

Tink's Blog

2019-09-07

《Nginx Cookbook 2019》中文版第一章基础知识

0.0 最前面话

最近看了一本关于NGINX的电子书，编排形式比较新颖，通过一问一答的方式来告诉读者如何快速上手NGINX，以及处理真实项目中的需求。书的内容简练概括，但涉及面较多，内容有4层和7层负载均衡，A/B测试，自动化部署，认证，Http2，debug和调优等，有些示例可以拿来直接用。由于原书暂无中文译本，姑且尝试翻译一下，90%机器翻译，10%人工修正。原书信息如下：

- 名称：Nginx cookbook-Advanced Recipes for High Performance Load Balancing
- 作者：Derek DeJonghe
- 下载地址：
<https://www.nginx.com/resources/library/complete-nginx-cookbook>

1.0 Introduction-简介

To get started with NGINX Open Source or NGINX Plus, you first need to install it on a system and learn some basics. In this chapter you will learn how to install NGINX, where the main configuration files are, and commands for administration. You will also learn how to verify your installation and make requests to the default server.

要开始使用开源版NGINX或NGINX Plus，你首先需要将在操作系统里面安装上并学习一些相关的基础知识。在本章中，你将学习如何安装nginx，其主要配置文件所在位置，以及管理命令。你还将学习如何验证你的安装并向默认服务器发送请求。

1.1 Installing on Debian/Ubuntu-在Debian/Ubuntu系统上安装

Problem-问题

You need to install NGINX Open Source on a Debian or Ubuntu machine.

你需要在Debian或Ubuntu系统的机器上安装开源版NGINX。

Solution-解决方案

Create a file named `/etc/apt/sources.list.d/nginx.list` that contains the following contents:

创建文件 `/etc/apt/sources.list.d/nginx.list`，包含以下内容：

```
deb http://nginx.org/packages/mainline/OS/ CODENAME nginx
deb-src http://nginx.org/packages/mainline/OS/ CODENAME nginx
```

文章目录

1. 0.0 最前面话
2. 1.0 Introduction-简介
3. 1.1 Installing on Debian/Ubuntu-在Debian/Ubuntu系统上安装
 - 3.1. Problem-问题
 - 3.2. Solution-解决方案
 - 3.3. Discussion-讨论
4. 1.2 Installing on RedHat/CentOS-在RedHat/CentOS系统上安装
 - 4.1. Problem-问题
 - 4.2. Solution-解决方案
 - 4.3. Discussion-讨论
5. 1.3 Installing NGINX Plus-安装Nginx Plus
 - 5.1. Problem-问题
 - 5.2. Solution-解决方案
 - 5.3. Discussion-讨论
6. 1.4 Verifying Your Installation-验证你的安装
 - 6.1. Problem-问题
 - 6.2. Solution-解决方案
 - 6.3. Discussion-讨论
7. 1.5 Key Files, Commands, and Directories-关键文件，命令和目录
 - 7.1. Problem-问题
 - 7.2. Solution-解决方案
 - 7.3. Discussion-讨论
8. 1.6 Serving Static Content-处理静态内容
 - 8.1. Problem-问题
 - 8.2. Solution-解决方案
 - 8.3. Discussion-讨论
9. 1.7 Graceful Reload-优雅重载
 - 9.1. Problem-问题
 - 9.2. Solution-解决方案
 - 9.3. Discussion-讨论

分类

Elasticsearch
Golang

- 项目总结

Golang源码分析系列
Golang设计模式系列
工具使用
并发编程
并发编程系列
开发语言
摘记
经济
编码与协议
翻译
读书
读书笔记

标签云

12-factor 2FA API API网关 Ansible
Awk CSRF Cheat Sheet Consul
Coredns Data Science Dbounce Devops
Docker Elasticsearch小技巧
Elasticsearch优化 Elasticsearch内存占用
GDB GMP模型 Golang Go通道 HTTP
HTTP Cache HTTP协议 HTTP状态码
JAVASCRIPT Jupyter notebook Linux命令
Linux命令大全 Linux配置 Logrotate Lua
Nginx Opcache PHP PHP-FPM
PHP优化 Pandas Prometheus QPS
RESTful Redis Redis主从复制
Redis系统服务 SQL注入 SSH Systemd
Throttle Webgrind Web安全 Wireshark
XSS Xdebug context docker golang
golang defer grpc iptables k8s nat
netstat nil通道 protobuf pyspark
python spark sublime sublime插件
tmux tow-factor authentication 东西流量
人生 代理 代码模式 代码质量 供给侧
函数选项 分布式 分布式ID 分析 南北流量
原创翻译 原子操作 双因素认证 反射
反射三定律 取模运算 吞吐量 完全指南
年度回顾 并发数 并发编程 微服务
性能优化 性能优化工具 性能指标 技术分享
持续集成 持续集成工具 效率神器 数学证明
日志管理 时区 服务发现 服务网格 格言
汇编 爱情 程序内存布局 算法 终端复用
统计学 统计思维 翻译 自动化部署 自旋锁
告改增 设计模式 调试工具 软件架构 通道
配置中心 配置管理 蓝构 锁 错误与异常
问题汇总 阅读习惯

归档

二月 2023
八月 2022
七月 2022
五月 2022
十二月 2021

Alter the file, replacing OS at the end of the URL with ubuntu or debian, depending on your distribution. Replace CODENAME with the code name for your distrobuton; jessie or stretch for Debian, or trusty, xenial, artful, or bionic for ubuntu. Then, run the following commands:

根据你的系统发行版本，使用ubuntu或debian替换上述文件中url网址中的 OS。使用你的系统发行版本的代号替换上述文件中的 CODENAME，Debian的代号有jessie或者stretch，ubuntu代号有trusty，xenial，artful，or bionic。接下来运行以下命令：

```
wget http://nginx.org/keys/nginx_signing.key
apt-key add nginx_signing.key
apt-get update
apt-get install -y nginx
/etc/init.d/nginx start
```

Discussion-讨论

The file you just created instructs the apt package management system to utilize the Official NGINX package repository. The commands that follow download the NGINX GPG package signing key and import it into apt. Providing apt the signing key enables the apt system to validate packages from the repository. The apt-get update command instructs the apt system to refresh its package listings from its known repositories. After the package list is refreshed, you can install NGINX Open Source from the Official NGINX repository. After you install it, the final command starts NGINX.

你刚刚创建的文件指示apt包管理系统使用正式的NGINX包仓库。接下来的命令将下载NGINX GPG包签名密钥，并将其导入apt中，apt包管理系统将签名密钥来验证来自仓库的文件。apt-get update 命令指示apt系统从其已知的参考库中更新其包列表。软件包列表刷新后，你可以从正式的NGINX仓库中安装开源版NGINX。安装后，最后的命令会启动NGINX。

1.2 Installing on RedHat/CentOS-在RedHat/CentOS系统上安装

Problem-问题

You need to install NGINX Open Source on RedHat or CentOS.

你需要在RedHat或者CentOS系统上安装开源版本NGINX

Solution-解决方案

Create a file named /etc/yum.repos.d/nginx.repo that contains the following contents:

创建文件 /etc/yum.repos.d/nginx.repo，包含以下内容：

```
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/mainline/OS/OSRELEASE/$basearch/
gpgcheck=0
enabled=1
```

Alter the file, replacing OS at the end of the URL with rhel or centos, depending on your distribution. Replace OSRELEASE with 6 or 7 for version 6.x or 7.x, respectively. Then, run the following commands:

创建上述文件之后，根据你的发行系统，使用rhel或者centos替换文件中url地址的 OS。对于版本6.x或7.x，分别用6或7替换 OSRELEASE。接下来运行以下命令：

```
yum -y install nginx
systemctl enable nginx
systemctl start nginx
firewall-cmd --permanent --zone=public --add-port=80/tcp
firewall-cmd --reload
```

Discussion-讨论

The file you just created for this solution instructs the yum package management system to utilize the Official NGINX Open Source package repository. The commands that follow install NGINX Open Source from the Official repository, instruct systemd to enable NGINX at boot time, and tell it to start it now. The firewall commands open

十一月 2021
九月 2021
四月 2021
三月 2021
二月 2021
十二月 2020
十一月 2020
十月 2020
九月 2020
八月 2020
七月 2020
六月 2020
五月 2020
四月 2020
三月 2020
二月 2020
一月 2020
十二月 2019
十一月 2019
十月 2019
九月 2019
八月 2019
七月 2019
六月 2019
五月 2019
三月 2019
十二月 2018
十一月 2018
十月 2018
八月 2018
六月 2018
五月 2018
三月 2018
二月 2018
一月 2018
十二月 2017
十一月 2017
九月 2017
八月 2017
七月 2017
六月 2017
四月 2017
三月 2017
二月 2017
十一月 2016
十月 2016
九月 2016
六月 2016
五月 2016

最新文章

y % x = y & (x - 1)的数学证明
Golang调度机制浅析
并发编程系列：谈谈锁的实现机制
并发编程系列：开篇
CGO使用指南

port 80 for the TCP protocol, which is the default port for HTTP. The last command reloads the firewall to commit the changes.

解决方案中创建的文件指示yum包管理系统使用官方的开源NGINX仓库。接下来的命令将从官方仓库中安装开源版Nginx，并设置系统在开机启动时启动NGINX。firewall命令将打开tcp协议的80端，它是HTTP服务的默认端口。最后的firewall命令将重载配置来提交更改

1.3 Installing NGINX Plus-安装Nginx Plus

Problem-问题

You need to install NGINX Plus.

你需要安装NGINX Plus

Solution-解决方案

Visit http://cs.nginx.com/repo_setup. From the drop-down menu, select the OS you're installing and then follow the instructions. The instructions are similar to the installation of the open source solutions; however, you need to install a certificate in order to authenticate to the NGINX Plus repository.

访问http://cs.nginx.com/repo_setup，从下拉按钮中，选择你要安装在的系统，然后按照说明操作。安装说明类似于开源版NGINX解决方案的安装；但是你需要安装一个证书才能认证到NGINX Plus仓库。

Discussion-讨论

NGINX keeps this repository installation guide up to date with instructions on installing the NGINX Plus. Depending on your OS and version, these instructions vary slightly, but there is one commonality. You must log in to the NGINX portal to download a certificate and key to provide to your system that are used to authenticate to the NGINX Plus repository.

NGINX保持最新的NGINX Plus仓库安装说明。根据你的操作系统和版本，这些说明略有不同，但有一个相同地方是你必须登录到NGINX网站，下载证书和密钥，以提供给您的系统用于对NGINX Plus仓库进行身份验证。

1.4 Verifying Your Installation-验证你的安装

Problem-问题

You want to validate the NGINX installation and check the version.

你想验证NGINX安装和检查版本

Solution-解决方案

You can verify that NGINX is installed and check its version by using the following command:

你可以使用下面命令验证NGINX是否安装和检查其版本

```
$ nginx -v
nginx version: nginx/1.15.3
```

As this example shows, the response displays the version. You can confirm that NGINX is running by using the following command:

正如上面显示，输出了NGINX版本。通过下面命令，你可以确认NGINX是否正在运行：

```
$ ps -ef | grep nginx
root      1738      1  0 19:54 ?    00:00:00 nginx: master process
nginx     1739   1738  0 19:54 ?    00:00:00 nginx: worker process
```

The ps command lists running processes. By piping it to grep, you can search for specific words in the output. This example uses grep to search for nginx. The result shows two running processes, a master and worker. If NGINX is running, you will always see a master and one or more worker processes. For instructions on starting NGINX, refer to the next section. To see how to start NGINX as a daemon, use the init.d or systemd methodologies.

`ps` 命令列出正在运行的进程。通过管道输出到 `grep` 命令，你可以搜索输出内容的关键字。如果NGINX正在运行，你监视看到一个主进程和一个或多个工作进程。我们将在下一节看到启动NGINX指导，如何使用`init.d`或者`systemd`方法启动NGINX作为一个守护进程

To verify that NGINX is returning requests correctly, use your browser to make a request to your machine or use `curl`:

你可以打开浏览器请求你本地地址或者使用`curl`命令来验证NGINX是否返回正确的请求。

```
$ curl localhost
```

You will see the NGINX Welcome default HTML site.

你将会看到NGINX默认的欢迎页面

Discussion-讨论

The `nginx` command allows you to interact with the NGINX binary to check the version, list installed modules, test configurations, and send signals to the master process. NGINX must be running in order for it to serve requests. The `ps` command is a surefire way to determine whether NGINX is running either as a daemon or in the foreground. The default configuration provided by default with NGINX runs a static site HTTP server on port 80. You can test this default site by making an HTTP request to the machine at local host as well as the host's IP and hostname.

`nginx` 命令允许你NGINX二进制文件进行交互来检查版本，列出安装模块，测试配置，以及发送信号给主进程。NGINX必须是运行状态才能提供处理请求。`ps` 命令是检测NGINX以守护进程或者前台方式运行的一个可靠方法。NGINX提供的默认配置会在80端口运行一个静态网站。你可以在本机使用localhost或者主机名字和主机IP来测试这个默认网站

1.5 Key Files, Commands, and Directories-关键文件，命令和目录

Problem-问题

You need to understand the important NGINX directories and commands.

你需要知道重要的NGINX目录和命令

Solution-解决方案

NGINX files and directories

NGINX文件和目录

- `/etc/nginx/`

The `/etc/nginx/` directory is the default configuration root for the NGINX server. Within this directory you will find configuration files that instruct NGINX on how to behave.

`/etc/nginx` 目录是NGINX服务默认配置的根目录。在这个目录中，您找到说明NGINX如何运行的配置文件。

- `/etc/nginx/nginx.conf`

The `/etc/nginx/nginx.conf` file is the default configuration entry point used by the NGINX service. This configuration file sets up global settings for things like worker process, tuning, logging, loading dynamic modules, and references to other NGINX configuration files. In a default configuration, the `/etc/nginx/ nginx.conf` file includes the top-level `http` block, which includes all configuration files in the directory described next.

`/etc/nginx/nginx.conf` 是NGINX服务默认配置的入口文件。该配置文件全局设置了工作进程数，调优，日志，加载动态模块以及导入其他NGINX配置文件。在默认配置中 `/etc/nginx/nginx.conf` 包含了顶级`http` 块。在`http` 块中引入了接下来的所有文件

- `/etc/nginx/conf.d/`

The `/etc/nginx/conf.d/` directory contains the default HTTP server configuration file. Files in this directory ending in `.conf` are included in the top-level `http` block from within the `/etc/ nginx/nginx.conf` file. It's best practice to utilize `include` statements and organize your configuration in this way to keep your configuration files concise. In some package repositories, this folder is named `sites-enabled`, and

configuration files are linked from a folder named `site-available`; this convention is deprecated.

`/etc/nginx/conf.d` 目录包含了默认的HTTP服务配置文件。在该目录下以 `.conf` 结尾的文件，被包含在 `/etc/nginx/nginx.conf` 文件的顶级http块中。最佳实践是利用`include`语句并以这种方式组织你的配置，以保持的配置文件简洁。在某些包仓库中，这个目录命名为 `sites-enabled`，配置文件是指向目录 `site-available` 下文件的软连接，这种转换已经废弃。

- `/var/log/nginx/`

The `/var/log/nginx/` directory is the default log location for NGINX. Within this directory you will find an `access.log` file and an `error.log` file. The access log contains an entry for each request NGINX serves. The error log file contains error events and debug information if the debug module is enabled.

`/var/log/nginx` 是NGINX默认的日志文件位置。在这个目录，你将看到`access.log`文件和`error.log`文件。`access.log`包含了每次请求NGINX服务的日志。`error.log`文件包含了错误和调试信息（如果debug模块开启了话）

NGINX commands

NGINX命令

- `nginx -h`

Shows the NGINX help menu.

显示NGINX帮助菜单

- `nginx -v`

Shows the NGINX version.

显示NGINX版本

- `nginx -V`

Shows the NGINX version, build information, and configuration arguments, which shows the modules built in to the NGINX binary.

显示了NGINX版本、构建信息和配置参数，其中显示了内置于NGINX二进制文件中的模块

- `nginx -t`

Tests the NGINX configuration.

测试NGINX配置

- `nginx -T`

Tests the NGINX configuration and prints the validated configuration to the screen. This command is useful when seeking support.

测试NGINX配置，并打印验证过的配置。此命令在寻求帮助时候很有用。

- `nginx -s signal`

The `-s` flag sends a signal to the NGINX master process. You can send signals such as `stop`, `quit`, `reload`, and `reopen`. The `stop` signal discontinues the NGINX process immediately. The `quit` signal stops the NGINX process after it finishes processing inflight requests. The `reload` signal reloads the configuration. The `reopen` signal instructs NGINX to reopen log files.

s标志向NGINX主进程发送信号。你可以发送停止(`stop`)、退出(`quie`)、重新加载(`reload`)和重新打开(`reopen`)等信号。停止信号会立即中断NGINX进程。退出信号在处理完正在进行的请求后停止NGINX进程。重新加载信号会重新加载配置。重新打开信号指示NGINX重新打开日志文件。

Discussion-讨论

With an understanding of these key files, directories, and commands, you're in a good position to start working with NGINX. With this knowledge, you can alter the default configuration files and test your changes by using the `nginx -t` command. If your test is successful, you also know how to instruct NGINX to reload its configuration using the `nginx -s reload` command.

通过了解关键文件、目录以及命令之后，你可以更好的开始使用NGINX。有了这些知识后，你可以更改默认配置文件并使用nginx -t 命令测试你的更改。如果你的测试是成功的，你也会知道如何使用nginx -s reload命令来指示NGINX重新加载其配置。

1.6 Serving Static Content-处理静态内容

Problem-问题

You need to serve static content with NGINX.
你需要NGINX处理静态内容

Solution-解决方案

Overwrite the default HTTP server configuration located in /etc/nginx/conf.d/default.conf with the following NGINX configuration example:

使用以下nginx配置示例覆盖 /etc/nginx/conf.d/default.conf 中的默认HTTP服务器配置：

```
server {
    listen 80 default_server;
    server_name www.example.com;
    location / {
        root /usr/share/nginx/html;
        # alias /usr/share/nginx/html;
        index index.html index.htm;
    }
}
```

Discussion-讨论

This configuration serves static files over HTTP on port 80 from the directory /usr/share/nginx/html/. The first line in this configuration defines a new server block. This defines a new context for NGINX to listen for. Line two instructs NGINX to listen on port 80, and the default_server parameter instructs NGINX to use this server as the default context for port 80. The server_name directive defines the hostname or names of which requests should be directed to this server. If the configuration had not defined this context as the default_server, NGINX would direct requests to this server only if the HTTP host header matched the value provided to the server_name directive.

上面配置通过HTTP的80端口处理来自 /usr/share/nginx/html 目录下的静态文件。配置的第一行定义了一个新的server块。它定义了一个新的NGINX监听的上下文。第二行设置NGINX监听80端口。server_name 指令定义了应该将哪些请求定向到该服务器的主机名或域名。如果配置没有将此上下文定义为默认服务器，那么只有当HTTP主机头与提供给server_name指令的值匹配时，NGINX才会将请求定向到此服务。

The location block defines a configuration based on the path in the URL. The path, or portion of the URL after the domain, is referred to as the URI. NGINX will best match the URI requested to a location block. The example uses / to match all requests. The root directive shows NGINX where to look for static files when serving content for the given context. The URI of the request is appended to the root directive's value when looking for the requested file. If we had provided a URI prefix to the location directive, this would be included in the appended path, unless we used the alias directory rather than root. Lastly, the index directive provides NGINX with a default file, or list of files to check, in the event that no further path is provided in the URI.

location块根据网址中的路径定义一个配置。路径或URL中域名后面的部分被称为URI。NGINX将最佳匹配URI请求到一个location块。示例中使用 / 来匹配所有请求。root 指令指示NGINX，为给指定的上下文提供内容时，在哪儿找到静态文件。当查找请求的文件时，请求的URI被附加到root指令的值。如果我们为location指令提供了URI前缀，这将被包含在附加路径中，除非我们使用alias目录而不是root目录。最后index指令提供了一个默认文件或者一系列文件，以防止URI中没有提供具体的文件。

1.7 Graceful Reload-优雅重载

Problem-问题

You need to reload your configuration without dropping packets.

你需要在不丢失数据包的情况下重新启动你的配置

Solution-解决方案

Use the reload method of NGINX to achieve a graceful reload of the configuration without stopping the server:

使用NGINX的reloa方法能够实现在不停止服务的情况下重新加载配置

```
$ nginx -s reload
```

This example reloads the NGINX system using the NGINX binary to send a signal to the master process.

本示例使用NGINX二进制文件以向主进程发送信号来重新加载NGINX配置

Discussion-讨论

Reloading the NGINX configuration without stopping the server provides the ability to change configurations on the fly without dropping any packets. In a high-uptime, dynamic environment, you will need to change your load-balancing configuration at some point. NGINX allows you to do this while keeping the load balancer online. This feature enables countless possibilities, such as rerunning configuration management in a live environment, or building an application- and cluster-aware module to dynamically configure and reload NGINX to meet the needs of the environment.

在不停止服务器的情况下重新加载NGINX配置提供了在不丢弃任何数据包的情况下动态更改配置的能力。在high-uptime、动态环境中，你需要在某个时间点更改负载均衡配置，NGINX可以这样做，且能同时保持负载均衡器在线。此特性支持无数种可能性，例如在实时环境中重新运行配置管理，或者构建一个应用程序和集群感知模块来动态配置和重新加载NGINX以满足环境需求。

#Nginx #翻译

留言 Share

NEWER OLDER

《Nginx Cookbook 2019》中文版第二章高性能负载均衡 Ansible如何通过跳板机连接目标机器

