LVS应用方案介绍

郭贤亮
13382030778

概 览

- LVS基本介绍
- LVS技术简介
- LVS三种部署模式
- LVS案例分享

LVS基本介绍

LVS基本介绍

LVS是什么？

LVS是Linux Virtual Server的简写，即**Linux虚拟服务器**，是一个虚拟的服务器集群系统。项目在1998年5月由章文嵩博士成立，是中国国内最早出现的开源软件项目之一。

LVS基本介绍

LVS宗旨（目标）

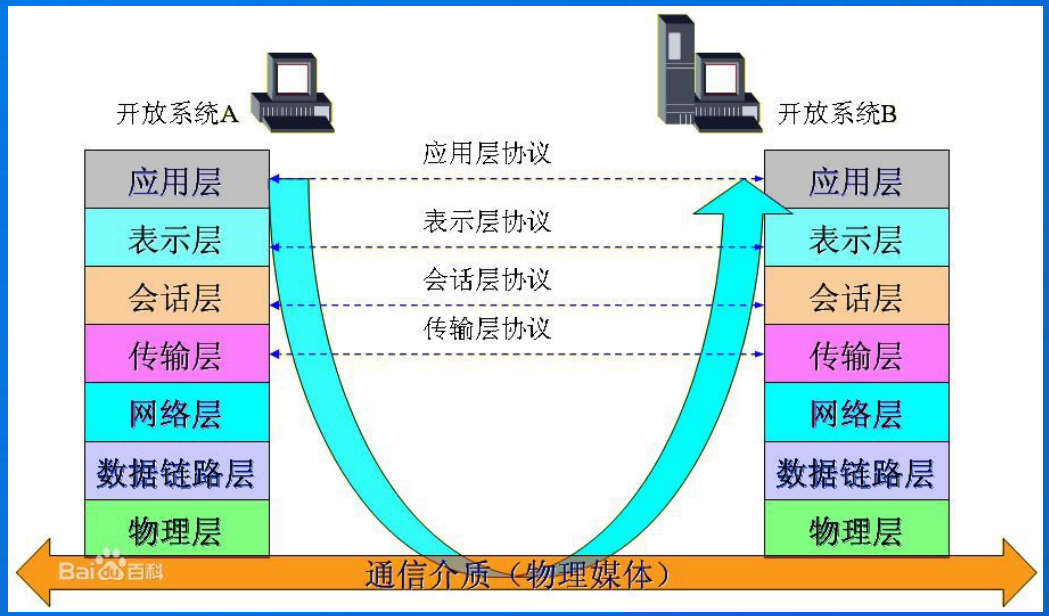
LVS使用集群技术和Linux操作系统实现一个高性能、高可用的服务器.

- 很好的可伸缩性（Scalability）
- 很好的可靠性（Reliability）
- 很好的可管理性（Manageability）

LVS技术简介

LVS技术简介

LVS集群采用IP负载均衡技术和基于内容请求分发技术。调度器具有很好的吞吐率，将请求均衡地转移到不同的服务器上执行，且调度器自动屏蔽掉服务器的故障，从而将一组服务器构成一个高性能的、高可用的虚拟服务器。整个服务器集群的结构对客户是透明的，而且无需修改客户端和服务器端的程序。具有透明性、可伸缩性、高可用性和易管理性。



物理层：物理接口规范，传输比特流,网卡是工作在物理层的。
数据层：成帧，保证帧的无误传输，MAC地址，形成EHTHERNET帧
网络层：路由选择，流量控制，IP地址，形成IP包
传输层：端口地址，如HTTP对应80端口。TCP和UDP工作于该层,还有就是差错校验和流量控制。
会话层：组织两个会话进程之间的通信,并管理数据的交换使用NETBIOS和WINSOCK协议。QQ等软件进行通讯因该是工作在会话层的。 表示层使得不同操作系统之间通信成为可能。
应用层：对应于各个应用软件

LVS技术特点

LVS采用IP负载均衡技术：

①通过网络地址转换（Network Address Translation）将一组服务器构成一个高性能的、高可用的虚拟服务器，称之为VS/NAT技术（Virtual Server via Network Address Translation）。

②在分析VS/NAT的缺点和网络服务的非对称性的基础上，通过IP隧道实现虚拟服务器的方法VS/TUN （Virtual Server via IP Tunneling）

③在分析VS/NAT的缺点和网络服务的非对称性的基础上，通过直接路由实现虚拟服务器的方法VS/DR（Virtual Server via Direct Routing）。

VS/NAT、VS/TUN和VS/DR技术是LVS集群中实现的三种IP负载均衡技术。

LVS技术原理

LVS对外提供一个虚拟IP供外部访问，一般会采用两台互为主备的机器，在两台机器的网卡上同时绑定这个虚拟IP，通过定时心跳检测，如果Master没有相应，Slave机器会接管虚拟IP的请求到本机。

原理即通过ARP协议，Master会通知所有同一子网内的机器，虚拟IP绑定当前MAC地址，每台子网内的机器都对这个ARP对应子网地址缓存。

举例说明：

(192.168.1.219) at 00:21:5A:DB:68:E8 [ether] on bond0
(192.168.1.217) at 00:21:5A:DB:68:E8 [ether] on bond0
(192.168.1.218) at 00:21:5A:DB:7F:C2 [ether] on bond0

192.168.1.217、192.168.1.218是两台真实主机地址，其中192.168.1.217为对外提供数据库服务的Master主机，192.168.1.218为热备的Slave主机。192.168.1.219为虚IP地址

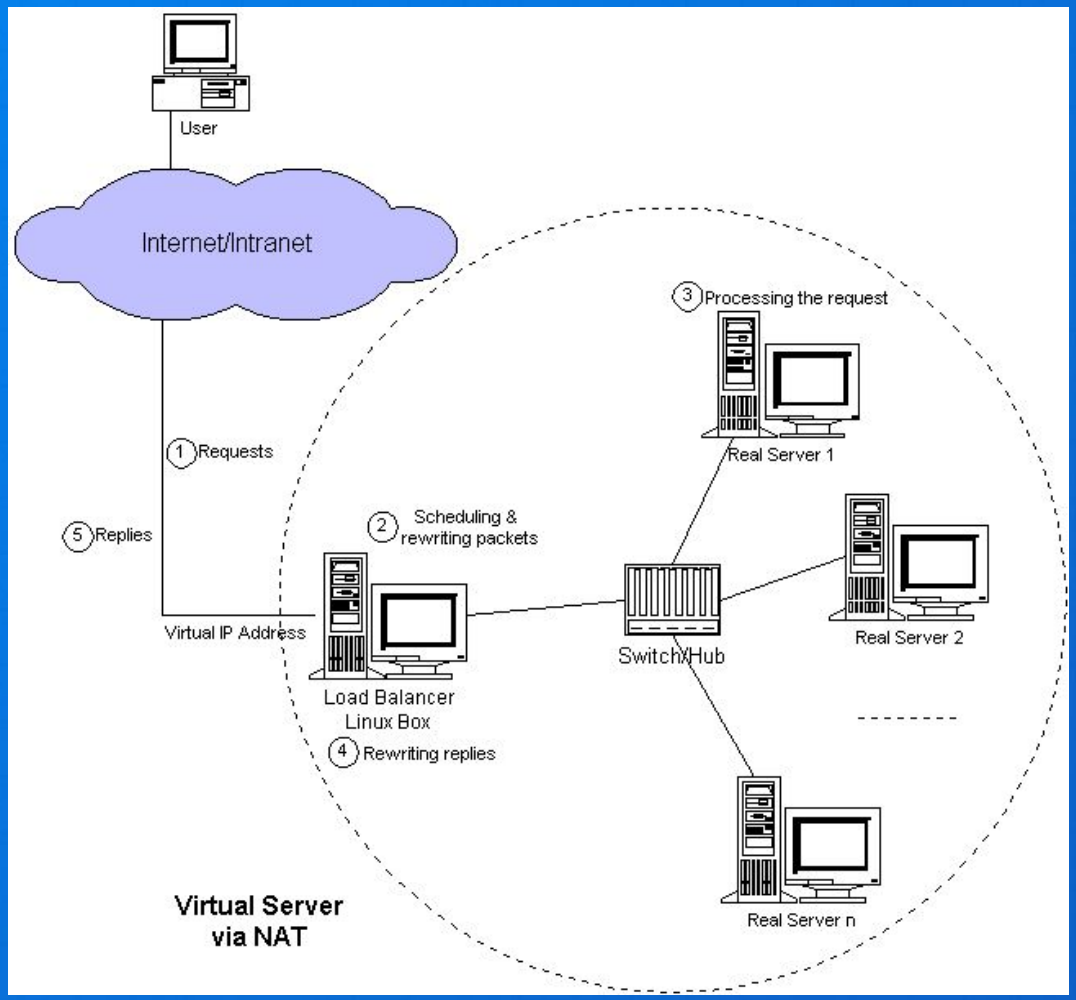
再看看217宕机后的arp缓存：

(192.168.1.219) at 00:21:5A:DB:7F:C2 [ether] on bond0
(192.168.1.217) at 00:21:5A:DB:68:E8 [ether] on bond0
(192.168.1.218) at 00:21:5A:DB:7F:C2 [ether] on bond0

当218 发现217宕机后会向网络发送一个ARP数据包，告诉所有主机192.168.1.219这个IP对应的MAC地址是00:21:5A:DB:7F:C2，这样所有发送到219的数据包都会发送到mac地址为00:21:5A:DB:7F:C2的机器，也就是218的机器

LVS三种部署模式

Virtual server via NAT（VS-NAT）

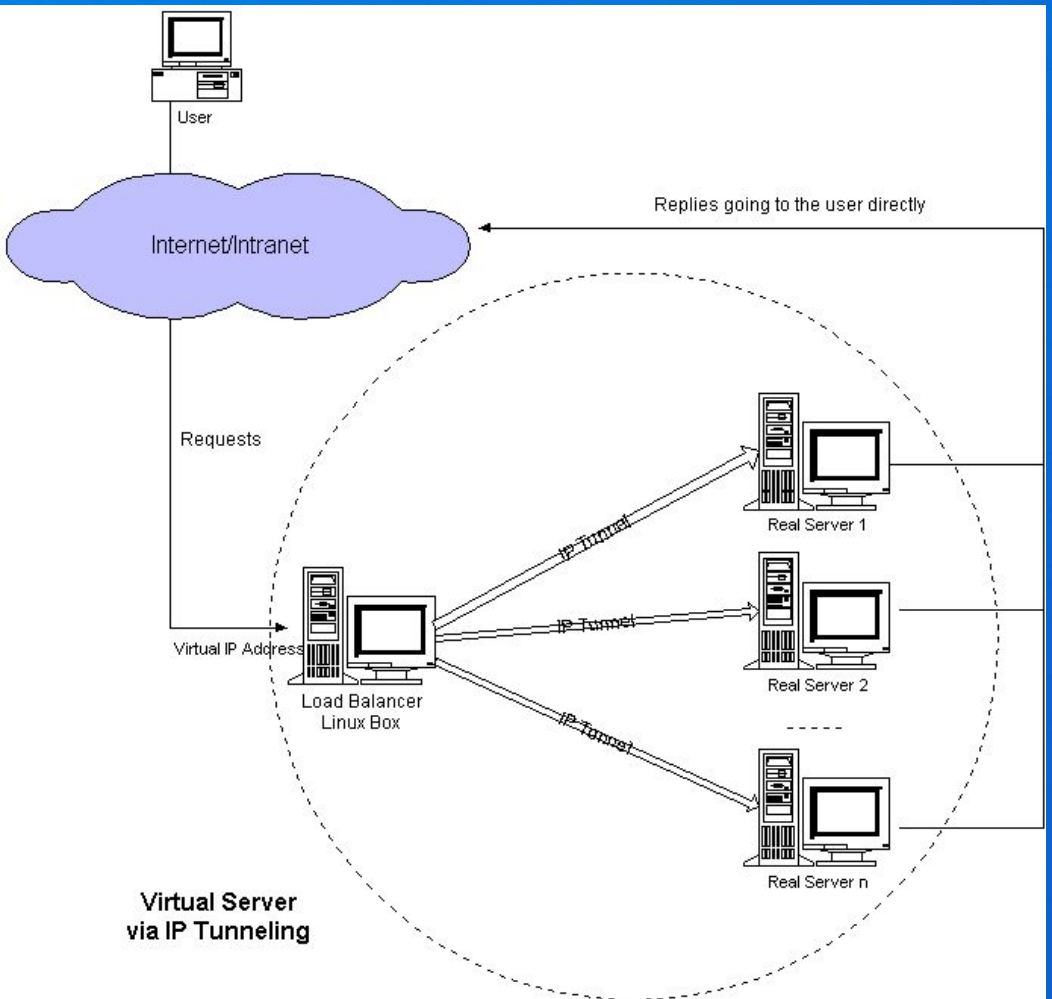


NAT模式

优点：集群中的物理服务器可以使用任何支持TCP/IP操作系统，物理服务器可以分配Internet的保留私有地址，只有负载均衡器需要一个合法的IP地址。

缺点：扩展性有限。当服务器节点（普通PC服务器）数据增长到20个或更多时,负载均衡器将成为整个系统的瓶颈，因为所有的请求包和应答包都需要经过负载均衡器再生。

Virtual server via IP tunneling（VS-TUN）

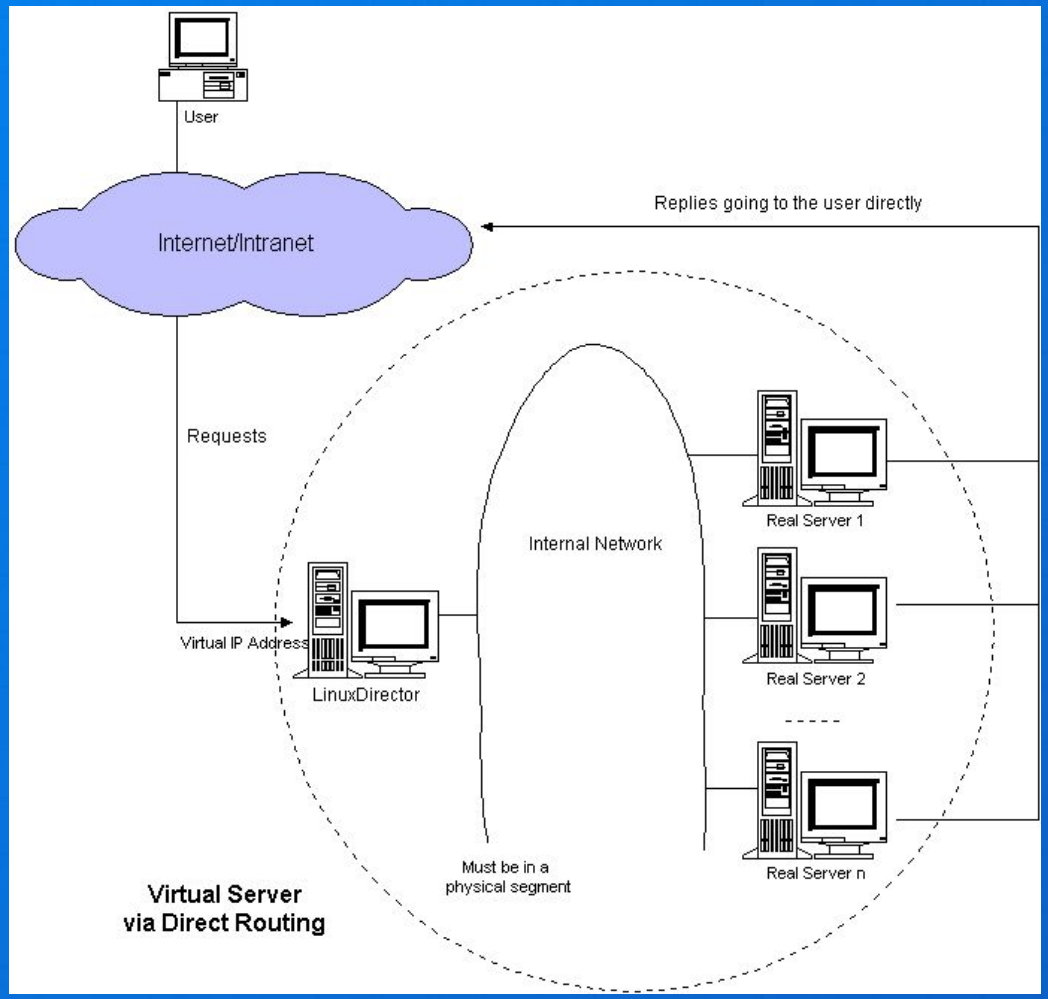


TUN模式

•**优点：**负载均衡器只负责将请求包分发给物理服务器，而物理服务器将应答包直接发给用户。所以，负载均衡器能处理很巨大的请求量，这种方式，一台负载均衡能为超过100台的物理服务器服务，负载均衡器不再是系统的瓶颈。

•**不足：**这种方式需要所有的服务器支持“IP Tunneling”(IP Encapsulation)协议，Linux支持，其他操作系统未知。

Virtual Server via Direct Routing（VS-DR）



DR模式

优点：和VS - TUN一样，负载均衡器也只是分发请求，应答包通过单独的路由方法返回给客户端。与VS-TUN相比，VS-DR这种实现方式不需要隧道结构，可以使用大多数操作系统做为物理服务器。

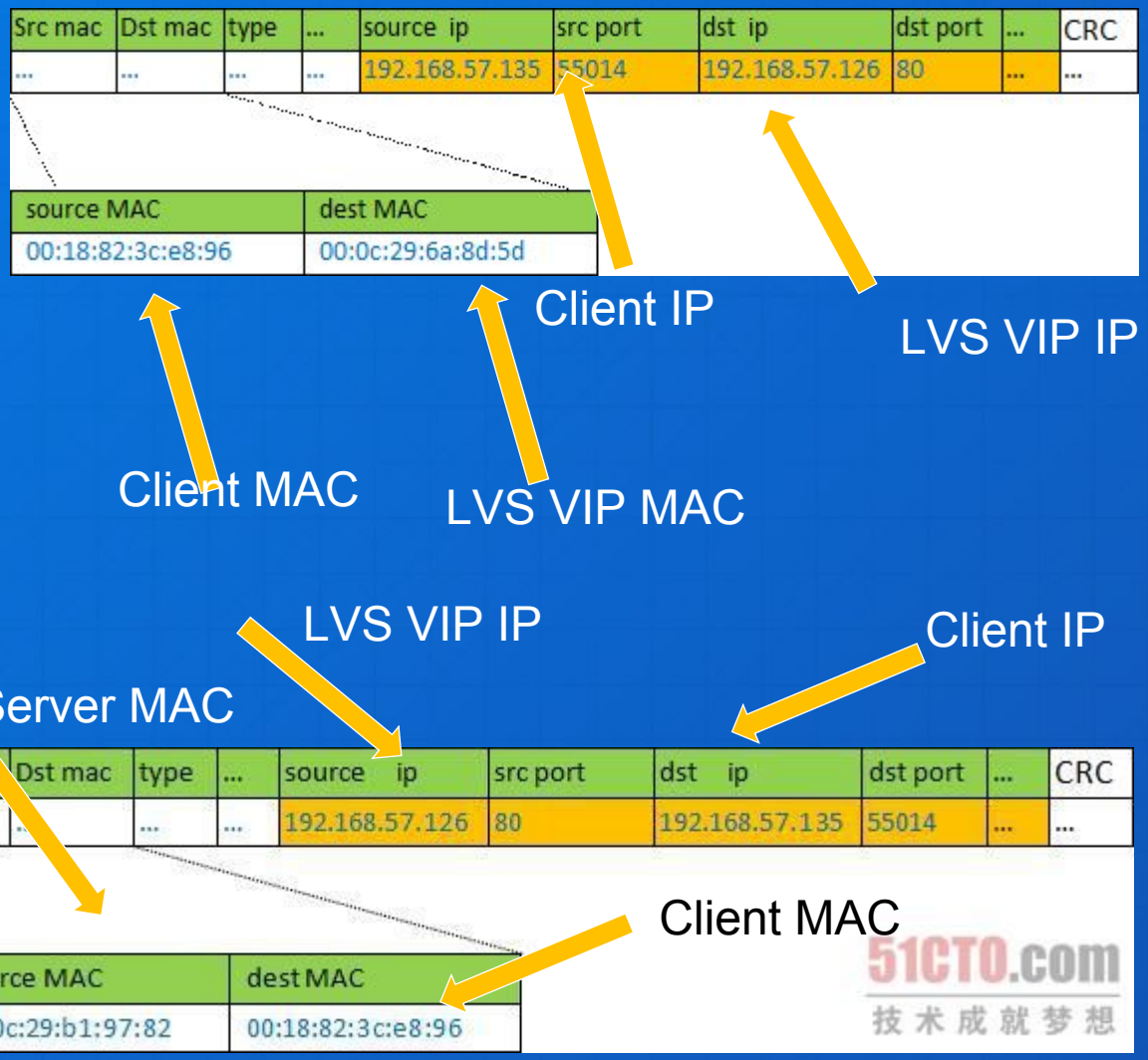
缺点：要求负载均衡器的网卡必须与物理网卡在一个物理段上。

VS-DR模式工作原理

1.client 发送一个pv请求给VIP；VIP收到这请求后会根据LVS设置LB算法选择一个realserver，然后把此请求的package 的MAC地址修改为realserver的MAC地址；dst mac 是LVS VIP的当前服务主机的网卡MAC。如右图：

2.ARP协议会把这个包发送给真正的realserver，并且修改dst mac 改成realserver的MAC 地址

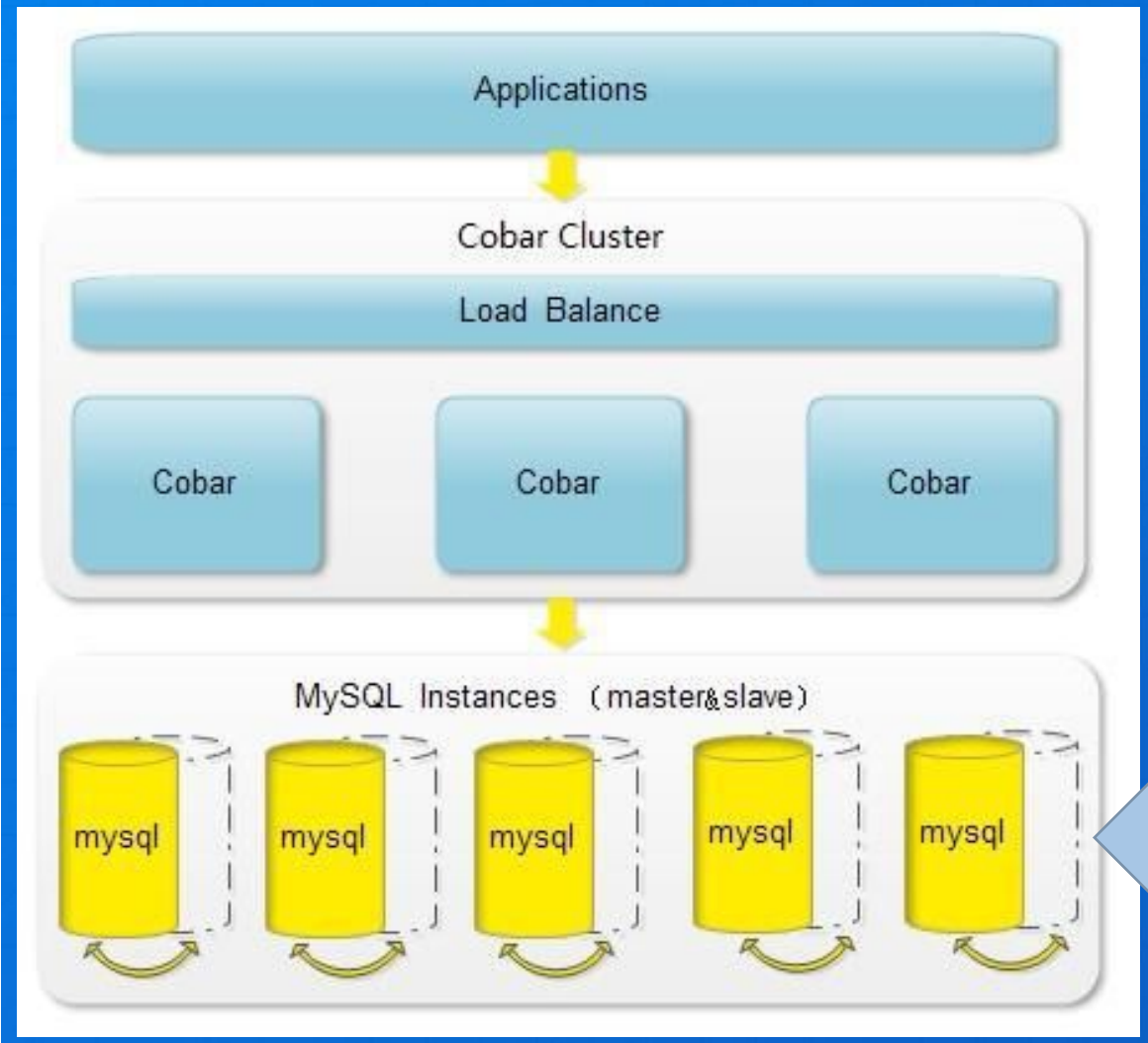
3.Realserver收到这个package后，会判断是否dst mac是否是自己的，不是则丢弃这个package。如果是自己的，则处理，并直接发送给client。发送包的package格式



三种模式比较

	NAT模式	IPIP模式	DR模式
对服务器结点要求	服务结点可以是任何操作系统	必须支持IP隧道模式，目前只有Linux	服务结点支持虚拟网卡设备，能够禁用设备的ARP响应
网络要求	拥有私有IP地址的局域网络	拥有合法IP地址的局域网或广域网	拥有合法IP地址的局域网，服务结点与均衡器必须在同一个网段
通常支持结点数	10~20个，试均衡器处理能力而定	较高，可以支持到100个服务结点	较高，可以支持到100个服务结点
网关	均衡器即为服务器结点网关	服务结点同自己的网关或者路由器连接，不经过均衡器	服务结点同自己的网关或者路由器连接，不经过均衡器
服务结点安全性	较好，采用内部IP，服务结点隐蔽	较差，采用公用IP地址，结点完全暴露	较差，采用公用IP地址，结点完全暴露
IP要求	仅需要一个合法IP地址作为VIP	除VIP外，每个服务结点需拥有合法的IP地址，可以直接路由至客户端	除VIP外，每个服务结点需拥有合法的IP地址，可以直接路由至客户端

LVS在阿里开源Cobar框架中的应用



在单个切片节点的Mysql主备环境中，需要做到单节点的Mysql高可用，这里必须使用LVS来做到高可用。

LVS案例分享

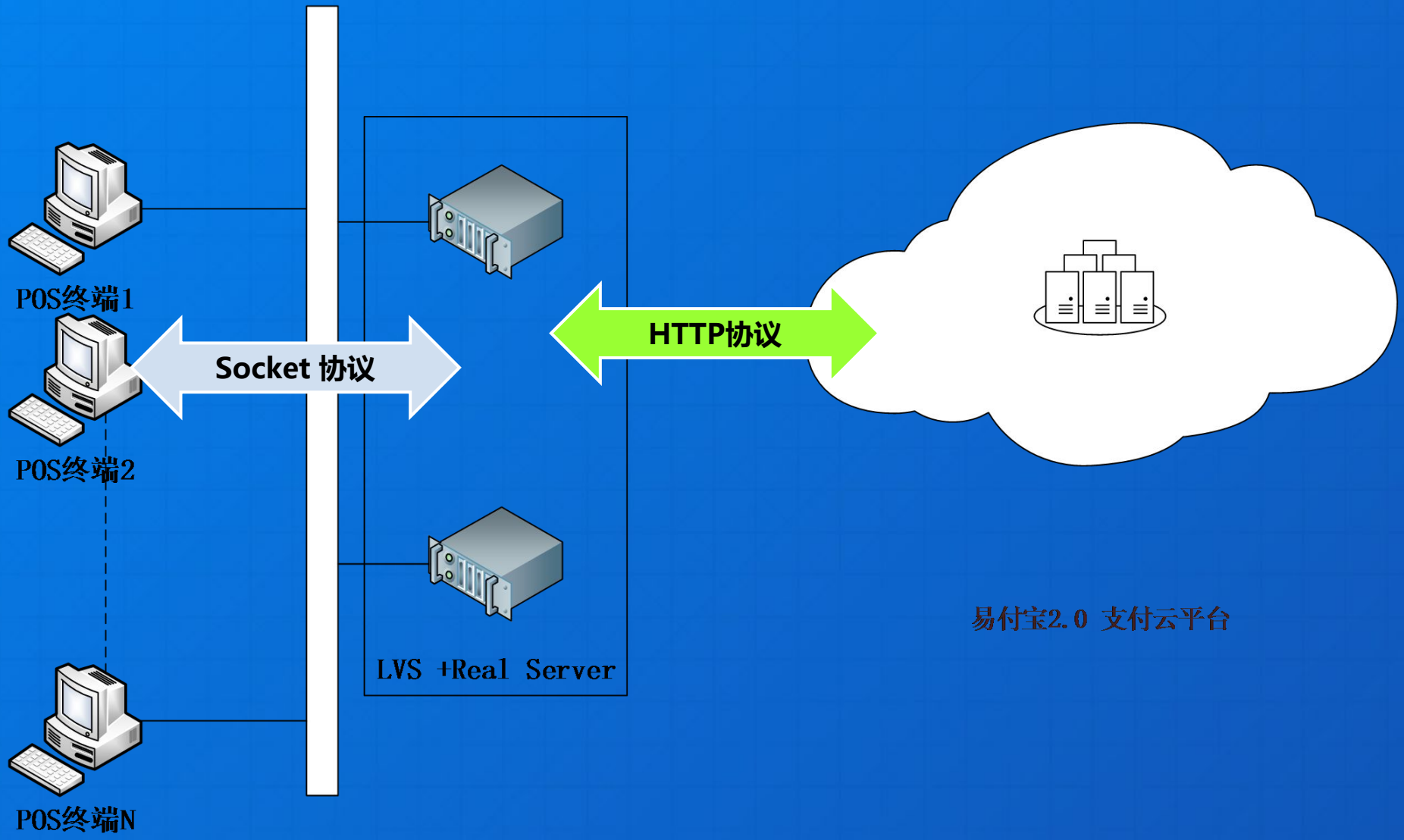
案例分享-需求背景

需求场景：易付宝2.0需要对接线下商户，例如超市，综合商场，餐饮连锁等；

由于易付宝2.0提供对外的接口是HTTP restful方式，而线下实体店是POS系统，实现两个系统对接，需要一个中间接口转换平台，这个接口转换平台需要高度可用，并且能达到花最少的投资，达到最好的收益。如果不计入成本，那么我们完全可以采用有四层交换设备达到硬件负载均衡和后端多服务器的部署模式，但实际上这样的一个商户一个中间平台，服务请求压力并不大，所以我们采用LVS来达到这样的高可用和负载均衡。商户具备以下特点：

- 线下商户众多，预期对接上千家商户
- 日常并发请求压力不大（每家POS终端并不是非常多，规模最大的数千台，最小的200台左右）
- 特殊节假日促销要能满足一定的高并发请求支付
- 要确保支付随时顺利进行

案例分享-线下XX超市


POS终端1
POS终端2
POS终端N
Socket 协议
LVS +Real Server
HTTP协议
易付宝2.0 支付云平台

案例分享-线下XX超市

线下XX超市有1000余台POS终端，这里采用的是前期采购的两台刀片服务器，LVS和 Real Server同时部署在两台机器上，节省了分离部署模式需要最少4台的要求，降低硬件投入成本，又确保高可用和负载均衡。

硬件环境：2U联想刀片机器，2*6核CPU，RAM 16GB，HD 500GB

OS：RHEL 6.3
软件包：ipvsadm-1.26.tar.gz、keepalived-1.2.12.tar.gz
物理机器IP：192.168.0.236,192.168.0.238 (同一子网) 服务端口：8898
虚拟IP(VIP)：192.168.0.239 端口：8898

网络逻辑图：

案例分享-线下XX超市

使用make，make install 安装完ipvsadm和keepalived后，修改/etc/keepalived/keepalived.conf，（“字样” 需要根据实际环境进行相应的修改） 内容如下：

```
! 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 172.0.0.1
   smtp_connect_timeout 30
   router_id LVS_SUNINGPOSINF_PRD
}

vrrp_instance VI_1 {
    state MASTER    ---A机器为MASTER,B机器为BACKUP,请注意机器上修改
    interface eth0
    virtual_router_id 51
    priority 100 ---主机A 100，备机B 99，请主机机器上修改
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
       192.168.0.239
    }
}

virtual_server 192.168.0.239 8898 {
    delay_loop 6
    lb_algo rr
    lb_kind DR
    persistence_timeout 50
    protocol TCP

    real_server 192.168.0.236 8898 {
        weight 1
        TCP_CHECK {
            connect_timeout 10
            nb_get_retry 3
            delay_before_retry 3
            connect_port 8898
        }
    }

    real_server 192.168.0.238 8898 {
        weight 1
        TCP_CHECK {
            connect_timeout 10
            nb_get_retry 3
            delay_before_retry 3
            connect_port 8898
        }
    }
}
```

案例分享-线下XX超市

分别在机器A和机器B上创建/etc/keepalived/realserver.sh脚本文件，对eth0端口绑定虚拟IP 10.10.3.102

```
#!/bin/bash
#description : start realserver
VIP=192.168.0.239
/etc/rc.d/init.d/functions
case "$1" in
start)
echo " start LVS of REALServer"
/sbin/ifconfig lo:0 $VIP broadcast $VIP netmask 255.255.255.255 up
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
;;
stop)
/sbin/ifconfig lo:0 down
echo "close LVS Directorserver"
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 "Usage: $0 {start|stop}"
exit 1
esac
```

chmod +x /root/realserver.sh, 分别启动A,B机器上的keepalived服务以及realserver.sh脚本,执行sh /root/realserver.sh start,service keepalived start。

进行测试，断开A或B的eth0网线，都可以访问到10.10.3.102:8898端口的服务，或者结合日志进行查看。启动lvs ：service keepalived start

验证日志：tail -f /var/log/messages

验证LVS负载状态： ipvsadm –ln 可以看到lvs分发的路由策略

案例分享-线下XX超市

评估：8.22日**线下XX**超市进行线下促销活动，支付请求量有一定的提升，可以看到两台机器上按照配置的平均分发的支付请求处理过程和对应的日志，满足预期的业务目标。

策略：后续如果碰到更大的POS终端数量的商户，可以采用LVS服务和部署中间转换程序的Real Server分离模式，这样2+2最小机器规模，或者2+N的可扩展机器方案，应对更高并发的支付请求。

案例分享-问题排查和解决

很不幸，双机同时承担分发和业务处理功能，碰到广播风暴了，开始可以连通VIP，并且能够应答，单过一段时间，便请求无法响应了，单独连两台物理IP和Port又是好的，这就是两台机器互相来回发数据包，甚至死循环下去。

解决：

1.备机的Keepalived服务在用状态，只是分发策略的weight=100分发到备机上的Real Server，weight=0分发到Master，即备机接到包后根据分发策略直接转发本机Real Server，分发策略上限制再次发给Master上的Real Server.

2.备机的Keepalived服务停止，同时eth0网卡不绑定VIP，通过定时任务不停检测master是否正常，可以通过ping检测，如果time out，则启动备机的keepalived和VIP，达到高可用的目的

3.启用第二块网卡，将请求转发Master和Backup的第二块网卡上，(未尝试)

案例分享-回顾

为什么没有采用以下方案？
1、Nginx/Apache
第七层网络协议应用软件，只能做http反向代理和负载均衡，此外也无法解决Nginx本身单点高可用
2、F5/四层交换设备
满足第四层到第七层网络协议，可以满足socket服务高可用和负载均衡，虽然性能和可靠性更好，但成本投入较大，不是最合适的

附录

参考资源

官方介绍：

http://www.linuxvirtuaLVSlserver.org/VS-NAT.html

http://www.linuxvirtualserver.org/VS-IPTunneling.html

http://www.linuxvirtualserver.org/VS-DRouting.html

Ipvsadm官网下载地址：http://www.linuxvirtualserver.org/software/index.html

Keepalived官网下载地址：http://www.keepalived.org/download.html

Thanks

关注「OneAPM 技术公开课」动态

下载讲师 PPT 及视频