HZ BOOKS
华章IT

Linux/Unix
技术丛书

Linux集群和自动化运维

余洪春 著

Linux Cluster and Automation Operation

- 高级运维架构师、资深系统运维工程师十余年工作经验总结，基于一线运维工作提炼，从Linux集群、Python自动化运维和亿级PV网站架构设计等多角度讲解，以实践案例指导读者掌握到Linux系统集群和自动化运维技巧。
- 姊妹篇《构建高可用Linux服务器》被《程序员》杂志和51CTO等权威媒体评为“10大最具技术影响力的图书”和“最受读者喜爱的原创图书”。

机械工业出版社
China Machine Press

Linux集群和自动化运维

Linux Cluster and Automation Operation

余洪春 著

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

Linux 集群和自动化运维 / 余洪春著 . —北京：机械工业出版社，2016.8
（Linux/Unix 技术丛书）

ISBN 978-7-111-54438-8

I. L… II. 余… III. Linux 操作系统 IV. TP316.89

中国版本图书馆 CIP 数据核字（2016）第 176055 号

Linux 集群和自动化运维

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：陈佳媛 杨绣国 责任校对：董纪丽
印 刷：北京市荣盛彩色印刷有限公司 版 次：2016 年 8 月第 1 版第 1 次印刷
开 本：186mm × 240mm 1/16 印 张：23.25
书 号：ISBN 978-7-111-54438-8 定 价：79.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426 88361066 投稿热线：（010）88379604
购书热线：（010）68326294 88379649 68995259 读者信箱：hzit@hzbook.com

Foreword 推荐序一

在全球“互联网＋”的大背景下，互联网创业企业的数量如雨后春笋般大量产生并得到了快速发展！对“互联网＋”最有力的支撑就是 Linux 运维架构师、云计算和大数据工程师，以及自动化开发工程师等！

但是，随着计算机技术的发展，企业对 Linux 运维人员的能力要求越来越高，这就使得很多想入门运维的新手不知所措，望而却步，甚至努力了很久却仍然徘徊在运维岗位的边缘；而有些已经工作了的运维人员也往往是疲于奔命，没有时间和精力去学习企业所需的新知识和新技能，从而使得个人的职业发展前景大大受限。

本书就是在这样的背景下诞生并致力于为上述问题提供解决方案的，本书是作者余洪春先生 10 多年来一线工作经验的“再”结晶，此前作者已经出版过 Linux 集群方向的图书（《构建高可用 Linux 服务器》），本次出版的书是作者对运维行业的再回馈。

书中不仅涵盖了入门运维人员必须了解的 IDC 和 CDN 服务的选型、Linux 系统及常见服务的优化实践内容，还有对于企业运维人员需要的大规模集群场景下必备的运维自动化 Shell 和 Python 企业开发应用实践案例、热门的自动化运维工具的企业应用实践、大规模集群及高可用的企业案例分享与安全防护等。

本书能够帮助运维人员掌握业内运维实战专家的网站集群的企业级应用经验的精髓，从而以较高的标准胜任各类企业运维的工作岗位，并提升自己的运维职业发展竞争力，值得一读！

——老男孩　老男孩 Linux 实战运维培训中心总裁
《跟老男孩学 Linux 运维：Web 集群实战》作者

推荐序二 *Forword*

本书作者余洪春先生和我相识于ChinaUnix举办的一次技术交流活动——“千万级PV高性能高并发网站架构与设计交流”，当时他已经在宣传自己的第一本著作——《构建高可用Linux服务器》，该书凝聚并整合了他多年来在一线工作的经验结晶，以至时至今日，该书仍是一本在国内非常经典的运维原创著作，现在已经更新到第三版，这种对技术不断进行完善的坚持及工匠精神让我深深折服。这次能受邀为他的新书《Linux集群和自动化运维》写推荐序，让我倍感荣幸。

本书覆盖了Linux集群服务的核心技术，同时还介绍了基于Python语言构建的主流自动化运维工具，包括Python脚本、Fabric、Ansible等，这些都是DevOps工具元素周期表中最闪亮的内容，也是运维人员必备的技能。本书中分享的案例是余洪春多年实战经验的精华，具有非常高的参考价值及借鉴意义。

书中内容从互联网业务平台构建及自动运维的场景出发，以常见的业务服务为基础，给出了大量的实战案例，这些都是作者在十余年的互联网运维工作中总结出来的宝贵经验，相信会给读者带来不少启发及思考。

更难能可贵的是，作者能从通俗易懂的角度出发，由浅入深地剖析自动运维管理之道。对于不同水平层次的读者来说，都能有效地阅读和吸收，也能根据实际需要各取所需。

最后，感谢余洪春给中国互联网从业者带来这么好的图书，我相信阅读本书的每一位读者都能从中获取提升的能量，为企业及行业做出自己的贡献。

腾讯高级工程师　刘天斯

Preface 前　言

为什么要写这本书

笔者从事系统运维和网站架构设计的工作已有 10 多年，现在在一家外企担任云平台架构师。云计算是现在的主流技术，未来也有很好的发展趋势，云计算的流行对于传统的运维知识体系来说，其实也造成了冲击，有很多读者经常向笔者咨询工作中的困惑，比如从事系统运维工作 3 ～ 5 年后就不知道该如何继续学习和规划自己的职业生涯了。因此笔者想通过此书，跟大家分享一下自己的工作经验和心得（包括传统运维和云平台运维工作的区别与对比），以期解决大家在工作中的困惑。本书提供了大量项目实践和线上案例，希望能让大家迅速了解 Linux 运维人员的工作职责，快速进入工作状态并找到成长方向。希望大家通过阅读此书，能够掌握 Linux 系统集群和自动化运维及网站架构设计的精髓，从而能够轻松愉快地工作，并提升自己的职业技能，这就是笔者写作此书的初衷。

运维架构师之路

在成为运维架构师之前，笔者从事过很长一段时间的系统集成、运维和管理工作，在 CDN 门户网站、电子广告、电子商务领域也有不少的沉淀和积累，在之前的《构建高可用 Linux 服务器》一书中已经跟大家分享了很多跟 Linux 集群有关的知识。笔者目前的主要工作职责是维护和优化公司的 DSP 电子广告业务平台，主要方向是云计算和大数据方面。需要维护的数据中心和机器数量非常之多，所以自动化运维和 DevOps 是目前的主要工作方向，此外，也会涉及网站架构设计及调优工作，因此在此书中特意将这部分工作经验分享出来，希望大家能从中学到新的知识体系，借以提升自己的职业技能。

读者对象

本书适合以下读者阅读。

- ❑ 中高级系统管理员
- ❑ 系统架构设计师

- 高级程序开发人员
- 运维开发工程师

如何阅读本书

本书是笔者对实际工作中积累的技术和经验所做的总结，涉及大量的知识点和专业术语。全书总共分为三大部分，第一部分包含第 1 章和第 2 章，主要讲解进行系统架构设计的软硬件环境，以及生产环境下的 Shell 脚本和 Python 脚本。其中，第 2 章的内容是以 Shell 为主，Python 为辅，Shell 部分讲得比较详细，Python 部分需要重点关注的地方也有所提及。之所以这样安排，主要是考虑到大多数搞开发的读者或 DevOps 工程师都是 Java 程序员出身，对 Shell 脚本语言不是很熟悉。第二部分包含第 3 章、第 4 章和第 5 章，主要讲自动化运维，包括 Fabric、Ansibel 和 Puppet 三大工具，大家可以结合自己的实际环境来选择对应的工具。第三部分包含第 6 章、第 7 章和第 8 章，主要讲的是 Linux 集群和网站架构设计，特别是第 8 章，分别以百万 PV、千万 PV 及亿级 PV 的网站为例来详细说明网站系统架构设计的相关技术，然后细分五层来解说网站的架构，并指出了设计网站的压力及关注点所在。

大家可以根据自己的职业发展和工作需求来选择不同的章节进行阅读或学习。

关于本书中的配置文件、Shell 脚本和 Python 脚本的编号，这里也略作说明，比如 1.5.3 节中有 1.sh，表示这是 1.5.3 节的第一个 Shell 脚本；如果是 2.py，则表示是 1.5.3 节的第二个 Python 脚本；其他依此类推，在哪个章节中出现的配置文件或脚本就在哪个章节中寻找，这样对照起来阅读理解会比较方便。此外，书中多次出现的 Nginx 配置文件 nginx.conf 也在对应的章节里。本书相关的 GitHub 地址为 http://github.com/yuhongchun/automation。

勘误

尽管笔者花费了大量的时间和精力来核对文件和语法，但书中难免还会存在一些错误和纰漏，如果大家发现有任何问题，都请及时反馈给我，相关信息可以发到个人邮箱 yuhongchun027@gmail.com。尽管无法保证对于每一个问题都会有一个正确答案，但我肯定会努力回答并且指出一个正确的方向。

致谢

感谢爱女媛媛的出生，你的降临是上天赐给我的最好礼物，是我进行写作的源泉和动力。

感谢我的家人，他们在生活上对我的照顾无微不至，让我有更多的精力和动力去工作和创作。

感谢好友三宝这么多年来对我的信任和支持，从始至终一直都在支持和信任我。

感谢机械工业出版社华章公司的编辑杨福川和杨绣国，在你们的信任、支持和帮助下，我才能如此顺利地完成全部书稿。

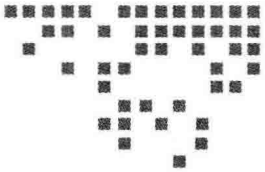
感谢好友老男孩和刘天斯，闲暇之余和你们一起交流开源技术和发展趋势，也是一种享受。

感谢 Linux 之父——Linus Torvalds，他不仅创造了 Linux 系统，而且还创造了 Git 这么神奇的版本管理软件。

余洪春（抚琴煮酒）

中国，武汉

目　录 *Contents*



第 1 章 Chapter 1

系统架构设计的构建基础

作为一名系统架构设计师，会面临着很多系统方面的架构设计工作，比如电子商务系统、CDN（内容分发网络）大型电子广告平台和 DSP 电子广告系统的运维方案确定及平台架构设计等，此外，还会涉及核心业务的系统在线优化及升级等工作。在以上这些工作中，又将包括多项选择：比如是考虑自建 CDN，还是租赁 CDN 系统；公司的业务系统所在的机房是考虑自建机房、托管机房，还是云计算平台，而如果选择托管机房，又会有更多的细节需要考虑，比如是选择电信机房、双线机房还是 BGP 机房，服务器应该如何选型，选择哪种操作系统等，这个时候系统架构设计师的经验和作用就体现出来了。他们应该在系统网站实施的初期就做好项目的成本预算和风险规避，并对系统的高可用及扩展性进行细致权衡，这些也是其工作职责所在。当然，在了解上述这些之前，首先应该了解一些网站架构设计相关的专业术语，下面就一起来看看。

1.1 网站架构设计相关术语

1.1.1 什么是 HTTP 1.1

HTTP 1.1（Hypertext Transfer Protocol Version 1.1），即超文本传输协议 – 版本 1.1，跟版本 1.0 是有区别的。

针对 HTTP 1.0 中 TCP 连接不能重复利用的情况，HTTP1.1 采用了效率更高的持续连接机制，即客户端和服务器建立 TCP 连接以后，后续相关联的 HTTP 请求可以重复利用已经建立起来的 TCP 连接。

HTTP 1.1 是用来在 Internet 上传送超文本的传送协议。它是运行在 TCP/IP 协议族之

上的 HTTP 应用协议，它可以使浏览器更加高效，并减少网络传输。任何服务器除了包括 HTML 文件以外，都还有一个 HTTP 驻留程序，用于响应用户请求。如果浏览器是 HTTP 客户，在向服务器发送请求时，向浏览器中输入一个开始文件或点击一个超级链接，浏览器就向服务器发送 HTTP 请求，此请求被送往由 URL 指定的 IP 地址。驻留程序接收到请求，在进行必要的操作后就会回送所要求的文件。

HTTP 1.1 支持持续连接。通过这种连接，就有可能在建立一个 TCP 连接后，发送请求并得到回应，然后发送更多的请求并得到更多的回应。由于把建立和释放 TCP 连接的开销分摊到了多个请求上，因此对于每个请求而言，由于 TCP 连接而造成的相对开销就被大大地降低了。而且，还可以发送流水线请求，也就是说在发送请求 1 的回应到来之前就发送请求 2。也可以认为，一次连接发送多个请求，由客户机确认是否关闭连接，而服务器会认为这些请求分别来自于不同的请求。

1.1.2 什么是 Web 2.0

Web 2.0，指的是利用 Web 的平台，由用户主导而生成内容的互联网产品模式，为了区别由网站雇员主导生成内容的传统网站而定义为 Web 2.0。Web 1.0 的盈利模式都基于一个共同点，即巨大的点击流量，无论是早期融资还是后期获利，依托的都是众多的用户和点击率，以点击率为基础融资上市或开展增值服务，充分体现了互联网的眼球经济色彩，例如早期的新浪、搜狐和网易等。

Web 2.0 是资源平等的体现。Web 2.0 的应用可以让人了解到目前万维网正在进行的一场改变——从一系列网站到一个成熟的、为最终用户提供网络应用的服务平台。这种概念的支持者期望 Web 2.0 服务在很多用途上能最终取代桌面计算机应用。虽然 Web 2.0 并不是一个技术标准，但是它包含了技术架构及应用软件。它的特点是鼓励信息的最终利用者通过分享，使得可供分享的资源变得更加丰富；相反的，过去网上的各种分享方式则显得支离破碎。

Web 2.0 是网络运用的新时代，网络成为了新的平台，内容因为每位用户的参与而产生，参与所产生的个人化内容，借由人与人（P2P）的分享，形成了现在的 Web 2.0 世界。

Web 2.0 的主要特点和基于这些特点所产生的具有代表性的服务如下。

1. 博客

博客（Blog）是 Web 2.0 最早期的服务之一，可使任何参与者拥有自己的专栏，成为网络内容的产生源，进而形成微媒体，为网络提供文字、图片、声音或视频信息。

2. 内容源

内容源（RSS）是伴随博客产生的简单文本协议，将博客产生的内容进行重新格式化输出，从而将内容从页面中分离出来，便于同步到第三方网站或提供给订阅者进行阅读。

3. Wiki

是一个众人协作的平台，方便编写百科全书、词典等。Wiki 指的是一种超文本系统，

这种超文本系统支持面向社区的协作写作，例如百度百科和维基百科。

4. 参与评论与评分的 Digg 机制

Web 2.0 最显著的特点之一是分享机制和去中心化，Digg 机制为更多的网络用户提供了参与网络建设的机制，无须进行内容贡献或创作，只要用户对网络内容进行评分或点评，即可参与到网络内容的建设过程当中。

5. 美味书签

美味书签（Delicious）不同于个人博客，用户可根据自己的喜好进行网络内容的收藏与转载，并将自己的收藏或转载整理成列表，分享给更多的用户，从而在网络上起到信息聚合与过滤的作用。

6. 社会化网络

社会化网络（SNS）从原有的以网站、内容为中心，转为侧重于以人与人之间的关联为中心，网络上每一个节点所承载的不再是信息，而是以具体的自然人为节点，形成的新型互联网形态。

7. 微博

微博（Microblog）作为博客的精简版，有较为严格的字数限制和政治立场限制。有更简单的发布流程和更随意（被限制的话题、领域）的写作方式，使得参与到网络内容贡献中的门槛降低，更大程度地推动了网络内容建设和个体信息贡献。

8. 基于位置信息的服务

基于位置信息的服务（LBS）是集地理信息系统（GIS）、微博（Twitter）和移动设备（Mobile）及 A-GPS 定位服务于一体的增强型微博系统，其主导思想是每一条信息除了利用时间为索引，还加入了地理经纬度的索引，从而实现不仅可以通过时间对信息进行筛选，还可以利用地理坐标对信息进行合理的筛选。

9. 即时通信

即时通信（IM）软件可以说是目前中国网上用户使用率最高的软件。聊天一直是网民们上网的主要活动之一，网上聊天的主要工具已经从初期的聊天室、论坛变为以 QQ 和 Skype 为主要代表的即时通信软件。

1.1.3　软件开发 C/S 结构与 B/S 结构的区别

C/S 结构是大家熟知的软件系统体系结构，即 Client/Server（客户机 / 服务器）结构，它通过将任务合理地分配到 Client 端和 Server 端，来降低系统的通信开销，不过需要安装客户端才可进行管理操作。B/S 结构，即 Browser/Server（浏览器 / 服务器）结构，是随着 Internet 技术的兴起，对 C/S 结构的一种变化或改进的结构。在这种结构下，用户界面可完全通过 WWW 浏览器来实现。像 QQ、Skype 这类即时通信软件就属于 C/S 结构；而像百度、

Google 这样的搜索引擎就属于 B/S 结构。

随着计算机技术的不断发展与应用，计算模式从集中式转向了分布式，尤为典型的是 C/S 结构。两层结构 C/S 模式，在 20 世纪 80 年代及 90 年代初得到了大量的应用，最直接的原因是可视化开发工具的推广。之后，它开始向三层结构发展。近年来，随着网络技术的不断发展，尤其是基于 Web 的信息发布和检索技术、Java 计算技术及网络分布式对象技术的飞速发展，导致了很多应用系统的体系结构从 C/S 结构向更加灵活的多级分布结构演变，使得软件系统的网络体系结构跨入一个新阶段，即 B/S 体系结构。基于 Web 的 B/S 模式其实也是一种客户机 / 服务器模式，只不过它的客户端是浏览器。为了区别于传统的 C/S 模式，才特意将其称为 B/S 模式的。了解这些结构的特征，对于系统的选型而言是很关键的。

下面是 C/S 结构与 B/S 结构的特点分析。

1. 系统的性能

在系统的性能方面，B/S 占有优势的是其异地浏览和信息采集的灵活性。任何时间、任何地点、任何系统，只要可以使用浏览器上网，就可以使用 B/S 系统的终端。

不过，采用 B/S 结构时，客户端只能完成浏览、查询、数据输入等简单功能，绝大部分工作由服务器承担，这就使得服务器的负担很重。采用 C/S 结构时，客户端和服务器端都能够处理任务，这虽然对客户机的要求较高，但因此可以减轻服务器的压力。而且，由于客户端使用浏览器，使得网上发布的信息必须是以 HTML 格式为主，其他格式的文件则多半是以附件的形式来存放的。而且 HTML 格式文件（也就是 Web 页面）不便于编辑修改，给文件的管理带来了许多不便。

2. 系统的开发

C/S 结构是建立在中间件产品基础之上的，要求应用开发者自己去处理事务管理、消息队列、数据的复制和同步、通信安全等系统级的问题。这对应用开发者提出了较高的要求，而且还会迫使应用开发者投入很多精力来解决应用程序以外的问题，这使得应用程序的维护、移植和互操作变得复杂。如果客户端是在不同的操作系统上，那么 C/S 结构的软件还需要开发不同版本的客户端软件。

但是，与 B/S 结构相比，C/S 技术的发展历史更为“悠久”。从技术成熟度及软件设计、开发人员的掌握水平来看，C/S 技术应是更成熟、更可靠的。

3. 系统的升级维护

C/S 系统的模块中只要有一部分发生改变，就会关联到其他模块的变动，这会使得系统的升级成本比较高。B/S 与 C/S 处理模式相比，则大大简化了客户端，只要客户端机器能上网就可以。对于 B/S 而言，开发、维护等几乎所有的工作也都集中在服务器端，当企业对网络应用进行升级时，只需更新服务器端的软件就可以了，这就降低了异地用户进行系统维护与升级的成本。如果客户端的软件系统升级比较频繁，那么 B/S 架构的产品优势就更明显——所有的升级操作只需要针对服务器进行即可，这对那些点多面广的应用是很有价

值的，例如一些招聘网站就需要采用 B/S 模式，客户端分散，且应用简单，只需要进行简单的浏览和少量信息的录入即可。

在系统安全维护上，B/S 则略显不足，B/S 结构尤其得考虑数据的安全性和服务器的安全性，毕竟现在的网络安全系数并不高。以 OA（办公自动化）软件为例，B/S 结构要实现办公协作过程中复杂的工作流控制与安全性控制，还有很多技术上的难点。因此，当前虽然出现了 B/S 结构的 OA 系统产品，但尚未大范围推广。

1.1.4　评估网站性能的专业术语

1. PV

PV（Page View）即访问量，中文翻译为页面浏览，即页面浏览量或点击量，用户每刷新一次就会被计算一次。PV 的具体度量方法就是从浏览器对网络服务器发出一个请求（Request），网络服务器接到这个请求后，会将该请求对应的一个网页（Page）发送给浏览器，从而产生一个 PV。在这里只要是请求发送给了浏览器，无论这个页面是否完全打开（下载完成），那么都被计为 1 个 PV。PV 反映的是浏览某网站的页面数，所以每刷新一次也算一次，也就是说 PV 与 UV（独立访客）的数量成正比，但 PV 并不是页面的来访者数量，而是网站被访问的页面数量。

2. UV

UV（Unique Visitor）即独立访问，访问网站的一台电脑客户端为一个访客，如果以天为计量单位，程序会统计 00 ： 00 至 24 ： 00 时间段内的电脑客户端。相同的客户端只被计算一次。一个电脑客户端可能有多个不同的自然人访问，但也只记一个 UV，通过不同的技术方法来记录，实际会有误差。如果企业内部通过 NAT 技术共享上网，那么出去的公网 IP 有且只有一个，这个时候在程序里面统计的话，也只能算是一个 UV。

3. 并发连接数

当一个网页被浏览，服务器就会和浏览器建立连接，每个连接表示一个并发。如果当前网页的页面中包含很多图片，图片并不是一个一个显示的，服务器会产生多个连接同时发送文字和图片以提高浏览速度。网页中的图片越多，那么服务器的并发连接数（Concurrent TCP Connections）就会越多。我们一般以此值作为衡量单台 Web 机器性能的参数。现在 Nginx 在网站中的应用比例非常大，可以参考 Nginx 的活动并发连接数。

4. 每秒查询率 QPS

QPS（Query Per Second）是对一个特定的查询服务器在规定时间内所处理流量多少的衡量标准，在因特网上，作为域名系统服务器的机器其性能经常用每秒查询率来衡量。对应的是 Fetches/Sec，即每秒的响应请求数，也称为最大吞吐能力。对于系统而言，QPS 数值是一个非常重要的参数，它是综合反映系统最大吞吐能力的衡量标准。它反映的不仅仅是 Web 层面的，还有缓存、数据库方面的，它反映的是系统的综合处理能力。