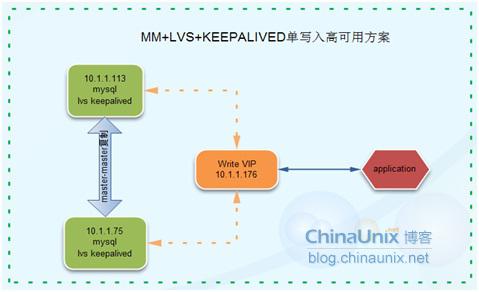
**MySQL高可用方案之LVS+keeplived +M/M**

1.1  方案简介

Lvs+keepalived作为目前比较流行的高可用解决方案，lvs提供负载均衡，keepalived作为故障转移，提高系统的可用性。但是一般的mysql高可用为了实现mysql数据的一致性，一般都是采用单点写入，本方案采用keepalived中的sorry\_server来实现写入数据库为单点的需求。本方案实现的功能是当网络有问题、mysql有问题、服务器宕机、keepalived服务停止后,服务器能自动跳转到备用机，当主服务器服务启动起来后会自动切换回来。

1.2  方案架构图



1.3  方案优缺点

优点：

ü  安装配置简单，实现方便，高可用效率好，可以根据服务与系统的可用性多方面进行切换。

ü  可以将写VIP和读VIP分别进行设置，为读写分离做准备。

ü  扩展不是很方便。

ü  可以在后面添加多个从服务器，并做到负载均衡。

缺点：

ü  在启动或者恢复后会立即替换掉定义的sorry\_server，因此如果要实现指定条件替换或者不替换需要通过其他方式实现，比如：临时更改mysql的端口等。

ü  切换需要1s左右的时间。

1.4  方案实战

1.4.1 适用场景

这个方案适用于只有两台数据库服务器并且还没有实现数据库的读写分离的情况，读和写都配置VIP。这个方案能够便于单台数据库的管理维护以及切换工作。比如进行大表的表结构更改、数据库的升级等都是非常方便的。

1.4.2 实战环境介绍

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 服务器名 | IP | VIP | 系统 | Mysql |
| Master | 10.1.1.113 | 10.1.1.176 | Centos 5.5 64bit | 5.1.63 |
| Backup | 10.1.1.75 | 10.1.1.176 | Centos 5.5 64bit | 5.1.63 |

1.4.3 Mysql的安装和配置

Mysql的安装和配置相对来讲非常简单，这里就不做介绍，有兴趣的朋友可以查看我博客中关于mysql 5.1.63版本自动安装的文章http://blog.chinaunix.net/uid-20639775-id-3168737.html

1.4.4 Mysql的主主同步配置

Mysql的主主同步这里也不做介绍了，有兴趣的话可以看一下我博文中关于mysql主从同步管理的介绍，主主同步和主从同步差不多，只是互为主从而已，链接如下：

http://blog.chinaunix.net/uid-20639775-id-3254611.html

1.4.5 Lvs的安装

在master、backup服务器都进行安装：

wget <http://www.linuxvirtualserver.org/software/kernel-2.6/ipvsadm-1.24.tar.gz>

ln -s /usr/src/kernels/2.6.18-164.el5-i686/ /usr/src/linux

tar zxvf ipvsadm-1.24.tar.gz

cd ipvsadm-1.24

make && make install

1.4.6 Keepalived的安装

在master、backup服务器都进行安装：

wget <http://www.keepalived.org/software/keepalived-1.1.19.tar.gz>  
tar zxvf keepalived-1.1.19.tar.gz  
cd keepalived-1.1.19  
./configure --prefix=/usr/local/keepalived  
make   
make install  
cp /usr/local/keepalived/sbin/keepalived /usr/sbin/  
cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/  
cp /usr/local/keepalived/etc/rc.d/init.d/keepalived /etc/init.d/  
mkdir /etc/keepalived

1.4.7 Keepalived的配置

1.4.7.1   Master的keepalived的配置

Master和backup不一样的地方已经标记为红色

vim /etc/keepalived/keepalived.conf

global\_defs {

notification\_email {

               zhangxy@test.com

        }

        notification\_email\_from jiankong@test.com

        smtp\_server mail.test.com

        smtp\_connect\_timeout 30

        router\_id LVS1

}

vrrp\_sync\_group test {

group {

        loadbalance

}

}

vrrp\_instance loadbalance {

        state MASTER

        interface eth0

        lvs\_sync\_daemon\_inteface eth0

        virtual\_router\_id 51

        priority 180

        advert\_int 1

authentication {

        auth\_type PASS

        auth\_pass 1111

}

virtual\_ipaddress {

        10.1.1.176 dev eth0 label eth0:1

}

}

virtual\_server 10.1.1.176 3306 {

        delay\_loop 6

        lb\_algo rr

        lb\_kind DR

        persistence\_timeout 20

        protocol TCP

        sorry\_server 10.1.1.75 3306

        real\_server 10.1.1.113 3306 {

        weight 3

        TCP\_CHECK {

        connect\_timeout 3

        nb\_get\_retry 3

        delay\_before\_retry 3

        connect\_port 3306

}

}

}

1.4.7.2   Backup的keepalived的配置

Master和backup不一样的地方已经标记为红色

vim /etc/keepalived/keepalived.conf

global\_defs {

notification\_email {

               zhangxy@test.com

        }

        notification\_email\_from jiankong@test.com

        smtp\_server mail.test.com

        smtp\_connect\_timeout 30

        router\_id LVS1

}

vrrp\_sync\_group test {

group {

        loadbalance

}

}

vrrp\_instance loadbalance {

        state BACKUP

        interface eth0

        lvs\_sync\_daemon\_inteface eth0

        virtual\_router\_id 51

        priority 150

        advert\_int 1

authentication {

        auth\_type PASS

        auth\_pass 1111

}

virtual\_ipaddress {

        10.1.1.176 dev eth0 label eth0:1

}

}

virtual\_server 10.1.1.176 3306 {

        delay\_loop 6

        lb\_algo rr

        lb\_kind DR

        persistence\_timeout 20

        protocol TCP

        sorry\_server 10.1.1.75 3306

        real\_server 10.1.1.113 3306 {

        weight 3

        TCP\_CHECK {

        connect\_timeout 3

        nb\_get\_retry 3

        delay\_before\_retry 3

        connect\_port 3306

}

}

}

1.4.8 Master和backup的realserver的配置

对于realserver的配置master和backup是一致的，脚本内容如下：

vim /etc/rc.d/init.d/realserver.sh

#!/bin/bash

# description: Config realserver lo and apply noarp

SNS\_VIP=10.1.1.176

/etc/rc.d/init.d/functions

case "$1" in

start)

       ifconfig lo:0 $SNS\_VIP netmask 255.255.255.255 broadcast $SNS\_VIP

       /sbin/route add -host $SNS\_VIP dev lo:0

       echo "1" >/proc/sys/net/ipv4/conf/lo/arp\_ignore

       echo "2" >/proc/sys/net/ipv4/conf/lo/arp\_announce

       echo "1" >/proc/sys/net/ipv4/conf/all/arp\_ignore

       echo "2" >/proc/sys/net/ipv4/conf/all/arp\_announce

       sysctl -p >/dev/null 2>&1

       echo "RealServer Start OK"

       ;;

stop)

       ifconfig lo:0 down

       route del $SNS\_VIP >/dev/null 2>&1

       echo "0" >/proc/sys/net/ipv4/conf/lo/arp\_ignore

       echo "0" >/proc/sys/net/ipv4/conf/lo/arp\_announce

       echo "0" >/proc/sys/net/ipv4/conf/all/arp\_ignore

       echo "0" >/proc/sys/net/ipv4/conf/all/arp\_announce

       echo "RealServer Stoped"

       ;;

\*)

       echo "Usage: $0 {start|stop}"

       exit 1

esac

exit 0

1.4.9 Master和backup的启动

启动master和backup的mysql以后，再在master和backup执行如下命令启动keepalived和realserver脚本：

/etc/rc.d/init.d/realserver.sh start

/etc/rc.d/init.d/keepalived start

并将keepalived和realserver的启动脚本加入到rc.local自启动中：

echo “/etc/rc.d/init.d/realserver.sh start” >> /etc/rc.local

echo “/etc/rc.d/init.d/keepalived start” >> /etc/rc.local

1.4.10     高可用方案测试

方案搭建好以后就要进行全方位的可靠性测试了，看看是否达到了我们的预期效果，大致测试步骤如下：

ü  停掉master上的mysql，看看能否自动切换到sorry\_server，使用如下命令查看：ipvsadm –ln。

ü  停掉master上的keepalived，看写VIP是否会迁移到backup。

ü  启动master上的mysql，看是否能切换回master。

ü  启动master上的keepalived，看VIP是否会迁移回master上。

ü  重启master的系统，看看切换过程是否OK