

源码之前，了无秘密

资深工程师多年研读代码、精心打磨之作
ChinaUnix社区、运维专家、一线工程师强烈推荐

深入剖析 Nginx

高群凯 著



 人民邮电出版社
POSTS & TELECOM PRESS

深入剖析 Nginx

高群凯 著



人民邮电出版社
北京

图书在版编目 (C I P) 数据

深入剖析Nginx / 高群凯著. -- 北京 : 人民邮电出版社, 2013.5
ISBN 978-7-115-30762-0

I. ①深… II. ①高… III. ①Web服务器 IV.
①TP393.09

中国版本图书馆CIP数据核字(2013)第008788号

内 容 提 要

Nginx 是一款功能强大的高性能 Web 和反向代理服务器,最初由俄罗斯程序员 Igor Sysoev 开发,而当前由 Igor Sysoev 领导的专业公司 Nginx, Inc.进行持续的维护与更新。Nginx 可以在大多数 UNIX 或类 UNIX 系统上编译运行比如 FreeBSD、Solaris、Linux 等,并且官方还提供有 Windows 下的可执行版本。目前, Nginx 在 Netflix、Wordpress.com、新浪、网易、腾讯、豆瓣等国内外众多知名网站中应用。

本书不是一本关于 Nginx 配置指令如何使用的介绍手册。本书重点在于通过剖析 Nginx 的源代码,探究其功能结构及其内部实现原理。全书共 14 章和 3 个附录。首先介绍了开始剖析 Nginx 源代码前的准备工作,以及跟踪和调试的方法;然后,分别深入分析了 Nginx 的进程模型、数据结构、配置指令、主要功能模块、I/O 事件处理、变量机制、客户端请求过程、Filter 模块实例、负载均衡策略以及 Handler 模块等。附录部分提供了 Nginx 的编译模块、运行配置等有用信息。

从源码剖析的角度出发,是程序员常用的学习和提高方法。本书是作者多年研读 Nginx 代码、深入思考和不断实践的结晶。本书适合系统程序员、软件开发工程师、Nginx 高级运维工程师阅读参考,对于有志从事相关工作的 IT 专业学生,更是不可多得的学习资料。

深入剖析 Nginx

- ◆ 著 高群凯
责任编辑 陈冀康
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 22
字数: 423 千字 2013 年 5 月第 1 版
印数: 1—3 500 册 2013 年 5 月北京第 1 次印刷

ISBN 978-7-115-30762-0

定价: 59.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223
反盗版热线: (010)67171154

前言

慕名对 Nginx 源码进行学习与研究是早在 2008 年的事情。当时正在为职业规划与未来发展困惑不已，一筹莫展之际不知从哪里得知高性能服务器是一个很有“前途”的努力方向，几经搜索又机缘偶合地得识 Lighttpd 与 Nginx。在逐步了解和熟悉它们的源码后，我开始感到自己的无知与浅薄，发现原来代码也可以写得如此优雅。

我已编著过一本《Lighttpd 源码分析》。先解析 Lighttpd 源码并没有什么特别的原因，只是因为当时 Lighttpd 比 Nginx 要火，应用得也较广；而近几年，凭借专注持续的更新与运作，Nginx 后来居上，在全世界的应用仅次于 Apache 与 Microsoft IIS¹，而且大有赶超 IIS 的势头。

Nginx 提供了一个安全、快速并且灵活的 Web Server 环境。与其他 Web Servers 相比，其主要特点是占有系统资源少、并发能力强、稳定性好，这些都是吸引网站架构人员重点关注以及对其进行广泛应用的巨大优点。然而，对于开发工程师而言，更有吸引力的地方却是 Nginx 是怎样做到如此强劲的功能与性能的，这从因特网上存在大量的 Nginx 源码分析的博客文章与论坛帖子可见一斑。

既然互联网上已有大量的 Nginx 源码分析文章，那我现在编写这本《Nginx 源码分析》是否多此一举？我会说不是，毕竟它较为完整地梳理了 Nginx 的相关核心流程。当然，这只是我给出的答案，最终还需要各位读者来回答。我在此阐述一下自己的看法。本书的原始素材是我在最近三四年里地对 Nginx 代码分析时断断续续地所做的笔记，只不过之前写的笔记过细地逐行注释代码，导致文档中充斥大量的代码。而在这次重新整理后，希望能从比较高的角度去解析 Nginx，把握全局，搞清楚整体实现原理而不是陷入细节。我个人认为，只要看清楚了 Nginx 整体的实现原理，对于一时半会没有触及到的细节，在真正遇到那个点时再去理解也是毫不费力，最多只不过还需去查一下 Man 手册，了解一下系统 API 而已。授人以鱼不如授人以渔，为了达到这个目标，本书力争尽量少贴代码多画图，当然一些必要的代码仍然是不可或缺的，所以读者还是会在本书看到源代码。虽然我的最初出发点是好的，但是在解析 Nginx 的某些功能时，我的确遇到了困难，特别是在解析到一大片代码却并没有孰轻孰重的情况下，实在让我很纠结，把代码全部贴上将占去大量的版面，会违背我少贴代

¹ <http://news.netcraft.com/archives/2012/08/02/august-2012-web-server-survey.html>。

码的初衷，不贴上又怕读者不知我所云何物而看不懂。这不仅是因为 Nginx 代码本身的耦合性较大，也在于我个人的文字描述能力有待提高，所以导致本书或多或少的各种缺点，比如啰嗦的地方、没讲到的细节等，还请各位读者多多包涵。

读者对象

我面试过不少刚本科毕业或研究生毕业的应届生，问得比较多的一句是：写过实际应用的代码么？答：没有。我又问：看过什么优秀的开源代码么？答：没有。这种情况很糟糕。限于国内大学的环境，除了较为简单的课程设计以外，很多大学生在校园内很少有机会写代码或参与实践问题的解决。针对这一点，至少我们可以去看一些有价值的优秀开源代码，毕竟在大学校园里，我们有时间、有地点、有资源（教室、图书馆、电子阅览室等）。所谓他山之石，可以攻玉，在理解这些开源代码的过程中，我们甚至可以尝试写一些扩展功能模块，逼迫自己去主动学习，培养扎实的理论基础知识，锻炼实际的动手编程能力，自己也会因此逐步成长许多。

本书的另一目标读者群为软件开发工程师，这是显而易见的事情。短小精悍的 Nginx 实现了如此强大的 HTTP 服务、反向代理服务以及邮件代理服务等，这些功能值得每一位软件开发工程师去学习和研究，况且国内也有不少项目在对 Nginx 进行二次开发或移植整合，有一本能帮助开发工程师快速开展工作的书籍也许能加快项目进度。

另外，鉴于国内对 Nginx 的广泛使用，虽然 Nginx 官网上有不少针对配置使用的文档，但是限于文字描述的简洁性与文档更新的滞后性，对于运维工程师而言，一些配置指令的使用是靠逐步验证来判断其具体使用用法的。通过本书的介绍，我们能从源代码上去找对应配置指令的用法并理解其真实的实际功能，即它让源代码本身就变成一本配置使用手册。所以，对于那些具有强烈意愿知其然又知其所以然的高级运维工程师，本书也是一个不错的选择。

我假设本书的读者已经掌握了 C 语言，并且对 Linux 系统有一定程度的了解。不过，即使在开始阅读本书之前，读者在这些方面存在一些不足也无关紧要，我会尽可能地把相关扩展链接标注出来，读者查询一下相关资料即可。

本书的读者对象主要是如下三类人群：

- ☐ 在校大学生；
- ☐ 软件开发工程师；

- ❑ Nginx 高级运维工程师。

Nginx 版本

本书基于 Nginx-1.2.0，该版本是我在重新开始做文档整理时 Nginx 的最新版本。Nginx 源码更新较为频繁，但更新的主要是 Bug 修正或增添新功能，而其主要架构是稳定的，所以不论哪个版本的 Nginx，本书基本都是适用的。

本书讨论环境

我是一名 Linux 开发工程师，所以本书的讨论环境也就是以 Linux 为主，使用的所有相关工具都是 Linux¹下的，比如调试工具 gdb、编译工具 gcc、测试命令 strace/pstack、测试工具 wget²、curl³等。

Nginx 本身提供的相关机制，对各种平台都进行了支持，比如 I/O 多路复用模型就支持 epoll、kqueue、eventport 等，但本书仍然以 Linux 平台上的机制为主要讨论对象，像 I/O 多路复用模型里就是 epoll。在没有明确说明的情况，默认的编译模块以附录 A 为准，且默认以附录 B 给出的配置运行 Nginx。

Linux 是广泛使用的操作系统，所以以它为主要实例也是最为方便且有效的方法。我在 Intel x86 机器上安装了一个 CentOS⁴ 6.2 的 32 位操作系统（系统设置全为默认，比如页大小 4KB，应用程序与内核的地址空间划分为 3:1 等），并且把 CentOS 6.2 所提供的相关软件开发包也都选择安装上了，这对于我在后继执行相关程序的编译与安装时提供了极大的方便，下面列出了相关系统环境：

- ❑ 操作系统：CentOS release 6.2 (Final)/kernel-2.6.32/32bit;
- ❑ 编译器：gcc⁵ version 4.4.6 20110731 (Red Hat 4.4.6-3) (GCC);

¹ 虽然有一些工具也的确能使用于 Windows 等其他平台下。

² <http://www.gnu.org/software/wget/>

³ <http://curl.haxx.se/>

⁴ <http://www.centos.org/>

⁵ <http://gcc.gnu.org/>

- ❑ 调试器: GNU gdb¹ (GDB) Red Hat Enterprise Linux (7.2-50.el6);
- ❑ Make: GNU Make² 3.81。

本书内容

本书不是关于 Nginx 配置指令如何使用的介绍手册, 关于那些内容在 Nginx 官网上有专门的帮助文档³, 甚至有对应的中文翻译⁴。因此本书内容的重点在于解析 Nginx 的内部实现原理。

虽然限于时间关系而无法做到方方面面的解析, 但我尽力把 Nginx 最核心的线条抽取出来并努力把它们以一种更容易理解的形式展现在各位读者面前。本书一共 14 章, 大部分章节在排列上并没有特别的先后顺序, 所以读者在翻阅本书时可以来回跳跃。各章内容简介如下:

第 1 章, 介绍开始本书内容前的准备工作。这是一些基础概念和工具使用, 比如什么是 Nginx、怎样安装运行 Nginx、如何快速方便地阅读 Nginx 源码以及有哪些相关的知识站点等。

第 2 章, 详细介绍了跟踪与调试的多种手段与技巧。通过实际案例可以看到, 这些知识能够极为方便地帮助我们理解 Nginx 程序内部的相关执行逻辑。

第 3 章, 从 3 个层次上介绍 Nginx 的进程模型。首先是最顶层, 通过 Nginx 整体架构框图从宏观上了解 Nginx; 其次是中间层, 对监控进程、工作进程、Cache 进程进行逐一介绍, 了解它们各自的主要执行逻辑; 最后是交互层, 即进程之间 (套接字、共享内存) 以及进程与用户之间 (信号) 的信息交换。

第 4 章, Nginx 封装了很多有用的数据结构, 大多较为简单一看即懂, 但其中的内存池、Hash 和 Radix tree 这 3 个数据结构较为复杂, 所以本章对它们做了详细介绍。

第 5 章, 配置指令是用户控制 Nginx 实际运行逻辑的主要手段, 如何将用户设置的配置值转换到 Nginx 内部并控制 Nginx 的具体执行, 是本章的解析重点。

¹ <http://www.gnu.org/software/gdb/>

² <http://www.gnu.org/software/make/>

³ <http://www.nginx.org/en/docs/>和 <http://wiki.nginx.org/DirectiveIndex>

⁴ <http://wiki.nginx.org/NginxChs>

第 6 章, Nginx 提供的丰富功能基本都是通过模块来实现的, 根据每个模块的具体功能不同而分为不同的类型, 比如 Handler 模块、Filter 模块、Load-balance 模块等。通过对它们的综合概述, 让读者从宏观上了解每类 Nginx 模块的功能。

第 7 章, 与事件相关的实现将在本章介绍。这包括 I/O 多路复用模型、I/O 读写事件、超时事件等。另外, 由于 Nginx 工作进程的主要任务就是处理事件, 所以各个工作进程之间的负载均衡也一并介绍。

第 8 章, 变量机制。这里所提到的变量主要是指 Nginx 用户在配置文件里所使用的可变符号, 这些符号大多会随着客户端请求的不同而不同, 类似于编程语言里的变量, 本章将详细阐述 Nginx 对它们的具体实现。

第 9 章, 介绍一个完整的客户端请求, 包括 Web 服务器端处理该请求并响应相关数据的过程。在这个过程中, Nginx 充当 Web 服务器直接处理客户端请求, 而不是转发到后端服务器。

第 10 章, 对于到达 Nginx 的客户端请求, Nginx 首先要做的就是对它进行定位, 也就是找到其所对应的 Server 与 Location, 从而提供处理该请求的正确上下文环境, 请求处理才能得以继续。这就是本章的主要内容。

第 11 章, 与第 9 章内容类似, 但本章提到的 Nginx 充当的是代理角色, 即它把请求转发给后端 PHP 服务器进行处理, 然后接收其响应并把响应数据发给最终客户端, 即一个完整的 Nginx+Fastcgi+PHP 的客户端请求处理响应过程。

第 12 章, 详细介绍一些 Filter 模块实例, 比如 ngx_http_not_modified_filter_module、ngx_http_headers_filter_module 等。

第 13 章, 详细介绍两种 Load-balance 策略的实现, 即加权轮询策略与 IP 哈希策略。

第 14 章, 详细介绍 Nginx 如何通过 Handler 模块实现对客户端的访问控制。

另外, 附录 A 给出的是默认 configure 下的 Nginx 编译模块, 附录 B 是编写本书时所默认使用的 Nginx 运行配置, 而附录 C 给出一个 HTTP 状态码简单介绍的列表, 方便读者查阅。

最后, 就我个人而言, 本书最大的遗憾就是其目前的层次仅达到 What 与 How 的程度, 即它只介绍了 Nginx 是什么, 有什么功能特性, 内部如何运作, 具体如何实现, 而没有介绍其为什么是这样而不是那样, 为什么这样的设计能达到高性能。限于时间关系, 这第三个层次 (即 Why) 的内容就暂留给各位读者自己去独立思考吧。不过在此之前, 需要充分理解和

掌握本书所介绍的前两个层次内容。

致谢

首先，感谢 Igor Sysoev¹大师以及 Nginx Inc.²为我们提供了如此优秀的开源项目，他们对 Nginx 的持续改进，让我们看到越来越强大的 Nginx。

感谢互联网上众多的知识分享者，帮助我对 Nginx 代码的理解，特别是在我开始重新整理 Nginx 文档时，互联网上已涌现出大量的 Nginx 相关文档，虽然比较零散，但的确加速了我对文档的整理进度。这些内容大多来自博客、论坛、问答等，虽然我无法把他们一一列出，但这并不妨碍我对他们的感激之情。部分参考链接如下。

<http://openresty.org/>

<http://tengine.taobao.org/book/index.html>

<http://blog.csdn.net/dingyujie/article/category/782920>

<http://www.aosabook.org/en/nginx.html>

<http://www.evanmiller.org/nginx-modules-guide.html>

<http://www.evanmiller.org/nginx-modules-guide-advanced.html>

<http://blog.lifeibo.com/?cat=4>

<http://www.pagefault.info/?cat=7>

感谢本书编辑对我的各种指导，使我的写作水平有了很大的提高。

最后，感谢在工作和生活中帮助过我的所有人，感谢你们，正是因为有了你们，才有了本书的面世。

本书的相关网站

我维护了一个包含本书相关信息的网站：http://lenky.info/ngx_book/，其中主要包括了本书

¹ <http://sysoev.ru/en/>

² <http://www.nginx.com/>

的勘误表、内容扩展和修改；同时，如果读者有关于本书或 Nginx 的任何问题，都可以在此页面进行留言，我会尽量经常查看并且及时回复读者提出的相关问题。另外，我的个人电子邮箱为：lenky0401@gmail.com。

作为本书的作者，我以兢兢业业的态度力求做到准确无误，但限于时间仓促和个人水平，书中难免会有一些纰漏，这不乏笔误，有叙述不清楚的地方，甚至有我个人理解的错误。针对这些可能存在的不足给读者带来的任何困惑与阻碍，在此提前说声抱歉，请各位读者和同行多多包涵，如果能提出宝贵的意见，我将洗耳恭听，感激不尽。最后，也请大家时时关注一下上面提到的这个网址，有任何问题，我都会及时在那公布。

作者简介



高群凯，重庆大学计算机硕士毕业，在技术领域的兴趣主要包括 Linux、Lighttpd、Nginx、Mongodb，以及 X86、MIPS（Cavium、RMI、Tilera）等硬件架构。毕业之后，一直在深信服科技有限公司¹工作，专职从事 Linux 相关研究，对系统底层性能优化、内核宕机疑难排查等相关技术领域具有一定的经验。曾著有《Lighttpd 源码分析》一书。

¹ <http://www.sangfor.com/>

目 录

第 1 章 源码分析的准备工作	1
1.1 主要特性	1
1.2 源码下载	4
1.3 源码目录结构	5
1.4 源码分析工具	6
1.5 测试辅助工具	7
1.6 编译与执行	8
1.7 其他准备	9
第 2 章 跟踪与调试	10
2.1 利用 gdb 调试	10
2.1.1 绑定 Nginx 到 gdb	10
2.1.2 gdb 的 watch 指令	13
2.1.3 Nginx 对 gdb 的支持	14
2.1.4 宏	16
2.1.5 cgdb	18
2.2 利用日志信息跟踪 Nginx	19
2.3 利用 strace/pstack 调试 Nginx	21
2.4 获得 Nginx 程序完整执行流程	23
2.5 加桩调试	29
2.6 特殊应用逻辑的调试	31
第 3 章 进程模型	35
3.1 整体架构	35
3.2 核心进程模型	36

3.2.1	监控进程.....	36
3.2.2	工作进程.....	38
3.3	Cache 进程模型.....	38
3.3.1	Cache 管理进程.....	40
3.3.2	Cache 加载进程.....	42
3.4	进程通信.....	43
3.5	共享内存.....	46
3.6	slab 机制.....	50
3.7	信号处理.....	59
3.7.1	准备工作.....	60
3.7.2	设置生效.....	61
3.7.3	处理实例.....	62
第 4 章	数据结构.....	65
4.1	内存池.....	65
4.2	Hash.....	72
4.3	Radix tree.....	77
第 5 章	配置解析.....	81
5.1	配置文件格式.....	81
5.2	配置项目解析准备.....	84
5.3	配置文件解析流程.....	86
5.4	配置信息组织结构.....	95
5.5	配置信息的继承.....	108
第 6 章	模块综述.....	113
6.1	Handler 模块.....	116
6.2	Filter 模块.....	124
6.3	Upstream 模块.....	127
6.4	Load-balance 模块.....	132

第 7 章 事件管理机制	134
7.1 I/O 多路复用模型	134
7.2 epoll 模型	138
7.3 事件处理	144
7.4 负载均衡	146
7.4.1 客户端请求均衡	146
7.4.2 多核绑定	152
7.5 超时管理	153
第 8 章 变量机制	160
8.1 初识变量	160
8.2 支撑机制	163
8.3 脚本引擎	174
8.4 执行顺序	180
第 9 章 请求处理与响应	182
9.1 创建监听套接口	182
9.2 创建连接套接口	188
9.3 请求处理	191
9.4 数据响应	197
9.5 子请求	201
9.6 连接关闭	214
9.6.1 keepalive 机制	217
9.6.2 延迟关闭机制	221
第 10 章 请求定位	228
10.1 Location 的生成	228
10.2 Location 的整理	232
10.3 Server 的定位	239
10.4 Location 的定位与使用	242

10.5	未命名 location 的使用	244
10.6	try_files 指令	248
第 11 章	动态页面请求处理	252
11.1	测试环境	252
11.2	客户端发起 Php 请求	255
11.3	请求转发	259
11.3.1	建立连接	261
11.3.2	发送请求数据	265
11.4	数据响应	267
11.4.1	接收并处理 Fastcgi 响应头	267
11.4.2	接收并发送 Fastcgi 响应体	273
第 12 章	过滤模块	283
12.1	ngx_http_not_modified_filter_module	283
12.1.1	304 状态码	283
12.1.2	412 状态码	285
12.2	ngx_http_headers_filter_module	288
12.2.1	HTTP 缓存功能的扩展	289
12.2.2	增加自定义响应头	290
12.3	ngx_http_gzip_filter_module	291
12.4	ngx_http_range_filter_module	293
第 13 章	负载均衡	299
13.1	负载均衡策略	300
13.2	加权轮询	301
13.2.1	准备工作	301
13.2.2	选择后端服务器	304
13.2.3	后端服务器权值计算	307
13.2.4	释放后端服务器	307
13.3	IP 哈希	310

选择后端服务器.....	310
13.4 两种策略对比.....	313
第 14 章 访问控制	315
14.1 禁止指定 IP 访问	315
14.2 密码认证访问	318
14.3 其他访问控制措施.....	327
14.3.1 禁止用户通过 IP 地址访问服务器.....	327
14.3.2 禁止对某目录或某类文件的访问	327
附录 A	329
附录 B	332
附录 C	333

第 1 章

源码分析的准备工作

从 Nginx（读作 engine x）的官方网站¹，我们可以看到如下介绍：Nginx 是 Igor Sysoev²编写的一款 HTTP 和反向代理服务器，另外它也可以当作邮件代理服务器。它一直被众多流量巨大的俄罗斯网站所使用，例如 Yandex³、Mail.Ru⁴、VKontakte⁵以及 Rambler⁶等。据 Netcraft 统计，截止到 2012 年 8 月份，世界上最繁忙的网站中有 11.48%⁷在使用 Nginx 作为其服务器或者代理服务器。部分典型成功案例有：Netflix⁸、Wordpress.com⁹和 FastMail.FM¹⁰。鉴于 Nginx 的强大性能与稳定性，在国内也有大量的高压力网站在使用 Nginx，如新浪、网易、腾讯、CSDN、酷六、水木社区、豆瓣等。

1.1 主要特性

作为轻量级 HTTP 服务的典型代表，Nginx 除了具备体积小、配置灵活、并发能力强、稳定等众所周知的特点以外，在官方网站还详细列出了 Nginx 的一些主要特性，我们来详细了解一下¹¹。

1. HTTP 服务基本特性

- 处理静态页面请求；

¹ <http://www.nginx.org/en/>

² <http://sysoev.ru/en/>

³ <http://www.yandex.ru/>

⁴ <http://www.mail.ru/>

⁵ <http://vkontakte.ru/>

⁶ <http://www.rambler.ru/>

⁷ <http://news.netcraft.com/archives/2012/08/02/august-2012-web-server-survey.html>

⁸ <https://signup.netflix.com/openconnect/software>

⁹ <http://www.nginx.com/cs/nginx-automattic.html>

¹⁰ <http://blog.fastmail.fm/2007/01/04/webimappop-frontend-proxies-changed-to-nginx/>

¹¹ <http://www.nginx.org/en/>