

双机keepalived高可用集群方案

v1.0.20220602.2

一，双机高可用方案。

1，主机环境。

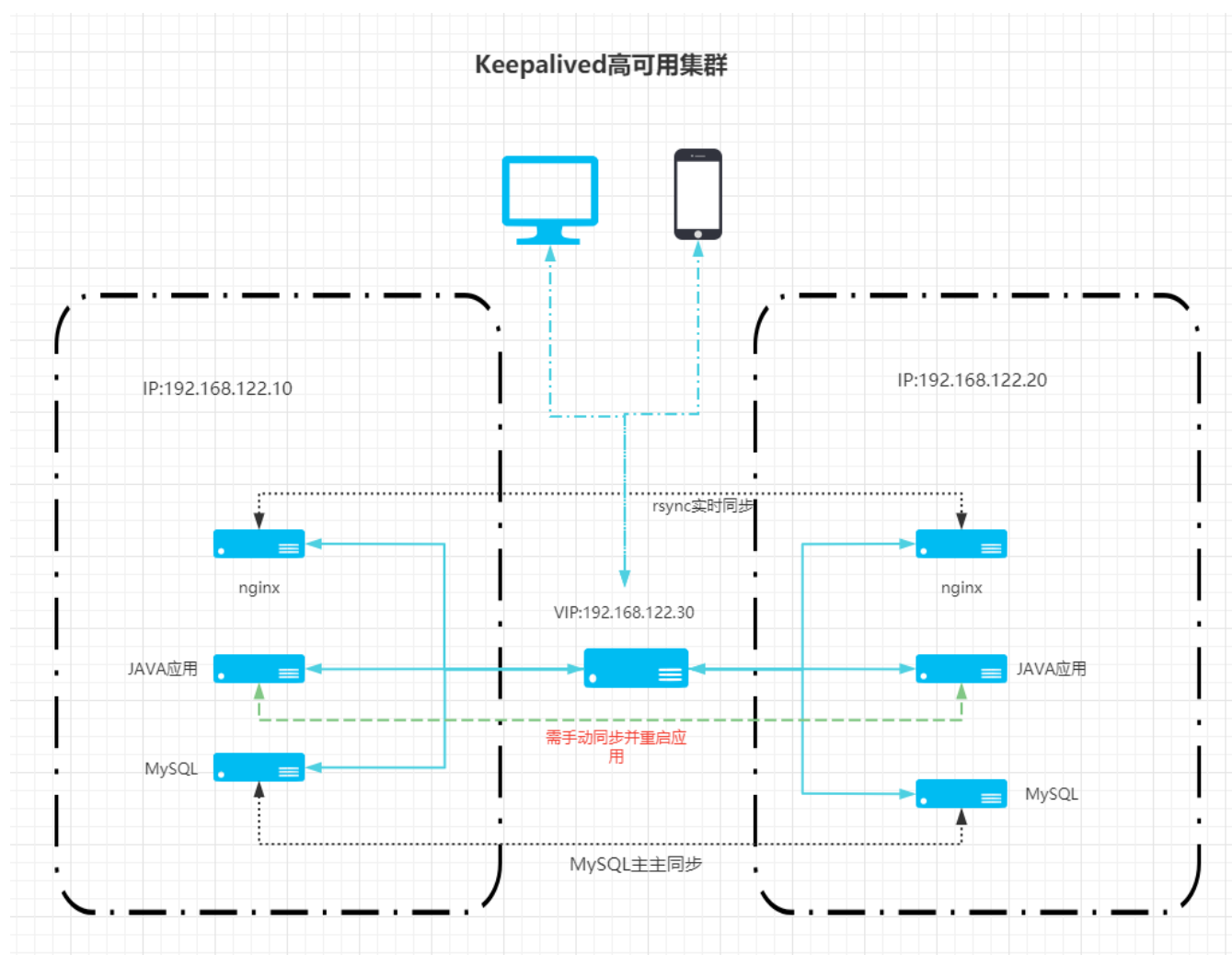
主机：IP: 192.168.122.10, Centos7, MySQL5.7, nginx, keepalived, rsync

备机：IP: 192.168.122.20, Centos7, MySQL5.7, nginx, keepalived, rsync

VIP: 192.168.122.30

实现效果：两台主机配置mysql双主同步，nginx+rsync实现web资源实时双向同步，vip提供高可用，JAVA应用需手动同步。

2，业务系统高可用拓扑图



二，keepalived安装与配置。

1, keepalived简介

简单地说, keepalived 就是通过管理 VIP 来实现机器的高可用的, 在使用 keepalived 的情况下, 只有一台服务器能够提供服务(通过 VIP 来实现), 当 Master 主机宕机后, VIP 会自动飘移到另一台服务器。

keepalived 采用 Master/Slave 模式, 在 Master 上设置配置文件的 VIP, 当 Master 宕机后, VIP 自动漂移到另一台 keepalived 服务器上。

keepalived 可以用来做各种软件的高可用集群, 它会一直检测服务器的状态, 如果有一台服务器宕机, 或工作出现故障, keepalived 将检测到, 并将有故障的服务器从系统中剔除, 同时使用其他服务器代替该服务器的工作, 当服务器工作正常后 keepalived 自动将服务器加入到服务器群中。

2, Yum 安装

```
yum install -y keepalived
```

3, 源码安装

```
# 安装依赖
yum install -y gcc popt-devel openssl openssl-devel libssl-dev libnl-devel popt-devel
libnfnetlink-devel

# 下载安装包
wget http://www.keepalived.org/software/keepalived-2.1.5.tar.gz

# 解压并安装
tar -xvz -f keepalived-2.1.5.tar.gz
cd keepalived-2.1.5
./configure --prefix=/usr/local/keepalived
make && make install

cp /usr/local/keepalived/sbin/keepalived /usr/sbin/
cp /usr/local/keepalived/etc/rc.d/init.d/keepalived /etc/init.d/
cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/
mkdir /etc/keepalived/
cp /usr/local/keepalived/etc/keepalived/keepalived.conf /etc/keepalived/
```

4, keepalived配置。

1) , 192.168.122.10主机/etc/keepalived.conf配置文件

```
! Configuration File for keepalived

global_defs {
    notification_email {
        acassen@firewall.loc
        failover@firewall.loc
        sysadmin@firewall.loc
    }
    notification_email_from Alexandre.Cassen@firewall.loc
    smtp_server 127.0.0.1
    smtp_connect_timeout 30
    router_id LVS_DEVEL
    vrrp_skip_check_adv_addr
```

```

vrrp_strict
vrrp_garp_interval 0
vrrp_gna_interval 0
}

vrrp_instance VI_1 {
    state BACKUP
    nopreempt
    # preempt_delay 30
    interface eth0
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.122.30
    }
}

virtual_server 192.168.122.30 80 {
    delay_loop 6
    lb_algo rr
    lb_kind DR
    persistence_timeout 50
    protocol TCP

    real_server 192.168.122.10 80 {
        weight 1
        notify_down /etc/keepalived/webshutdown.sh
        HTTP_GET {
            url {
                path /index.html
                status_code 200
            }
            connect_port 80
            connect_timeout 3
            nb_get_retry 3
            delay_before_retry 3
        }
    }
}

virtual_server 192.168.122.30 3306 {
    delay_loop 7
    lb_algo rr
    lb_kind DR
    persistence_timeout 50
    protocol TCP

    real_server 192.168.122.10 3306 {
        weight 1
        notify_down /etc/keepalived/dbshutdown.sh
    }
}

```

```

    TCP_CHECK {
        connect_port 3306
        connect_timeout 3
        nb_get_retry 3
        delay_before_retry 3
    }
}
}

```

2) , 192.168.122.10主机/etc/keepalived.conf配置文件

```

! Configuration File for keepalived

global_defs {
    notification_email {
        acassen@firewall.loc
        failover@firewall.loc
        sysadmin@firewall.loc
    }
    notification_email_from Alexandre.Cassen@firewall.loc
    smtp_server 127.0.0.1
    smtp_connect_timeout 30
    router_id LVS_DEVEL
    vrrp_skip_check_adv_addr
    vrrp_strict
    vrrp_garp_interval 0
    vrrp_gna_interval 0
}

vrrp_instance VI_1 {
    state BACKUP
    nopreempt
    # preempt_delay 30
    interface eth0
    virtual_router_id 51
    priority 99
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.122.30
    }
}

virtual_server 192.168.122.30 80 {
    delay_loop 6
    lb_algo rr
    lb_kind DR
    persistence_timeout 50
    protocol TCP

    real_server 192.168.122.20 80 {

```

```

weight 1
notify_down /etc/keepalived/webshutdown.sh
HTTP_GET {
    url {
        path /index.html
        status_code 200
    }
    connect_port 80
    connect_timeout 2
    nb_get_retry 2
    delay_before_retry 3
}
}

virtual_server 192.168.122.30 3306 {
    delay_loop 7
    lb_algo rr
    lb_kind DR
    persistence_timeout 50
    protocol TCP

    real_server 192.168.122.20 3306 {
        weight 1
        notify_down /etc/keepalived/dbshutdown.sh
        TCP_CHECK {
            connect_port 3306
            connect_timeout 2
            nb_get_retry 2
            delay_before_retry 3
        }
    }
}

```

3) , 两台主机都是BACKUP角色, priority也可以一样, 只是先启动keepalived的主机占用vip。配置文件中也可以增加JAVA应用端口的检测。

4) , 配置shutdown.sh脚本, 脚本内容可根据情况调整。主要为实现检测到mysql或是nginx异常时, 重启服务, 如无法恢复就停keepalived服务。

```

[root@vm10 ~]# cat /etc/keepalived/dbshutdown.sh
#!/bin/bash
systemctl stop keepalived

[root ~ ]# chmod u+x /etc/keepalived/dbshutdown.sh
# 给关闭keepalived服务的脚本加执行权限

# nginx检测脚本样例

#!/bin/bash
A=`ps -C nginx --no-header |wc -l`
# 判断nginx是否宕机, 如果宕机了, 尝试重启
if [ $A -eq 0 ];then

```

```
systemctl restart nginx
# 等待一小会再次检查nginx, 如果没有启动成功, 则停止keepalived, vip漂移
sleep 3
if [ `ps -C nginx --no-header |wc -l` -eq 0 ];then
    systemctl stop keepalived
fi
fi
```

三, 启动keepalived并测试集群。

1, 配置防火墙。

```
firewall-cmd --direct --permanent --add-rule ipv4 filter INPUT 0 --in-interface eth0 --
destination 224.0.0.18 --protocol vrrp -j ACCEPT
firewall-cmd --direct --permanent --add-rule ipv4 filter OUTPUT 0 --out-interface eth0 --
destination 224.0.0.18 --protocol vrrp -j ACCEPT
firewall-cmd --reload
```

2, 启动keepalived服务。

```
for i in vm10 vm20; do ssh $i 'systemctl enable keepalived && systemctl restart
keepalived';done

[root@vm20 ~]# ip addr
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default
qlen 1000
    link/ether 52:54:00:4f:f2:73 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.20/24 brd 192.168.122.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever
    inet 192.168.122.30/32 scope global eth0
        valid_lft forever preferred_lft forever
# 查看vip 192.168.122.30 已经绑定到优先级高或是先启动keepalived服务的主机上。
```

3, 测试vip漂移是否正常。

在一台主机关闭nginx或是mysql服务, 查看vip是否漂移到另外一台主机上面, 故障主机的keepalived是否关闭。