

目录

1	大概思路.....	1
2	了解 Nginx 及文档资源.....	1
3	Nginx 命令模块及进程结构.....	2
4	解读 Nginx 配置	3
5	让“百度”成为自己的“搬运工”	5
6	服务器负载均衡集群.....	7
7	域名解析实现负载均衡.....	8
8	总结.....	10

1 大概思路

- 了解 Nginx 及文档资源
- Nginx 命令模块及进程结构
- 解读 Nginx 配置
- 让“百度”成为自己的“搬运工”
- 服务器负载均衡集群
- 域名解析实现负载均衡
- 总结

2 了解 Nginx 及文档资源

Nginx 是一款轻量级的 Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器，并在一个 BSD-like 协议下发行。其特点是占有内存少，并发能力强，事实上 nginx 的并发能力确实在同类型的网页服务器中表现较好，中国大陆使用 nginx 网站用户有：百度、京东、新浪、网易、腾讯、淘宝等。

中文文档官网：

<http://www.nginx.cn/doc/>

中文文档官网上，拥有了一些示例：

安装

[nginx 在 windows 上安装](#)

[nginx 在 freebsd 上安装](#)

[nginx 在 ubuntu 上安装](#)

[nginx 在 fedora 上安装](#)

[nginx 在 centos 上安装](#)

[nginx php-fpm 安装配置](#)

配置示例和方法

[完整示例](#)

[完整示例（来自 Nginx 站点）](#)

[虚拟主机](#)

[负载均衡示例](#)

[防盗链](#)

[X-Sendfile \(=> X-Accel-Redirect\) in Nginx](#)

[Disable the errorlog entries caused by the HW-Loadbalancer](#)

3 Nginx 命令模块及进程结构

基本命令

命令	描述
nginx -s stop	立即停止守护进程（使用 TERM 信号）
nginx -s quit	温和地停止守护进程（使用 QUIT 信号）
nginx -s reopen	重新打开日志文件
nginx -s reload	重新载入配置文件
nginx -t	测试配置文件
kill/killall nginx	配置服务无法停止，可用该命令终止该进程

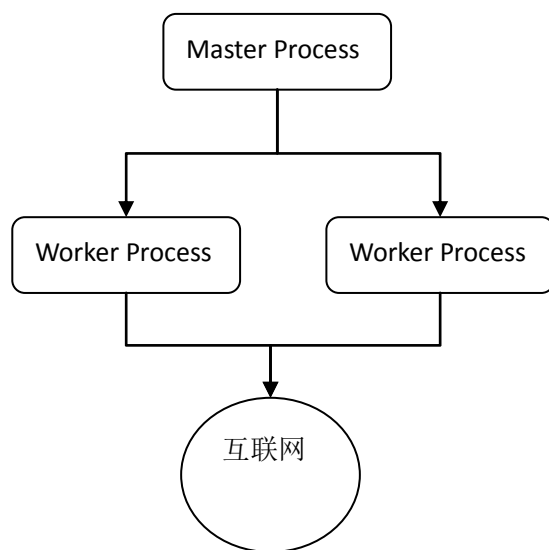
高级的配置，一般有 5 个文件

标准名称	描述
nginx.conf	应用程序的基本配置文件
mime.types	一个文件扩展列表文件，与 MIME 类型关联
fastcgi.conf	与 FastCGI 相关的配置文件
proxy.conf	与 Proxy 相关的配置文件
sites.conf	配置 Nginx 提供的网站，最好每个域建文件

基本模块

名称	描述
核心模块（Core module）	基本特征和指令，例如进程管理和安全
事件模块（Events module）	让你在 Nginx 内部机制配置网络使用能力
配置模块（Configuration module）	提供包含机制

启动 Nginx，一个独一无二的进程存在于内存——master 进程，如果该服务在系统启动时有 init 脚本启动，它会使用当前的用户和用户组来运行——通常为 root/root，master 进程本身不处理任何客户端的请求，它是用来产生进程——worker 进程，这些 worker 进程在配置文件中指定的用户和用户组来运行，定义大量 worker 进程以及每个 worker 进程的最大连接数。



硬件选择

低流量的配置	标准配置	高流量配置
CPU: 双核 RAM: 2GB 请求: ~1/s	CPU: 四核 RAM: 4GB 请求: ~50/s	CPU: 八核 RAM: 12GB 请求: ~1000/s
推荐值		
worker_processes 2; worker_rlimit_nofile 1024; worker_priority -5; worker_cpu_affinity 01 10; events{ multi_accept on; worker_connections 128; }	worker_processes 4; worker_rlimit_nofile 8192; worker_priority 0; worker_cpu_affinity 0001 0010 0100 1000; events{ multi_accept off; worker_connections 1024; }	worker_processes 8; worker_priority 0; events{ multi_accept off; worker_connections 8192; }

4 解读 Nginx 配置

核心模块（Core module）指令

指令和使用环境	描述
master_process	语法: on 或 off 若为 on, Nginx 将开启多个进程, 一个主进程 (即 master 进程) 和 worker 进程; 若为 off, Nginx 会以独一无二的进程来运行。该指令仅被用于测试。
worker_processes	语法: 数字 定义 worker 进程的数量, Nginx 可将请求分到多个 worker 进程
error_log	语法: error_log /file/path level 能够提供不同的错误日志级别: 应用程序、HTTP 服务、虚拟主机和虚拟主机目录
pid	语法: 文件路径 用于存放 Nginx 守护进程的 pid 文件路径。默认值为编译时配置的路径

事件模块（Event Module）指令

指令和使用环境	描述
worker_connections	语法: 数字 定义一个 worker 进程能够同时连接的数量
multi_accept	语法: on 或 off 定义 Nginx 是否立刻从所有监听队列进入的连接

配置模块（Configuration module）指令（涉及的主要是 HTTP 核心模块，下面再补充）

Nginx.conf 解读：

```
#user nobody;
#定义 worker 进程的数量，Nginx 可将请求分到多个 worker 进程
worker_processes 1;

#能够提供不同的错误日志级别：应用程序、HTTP 服务、虚拟主机和虚拟主机目录
#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;

#用于存放 Nginx 守护进程的 pid 文件路径。默认值为编译时配置的路径
#pid logs/nginx.pid;

#定义一个 worker 进程能够同时连接的数量
events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;

    server {
        listen 80;
        server_name localhost;
        location / {
            root html;
            index index.html index.htm;
        }
        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
            root html;
        }
    }
}
```

Nginx 以上面配置为基础，提供 Web 服务

- 通过 80 监听 tcp 套接字
- 访问地址 <http://localhost/>
- 主页为 index.html

在 windows 的 cmd 下，进入 nginx 目录，输入：

```
D:\DTLDownloads\nginx-1.10.2>start nginx
```

```
D:\DTLDownloads\nginx-1.10.2>nginx -t
```

```
nginx: the configuration file D:\DTLDownloads\nginx-1.10.2/conf/nginx.conf syntax is ok
```

```
nginx: configuration file D:\DTLDownloads\nginx-1.10.2/conf/nginx.conf test is successful
```

```
D:\DTLDownloads\nginx-1.10.2>
```

运行效果如下：



5 让“百度”成为自己的“搬运工”

如何使用 Nginx 代理，使“百度”成为搬运工，主要设置 `server_name` 和 `proxy_pass` 属性。这里涉及了配置模块，主要是 HTTP 核心模块。

HTTP 核心模块包含 HTTP 服务器所有基本的区段（block）、指令和变量。主要有 `http`，`server` 和 `location` 这三个主要的区段结构。

- `http` 该区段嵌入配置文件的根部。
- `server` 这个区段允许你声明一个站点，能够设定网站（通过主机名）。这样通过 Nginx 实现公认的服务器。该区段只能用在 `http` 区段。
- `location` 定义一组设置，应用于网站的一个特定位置。该区段能够用于 `server` 区段，也能嵌套在其它 `location` 中。

指令和使用环境	描述
<code>server_name</code>	语法： <code>server_name hostname1[hostname2...]</code> 在 <code>server</code> 区段定义一个或者多个主机名
<code>sendfile</code>	语法： <code>on</code> 或 <code>off</code> Nginx 将使用 <code>sendfile</code> 内核来调用处理文件传递。
<code>root</code>	语法： 目录路径

	定义文档根目录
keepalive_timeout	语法: keepalive_timeout time1[time2...] keep-alive 能够使用客户端服务器的连接在一定时间内持续有效

Nginx.conf 配置如下:

```
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;
    server {
        listen 80;
        server_name localhost zhyongfeng;
        location / {
            proxy_pass https://www.baidu.com;
        }
        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
            root html;
        }
    }
}
```

运行效果:





6 服务器负载均衡集群

不同主机端口, 通过 localhost:80 端口进行负载均衡集群访问, 使用 localhost 对 WEB API 配置如下:

```
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;

    upstream zhyongfeng.com {
        #可进行 down 测试;
        server 10.92.202.56:5600;
        server 10.92.202.57:5700;
        server 10.92.202.58:5800;
    }

    server {
        listen 80;
        server_name localhost;
        location / {
            proxy_pass http://zhyongfeng.com;
        }
        error_page 500 502 503 504 /50x.html;
```



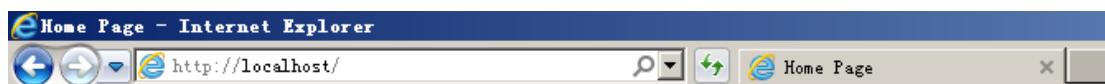
```
location = /50x.html {  
    root    html;  
}  
}
```

运行 CMD:

```
D:\DTLDownloads\nginx-1.10.2>start nginx
```

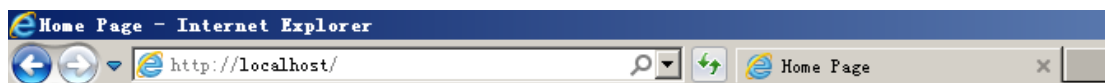
```
D:\DTLDownloads\nginx-1.10.2>nginx -s reload
```

访问: <http://localhost/>, 运行结果:



这是10.92.202.57的机器

© 2017 - My ASP.NET Application



这是10.92.202.56的机器

© 2017 - My ASP.NET Application



这是10.92.202.58的机器

© 2017 - My ASP.NET Application

7 域名解析实现负载均衡

不同主机端口, 通过自定义域名 `zhyongfeng.com: 80` 端口进行负载均衡集群访问, 则访问 `C:\Windows\System32\drivers\etc\hosts`, 添加下列“本机 IP 自定义的域名”:

```
10.93.85.66 zhyongfeng.com
```

使用 `zhyongfeng.com` 对 WEB API 配置如下:

```
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;

    upstream zhyongfeng.com {
        server 10.92.202.56:5600;
        server 10.92.202.57:5700;
        server 10.92.202.58:5800;
    }
    server {
        listen 80;
        server_name zhyongfeng.com;
        location / {
            proxy_pass http://zhyongfeng.com;
        }
    }
}
```

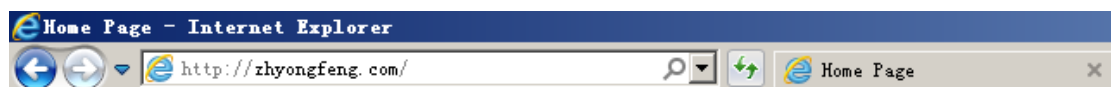
运行 CMD，重新 reload 加载配置，停止后重新启动：

```
D:\DTLDownloads\nginx-1.10.2>nginx -s reload
```

```
D:\DTLDownloads\nginx-1.10.2>nginx -s stop
```

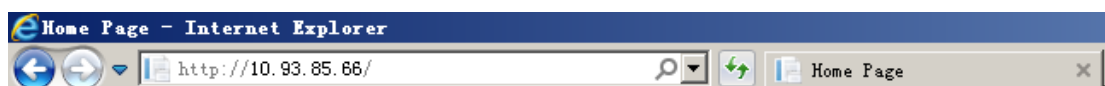
```
D:\DTLDownloads\nginx-1.10.2>start nginx
```

访问：<http://zhyongfeng.com/>，<http://localhost/>，<http://10.93.85.66/>的运行结果：



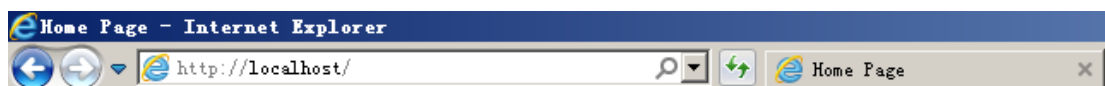
这是10.92.202.58的机器

© 2017 - My ASP.NET Application



这是10.92.202.57的机器

© 2017 - My ASP.NET Application



这是10.92.202.58的机器

© 2017 - My ASP.NET Application

8 总结

Nginx 可以使用 Rewrite 模块建立高级重写规则，可以和 PHP、Python 结合，也可以替换 Apache，或者作用于现有 HTTP 服务器前端，能够轻松建立虚拟主机配置，既能反向代理又可以是电子邮件（IMAP/POP3）代理服务器，功能强大，可以深入学习。

Windows 的 Nginx 资源下载：

http://download.csdn.net/download/ruby_matlab/10117702

PDF 下载：

[Nginx 集群及代理的应用.pdf](#)