跃度等方面进行甄别。本书不仅对市面上常见的连接池组件进行了全方位比较和分析，还以实战的角度深入介绍了高性能 HikariCP 连接池的使用、原理与维护。政科是我多年的好友，他在技术上面严谨、执着和一丝不苟，传承的是一种工匠精神，相信读者通过本书会受益匪浅。

程超

《高可用可伸缩微服务架构》联合作者

《深入分布式缓存》联合作者

本书从 HikariCP 入手，串起点、线、面，行文生动，内容翔实丰满。本书以一个性能相关案例开篇，接着比较了各大数据库连接池，最后对 HikariCP 配置、性能揭秘、监控等多方面展开描述。本书较好地诠释了从原理到实践这一思想，对于使用 HikariCP 的朋友可以成为居家旅行必备，对于想了解整体池化技术或者性能优化的朋友，亦有他山之石之效。

于君泽

《深入分布式缓存》联合作者

公众号《技术琐话》运营者

HikariCP 作为 Spring Boot 2.0 的默认数据库连接池，实现了从不为人知到一日爆红。乘着 Spring Boot 这艘大船，其必定会成为 Java 开发的主流数据库连接池。

本书通过庖丁解牛式的技术分解，对 JDBC 和数据库连接池涉及的技术实现细节及一些常见问题进行了梳理。通过阅读本书可以有更深入、系统地掌握 Java 数据库开发技术。

构建一个高性能的网络应用需要从网络协议层、应用协议层进行全链路推敲，另外连接

池的设计和线程工作模型的设计是关键点。数据库连接池麻雀虽小五脏俱全。通过本书可学习一个优秀的数据库连接池产品的使用方法及其相关的技术，这是掌握高性能网络开发相关技术的一种非常好的学习方式。

随着开源软件的成熟，作为淘宝去 IOE[⊖]后处理海量数据请求的最重要技术——分库分表技术不再是大型互联网公司的专享技术。如果你正遇到大数据量导致的数据库性能瓶颈，或者希望到大型互联网公司去从事相关工作，那么“自研集成 HikariCP 和 Sharding-JDBC 数据库中间件”这一节应重点关注。

本书不仅是一本告诉读者如何使用 HikariCP 数据库连接池的入门书籍，更是一本围绕数据库连接池相关技术展开的高性能 Java 开发技术实战书，任何一个对技术有好奇心的 Java 开发者都可以从本书中收获良多。

杨俊

阿里技术专家

2014 年阿里中间件性能挑战赛 JAVA 组第二名

资源的复用能够提升资源利用率，降低成本，其中数据库连接池的作用便是对数据库连接资源进行复用。对数据库连接池原理的了解及使用，基本上已经成为软件从业人员必备技能。目前市面上已经有很多数据库连接池的组件，在数据库连接池红海市场中，HikariCP 作为后起之秀之所以能够实现浴血突围，是因为其有非常多的独到之处。本书全面描述了 HikariCP 的独到设计，在软件的设计和性能优化等方面会给你打开一扇新的窗口。

何涛

网联架构师

第一次看到本书的名字时，就在想连接池有啥可讲的，我天天都在使用。当我翻开本书阅读时却被作者讲述的阿里中间件实战经验深深吸引了。书中的案例个个都是鲜活的高并发案例，而支撑这些高并发的就是数据库连接池性能。在本书中，作者从 Linux 内核网络参数调优和 TCP 原理入手，通过 JDBC 剖析、SPI 实战、动态代理与字节码技术等，由浅入深地介绍了 HikariCP “简单、极致、可靠”的设计哲学。

杨彪

《分布式服务架构：原理、设计与实战》作者

《可伸缩服务架构：框架与中间件》作者

原蚂蚁金服专家，高级架构师

⊖ 淘宝去 IOE 就是去掉 IBM 的小型机、Oracle 数据库、EMC 存储设备，代之以在开源软件基础上开发的系统。

为什么要写这本书

2017年我开始从事数据库连接池中间件的研发和架构工作，并在公司内部写了一系列数据库连接池 HikariCP 源码解析和实战的文章。在老板张聪的建议下，我申请了微信公众号并坚持分享技术，公众号的初心是：“写的内容要有人阅读，这样才会引发互动交流；有朋友就不会觉得孤独，而且互相探讨也有助于自身提高”。

我从事的底层数据库中间件采用了 HikariCP 这款数据库连接池，这我和 HikariCP 结下了不解之缘。非常幸运的是，在长达近一年的时间里，每天都有大量的业务开发人员向我咨询数据库连接池的问题，最多的时候可以达到十多位同学排队咨询，使我颇有老中医开堂坐诊的感觉。我是一个细心的人，大家的疑问和最后的解决方式都会一一记录下来，我也会协助网友以及其他公司的技术人员分析定位数据库连接池问题，帮助他们的项目顺利上线我也同样感到喜悦，这段经历是我技术之路上宝贵的财富。

我毕业于哈工大，哈工大的校训是“规格严格，功夫到家”，母校影响着我的做事风格，我是一个追求极致的人。但每当我协助业务开发人员排查问题时，却常常发现糟糕的事情。比如很多业务项目里充斥着 HikariCP、Druid、BoneCP、c3p0、Tomcat JDBC 等多种数据库连接池依赖项，而开发人员却并不知道自己究竟使用的是哪款数据库连接池，而将依赖做到精简是中间件岗位最基本的素养。美国童子军有一条“让营地比你来时更干净”的军规，这也是《Clean Code》这本经典书强烈建议的。比如，我发现很多业务开发同学咨询的问题如果总结下来是可以分门别类的，但是部分人解决了问题却不会分享、传播、传承，进而造成其他人接二连三地因为类似的问题再度陷入费时费力的排查之中；再比如，很多技术人员在学习使用数据库连接池之后想对更深入的技术进行探索，然而市面上却没有系统、深入地介绍数据库连接池的书，导致他们不得不投入到其他项目中从而没有实现深入了解数据库连接池的梦。

因此，我决定撰写一本相关的图书。这是一本介绍数据库连接池的书，由于这本书中更多凝结的是我多年的实践经验、我对于社区数据库连接池上千条问题讨论的研读等，所以这本书称为“HikariCP 数据库连接池实战”。

Hikari 在日语中的意思是光，代表了极致。极致无处不在，极致可以小到一次代码提交，也可以大到整个研发流程；极致可以小到一行 JDK 底层代码，也可以大到整个电商系统架构；极致可以运用在技术选型，也可以运用在团队管理。极致也是分场景的，有些是大规模高并发的要求，有些原则上不到一定的数量级不能用。极致的追求，不仅需要的基础技术原理有足够深入的了解，还需要配套的中间件支持，相关的业务协助改造。极致之路，没有银弹，不要想一招毙命，一口吃成个胖子，而是要审时度势，根据当前的现状来裁定。在和 HikariCP 打交道的期间，我发现这是数据库连接池界一款以追求极致为初心的艺术作品。不论是从相关文献资料、问题讨论、源码更新，还是从 HikariCP 多年来不辞辛劳一步一个脚印的社区贡献，我都能深深感受到 HikariCP 在极致之路上一次又一次地超越自我，以追求卓越的状态面对未来的技术发展方向。

光不但是极致，更是希望。这些年，尤其是写这本书的期间，我承受了不少的委屈，也遭遇了很多的坎坷，经历了生离，也经历了死别。然而我坚持将自己的梦实现，希望能活成一束光，能够温暖别人，也能够照亮别人，这才是真正的正能量。《HikariCP 数据库连接池实战》，是我的孩子，也是我梦想的实现，但我希望它不再是属于我一个人的梦，更希望能够将 HikariCP 的光亮照进每一名读者心中。

最后送大家一段话：在一无所有时，也请坚持做一名追光者，愿大家都能找到自己内心深处的光，接近并成为它，去发光发亮，成为别人眼中的和照亮别人的光，用更亮的光芒照亮世界的每一个角落。你就是太阳，蒸发了彷徨，有你的地方就是天堂。

读者对象

本书适合所有 Java 程序员阅读，尤其适合以下读者：

- Java 初中级开发者。
- 系统架构师。
- 中间件开发者。
- Java 多线程、并发开发者。
- 使用 HikariCP 参与工作流程的公司与集体。
- 开设 Java 相关课程的大专院校师生。
- 其他对数据库连接池技术感兴趣的人员。

本书特色

在本书写作的过程中，我尽量避免让这本书成为“代码书”或者“概念书”，用自己的实战经验和大量文献涉猎将本书凝结为一本高内涵、高价值的实战书籍。这本书准确地说写了两年的时间，第一年我在数据库连接池方面进行实践并形成零散的文字记录，第二年较为体系化地将各种经验、总结、记录梳理成册。本书的目录结构也经过了好几个版本的迭代修改，旨在让读者用最短的时间循序渐进地学完相关知识点。本书的篇幅也做过很多的压缩、删减，目的是聚焦核心内容，像“瑞士军刀”一样短小锋利，可以帮助读者快速学习并解决问题。

本书以阿里大型消息中间件的无池化 TCP 调优开篇，这也是我在阿里晋升答辩的内容之一，在扩展性能调优知识的同时顺带引出 Java 平台下性能调优的利器——数据库连接池。这样的开篇可以让读者真真切切感受到数据库连接池出现的必要性。接着，从数据库连接池出现的背景、原因入手，循序渐进地为大家盘点了主流的数据库连接池并进行了对比，这也弥补了市面上没有一本书体系化介绍各种数据库连接池的遗憾。

当然，这还只是“前戏”，本书还剖析了 HikariCP 这款数据库连接池的设计理念和原理，并对其重点功能和模块逐一进行了讲解。在分析之余，这本书还对 HikariCP 的关联技术、具体实战、扩展内容等进行了补充说明，旨在帮助读者举一反三、触类旁通。本书最后以一个疑难问题为例，通过排查问题，帮助大家一起对 HikariCP 的相关内容回顾和展望，以求融会贯通，温故而知新。

如何阅读本书

本书分为四大部分：

第一部分为准备篇，以阿里连接池极致优化案例引入，帮助读者了解数据库连接池的背景、原理，并全面了解主流的数据库连接池，包含本书的主角 HikariCP。

第二部分为基础篇，着重讲解 HikariCP 参数配置、JDBC 等核心模块。这些内容都是 HikariCP 重中之重。除了原理性介绍以外，为了帮助读者夯实数据库连接池的基础，本书还特地对技术细节、设计理念、实战扩展等进行了补充。

第三部分为原理篇，首先，全方位剖析了 HikariCP 性能极致的原因，这些都会成为读者面试、工作中的谈资；其次，将连接原理、参数原理、动态代理与字节码技术这三个维度作为切入点，对 HikariCP 进行源码级别的解析，这样全方位地进行剖析不同于单视角的源码解析，更能将 HikariCP 的原理立体化地展示在读者面前。

第四部分为实战篇，这是本书精华的部分。HikariCP 的监控、扩展技术、常见问题等都

是读者工作中较为常用的知识点，这些内容大多来自我实战经验的积累及大量文献资料的收集整理。本部分可以独立阅读，作为案头的工具书使用。

勘误和支持

由于作者的水平有限，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。为此，我在 GitHub 网站上专门新建了一个 issue 项目（<https://github.com/CharlesMaster/Hikari/issues>）。你可以将书中的错误、疑问、改进建议以 GitHub Issue 的形式发布在 Bug 勘误表页面中，我将尽量在线上为读者提供最满意的解答。当然，更希望大家可以通过 issue 对 HikariCP 展开讨论，互相切磋和共同成长。如果你有更多的宝贵意见，也欢迎发送邮件至邮箱 just_love_zzk@126.com，期待能够得到你的真挚反馈。

致谢

首先要感谢伟大的 HikariCP 的作者 Brett Wooldridge，他开创了一款影响我整个人生的软件。

感谢 Apache ShardingSphere PPMC 张亮、Apache SkyWalking PPMC 吴晟在我从事数据库连接池工作过程中给予的指导与帮助。感谢奇点云应用研发总监、首席架构师张聪，其作为引路人将我领入了数据库连接池领域，同时也是在他的支持与鼓励下才促成了这本书的撰写与出版。

感谢技术圈子中每一位充满创意和活力的朋友——何涛、程超、右军、李艳鹏、杨俊，以及名单之外的更多朋友，感谢你们长期对我的帮助与信任。

感谢笙笙对我写本书的支持。

感谢机械工业出版社华章公司的编辑杨福川老师和孙海亮老师，在这一年多的时间中始终支持我的写作，你们的鼓励和帮助引导我顺利完成全部书稿。

感谢王玉婵老师为本书付出的劳动。

最后感谢我的爸爸、妈妈、小姨、姨夫、爷爷、奶奶、外公、外婆，感谢你们将我培养成人，并时时刻刻为我灌输着信心和力量！

谨以此书献给我最亲爱的家人，以及众多热爱 HikariCP 的朋友们！

Contents 目 录

赞誉
前言

第一篇 准备篇

第1章 阿里中间件实战，第一个案例	2
1.1 物联网 MQTT 单机压测 130 万参数调优	2
1.2 阿里中间件 TCP 四次挥手性能调优实战	4
1.2.1 亿级消息网关 Rowan 架构	4
1.2.2 人脸识别服务：异曲同工的架构	6
1.2.3 “双十一大促”全链路压测发现 TCP 问题	9
1.2.4 Linux 内核网络参数调优	11
1.2.5 Linux TCP 参数调优	12
1.2.6 一行代码大幅提升 QPS	13
1.3 技术驱动业务，结果为导向	16
1.4 本章小结	17

第2章 数据库连接池江湖	19
2.1 为什么使用数据库连接池	19
2.2 数据库连接池原理	22
2.3 数据库连接池百晓生《兵器谱》	23
2.3.1 c3p0	24
2.3.2 Proxool	29
2.3.3 XAPool	30
2.3.4 DBCP	32
2.3.5 Tomcat JDBC Pool	34
2.3.6 BoneCP	36
2.3.7 Druid	38
2.4 主流数据库连接池对比	41
2.4.1 性能对比	41
2.4.2 代码复杂度	42
2.4.3 功能对比	42
2.4.4 数据库中断	43
2.5 本章小结	46
第3章 初识HikariCP	47
3.1 Hikari 背景、特色及前景	47
3.2 SpringBoot 数据库连接池加载顺序剖析	51

3.3	SpringBoot 整合 HikariCP 实战	53
3.3.1	Spring Initializr	53
3.3.2	添加 HikariCP 依赖	55
3.3.3	JdbcTemplate	56
3.3.4	Database Initialization	57
3.3.5	启动运行	59
3.4	本章小结	60

第二篇 基础篇

第4章	HikariCP参数配置	64
4.1	校时	64
4.2	HikariCP 配置手册	68
4.2.1	必需配置	69
4.2.2	非必需配置	72
4.3	HikariCP 连接池配置多大合适	78
4.4	Fixed Pool Design 思想	81
4.5	MySQL 高性能配置	82
4.6	Hibernate 配置	83
4.7	JNDI 配置	85
4.8	本章小结	86
第5章	HikariCP与JDBC	87
5.1	HikariCP JDBC Logging	87
5.2	JDBC	90
5.2.1	JDBC 定义	91
5.2.2	JDBC 实战案例	93
5.2.3	JDBC 剖析	95
5.2.4	PreparedStatement 和 Statement	100
5.3	JDBC 与 SPI	101

5.3.1	SPI 简介	101
5.3.2	SPI 实战案例	102
5.3.3	JDBC 的 SPI 机制	104
5.3.4	SPI 高级实战：基于 Dubbo 的分布式日志链路 TraceID 追踪	106
5.4	拓展：线程池技术	109
5.4.1	MySQL 线程池简介	110
5.4.2	MySQL 线程池技术内幕	111
5.4.3	MySQL 线程池实战	114
5.5	本章小结	117

第三篇 原理篇

第6章	HikariCP性能揭秘	120
6.1	华山论剑	120
6.2	第三方测评	124
6.2.1	环境配置	125
6.2.2	获取关闭连接性能测试	125
6.2.3	查询一条语句性能测试	127
6.2.4	psCache 性能对比	128
6.2.5	测试结论	129
6.3	HikariCP 为什么这么快	129
6.3.1	精简字节码	130
6.3.2	FastList	132
6.3.3	ConcurrentBag	134
6.4	本章小结	147
第7章	HikariCP连接原理	148
7.1	获取连接	148
7.2	归还连接	155

7.3	关闭连接	158	10.2	为什么需要数据库连接池 监控	228
7.4	生成连接	163	10.3	HikariCP 监控指标	231
7.5	扩展阅读: DCL 为什么要加 volatile	166	10.3.1	hikaricp_pending_threads	233
7.6	扩展阅读: Log4j2 为何性能 优秀	169	10.3.2	hikaricp_connection_acquired_ nanos	234
7.7	本章小结	173	10.3.3	hikaricp_idle_connections	235
第8章	HikariCP参数源码解析	174	10.3.4	hikaricp_active_connections	235
8.1	SpringBoot 2.x HikariCP 参数 加载原理	174	10.3.5	hikaricp_connection_usage_ millis	235
8.2	allowPoolSuspension	181	10.3.6	hikaricp_connection_timeout_ total	235
8.3	validationTimeout	187	10.3.7	hikaricp_connection_creation_ millis	236
8.4	leakDetectionThreshold	189	10.4	HikariCP 监控指标实战	236
8.5	本章小结	202	10.4.1	连接风暴	236
第9章	HikariCP动态代理与字节码 技术	203	10.4.2	慢 SQL	237
9.1	HikariCP 字节码工程	203	10.4.3	监控指标与参数配置	237
9.1.1	HikariCP 的字节码技术	204	10.5	SpringBoot 2.0 暴露 HikariCP Metrics	238
9.1.2	代理技术原理	209	10.6	SpringBoot 2.0 监控 HikariCP JMX	241
9.1.3	HikariCP JIT 方法内联优化	213	10.7	微服务架构下的监控平台 选型	246
9.2	JMH 基准测试	216	10.8	本章小结	250
9.2.1	JMH 常用注解	216	第11章	从HikariCP Metrics谈微服务 监控架构实战	252
9.2.2	JMH 实战案例 Orika	219	11.1	HikariCP Metrics	253
9.3	本章小结	223	11.2	Micrometer	258
第四篇 实战篇					
第10章	HikariCP监控实战	226			
10.1	监控体系层次	226			

11.3	SpringBoot 2.x 自定义埋点 实战	260	12.5	本章小结	311
11.3.1	引入依赖	260	第13章 HikariCP常见问题		312
11.3.2	配置 application. properties	261	13.1	HikariCP 故障分析技巧	312
11.3.3	注册 Metrics	261	13.2	leakDetectionThreshold 参数解决 Spark/Scala 连接池泄露	318
11.3.4	支付接口埋点统计 总次数	262	13.3	详解 JDBC 超时	320
11.3.5	查看指标数据	262	13.4	快速恢复	323
11.4	SpringBoot 2.x 集成 Micrometer 源码解析	266	13.5	Oracle Connection Reset 问题	327
11.5	SpringBoot 1.5.x 自定义埋点 实战	269	13.6	HikariCP 关闭连接的 5 种 情况	329
11.5.1	引入 Micrometer 依赖	269	13.7	如何获取 HikariDataSource 的 active connection	330
11.5.2	创建 Bean 并注册 Metrics 指标	270	13.8	如何对 HikariCP 配置文件中的服 务器名、用户名、密码加密	330
11.5.3	在 Controller 中进行支付业务 埋点	271	13.9	HikariCP 神奇的配置 dataSourceProperties	330
11.6	监控架构重点	272	13.10	如何获取 HikariCP 连接池中的 原始连接	332
11.6.1	Prometheus	272	13.11	HikariCP 并不是万能工具	332
11.6.2	Grafana	275	13.12	本章小结	333
11.7	本章小结	278	第14章 HikariCP诡案实录		334
第12章 HikariCP扩展技术		280	14.1	问题描述	334
12.1	Flexy-Pool	280	14.2	Brett 经典回答	336
12.2	Apache ShardingSphere	284	14.3	另一个类似的案例	337
12.3	自研集成 HikariCP 和 Sharding- JDBC 数据库中间件	295	14.4	分析问题	338
12.4	时钟回拨	300	14.5	解决问题	344
			14.6	本章小结	344



第一篇 *Part 1*

准 备 篇

- 第1章 阿里中间件实战，第一个案例
- 第2章 数据库连接池江湖
- 第3章 初识HikariCP

阿里中间件实战，第一个案例

在应用系统开发过程中，池化技术，如对象池、连接池、线程池等，通过复用对象以减少创建、释放连接的消耗来提升性能。由于 TCP 连接的创建开支十分昂贵，数据库所能承载的 TCP 并发连接数也有限制，针对这种场景，数据库连接池应运而生。数据库连接池的实现有很多，如 c3p0、DBCP、Druid 等，也包括本书将详细介绍的、号称性能最好的数据库连接池——HikariCP。

本章首先从一个有趣的实例说起。在此过程中，我将告诉你很多和连接相关的知识，并且让你对系统的调优过程产生一点共鸣，然后我才能按通常的方式向你展开原理介绍。这个例子是我在阿里巴巴多年间对一款中间件调优的真实经历，这款中间件经历过很多打磨，比如全链路压测、3 年双十一大促、多种监控系统接入、数次整体结构重构改造、重大问题修复等。这款中间件是纯 TCP 的，再回首，我并不为当年所做的那些 TCP 性能优化感到自豪，相反，我认为一款池化的中间件如果能提炼出来，会更有价值，因为它更加有利于技术产品的快速落地及运维，进而更好地提升研发效能。

希望读者通过这个实例可以了解到在没有池化技术的情况下 TCP 调优是如何进行的，同时也能理解 TCP 调优的复杂性，体会到池化技术出现的必然性。数据库连接池同样是池化技术，大家也可以深入思考并想象一下数据库连接池中间件需要解决哪些问题。

1.1 物联网 MQTT 单机压测 130 万参数调优

2019 年 1 月，我的工作之一就是为公司搭建一套独立于阿里云之外的自己的物联网 MQTT 集群。MQTT 非常适合物联网场景，它可以保持设备与服务器的长连接，避免反复

轮询, 并支持推送。和数据库连接池一样, 它同样运行于 TCP 协议之上, MQTT 相比于 HTTP 具有协议开销低、容忍弱网络、低功耗、百万并发等优点。但是由于当前的阿里云 MQTT 不支持 will、retain msg、QOS2, 并且存在限流、断连等不稳定因素, 自建 MQTT 集群就显得格外重要。

在调研了 Eclipse Mosquitto 后, 我又在 EMQ (Erlang/Enterprise/Elastic MQTT Broker) 的官网^①上发现了下列描述: “EMQ 消息服务器 1.x 版本 MQTT 连接压力测试到 130 万, 在 1 台 8 核心、32G 内存的 CentOS 服务器上。”经过梳理后我认为, 它的服务端调优主要分为 Linux 操作系统参数调优和 TCP 协议栈网络参数调优两部分^②。

Linux 操作系统参数调优如下:

1) 系统全局允许分配的最大文件句柄数。

```
sysctl -w fs.file-max=2097152
sysctl -w fs.nr_open=2097152
echo 2097152 > /proc/sys/fs/nr_open
```

2) 允许当前会话或进程打开文件句柄数。

```
ulimit -n 1048576
```

TCP 协议栈网络参数调优如下:

1) 并发连接 backlog 设置。

```
sysctl -w net.core.somaxconn=32768
sysctl -w net.ipv4.tcp_max_syn_backlog=16384
sysctl -w net.core.netdev_max_backlog=16384
```

2) TCP Socket 读写 Buffer 设置。

```
sysctl -w net.core.rmem_default=262144
sysctl -w net.core.wmem_default=262144
sysctl -w net.core.rmem_max=16777216
sysctl -w net.core.wmem_max=16777216
sysctl -w net.core.optmem_max=16777216
#sysctl -w net.ipv4.tcp_mem='16777216 16777216 16777216'
sysctl -w net.ipv4.tcp_rmem='1024 4096 16777216'
sysctl -w net.ipv4.tcp_wmem='1024 4096 16777216'
```

① <http://www.emqt.com>, 2019 年 1 月 25 日已经更新到 3.0.1 版本, EMQ X R3.0 完整支持 MQTT V3.1/V3.1.1/V5.0 版本协议规范, 并扩展支持 MQTT-SN、WebSocket、CoAP、Stomp 或私有 TCP/UDP 协议。EMQ X 3.0 消息服务器支持单节点百万连接与多节点分布式集群。

② 100 万连接测试所需参数包含 Linux 内核参数、网络协议栈参数、Erlang 虚拟机参数, 本书由于主要介绍数据库连接池 Linux 内核参数和网络协议栈参数的共通部分, 所以 Erlang 虚拟机参数调优及 EMQX 消息服务器参数调优部分会略去。