小米集团训练营SRE大作业

作业简介和功能介绍

作业简介：

- 场景1.用户通过www.example.com的方式访问lvs服务，lvs将请求均衡转发到nginx服务，然后根据不同请求路径/静态文件，nginx返回不同html文件内容
- 场景2.通过shell或者python脚本，去分析2台nginx机器上的access 日志，把qps（每秒的请求数）和http code为200 和500+的数量按照1分钟统计出来， 都保存到mysql数据库中。
- 功能介绍:

通过搭建lvs负载均衡集群，（我采用的是lvs四层负载均衡NAT模式，调度算法为轮询）能够将来自客户端的请求通过lvs均衡转发到后端的nginx服务器中，服务器能够根据客户端的不同请求返回对应的文件内容。同时，通过编写shell脚本，对后端nginx服务器的日志进行分析，将分析结果返回到mysql数据库中，通过对mysql数据库中的数据进行分析，来判断后端nginx服务器的状态。分析结果包含nginx服务器每分钟接收到的请求数，以每分钟http code（状态码）为200和500+的数量。

环境搭建

环境需求：

centos7系统

客户端虚拟主机： IP:172.30.154.110(桥接模式)

LVS服务器主机：

- 网络配置: VIP:172.30.154.100(桥接模式)

  DIP:192.168.222.132(NAT模式)

- 软件安装：ipvsadm工具，mysql5.7.44

nginx服务器主机：

- 网络配置：

  nginx服务器主机1:IP:192.168.222.135(NAT模式)

  nginx服务器主机2:IP:192.168.222.136(NAT模式)

- 软件安装:mysql5.7.44以及nginx

环境安装与启动

MySQL安装与启动：

- 卸载内置环境

```
ps axj | grep mariadb #检查是否有mariadb
systemctl stop mariadb.service #停止mariadb服务如果有的话
rpm -qa | grep mariadb
rpm -qa | grep mysql #检查系统的安装包
yum remove +安装包名 #卸载搜索出来的对应的安装包
wget http://repo.mysql.com/mysql57-community-release-el7-10.noarch.rpm #获取mysql官方yum源
```

- 安装MySQL yum源

```
rpm -Uvh mysql57-community-release-el7-10.noarch.rpm
```

- 开始安装

```
yum install -y mysql-community-server
```

- 启动MySQL服务并检查启动情况

```
systemctl start mysqld.service
systemctl status mysqld.service
```

MySQL登入并在LVS的mysql中设置可远程连接

- 获取MySQL临时密码

```
grep 'temporary password' /var/log/mysqld.log
```

- 使用临时密码登入MySQL

```
mysql -uroot -p
```

- 修改MySQL密码(其中root为用户名，localhost为主机名)

```
use mysql;
alter user 'root'@'localhost' identified by 'newpassword';
```

- 设置远程连接

```
update user set host = "%" where user = "root";
flush privileges;
```

ipvsadm工具安装以及相关模块加载：

- ipvsadm安装：

```
yum install ipvsadm
```

- 加载相关模块

```
modprobe ip_vs #加载ip_vs模块
sysctl -w net.ipv4.ip_forward=1 #启动临时转发
```

nginx服务器安装：

- ```
  yum install yum-utils
  vim /etc/yum.repos.d/nginx.repo
  yum install nginx
  ```

将以下内容放入到编辑文件中：

[nginx-stable]

name=nginx stable repo

baseurl=http://nginx.org/packages/centos/$releasever$/basearch/

gpgcheck=1

enabled=1

gpgkey=https://nginx.org/keys/nginx_signing.key

module_hotfixes=true

[nginx-mainline]

name=nginx mainline repo

baseurl=http://nginx.org/packages/mainline/centos/$releasever$/basearch/

gpgcheck=1

enabled=0

gpgkey=https://nginx.org/keys/nginx_signing.key

module_hotfixes=true

网卡配置

LVS服务器网卡配置

```
vim /etc/sysconfig/network-scripts/ifcfg-ens33
```
```

```
TYPE="Ethernet"
BOOTPROTO="static"
NAME="ens33"
DEVICE="ens33"
ONBOOT="yes"
IPADDR="192.168.222.132"
DNS1="8.8.8.8"
DNS2="114.114.114.114"
NETMASK="255.255.255.0"
GATEWAY="192.168.222.2"
```

```
vim /etc/sysconfig/network-scripts/ifcfg-ens36
```

```
TYPE="Ethernet"
BOOTPROTO="static"
NAME="ens36"
DEVICE="ens36"
ONBOOT="yes"
IPADDR="172.30.154.100"
DNS1="8.8.8.8"
DNS2="114.114.114.114"
NETWORK=255.255.0.0
```

nginx服务器网卡配置

```
vim /etc/sysconfig/network-scripts/ifcfg-ens33
```

```
TYPE="Ethernet"
BOOTPROTO="static"
NAME="ens33"
IPADDR="192.168.222.135"
DEVICE="ens33"
ONBOOT="yes"
DNS1=8.8.8.8
DNS2=114.114.114.114
GATEWAY="192.168.222.132"
~
```

另一台将ip改成192.168.222.136即可

客户端网卡配置

```
vim /etc/sysconfig/network-scripts/ifcfg-ens33
```

```
TYPE="Ethernet"
BOOTPROTO="static"
NAME="ens33"
IPADDR="172.30.154.110"
DEVICE="ens33"
ONBOOT="yes"
DNS1=8.8.8.8
DNS2=114.114.114.114
NETMASK=255.255.0.0
```

实现过程

实现思路：

- 在客户端主机进行本地域名解析，能够通过www.crz.com访问LVS服务
- 后端nginx服务器主机上，对nginx配置文件进行修改，使其能够监听从lvs服务器转发过来的信息，并根据不同的请求路径返回对应的文件内容。
- 在nginx服务器主机上，设置免密连接LVS主机。
- 通过在lvs服务器上使用ipvsadm工具，搭建集群，使得客户端流量到达LVS服务时，LVS能够将客户端的请求均衡转发到后端服务器上。我搭建的LVS负载均衡为NAT模式，使用轮询调度算法。
- 通过在nginx服务器上运行编写好的shell脚本，对后端nginx服务器的日志文件进行分析，分析结果包含nginx服务器每分钟接收到的请求数，以每分钟http code（状态码）为200和500+的数量，将分析结果存入到LVS服务器地址的root用户的MySQL库中，在LVS主机上查看分析结果。

客户端主机进行本地域名解析

在文件中添加内容：172.30.154.100 www.crz.com

```
vim /etc/hosts
```

nginx配置文件修改

```
vim /etc/nginx/nginx.conf
```

在http块下将server块改成

```
server {
    listen   80;
    server_name    localhost;

location = /mi.html{
    root  /usr/share/nginx/html;
}
location = /hello.html{
    root /usr/share/nginx/html;
}
location /{
    root   /usr/share/nginx/html;
    index   index.html;
}
}
```

修改完成后，检查配置文件信息并启动nginx服务

```
cd /usr/sbin/
./nginx -t
./nginx
```

```
[root@crz sbin]# ./nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
[root@crz sbin]# ./nginx
```

在lvs上curl测试以下是否成功

```
curl 192.168.222.135
curl 192.168.222.136
```

```
[root@localhost ~]# curl 192.168.222.135
I am xiaodi1!
[root@localhost ~]# curl 192.168.222.136
I am xiaodi2!
```

如果客户端访问www.crz.com 则将会返回 I am xiaodi1！ 或者 I am xiaodi2！

若访问www.crz.com/mi.html 则将会返回 I love xiaomi！ I am xiaodi1！ 或者 I love xiaomi！ I am xiaodi2！

若访问www.crz.com/hello.html 则将会返回 Hello World！ I am xiaodi1！ 或者 Hello World！ I am xiaodi2！

在nginx服务器主机上，设置免密连接LVS主机。

```
ssh-keygen #生成ssh密匙
ssh-copy-id root@192.168.222.132 #免密连接lvs主机
```

```
Number of key(s) added: 1

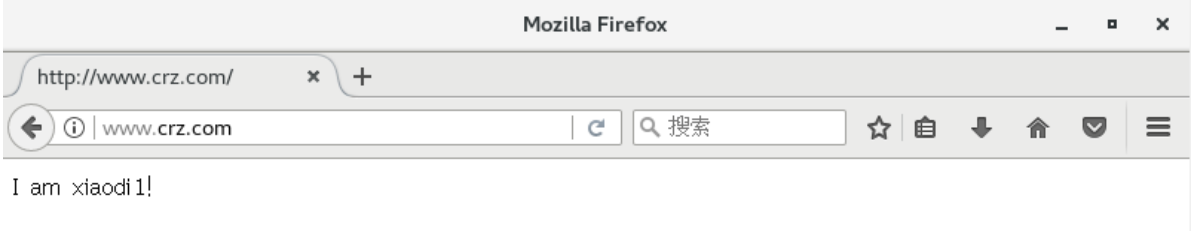
Now try logging into the machine, with:   "ssh 'root@192.168.222.132'"
and check to make sure that only the key(s) you wanted were added.
```

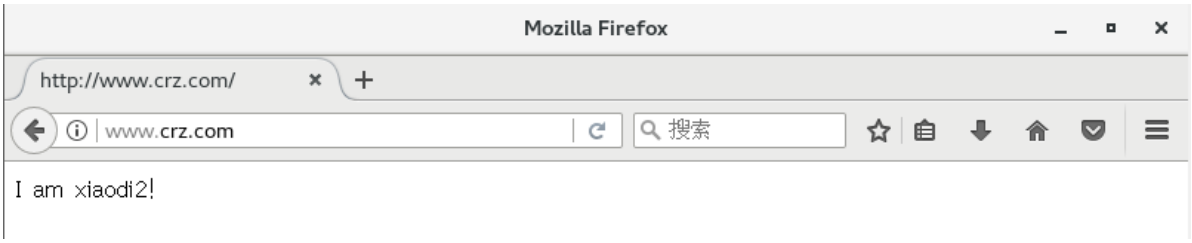
LVS负载均衡：

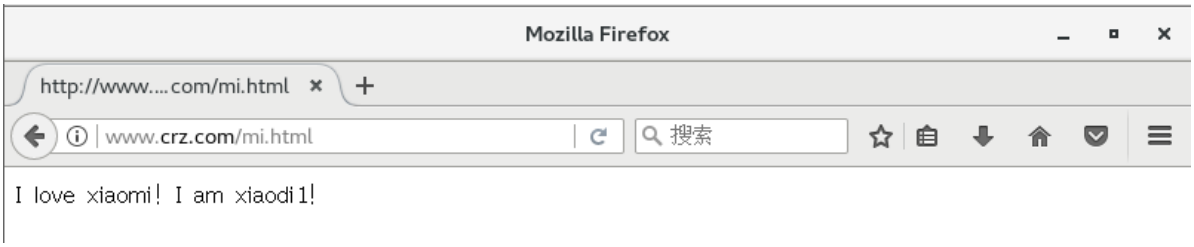
```
modprobe ip_vs #加载ip_vs模块
sysctl -w net.ipv4.ip_forward=1 #启动临时转发
ipvsadm -C #清除原有规则
ipvsadm -A -t 172.30.154.100:80 -s rr #rr表示轮询
ipvsadm -a -t 172.30.154.100:80 -r 192.168.222.135 -m
ipvsadm -a -t 172.30.154.100:80 -r 192.168.222.136 -m #-m表示NAT模式
ipvsadm -Ln #查看集群
```

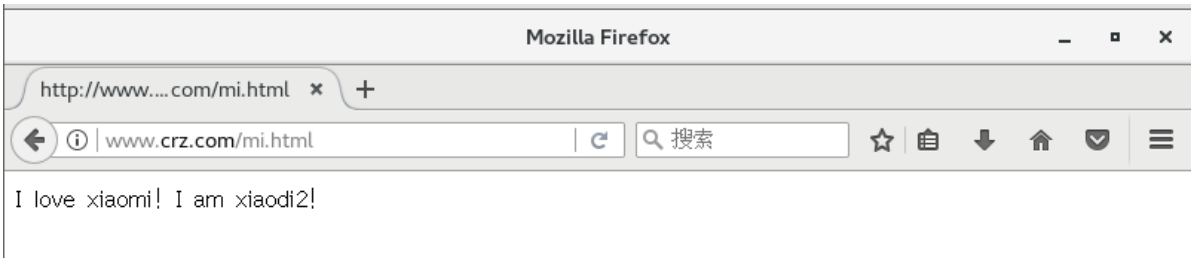
```
[root@localhost ~]# ipvsadm -Ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  172.30.154.100:80 rr
  -> 192.168.222.135:80           Masq    1      0          0
  -> 192.168.222.136:80           Masq    1      0          0
```

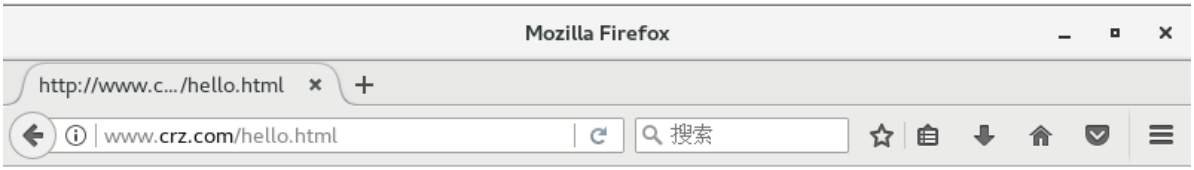
在客户端浏览器访问lvs集群查看负载均衡功能的实现：

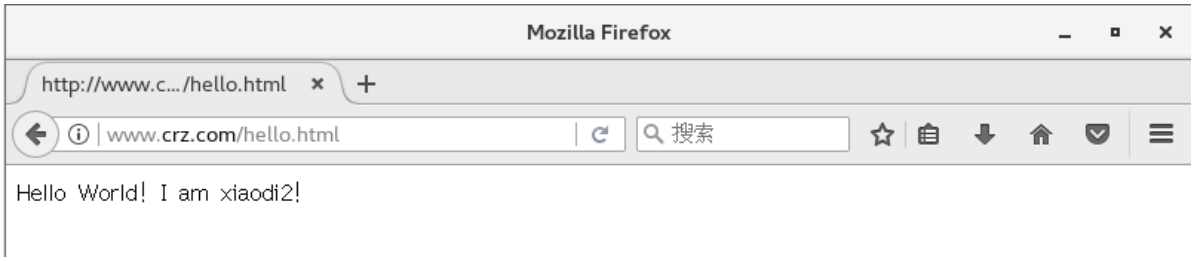



Mozilla Firefox
http://www.crz.com/
www.crz.com
搜索
I am xiaodi2!


Mozilla Firefox
http://www....com/mi.html
www.crz.com/mi.html
搜索
I love xiaomi! I am xiaodi1!


Mozilla Firefox
http://www....com/mi.html
www.crz.com/mi.html
搜索
I love xiaomi! I am xiaodi2!


Mozilla Firefox
http://www.c.../hello.html
www.crz.com/hello.html
搜索
Hello World! I am xiaodi1


Mozilla Firefox
http://www.c.../hello.html
www.crz.com/hello.html
搜索
Hello World! I am xiaodi2!

运行脚本，查看分析结果：

代码

###shell脚本

```
#!bin/bash
touch /var/lib/mysql-files/nginx.logs
touch /root/nginx.log
> /root/nginx.log
> /var/lib/mysql-files/nginx.logs
awk '
{
match($0,/\[([^\]]+)/,arr);#提取时间戳
full_timestamp = substr(arr[1],1,length(arr[1]));#去掉默认格式的方括号
split(full_timestamp,timeparts,"[: /]");#提取年月日小时分钟

#构造时间键
day = timeparts[1];
if(timeparts[2] == "Jun"){
    mon = 6;
}
if(timeparts[2] == "Jan"){
    mon = 1;
}
if(timeparts[2] == "Feb"){
    mon = 2;
}
if(timeparts[2] == "Mar"){
    mon = 3;
}
if(timeparts[2] == "Apr"){
    mon = 4;
}
if(timeparts[2] == "May"){
    mon = 5;
}
if(timeparts[2] == "Jul"){
    mon = 7;
}
if(timeparts[2] == "Aug"){
    mon = 8;
}
if(timeparts[2] == "Sep"){
    mon = 9;
}
if(timeparts[2] == "Oct"){
    mon = 10;
```

```
}
if(timeparts[2] == "Nov"){
    mon = 11;
}
if(timeparts[2] == "Dec"){
    mon = 12;
}
year = timeparts[3];
hour = timeparts[4];
minute = timeparts[5];
time_key = year"-"mon"-"day" "hour":"minute;
IP = $1;#后端服务器ip
status = $9#状态码

#更新每分钟的请求数以及对应的状态码
requests[time_key]++
if(status == 200){
    status_200[time_key]++;
}
else if(status >= 500){
    status_500[time_key]++;
}
}
#遍历数组并打印结果
END {
for(t in requests){
    qps = requests[t];
    status_200_count = status_200[t] + 0;
    status_500_count = status_500[t] + 0;
    print IP,"/",t,"/",qps,"/",status_200_count,"/",status_500_count;
}
}
' /var/log/nginx/access.log > /root/nginx.log

sort -t: -k1,1n /root/nginx.log > /var/lib/mysql-files/nginx.logs
cat /var/lib/mysql-files/nginx.logs
scp /var/lib/mysql-files/nginx.logs root@192.168.222.132:/var/lib/mysql-files/
mysql -h192.168.222.132 -uroot -pCrz666.. < /root/sql_file
```

###sql语句代码：

```
create database if not exists logs;
use logs;
create table if not exists log(
    id int AUTO_INCREMENT PRIMARY KEY,
    IP varchar(20),
    time datetime,
    qps int,
    status_200_count int,
    status_500_count int
    );
load data infile '/var/lib/mysql-files/nginx.logs'
    into table log
```

```
    fields terminated by '/'
    lines terminated by '\n'
    ignore 1 rows
    (
    IP,
    @time,
    qps,
    status_200_count,
    status_500_count
    )
set time = str_to_date(@time, '%Y-%m-%d %H:%i');
select * from log;
```