|  |
| --- |
| 文章标题  nginx反向代理 |
| **作者：**  **归档：学习笔记** |
| **快捷键：**  Ctrl + 1 标题1  Ctrl + 2 标题2  Ctrl + 3 标题3  Ctrl + 4 实例  Ctrl + 5 程序代码  Ctrl + 6 正文 |
| **格式说明：**  蓝色字体：注释  黄色背景：重要  绿色背景：注意 |

**的起点！**

LNMP课程回顾部分

①. 部署流程要熟悉

Linux nginx（编译安装） mysql（二进制包安装） php（编译安装）

nginx+PHP结合：设置fastcgi的配置

nginx+mysql结合：编写程序代码（wordpress初始化即可让nginx与mysql建立连接）

② 排错过程说明

· nginx是否能用

· nginx与PHP测试（phpinfo）

· nginx与MySQL测试（test\_mysql）---mysql用户授权问题

如果不在一台机器上，测试过程有所不同，需要修改test\_mysql.php测试文件内容

回顾LNMP整体架构部署流程，根据部署流程图回顾

## Nginx反向代理知识

### Nginx反向代理环境部署准备：（迎宾小姐）

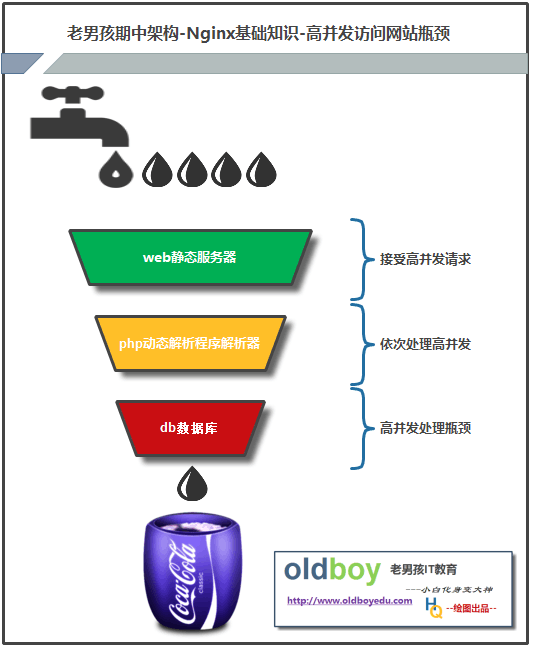
准备好lb01 10.0.0.5 安装好nginx 1.10.3 编译参数和web01上面一样

准备好三台web服务器，三台web服务器安装好nginx即可

web01（10.0.0.8） web02（10.0.0.7） web03（10.0.0.9）

安装好wireshark软件（群文件中 wireshark-win64-2.2.2）

### 集群的概念



#### 集群概念介绍说明：一堆干相同事情的服务器，称为集群

#### 集群作用：

01. 处理高性能（Performance）问题

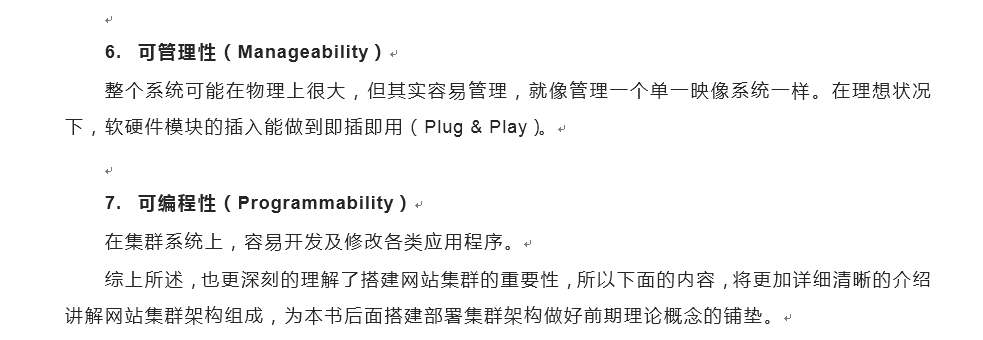
02. 价格有效性（Cost-effectiveness）

03. 可伸缩性（Scalability）

04. 高可用性（Availability）

05. 透明性 ：多个独立计算机组成的松耦合集群系统构成一个虚拟服务器。用户或客户端程序访问集群系统时，就像访问一台高性能、高可用的服务器一样，集群中一部分服务器的上线、下线不会中断整个系统服务，这对用户也是同名的。

06．客观理性



#### 集群概念的特点说明：高可用 高性能 （核心特点说明）

#### 集群种类：

负载均衡集群（Load balancing clusters），简称LBC或者LB。 ---解决调度问题（迎宾）

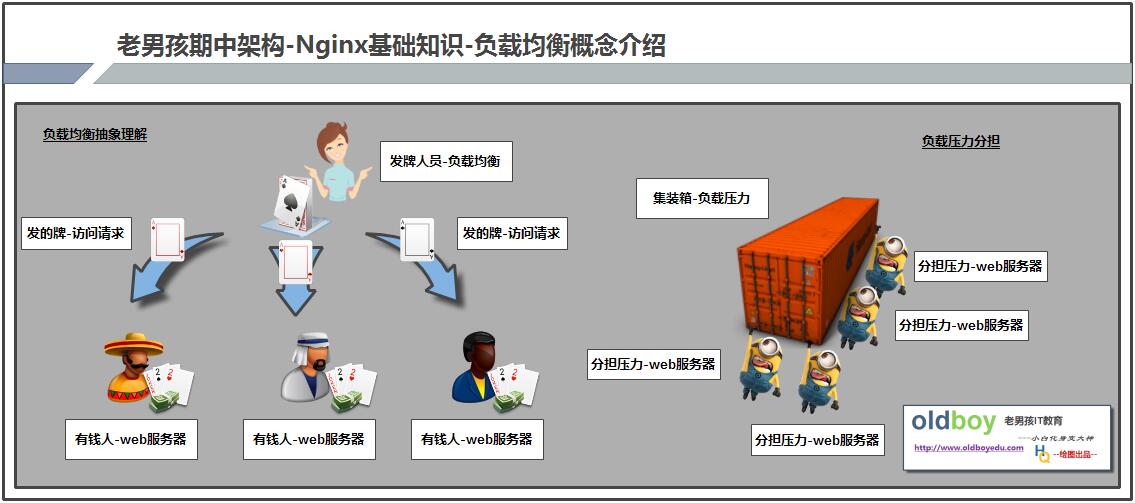
? 高可用性集群（High-availability（HA）clusters），简称HAC。 ---解决单点故障问题（keepalived）

? 高性能计算集群（High-performance（HP）clusters），简称HPC。

? 网络计算（Grid computing）集群。

#### 集群的分类说明：负载均衡集群 高可用集群

### 负载均衡概念



#### 负载均衡集群概念说明：前面是负载均衡器，后面是集群的节点

#### 负载均衡集群：会出现雪崩效应

#### 高可用集群：利用keepalived实现，一个机器宕机了，另一个机器可以顶替上

#### 反向代理实现软件为：nginx（7层http https ）层次进行回顾---Nginx支持7层，1.9以后也支持4层了

LVS支持4层，负载均衡架构学习haproxy(4层和7层 属于反向代理软件)

#### 硬件负载均衡设备说明：

硬件和软件使用场景（企业中软硬件选型），中小型场景;

门户网站 （LVS Tnginx）大型企业网站;

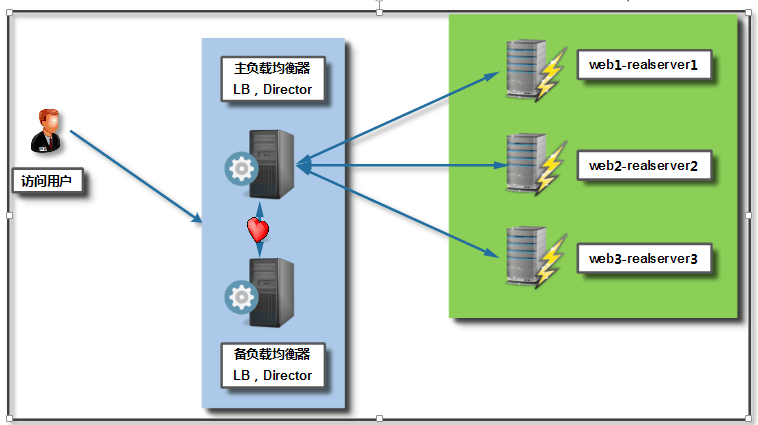
授课选择Nginx进行负载均衡学习即可

#### nginx常用的模块：

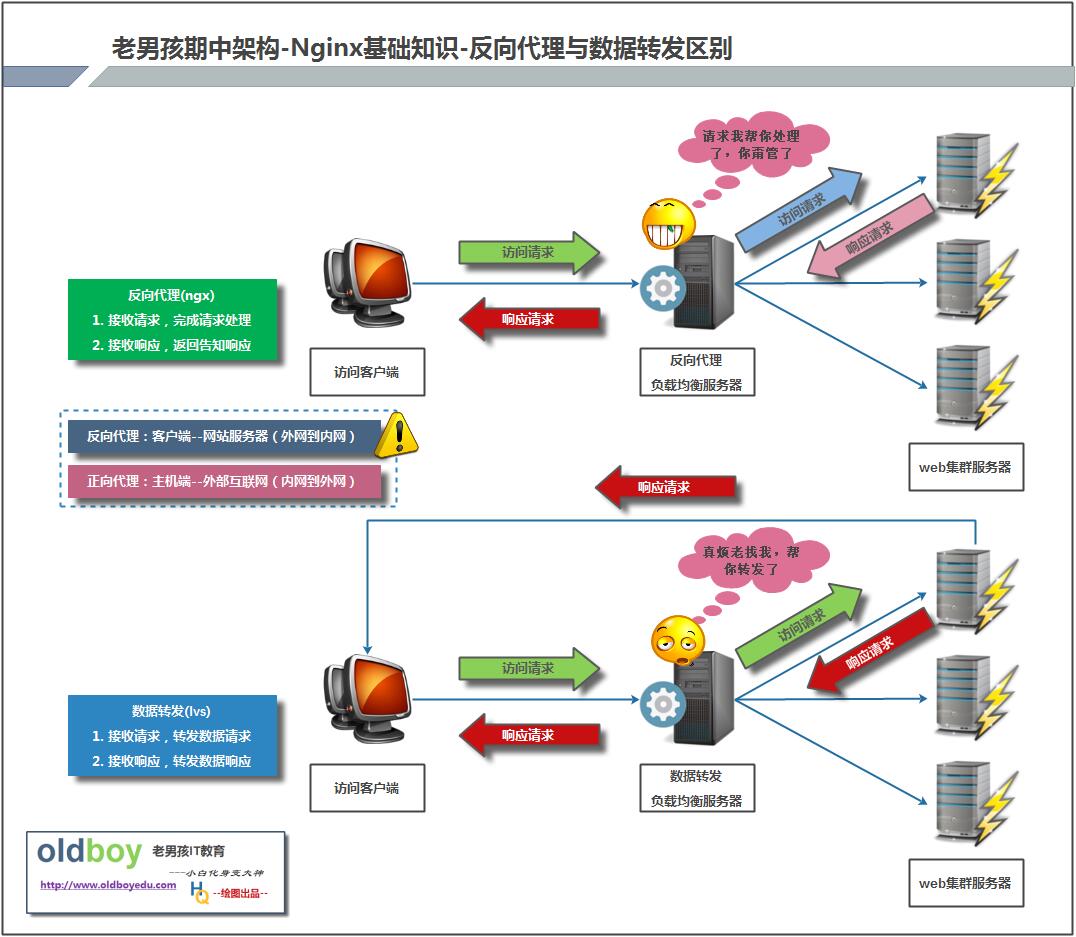
upstream 实现负载均衡和反向代理

proxy

rewrite



## Nginx反向代理实战部署过程



### 安装nginx

#### 执行安装脚本

#!/bin/bash

#yum

yum install -y pcre-devel openssl-devel

#mkdir

if [ ! -d /home/oldboy/tools ]

then

mkdir -p /home/oldboy/tools

fi

cd /home/oldboy/tools

wget -q http://nginx.org/download/nginx-1.10.3.tar.gz

#tar xf

tar xf nginx-1.10.3.tar.gz

#cd dir

cd nginx-1.10.3

#useradd

useradd -s /sbin/nologin -M www

#anzhuang

cd /server/tools/nginx-1.10.3

./configure --prefix=/application/nginx-1.10.3 --user=www --group=www --withule

#make

cd /server/tools/nginx-1.10.3

make && make install

#ln

ln -s /application/nginx-1.10.3 /application/nginx

#### 启动服务

/application/nginx/sbin/nginx

#### 检查语法

/application/nginx/sbin/nginx -t

### 统一编写nginx配置（web服务器端）

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" "$http\_x\_forwarded\_for"';

server {

listen 80;

server\_name bbs.etiantian.org;

location / {

root html/bbs;

index index.html index.htm;

}

access\_log logs/access\_bbs.log main;

}

server {

listen 80;

server\_name www.etiantian.org;

location / {

root html/www;

index index.html index.htm;

}

access\_log logs/access\_www.log main;

}

}

### 统一nginx测试环境（web服务器端），

创建站点目录文件

mkdir -p /application/nginx/html/{www,bbs}

站点目录文件输入内容

for name in www bbs; do echo $name `hostname` >/application/nginx/html/$name/bingbing.html;done

for name in www bbs; do cat /application/nginx/html/$name/bingbing.html;done

说明：在浏览器测试前，重启或启动nginx服务程序（lb01 web01 web02）

### 浏览器测试web服务（利用IP地址进行访问）

### 编辑配置负载服务文件

#### nginx常用模块

nginx常用模块：

ngx\_http\_status\_module

ngx\_http\_ssl\_module

ngx\_http\_log\_module

ngx\_http\_upstream\_module

ngx\_http\_proxy\_module

ansible:

### 负载均衡机lb01 nginx.conf配置

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

upstream server\_pools { 虚拟主机服务池

server 10.0.0.7:80;

server 10.0.0.8:80;

}

server {

listen 80;

server\_name bbs.etiantian.org; 虚拟主机名称

location / {

proxy\_pass http://server\_pools;

}

}

}

#### 负载均衡配置完毕后，进行负载测试

[root@lb01 conf]# curl 10.0.0.5

web01 bbs.etiantian.org

### 排错思路

01. 配置没有问题

02. 服务也正常启动了

03. 错误日志没有错误信息

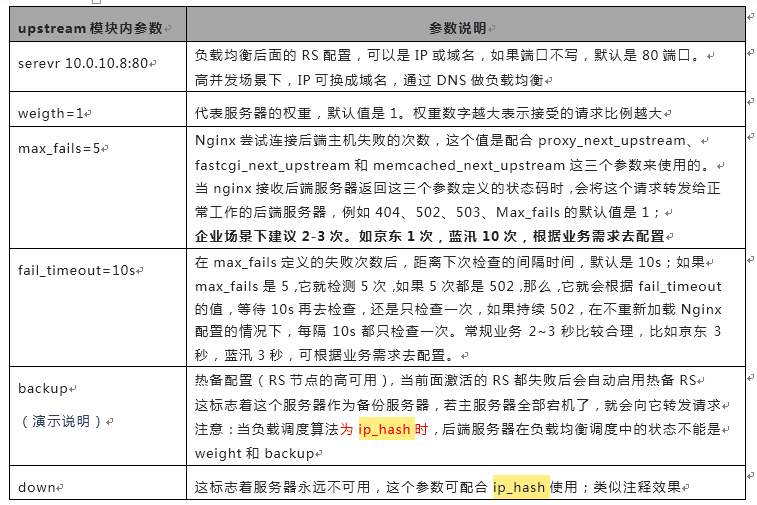
#### 发大招排查

服务或系统重启

重新配置

重新安装（系统重新安装）

## 负载均衡参数介绍-深入参数说明



lb01 nginx.conf multi hosts

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" "$http\_x\_forwarded\_for"';

upstream server\_pools {

server 10.0.0.7 weight=1 max\_fails=3 fail\_timeout=10s;

server 10.0.0.8 weight=1 max\_fails=3 fail\_timeout=10s;

server 10.0.0.9 weight=1 max\_fails=3 fail\_timeout=10s;

}

server {

listen 80;

server\_name www.etiantian.org;

location / {

proxy\_pass http://server\_pools;

}

access\_log logs/access\_www.log main;

}

server {

listen 80;

server\_name blog.etiantian.org;

location / {

proxy\_pass http://server\_pools;

}

access\_log logs/access\_blog.log main;

}

}

### 编写负载均衡多虚拟主机配置

说明：负载配置信息可以参见官方说明 <http://nginx.org/en/docs/http/load_balancing.html>

#### ####lb01 nginx.conf 配置文件

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

upstream server\_pools {

server 10.0.0.7:80;

server 10.0.0.8:80;

}

server {

listen 80;

server\_name bbs.etiantian.org

location / {

proxy\_pass http://server\_pools;

proxy\_set\_header Host $host

}

}

server {

listen 80;

server\_name www.etiantian.org

location / {

proxy\_pass http://server\_pools;

proxy\_set\_header Host $host

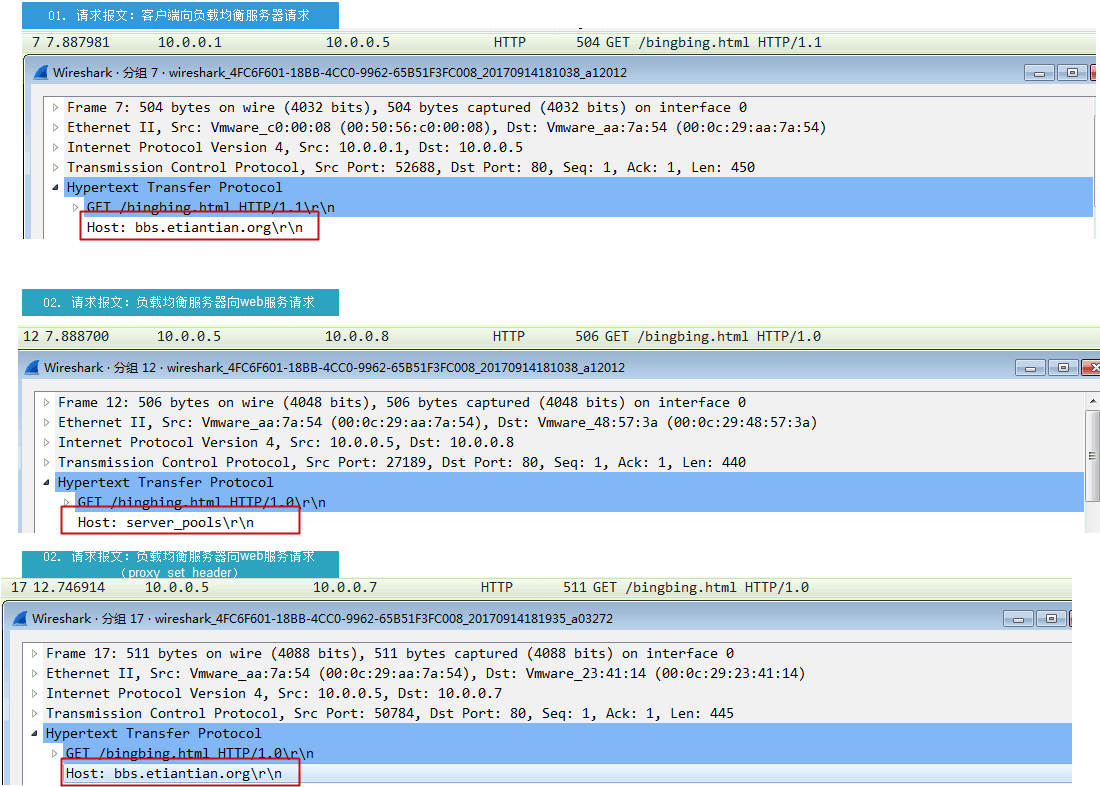
}

}

}

说明：多虚拟主机信息配置完毕，需要修改windows的hosts文件进行域名映射

#### 进行抓包，通过host参数可以看出登录不同虚拟主机时，显示虚拟主机的名称



## Nginx反向代理实战部署过程：

### cookie和session的区别

#### 都可以存放用户的信息

#### cookie 客户端浏览器

1）变量名字和变量的内容

2）开发人员制定

3）每个网站（域名）对应着 cookie是相同

4）钥匙（字符串 文本）

#### session 服务器

1）用户密码 登录信息（登录） ----打包 加密---锁头

2）不用重新登录

### Nginx反向代理实战抓包分析

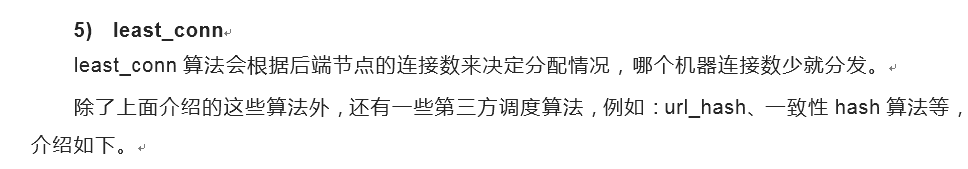
①. windows上进行抓包

②. 指定抓包的网卡信息

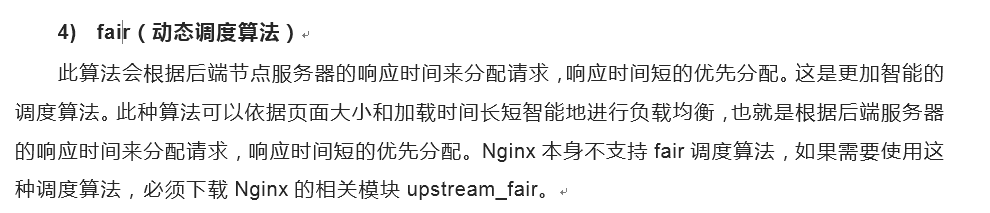
### 模块调度算法

解决用户多次登录的问题，登录不同的web服务器，需要多次登录的问题。

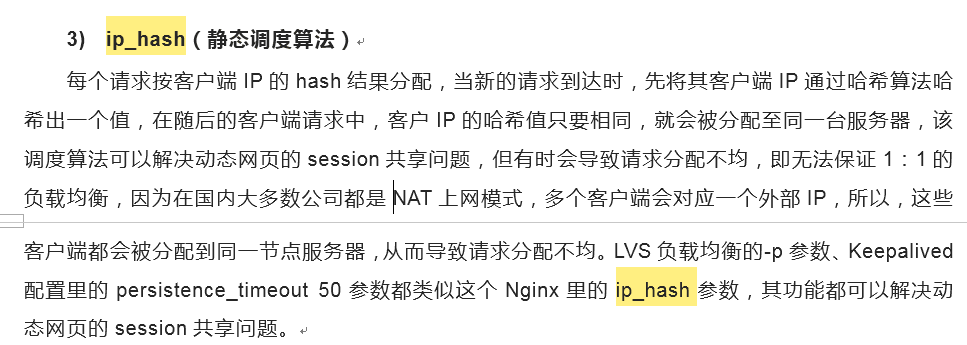
#### least\_conn 算法



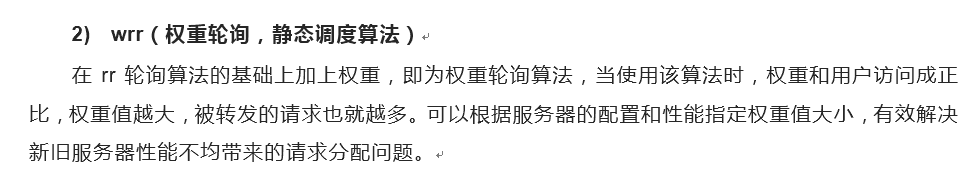
#### fair 算法



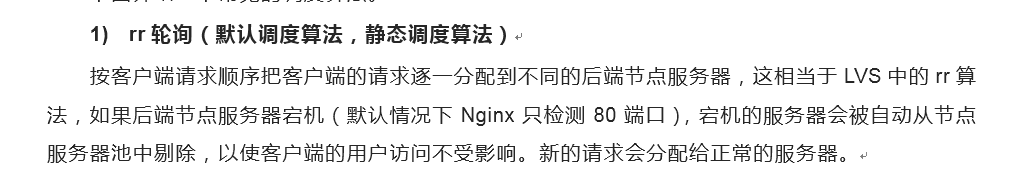
#### ip\_hash 算法



#### wrr 算法



#### rr 算法



## Nginx反向代理与负载均衡深入讲解

### nginx反向代理实现web服务器获悉真实用户信息

#### 部署lb01 配置文件

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

upstream server\_pools {

server 10.0.0.7:80;

server 10.0.0.8:80;

}

server {

listen 80;

server\_name bbs.etiantian.org;

location / {

proxy\_pass http://server\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

}

server {

listen 80;

server\_name www.etiantian.org;

location / {

proxy\_pass http://server\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

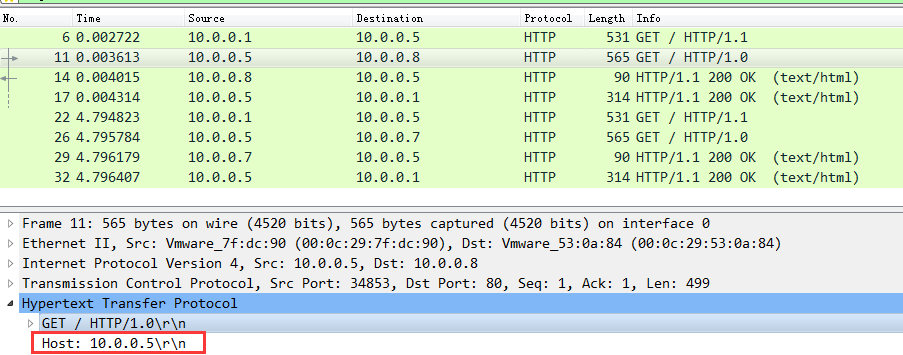
}

}

#### 抓包查看

proxy\_set\_header Host $host; 显示虚拟主机名称

proxy\_set\_header X-Forwarded-For $remote\_addr; 访问端IP



### nginx.conf lb01 基于目录（uri）进行转发

#### 第一个里程碑：规划 分类

/upload 10.0.0.8:80 html/www/upload upload服务器

/static 10.0.0.7:80 html/www/static static静态服务器

/ 10.0.0.9:80 html/www 默认

#### 第二个里程碑：创建upstream负载信息

upstream upload\_pools {

server 10.0.0.8:80;

}

upstream static\_pools {

server 10.0.0.7:80;

}

upstream default\_pools {

server 10.0.0.9:80;

}

#### 第三个里程碑：如何调用upstream信息

注意：

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

必须在location 模块下使用才可以生效

location /static/ {

proxy\_pass http://static\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

location /upload/ {

proxy\_pass http://upload\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

location / {

proxy\_pass http://default\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

#### 第四个里程碑：配置lb负载均衡集群

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" "$http\_x\_forwarded\_for"';

upstream upload\_pools {

server 10.0.0.8:80;

}

upstream static\_pools {

server 10.0.0.7:80;

}

upstream default\_pools {

server 10.0.0.9:80;

}

server {

listen 80;

server\_name www.etiantian.org;

location /static/ {

proxy\_pass http://static\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

location /upload/ {

proxy\_pass http://upload\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

location / {

proxy\_pass http://default\_pools;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

}

access\_log logs/access\_www.log main;

}

}

#### 第五个里程碑-创建环境

www.etiantian.org/bb.html

www.etiantian.org/upload/bb.html

www.etiantian.org/static/bb.html

##web01

mkdir -p /application/nginx/html/www/upload

echo "web01 upload" >/application/nginx/html/www/upload/xiaoyu.html

##web02

mkdir -p /application/nginx/html/www/static

echo "web02 static" >/application/nginx/html/www/static/xiaoyu.html

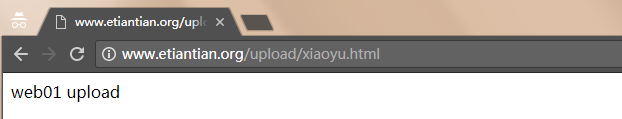
##web03

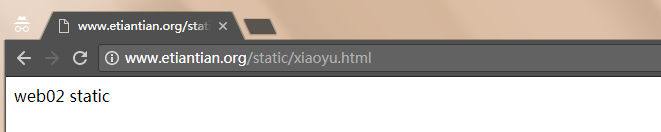
mkdir -p /application/nginx/html/www/

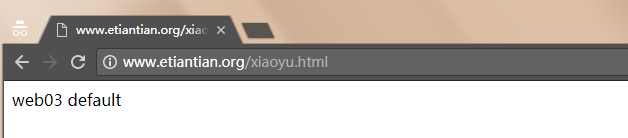
echo "web03 default" >/application/nginx/html/www/xiaoyu.html

#### 第六个里程碑-进行测试

登录浏览器测试







本地测试

[root@lb01 conf]# curl www.etiantian.org/upload/xiaoyu.html

web01 upload

[root@lb01 conf]# curl www.etiantian.org/static/xiaoyu.html

web02 static

[root@lb01 conf]# curl www.etiantian.org/xiaoyu.html

web03 default

### 根据用户不同的浏览器 客户端进行转发

不同的客户端 访问不同的网站

手机和电脑 访问相同的网站----结果不同

#### 设置变量

> GET / HTTP/1.1

> User-Agent: curl/7.19.7 (x86\_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.21 Basic ECC zlib/1.2.3 libidn/1.18 libssh2/1.4.2

> Host: www.baidu.com

> Accept: \*/\*

User-Agent-------的内容-----nginx 有一个变量$http\_user\_agent

#### 修改配置文件 nginx.conf lb01

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

upstream upload\_pools {

server 10.0.0.8:80;

}

upstream static\_pools {

server 10.0.0.7:80;

}

upstream default\_pools {

server 10.0.0.9:80;

}

server {

listen 80;

server\_name www.etiantian.org;

location / {

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $remote\_addr;

if ($http\_user\_agent ~\* "Chrome")

{

proxy\_pass http://static\_pools;

}

if ($http\_user\_agent ~\* "iphone")

{

proxy\_pass http://upload\_pools;

}

proxy\_pass http://default\_pools;

}

}

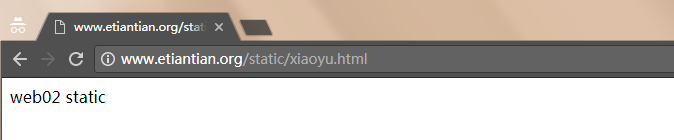
}

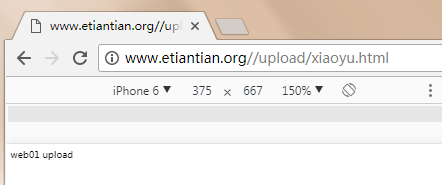




#### 进行测试

浏览器测试





本地测试

curl -A 指定user\_agent -A/--user-agent <string> User-Agent to send to server (H)

[root@lb01 conf]# curl -A Chrome www.etiantian.org/static/xiaoyu.html

web02 static

[root@lb01 conf]# curl -A iphone www.etiantian.org/upload/xiaoyu.html

web01 upload