一、集群搭建

1、部署 nginx 反向代理三个 web 服务，调度算法使用加权轮询：

服务器端：

yum -y install epel-*　　#安装 epel 源

yum -y install nginx　　#安装 nginx

systemctl stop firewalld

setenforce 0　　#关闭防火墙和 SELinux

vim /etc/nginx/nginx.conf　　#配置文件

```
http {
    upstream love {
        server 192.168.43.81:80 weight=3 ;
        server 192.168.43.82:80;
        server 192.168.43.83:80;
    }
```

```
    server {
        listen       80 default_server ;
        listen       [::]:80 default_server;
        server_name  _;


        include /etc/nginx/default.d/*.conf;

        location / {
            proxy_pass http://love;

        }
```

systemctl start nginx　　#启动服务

web 端：

yum -y install epel-*　　#安装 epel 源

yum -y install nginx　　#安装 nginx

systemctl stop firewalld

setenforce 0　　#关闭防火墙和 SELinux

systemctl start nginx　　#启动服务

2、所有 web 服务使用共享存储 nfs，保证所有 web 都对其有读写权限，保

证数据一致性

服务端

yum -y install rpcbind nfs-utils　#安装

vim /etc/exports　　#配置

```
/share 192.168.43.0/24(rw,sync,fsid=0)
```

chmod　-R o+w /share #加权限

systemctl enable rpcbind.service

systemctl enable nfs-server.service #开机启动

systemctl start rpcbind.service

systemctl start nfs-server.service　#启动

```
[root@bogon ~]# showmount -e 192.168.43.80
Export list for 192.168.43.80:
/share 192.168.43.0/24
```

#检查

客户端

yum -y install rpcbind nfs-utils　#安装

systemctl enable rpcbind.service #开机启动

systemctl start rpcbind.service #启动

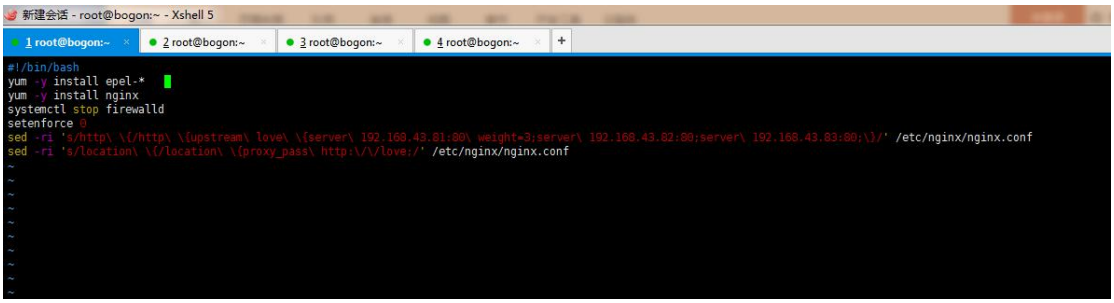
mount -t nfs 192.168.43.80:/share /var/www/html #挂载

开发脚本自动部署及监控

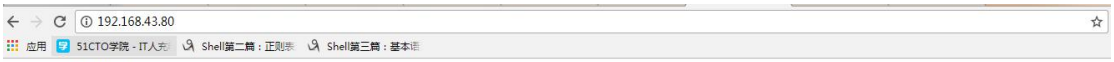
1.编写脚本自动部署反向代理、web、nfs；

1)反向代理

执行脚本成功



```
[root@bogon ~]# ./assigment1.sh
已加载插件：fastestmirror
file:///opt/repodata/repomd.xml: [Errno 14] curl#37 - "Couldn't open file /opt/repodata/repomd.xml"
正在尝试其它镜像。
Loading mirror speeds from cached hostfile
 * base: mirrors.cn99.com
 * epel: mirror.lzu.edu.cn
 * extras: mirrors.cn99.com
 * updates: mirrors.cn99.com
软件包 epel-release-7-10.noarch 已安装并且是最新版本
软件包 epel-rpm-macros-7-13.noarch 已安装并且是最新版本
无须任何处理
已加载插件：fastestmirror
file:///opt/repodata/repomd.xml: [Errno 14] curl#37 - "Couldn't open file /opt/repodata/repomd.xml"
正在尝试其它镜像。
Loading mirror speeds from cached hostfile
 * base: mirrors.cn99.com
 * epel: mirror.lzu.edu.cn
 * extras: mirrors.cn99.com
 * updates: mirrors.cn99.com
软件包 1:nginx-1.10.2-2.el7.x86_64 已安装并且是最新版本
无须任何处理
setenforce: SELinux is disabled
[root@bogon ~]#
```



web 端 同理
vim web.sh
yum -y install epel-*
yum -y install nginx
systemctl stop firewalld
setenforce 0
systemctl start nginx
chmod +x web.sh
nfs
服务端
脚本执行成功

```
#!/bin/bash
yum -y install rpcbind nfs-utils
echo '/share 192.168.43.0/24(rw,sync,fsid=0)' > /etc/expots
chmod  -R o+w /share
systemctl enable rpcbind.service
systemctl enable nfs-server.service
systemctl start rpcbind.service
systemctl start nfs-server.service
~
~
~
~
~
~
~
~
~
~
```

```
[root@bogon ~]# vim assignment2.sh
[root@bogon ~]# chmod +x assignment2.sh
[root@bogon ~]# ./assignment2.sh
已加载插件：fastestmirror
file:///opt/repodata/repomd.xml: [Errno 14] curl#37 - "Couldn't open file /opt/repodata/repomd.xml"
正在尝试其它镜像。
Loading mirror speeds from cached hostfile
 * base: mirrors.cn99.com
 * epel: mirror.lzu.edu.cn
 * extras: mirrors.cn99.com
 * updates: mirrors.cn99.com
软件包 rpcbind-0.2.0-42.el7.x86_64 已安装并且是最新版本
软件包 1:nfs-utils-1.3.0-0.48.el7.x86_64 已安装并且是最新版本
无须任何处理
[root@bogon ~]# systemctl status rpcbind.service
● rpcbind.service - RPC bind service
   Loaded: loaded (/usr/lib/systemd/system/rpcbind.service; indirect; vendor preset: enabled)
   Active: active (running) since 一 2017-10-09 09:58:01 CST; 6h ago
 Main PID: 945 (rpcbind)
   CGroup: /system.slice/rpcbind.service
           └─945 /sbin/rpcbind -w

10月 09 09:58:01 bogon systemd[1]: Starting RPC bind service...
10月 09 09:58:01 bogon systemd[1]: Started RPC bind service.
```

客户端

脚本执行成功

```
yum -y install rpcbind nfs-utils
systemctl enable rpcbind.service
systemctl start rpcbind.service
mount -t nfs 192.168.43.80:/share /var/www/html
~
~
~
~
~
```

```
[root@bogon ~]# ./web2.sh
[root@bogon ~]# df
文件系统                 1K-块    已用     可用 已用% 挂载点
/dev/mapper/cl-root  17811456 1362272 16449184    8% /
devtmpfs               493196       0   493196    0% /dev
tmpfs                  504196       0   504196    0% /dev/shm
tmpfs                  504196    6912   497284    2% /run
tmpfs                  504196       0   504196    0% /sys/fs/cgroup
/dev/sda1             1038336  141728   896608   14% /boot
tmpfs                  100840       0   100840    0% /run/user/0
192.168.43.80:/share 17811456 1363456 16448000    8% /var/www/html
[root@bogon ~]# vim web2.sh
```

2.编写监控脚本，监控集群内所有服务存活状态，内存、磁盘剩余率检测，异常则发送报警邮件

准备发送邮件的工具

vim /usr/bin/my_mail

复制发邮件代码，修改

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-
import sys
import smtplib
import email.mime.multipart
import email.mime.text

server = 'smtp.163.com'
port = '25'

def sendmail(server,port,user,pwd,msg):
    smtp = smtplib.SMTP()
    smtp.connect(server,port)
    smtp.login(user, pwd)
    smtp.sendmail(msg['from'], msg['to'], msg.as_string())
    smtp.quit()
    print('邮件发送成功email has send out !')


if __name__ == '__main__':
    msg = email.mime.multipart.MIMEMultipart()
    msg['Subject'] = '最后的作业'
    msg['From'] = 'xinghuaikang2@163.com'
    msg['To'] = 'xinghuaikang2@163.com'
    user = 'xinghuaikang2'
    pwd = 'xhk2623304075'
    content='%s\n%s' %('\n'.join(sys.argv[1:4]),' '.join(sys.argv[4:]))

    txt = email.mime.text.MIMEText(content, _charset='utf-8')
    msg.attach(txt)

    sendmail(server,port,user,pwd,msg)
~
~
~
~
~
~
~
"/usr/bin/my_mail" 32L, 852C
```

然后新建监控脚本 servermonitor.sh

vim servermonitor.sh

```
            msg="TIME:$(date +%F_%T)
                 HOSTNAME:$(hostname)
                 IPADDR:$(ifconfig |awk 'NR==2{print $2}')
                 MSG:Memory usage exceeds the limit,current value is ${mem_per}%"
            echo $msg
            /usr/bin/my_mail $msg
    fi
}

function monitor_disk_inode(){
    inode_use=`df -i $disk |awk 'NR==2{print $5}' |cut -d% -f1`
    if [ $inode_use -gt $disk_inode_limit ]
        then
            msg="TIME:$(date +%F_%T)
                 HOSTNAME:$(hostname)
                 IPADDR:$(ifconfig |awk 'NR==2{print $2}')
                 MSG:Disk inode usage exceeds the limit,current value is ${inode_use}%"
            echo $msg
            /usr/bin/my_mail $msg
    fi
}

function monitor_disk_space(){
    space_use=`df $disk |awk 'NR==2{print $5}'|cut -d% -f1`
    if [ $space_use -gt $disk_space_limit ]
        then
            msg="TIME:$(date +%F_%T)
                 HOSTNAME:$(hostname)
                 IPADDR:$(ifconfig |awk 'NR==2{print $2}')
                 MSG:Disk space usage exceeds the limit,current value is ${space_use}%"
            echo $msg
            /usr/bin/my_mail $msg
    fi
}

monitor_cpu &>> /tmp/monitor.log
monitor_mem &>> /tmp/monitor.log
monitor_disk_inode &>> /tmp/monitor.log
monitor_disk_space &>> /tmp/monitor.log
"servermonitor.sh" 66L, 2353C
```

3.编写计划任务，定时运行监控脚本，完成监控操作

crontab -e

* * * * *　/root/servermonitor.sh