HA Cluster和keepalived主从,主主高可用设置以及varnish缓存机制（一）

家有猪宝_ (/u/41944084a17b)　+关注

2019.01.23 11:43　字数 1441　阅读 1　评论 0　喜欢 0

(/u/41944084a17b)

一、简述HA cluster原理

> 高可用集群，英文原文为High Availability Cluster，简称HA Cluster；集群（cluster）就是一组计算机，它们作为一个整体向用户提供一组网络资源。这些单个的计算机系统就是集群的节点（node）。高可用性集群（HA cluster）是指如单系统一样地运行并支持（计算机）持续正常运行的一个主机群。

> 高可用集群的出现是为了使集群的整体服务尽可能可用，从而减少由计算机硬件和软件易错性所带来的损失。如果某个节点失效，它的备援节点将在几秒钟的时间内接管它的职责。因此，对于用户而言，集群永远不会停机。高可用集群软件的主要作用就是实现故障检查和业务切换的自动化。

简单的说高可用集群就是为了解决集群中的单点故障（SPoF），保证服务不间断运行的冗余（redundant）手段。

- SPoF：Single Point of Failure；单点故障
- 冗余（redundant）：在两个节点上装一个软件程序，根据判断状态完成资源转移；

高可用集群的衡量标准

通常用平均无故障时间（MTTF）来度量系统的可靠性，用平均故障维修时间（MTTR）来度量系统的可维护性。于是可用性被定义为：`HA=MTTF/(MTTF+MTTR)*100%`

- 可用性衡量指标：
  · 基本可用性：2个9；99%；年度停机时间87.6小时
  · 较高可用性：3个9；99.9%；年度停机时间8.8小时
  · 具有故障自动恢复能力的可用性：4个9；99.99%；年度停机时间53分钟
  · 极高可用性：5个9；99.999%；年度停机时间5分钟

二、keepalived

1. KeepAlived主要有两个功能：

- (1).能够对RealServer进行健康状况检查，支持4层、5层和7层协议进行健康检查；

- (2).对负载均衡调度器实现高可用，防止Director单点故障。

2. KeepAlived工作过程：

　　keepalived实现故障转移的功能是通过VRRP（virtual router redundancy protocol虚拟路由器冗余协议）协议来实现的。在keepalived正常工作的时候，主节点（master）会不断的发送心跳信息给备节点（backup），当备节点不能在一定时间内收到主节点的心跳信息时，备节点会认为节点宕了，然后会接管主节点上的资源，并继续向外提供服务保证其可用性。当主节点恢复的时候，备节点会自动让出资源并再次自动成为备节点。

3. KeepAlived基于vrrp协议的软件实现，原生设计的目的为了高可用ipvs服务；

- 基于vrrp协议完成地址流动；

- 为vip地址所在的节点生成ipvs规则（在配置文件中预先定义）；

- 为ipvs集群的各RS做健康状态检测；

- 基于脚本调用接口通过执行脚本完成脚本中定义的功能，进而影响集群事务；

4. HA Cluser的配置前提：

　　(1) 各节点时间必须同步；ntp，chrony

　　(2) 确保iptales及selinux不会成为阻碍；

　　(3) 各节点之间可通过主机名互相通信（对KA并非必须）；

　　建议使用/etc/hosts文件实现；

　　(4) 确保各节点的用于集群服务的接口支持MULTICAST通信；

　　D类：224-239

5. Keepalived安装配置：

　　在CentOS6.4以后，keepalived随base仓库提供；

- 程序环境：

　　主配置文件：/etc/keepalived/keepalived.conf

　　主程序文件：/usr/sbin/keepalived

　　nit File：keepalived.service

　　Unit File的环境配置文件：/etc/sysconfig/keepalived

- 配置文件组件部分：

　　TOP HIERACHY

　　- GLOBAL CONFIGURATION

　　- Global definitions

　　- Static routes/addresses

　　- VRRPD CONFIGURATION

　　- VRRP synchronization group(s)：vrrp同步组；

　　- VRRP instance(s)：每个vrrp instance即一个vrrp路由器；

- LVS CONFIGURATION

- Virtual server group(s)

- Virtual server(s)：ipvs集群的vs和rs；

- 配置语法 ：

- 配置虚拟器：

`vrrp_instance <STRING> { ...... }`

- 专用参数：

`state MASTER|BACKUP`：当前节点在此虚拟路由器上的初始状态；只能有一个是MASTER，余下的都应该为BACKUP；

`interface IFACE_NAME`：绑定为当前虚拟路由器使用的物理接口；

`virtual_router_id VRID`：当前虚拟路由器的唯一标识，范围是0-255；

`priority 100`：当前主机在此虚拟路由器中的优先级；范围1-254；

`advert_int 1`：vrrp通告的时间间隔；

```
    authentication {
        auth_type AH|PASS
        auth_pass <PASSWORD>
    }
    virtual_ipaddress {
        <IPADDR>/<MASK> brd <IPADDR> dev <STRING> scope <SCOPE> label <LABEL>
        192.168.200.16/24 dev eth1
        192.168.200.17/24 dev eth2 label eth2:1
    }
    track_interface {
        eth0
        eth1
        ...
    }
# 配置要监控的网络接口，一旦接口出现故障，则转为FAULT状态；
# nopreempt: 定义工作模式为非抢占模式；
# preempt_delay 300: 抢占模式下，节点上线后触发新选举操作的延迟时长；
```

- 定义通知脚本:

```
notify_master <STRING>|<QUOTED-STRING>：当前节点成为主节点时触发的脚本；
notify_backup <STRING>|<QUOTED-STRING>：当前节点转为备节点时触发的脚本；
notify_fault <STRING>|<QUOTED-STRING>：当前节点转为"失败"状态时触发的脚本；
notify <STRING>|<QUOTED-STRING>：通用格式的通知触发机制，一个脚本可完成以上三种状态的转
```

- 虚拟服务器:

配置参数:

```
virtual_server IP port |
virtual_server fwmark int
{
    ...
    real_server {
        ...
    }
    ...
}
```

(/apps/
utm_sc
banner

常用参数：

`delay_loop <INT>`：服务轮询的时间间隔；

`lb_algo rr|wrr|lc|wlc|lblc|sh|dh`：定义调度方法；

`lb_kind NAT|DR|TUN`：集群的类型；

`persistence_timeout <INT>`：持久连接时长；

`protocol TCP`：服务协议，仅支持TCP；

`sorry_server <IPADDR> <PORT>`：备用服务器地址；

```
real_server <IPADDR> <PORT>
{
    weight <INT>
    notify_up <STRING>|<QUOTED-STRING>
    notify_down <STRING>|<QUOTED-STRING>
    HTTP_GET|SSL_GET|TCP_CHECK|SMTP_CHECK|MISC_CHECK { ... }：定义当前主机的健康x
                }
```

(https:/
click.yo
slot=30
7f8b-4
92de0

- HTTP_GET|SSL_GET：应用层检测

```
HTTP_GET|SSL_GET {
    url {
        path <URL_PATH>：定义要监控的URL；
        status_code <INT>：判断上述检测机制为健康状态的响应码；
        digest <STRING>：判断上述检测机制为健康状态的响应的内容的校验码；
    }
    nb_get_retry <INT>：重试次数；
    delay_before_retry <INT>：重试之前的延迟时长；
    connect_ip <IP ADDRESS>：向当前RS的哪个IP地址发起健康状态检测请求
    connect_port <PORT>：向当前RS的哪个PORT发起健康状态检测请求
    bindto <IP ADDRESS>：发出健康状态检测请求时使用的源地址；
    bind_port <PORT>：发出健康状态检测请求时使用的源端口；
    connect_timeout <INTEGER>：连接请求的超时时长；
}
```

- TCP_CHECK：传输层检测

```
TCP_CHECK {
    connect_ip <IP ADDRESS>：向当前RS的哪个IP地址发起健康状态检测请求
    connect_port <PORT>：向当前RS的哪个PORT发起健康状态检测请求
    bindto <IP ADDRESS>：发出健康状态检测请求时使用的源地址；
    bind_port <PORT>：发出健康状态检测请求时使用的源端口；
    connect_timeout <INTEGER>：连接请求的超时时长；
}
```

三、Keepalived实现主从、主主架构

1. 主从配置：

准备2个节点：node1：192.168.80.136；node2：192.168.80.230

同步时间：`[root@node1 ~]# ntpdate 192.168.80.1`

安装配置keepalived：

在node1如下配置

```
[root@node1 ~]# yum install -y keepalived     #安装keepalived
[root@node1 ~]# cd /etc/keepalived/
[root@node1 keepalived]# cp keepalived.conf{,.bak}     #备份keepalived原始配置文件
[root@node1 keepalived]# vim keepalived.conf
# 在打开的文件中配置如下内容
! Configuration File for keepalived

global_defs {
    notification_email {
      root@localhost
    }
    notification_email_from keepalived@localhost
    smtp_server 127.0.0.1
    smtp_connect_timeout 30
    router_id node1
     vrrp_mcast_group4 224.1.105.33
}

vrrp_instance VI_1 {
    state MASTER     #当前节点在此虚拟路由器上的初始状态；只能有一个是MASTER，余下的都应
    interface eth33
    virtual_router_id 33
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
}
```

在node2节点上如下配置：

```
[root@node2 ~]# yum install -y keepalived     #安装keepalived
[root@node2 ~]# cd /etc/keepalived
[root@node2 keepalived]# cp keepalived.conf{,.bak}   #备份keepalived原始配置文件
[root@node2 keepalived]# vim keepalived.conf
# 在打开的文件中配置如下内容
! Configuration File for keepalived

global_defs {
    notification_email {
     root@localhost
    }
    notification_email_from keepalived@localhost
    smtp_server 127.0.0.1
    smtp_connect_timeout 30
    router_id node2
     vrrp_mcast_group4 224.1.105.33
}

vrrp_instance VI_1 {
    state BACKUP
    interface ens33
    virtual_router_id 33
    priority 96
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
}
```

启动node2节点keepalived测试

```
[root@node2 keepalived]# systemctl start keepalived
[root@node2 keepalived]# ifconfig
...
ens33:0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.80.93  netmask 255.255.255.255  broadcast 0.0.0.0
        ether 00:0c:29:40:ee:7c  txqueuelen 1000  (Ethernet)
...
[root@node2 keepalived]# systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vend
   Active: active (running) since Wed 2019-01-16 12:24:22 CST; 5s ago
  Process: 3069 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exit
 Main PID: 3070 (keepalived)
    Tasks: 3
   CGroup: /system.slice/keepalived.service
           ├─3070 /usr/sbin/keepalived -D
           ├─3071 /usr/sbin/keepalived -D
           └─3072 /usr/sbin/keepalived -D

Jan 16 12:24:22 node2 Keepalived_healthcheckers[3071]: Activating healthcheck
Jan 16 12:24:25 node2 Keepalived_vrrp[3072]: VRRP_Instance(VI_1) Transition t
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: VRRP_Instance(VI_1) Entering MAS
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: VRRP_Instance(VI_1) setting prot
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: Sending gratuitous ARP on ens33
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: VRRP_Instance(VI_1) Sending/queu
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: Sending gratuitous ARP on ens33
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: Sending gratuitous ARP on ens33
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: Sending gratuitous ARP on ens33
Jan 16 12:24:26 node2 Keepalived_vrrp[3072]: Sending gratuitous ARP on ens33

# 在node1节点上抓包测试
[root@node1 keepalived]# tcpdump -i ens33 -nn host 224.1.105.33
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
12:25:16.821399 IP 192.168.80.230 > 224.1.105.33: VRRPv2, Advertisement, vrid
12:25:17.822579 IP 192.168.80.230 > 224.1.105.33: VRRPv2, Advertisement, vrid
```

启动node1节点keepalived：

```
[root@node1 keepalived]# systemctl start keepalived
[root@node1 keepalived]# ifconfig

...
ens33:0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.80.93  netmask 255.255.255.255  broadcast 0.0.0.0
        ether 00:0c:29:44:bc:b6  txqueuelen 1000  (Ethernet)
...
[root@node1 keepalived]# systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vend
   Active: active (running) since Wed 2019-01-16 16:42:49 CST; 5s ago
  Process: 6090 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exit
 Main PID: 6091 (keepalived)
    Tasks: 3
   CGroup: /system.slice/keepalived.service
           ├─6091 /usr/sbin/keepalived -D
           ├─6092 /usr/sbin/keepalived -D
           └─6093 /usr/sbin/keepalived -D

Jan 16 16:42:49 node1 Keepalived_vrrp[6093]: VRRP_Instance(VI_1) forcing a ne
Jan 16 16:42:50 node1 Keepalived_vrrp[6093]: VRRP_Instance(VI_1) Transition t
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: VRRP_Instance(VI_1) Entering MAS
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: VRRP_Instance(VI_1) setting prot
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: Sending gratuitous ARP on ens33
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: VRRP_Instance(VI_1) Sending/queu
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: Sending gratuitous ARP on ens33
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: Sending gratuitous ARP on ens33
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: Sending gratuitous ARP on ens33
Jan 16 16:42:51 node1 Keepalived_vrrp[6093]: Sending gratuitous ARP on ens33
Hint: Some lines were ellipsized, use -l to show in full.

# node2节点抓包测试
[root@node2 keepalived]# tcpdump -i ens33 -nn host 224.1.105.33
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
16:45:21.875150 IP 192.168.80.136 > 224.1.105.33: VRRPv2, Advertisement, vrid
16:45:22.876093 IP 192.168.80.136 > 224.1.105.33: VRRPv2, Advertisement, vrid
```

2. 双主模式配置

```
# node1节点上修改keepalived.cnf配置文件，在最后添加如下内容
vrrp_instance VI_2 {
    stat BACKUP
    interface ens33
    virtual_router_id 34
    priority 96
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass XXXX1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
}

# node2节点上修改keepalived.conf配置文件，在最后添加如下内容
vrrp_instance VI_2 {
    state MASTER
    interface ens33
    virtual_router_id 34
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass XXXX1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
}

# 停止keepalived服务，再重新启动
[root@node2 keepalived]# systemctl stop keepalived
[root@node2 keepalived]# systemctl start keepalived
[root@node2 keepalived]# ip a l
...
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UI
    link/ether 00:0c:29:40:ee:7c brd ff:ff:ff:ff:ff:ff
    inet 192.168.80.230/24 brd 192.168.80.255 scope global noprefixroute dynar
        valid_lft 62510sec preferred_lft 62510sec
    inet 192.168.80.93/32 scope global ens33:0
        valid_lft forever preferred_lft forever
    inet6 fe80::9c20:6c3a:b648:5b22/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
    inet6 fe80::5291:5f99:50eb:805/64 scope link tentative noprefixroute dadf
        valid_lft forever preferred_lft forever
...

[root@node2 keepalived]# systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vend
   Active: active (running) since Wed 2019-01-16 17:37:47 CST; 6min ago
  Process: 6300 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exit
 Main PID: 6302 (keepalived)
    Tasks: 3
   CGroup: /system.slice/keepalived.service
           ├─6302 /usr/sbin/keepalived -D
           ├─6303 /usr/sbin/keepalived -D
           └─6304 /usr/sbin/keepalived -D

Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Adding sorry server [1!
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Removing alive servers
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Remote SMTP server [12
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: SMTP alert successfully
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Timeout connecting ser
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Check on service [192.
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Removing service [192.
```

```
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Lost quorum 1-0=1 > 0 :
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Remote SMTP server [12]
Jan 16 17:38:15 node2 Keepalived_healthcheckers[6303]: SMTP alert successfull
Hint: Some lines were ellipsized, use -l to show in full.
You have new mail in /var/spool/mail/root

# 重新启动node1节点的keepalived服务
[root@node1 keepalived]# systemctl start keepalived
[root@node1 keepalived]# ip a l
...
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UF
    link/ether 00:0c:29:44:bc:b6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.80.136/24 brd 192.168.80.255 scope global noprefixroute dynan
       valid_lft 62131sec preferred_lft 62131sec
    inet 192.168.80.93/32 scope global ens33:0
       valid_lft forever preferred_lft forever
    inet6 fe80::5291:5f99:50eb:805/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
...

[root@node1 keepalived]# systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vend
   Active: active (running) since Wed 2019-01-16 17:44:08 CST; 10s ago
  Process: 6681 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exit
 Main PID: 6682 (keepalived)
    Tasks: 3
   CGroup: /system.slice/keepalived.service
           ├─6682 /usr/sbin/keepalived -D
           ├─6683 /usr/sbin/keepalived -D
           └─6684 /usr/sbin/keepalived -D

Jan 16 17:44:15 node1 Keepalived_healthcheckers[6683]: Timeout connecting serv
Jan 16 17:44:15 node1 Keepalived_healthcheckers[6683]: Timeout connecting serv
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: Sending gratuitous ARP on ens33 1
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: VRRP_Instance(VI_1) Sending/queue
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: Sending gratuitous ARP on ens33 1
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: Sending gratuitous ARP on ens33 1
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: Sending gratuitous ARP on ens33 1
Jan 16 17:44:16 node1 Keepalived_vrrp[6684]: Sending gratuitous ARP on ens33 1
Jan 16 17:44:17 node1 Keepalived_healthcheckers[6683]: Timeout connecting serv
Jan 16 17:44:17 node1 Keepalived_healthcheckers[6683]: Timeout connecting serv
Hint: Some lines were ellipsized, use -l to show in full.
[root@node1 keepalived]# vim keepalived.conf
You have new mail in /var/spool/mail/root

# 在node2节点上status查看状态
[root@node2 keepalived]# systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vend
   Active: active (running) since Wed 2019-01-16 17:37:47 CST; 6min ago
  Process: 6300 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exit
 Main PID: 6302 (keepalived)
    Tasks: 3
   CGroup: /system.slice/keepalived.service
           ├─6302 /usr/sbin/keepalived -D
           ├─6303 /usr/sbin/keepalived -D
           └─6304 /usr/sbin/keepalived -D

Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: SMTP alert successfully
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Timeout connecting serv
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Check on service [192.:
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Removing service [192.:
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Lost quorum 1-0=1 > 0 :
Jan 16 17:38:14 node2 Keepalived_healthcheckers[6303]: Remote SMTP server [12]
Jan 16 17:38:15 node2 Keepalived_healthcheckers[6303]: SMTP alert successfull
Jan 16 17:44:09 node2 Keepalived_vrrp[6304]: VRRP_Instance(VI_1) Received adve
Jan 16 17:44:09 node2 Keepalived_vrrp[6304]: VRRP_Instance(VI_1) Entering BACk
Jan 16 17:44:09 node2 Keepalived_vrrp[6304]: VRRP_Instance(VI_1) removing prot
```

```
Hint: Some lines were ellipsized, use -l to show in full.
```

3. 通知脚本使用方式

```bash
#编辑通知脚本
#!/bin/bash
#keepalived 邮件通知脚本
#date:2019-1-16
contact = 'root@localhost'
notify () {
    local mailsubject="$(hostname) to be $1 vip floating"
    local mailbody="$(date + '%F %T'): vrrp transition, $(hostname) changed t(
    echo "$mailbody" | mail -s "$mailsubject" $contact
}

case $1 in
master)
    notify master
    ;;
backup)
    notify backup
    ;;
fault)
    notify fault
    ;;
*)
    echo "Usage: $(basename $0) {master|backup|fault}"
    exit 1
    ;;
esac

# 在keepalived.conf中的vrrp实例中添加如下内容
vrrp_instance VI_1 {
    state BACKUP
    interface ens33
    virtual_router_id 33
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
    notify_master "/etc/keepalived/notify.sh master"
    notify_backup "/etc/keepalived/notify.sh backup"
    notify_fault "/etc/keepalived/notify.sh fault"
}
```

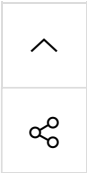## 4. 以dr集群架构配置示例

(/apps/
utm_so
banner

dr架构.png

(https:/
click.yo
slot=3(
7f8b-4
92de0:

```
[root@node1 keepalived]# yum install -y ipvsadm     #安装ipvsadm以便查看生成的规则
# 编辑keepalived.conf为node1和node2生成规则
[root@node1 keepalived]# vim keepalived.conf

! Configuration File for keepalived

global_defs {
    notification_email {
     root@localhost
    }
    notification_email_from keepalived@localhost
    smtp_server 127.0.0.1
    smtp_connect_timeout 30
    router_id node1
     vrrp_mcast_group4 224.1.105.33
}

vrrp_instance VI_1 {
    state MASTER
    interface ens33
    virtual_router_id 33
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.80.93 dev ens33 label ens33:0
    }
    notify_master "/etc/keepalived/notify.sh master"
    notify_backup "/etc/keepalived/notify.sh backup"
    notify_fault "/etc/keepalived/notify.sh fault"
}

virtual_server 192.168.80.93 80 {
    delay_loop 1
    lb_algo wrr
    lb_kind DR
    protocol TCP
    sorry_server 127.0.0.1 80

    real_server 192.168.80.176 80 {
        weight 1
        HTTP_GET {
            url {
              path /index.html
              status_conde 200
            }
            connect_timeout 3
            nb_get_retry 3
            delay_before_retry 3
        }
    }

    real_server 192.168.80.85 80 {
        weight 1
        HTTP_GET {
            url {
              path /index.html
              status_conde 200
            }
            connect_timeout 3
            nb_get_retry 3
            delay_before_retry 3
        }
    }
}
```

```
# 将此配置文件拷贝到node2节点,并修改以下几行
    router_id node2
    state BACKUP
    priority 96
# 重新启动node2节点的keepalived服务
[root@node2 keepalived]# systemctl stop keepalived
[root@node2 keepalived]# systemctl start keepalived
[root@node2 keepalived]# ifconfig
...
ens33:0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.80.93  netmask 255.255.255.255  broadcast 0.0.0.0
        ether 00:0c:29:40:ee:7c  txqueuelen 1000  (Ethernet)
...

[root@node2 keepalived]# ipvsadm -ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port            Forward Weight ActiveConn InActConn
TCP  192.168.80.93:80 wrr
  -> 192.168.80.85:80              Route   1      0          0
  -> 192.168.80.176:80             Route   1      0          0

# 启动node1的keepalived服务,通过下面查看ip和status后看到node1已经成功上线
[root@node2 keepalived]# ipvsadm -ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port            Forward Weight ActiveConn InActConn
TCP  192.168.80.93:80 wrr
  -> 192.168.80.85:80              Route   1      0          0
  -> 192.168.80.176:80             Route   1      0          0

# 使用client访问服务正常
[root@localhost ~]# curl http://192.168.80.93
<h1>RealServer 1</h1>
[root@localhost ~]# curl http://192.168.80.93
<h1>RealServer 2</h1>
```
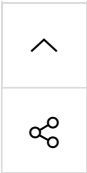
小礼物走一走，来简书关注我

赞赏支持

📕 日记本 (/nb/28535672)

举报文章　　© 著作权归作者所有

家有猪宝_ (/u/41944084a17b)

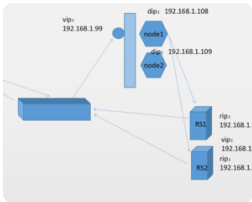写了 46020 字，被 0 人关注，获得了 1 个喜欢
(/u/41944084a17b)

＋关注

喜欢

更多分享

(/p/428251ede1aa)

(/p/6610251e8aec?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend

HA Cluster 高可用集群keepalived (/p/6610251e8aec?utm_campaign=m…

一、简述HA Cluster原理 高可用集群，英文原文为High Availability Cluster，简称HA Cluster；集群（cluster）就是一组计算机，它们作为一个整体向用户提供一组网络资源。这些单个的计算机系统 就是集…

Net夜风 (/u/db980997dc98?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

20171029 KeepAlived (/p/d2838d66f44b?utm_campaign=maleskine&ut…

一、高可用集群 （一）提升系统高可用性的解决方案：冗余(redundant) 工作模式active/passive：主备active/active：双主 以心跳方式通告active --> HEARTBEAT --> passiveactive <--> HEARTBEAT...
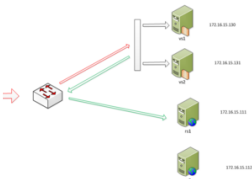
哈喽别样 (/u/f6a0c03fb91d?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

keepalive (/p/793744084c4d?utm_campaign=maleskine&utm_content=…

转载自：http://blog.csdn.net/u010391029/article/details/48311699 1. 前言 VRRP(Virtual Router Redundancy Protocol)协议是用于实现路由器冗余的协议，最新协议在RFC3768中定...

Chris_Zheng (/u/bd0971d7f15b?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

(/p/fea4a4ae080c?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend

keepalived实现主从、主主架构 (/p/fea4a4ae080c?utm_campaign=males…

一、HA Cluster实现方案: 1、vrrp协议的实现VRRP（Virtual Router Redundancy Protocol，虚拟路由冗余协议）是一种容错协议。通常，一个网络内的所有主机都设置一条缺省路由，这样，主机发出的目的地址不…

任总 (/u/a6304912144e?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend
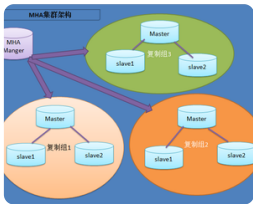
(/p/3492e7d5d2aa?

(/apps/
utm_so
banner

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend

MySQL 高可用架构MHA 多Master (/p/3492e7d5d2aa?utm_campaign=m...

转载：https://www.cnblogs.com/gomysql/p/3675429.html 简介： MHA（MasterHigh Availability）目前在
MySQL高可用方面是一个相对成熟的解决方案，它由日本DeNA公司youshimaton（现就职于Fac...

SkTj (/u/92422fe74fa9?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

2018-10-10 (/p/359ac13d0e8f?utm_campaign=maleskine&utm_content...

才深刻体会到，孙总是位研究型学者+总裁！技术型须导！牛[强][强][强]与胡中海的风格一脉相承！前几天与
雅丽学姐同修聊，也感受到，新巅峰的团队不是在为金钱打工，而是在为教育奉献！前两年，我亲眼看到...

每个人的孟母堂 (/u/f5d9fb306c79?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recom(https:/
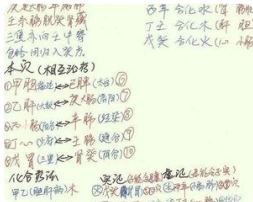click.yc
slot=30
7f8b-4
92de02

(/p/d9f6fa48da98?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend

心若在，远近都相宜！ (/p/d9f6fa48da98?utm_campaign=maleskine&ut...

人的心本身就是一处 四季春意盎然的花园， 芬芳四溢，蝴蝶自来，无论远近 。 壹 远近相宜，不远也不近的
距离，不觉疲惫。 最远的距离就是一步之遥。竟，不可触及。 最近的距离就是朝夕相处。然，心隔千里。...

古今通鉴 (/u/877126a0a50c?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

(/p/a9f0e7bbda9c?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend

五门十变法 (/p/a9f0e7bbda9c?utm_campaign=maleskine&utm_content...

今天学习五门十变法，经过仔细端详，认真考虑，总结如下，希望我的朋友们能看懂： 化合之物纯天然，故
相关经络可以自由取舍，只有是同本性的经络，实症用之都一样，虚证也一样。 例如：把脉某男，右寸浮...

玄智狂人 (/u/dd7b7c2025a0?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

(/p/d0a731fc8c41?

(/apps/
utm_so
banner

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommenc

这样给孩子选书，想让他停下来不读都难！ (/p/d0a731fc8c41?utm_camp…

文／乌江子 一个爱书的人，他必定不致缺少一个忠实的朋友，一个良好的老师，一个可爱的伴侣，一个优婉

的安慰者。——伊萨克•巴罗 读一本好书就是和许多高尚的人谈话。——歌德 阅读素养是一种根据社会需…

乌江子 (/u/fee1ff69f59d?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommenc

你想取得成功？看看自己是不是缺少这个条件！ (/p/0ee22ddbe4d6?utm_c…

原创/励未扬 2018年12月24日，未扬在洛阳向你问好！ 作者简介：龙起风尘末，虎啸未扬时。手执笔杆一

枝，怀揣梦想一个。用双手书写文字，用梦想丈量人生！ 01 专业性，是成功的必需条件 有多少人忘了当…

励未扬 (/u/9cc52b0b7dc1?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommenc

(https:/
click.yc
slot=3(
7f8b-4∠
92de0: