Nginx+Keepalived实现高可用

# 软件的安装

## Nginx的安装

|  |
| --- |
| #安装依赖库  yum -y install gcc pcre-devel zlib-devel  #编译安装nginx  cd /usr/local/src  tar -zxvf nginx-1.8.0.tar.gz  cd /usr/local/src/nginx-1.8.0  ./configure --user=www --group=www --prefix=/usr/local/nginx --with-http\_stub\_status\_module --with-http\_gzip\_static\_module  make && make install  #配置环境变量  cat >> /etc/profile<<EOF  export NGINX\_HOME=/usr/local/nginx  export PATH=\$PATH:\$NGINX\_HOME/sbin  EOF  source /etc/profile  #备份和修改配置  useradd -s /sbin/nologin www  mkdir /data  cp -f /usr/local/nginx/conf/nginx.conf /usr/local/nginx/conf/nginx.bak  cat > /usr/local/nginx/conf/nginx.conf <<EOF  user www;  worker\_processes 4;  error\_log logs/error.log notice;  pid /var/run/nginx.pid;  worker\_rlimit\_nofile 65536;  events {  use epoll;  worker\_connections 65536;  }  http {  include mime.types;  default\_type application/octet-stream;  charset UTF-8;  limit\_conn\_zone \$binary\_remote\_addr zone=one:10m;  log\_format main '\$remote\_addr - \$remote\_user [\$time\_local] '  '"\$request" \$status \$bytes\_sent '  '"\$http\_referer" "\$http\_user\_agent" '  '"\$gzip\_ratio"';  log\_format download '\$remote\_addr - \$remote\_user [\$time\_local] '  '"\$request" \$status \$bytes\_sent '  '"\$http\_referer" "\$http\_user\_agent" '  '"\$http\_range" "\$sent\_http\_content\_range"';  access\_log logs/access.log main;  client\_max\_body\_size 20m;  client\_header\_buffer\_size 32k;  large\_client\_header\_buffers 4 128k;  sendfile on;  tcp\_nopush on;  tcp\_nodelay on;  keepalive\_timeout 60;  client\_header\_timeout 10;  client\_body\_timeout 10;  send\_timeout 10;  client\_body\_buffer\_size 512k;  proxy\_connect\_timeout 5;  proxy\_read\_timeout 60;  proxy\_send\_timeout 5;  proxy\_buffer\_size 16k;  proxy\_buffers 4 64k;  proxy\_busy\_buffers\_size 128k;  proxy\_temp\_file\_write\_size 64k;  gzip on;  gzip\_min\_length 1k;  gzip\_buffers 4 16k;  gzip\_http\_version 1.1;  gzip\_comp\_level 2;  gzip\_types text/plain text/css application/json application/x-javascript text/xml application/xml application/xml+rss text/javascript;  gzip\_vary on;  proxy\_cache\_path /data/cache levels=1:2 keys\_zone=cache\_one:4096m max\_size=3g inactive=600m;  proxy\_temp\_path /data/temp;  open\_file\_cache max=65536 inactive=20s;  open\_file\_cache\_valid 30s;  open\_file\_cache\_min\_uses 2;  open\_file\_cache\_errors on;  upstream webServer{  server 172.16.19.242:80 weight=3 max\_fails=3 fail\_timeout=20s;  server 172.16.19.243:80 weight=3 max\_fails=3 fail\_timeout=20s;    }  server{  listen 80;  server\_name helpedu.com 218.244.141.167;  index index.html index.htm index.jsp  #root /home/k12;  charset UTF-8;  access\_log logs/access.log main;  #错误页面处理  error\_page 404 500 502 503 504 /404.html;    location /NginxStatus {  stub\_status on;  access\_log off;  #allow可以有多个  allow 122.225.227.163;  deny all;  }  location /{  proxy\_cache cache\_one;  proxy\_cache\_key \$host\$uri\$is\_args\$args;  proxy\_redirect off;  proxy\_set\_header HOST \$host;  proxy\_set\_header X-Real-IP \$remote\_addr;  proxy\_set\_header X-Forwarded-For \$proxy\_add\_x\_forwarded\_for;  proxy\_pass http://webServer;  proxy\_next\_upstream http\_500 http\_502 http\_503 error timeout invalid\_header;  }  }  }  EOF  #启动nginx  nginx -c /usr/local/nginx/conf/nginx.conf |

## Keepalived的安装

|  |
| --- |
| yum -y install gcc gcc+ gcc-c++  yum -y install popt-devel openssl openssl-devel libssl-dev libnl-devel popt-devel  cd /usr/local/src  tar zxvf keepalived-1.2.19.tar.gz  cd keepalived-1.2.19  ./configure --sysconf=/etc  make && make install  ln -s /usr/local/sbin/keepalived /sbin/  chkconfig --add keepalived  chkconfig --level 35 keepalived on |

# Nginx+Keepalived实现高可用负载均衡

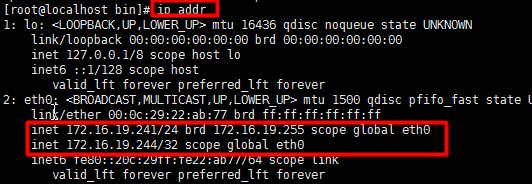
热备：（MASTER）-（BACKUP ）模式，备份机平常就是个摆设，浪费资源

互备：（MASTER、BACKUP）-（BACKUP、MASTER）模式，备份机也是主机，主机也是备份机，资源利用率高

## Nginx主从的配置

vi /etc/keepalived/keepalived.conf

|  |
| --- |
| #master  ! Configuration File for keepalived  global\_defs {  notification\_email {  xiaoxiazhang1@hengtiansoft.com  }    notification\_email\_from mail@example.org  smtp\_server 127.0.0.1  smtp\_connect\_timeout 30  router\_id LVS\_DEVEL  }  vrrp\_script chk\_nginx {  script "</dev/tcp/127.0.0.1/80"  interval 1 #检测脚本执行的间隔  weight -2 #权重  }    vrrp\_instance VI\_1 {  state MASTER  interface eth0  virtual\_router\_id 51  priority 101  advert\_int 1  mcast\_src\_ip 172.16.19.241    authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.244  }    track\_script {  chk\_nginx #调用检测脚本  }  } |
| #backup  ! Configuration File for keepalived  global\_defs {  notification\_email {  xiaoxiazhang1@hengtiansoft.com  }    notification\_email\_from mail@example.org  smtp\_server 127.0.0.1  smtp\_connect\_timeout 30  router\_id LVS\_DEVEL  }  vrrp\_script chk\_nginx {  script "</dev/tcp/127.0.0.1/80"  interval 1 #检测脚本执行的间隔  weight -2 #权重  }      vrrp\_instance VI\_1 {  state BAKEUP  interface eth0  virtual\_router\_id 51  priority 100  advert\_int 1  mcast\_src\_ip 172.16.19.242      authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.244  }    track\_script {  chk\_nginx #调用检测脚本  }  } |



## Nginx双主的配置

生产中两台主从架构会导致只有其中一台机器在运行，而另一台机器处于闲置状态。我们可以通过将第一台机器设置被第二台机器的备机，第二台机器设置成第一台机器的备机。然后申请一个域名分别绑定两个vip。

vi /etc/keepalived/keepalived.conf

|  |
| --- |
| #master\_1  ! Configuration File for keepalived  global\_defs {  notification\_email {  xiaoxiazhang1@hengtiansoft.com  }    notification\_email\_from mail@example.org  smtp\_server 127.0.0.1  smtp\_connect\_timeout 30  router\_id LVS\_DEVEL  }  vrrp\_script chk\_nginx {  script "</dev/tcp/127.0.0.1/80"  interval 1 #检测脚本执行的间隔  weight -2 #权重  }    vrrp\_instance VI\_1 {  state MASTER  interface eth0  virtual\_router\_id 51  priority 101  advert\_int 1  mcast\_src\_ip 172.16.19.241    authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.244  }    track\_script {  chk\_nginx #调用检测脚本  }  }  vrrp\_instance VI\_2 {  state BACKUP  interface eth0  virtual\_router\_id 52  priority 99  advert\_int 1  mcast\_src\_ip 172.16.19.242    authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.245  }    track\_script {  chk\_nginx #调用检测脚本  }  } |
| #master\_2  ! Configuration File for keepalived  global\_defs {  notification\_email {  xiaoxiazhang1@hengtiansoft.com  }    notification\_email\_from mail@example.org  smtp\_server 127.0.0.1  smtp\_connect\_timeout 30  router\_id LVS\_DEVEL  }  vrrp\_script chk\_nginx {  script "</dev/tcp/127.0.0.1/80"  interval 1 #检测脚本执行的间隔  weight -2 #权重  }    vrrp\_instance VI\_1 {  state BACKUP  interface eth0  virtual\_router\_id 51  priority 99  advert\_int 1  mcast\_src\_ip 172.16.19.241    authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.244  }    track\_script {  chk\_nginx #调用检测脚本  }  }  vrrp\_instance VI\_2 {  state MASTER  interface eth0  virtual\_router\_id 52  priority 100  advert\_int 1  mcast\_src\_ip 172.16.19.242    authentication {  auth\_type PASS  auth\_pass p@$$w0rd  }    virtual\_ipaddress {  172.16.19.245  }    track\_script {  chk\_nginx #调用检测脚本  }  } |