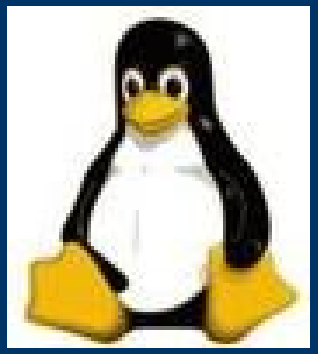
Linux 下的 Cluster 实现

啜立明



Linux 中
实现集群技术

实现负载均衡集群

LB 集群实现之 LVS

- LVS:Linux Virtual Server
- LVS 是由中国国防科技大学章文嵩博士于 1998 年 5 月创立
- LVS 的目标是创建 Linux 下同时具备良好灵活性、可靠性、可管理性的负载均衡软件

LVS

- LVS 包括 IPVS 与 KTCPVS
- IPVS 是基于 IP 层的负载均衡，同时 IPVS 已经嵌入至 Linux Kernel 中。
- 确保 Kernel 中已经包含 IPVS
- Networking->Networking Options->IP: Virtual Server Configuration

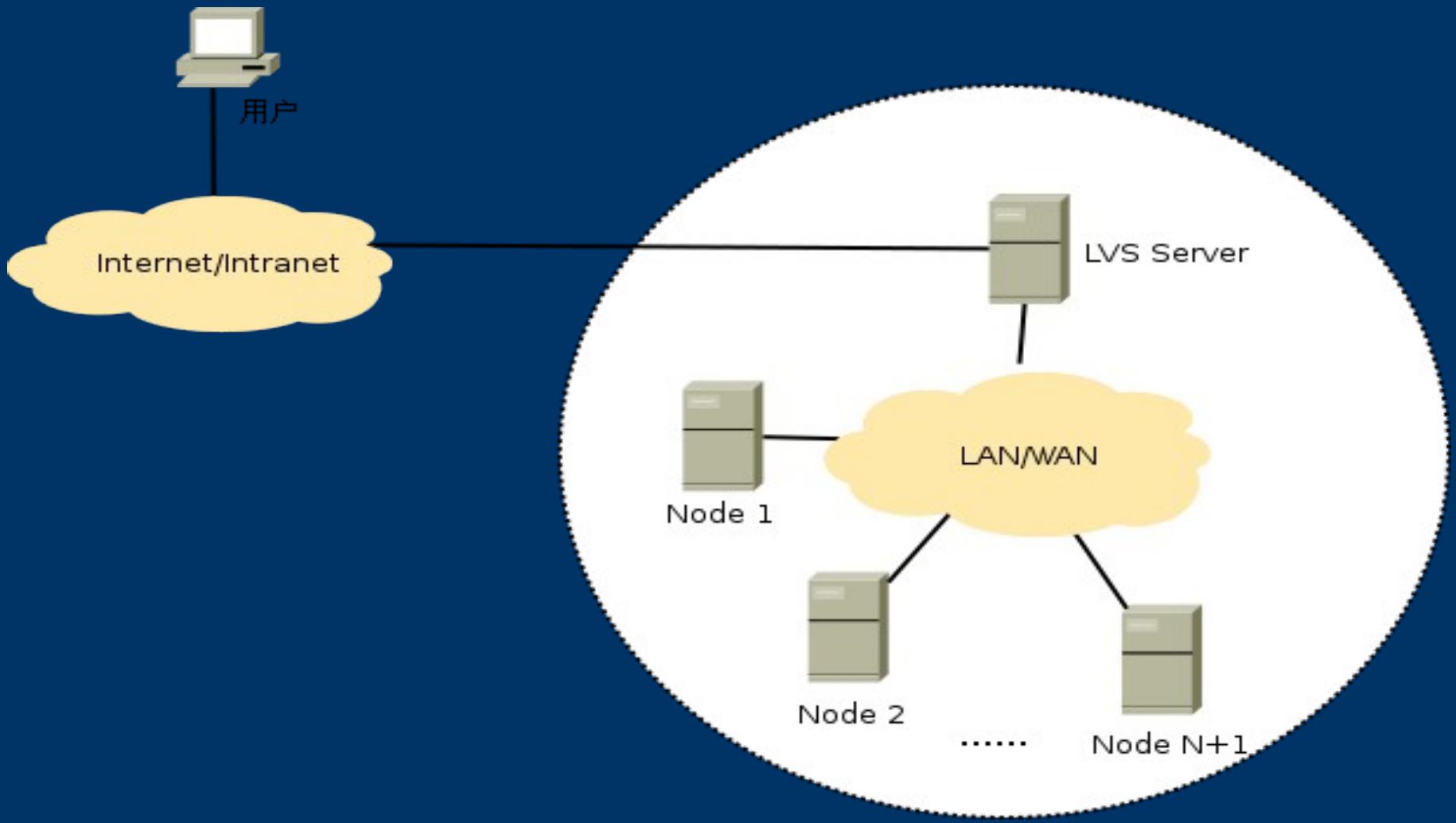
LVS 工作原理

- LVS 需要一个 IP 网关。即对外部网络只能看到 LVS 服务器自己。
- LVS 服务器在接收到用户的请求时，LVS 服务器将根据具体情况在把这些请求分发到各个真实服务器节点

LVS 工作原理

- 真实服务器节点在完成请求后把响应的结果返回给 LVS 服务器
- 最终再由 LVS 服务器将结果返还给用户

LVS 工作原理（图）



LVS 工作模式

•LVS 服务器与服务节点间的工作模式有：

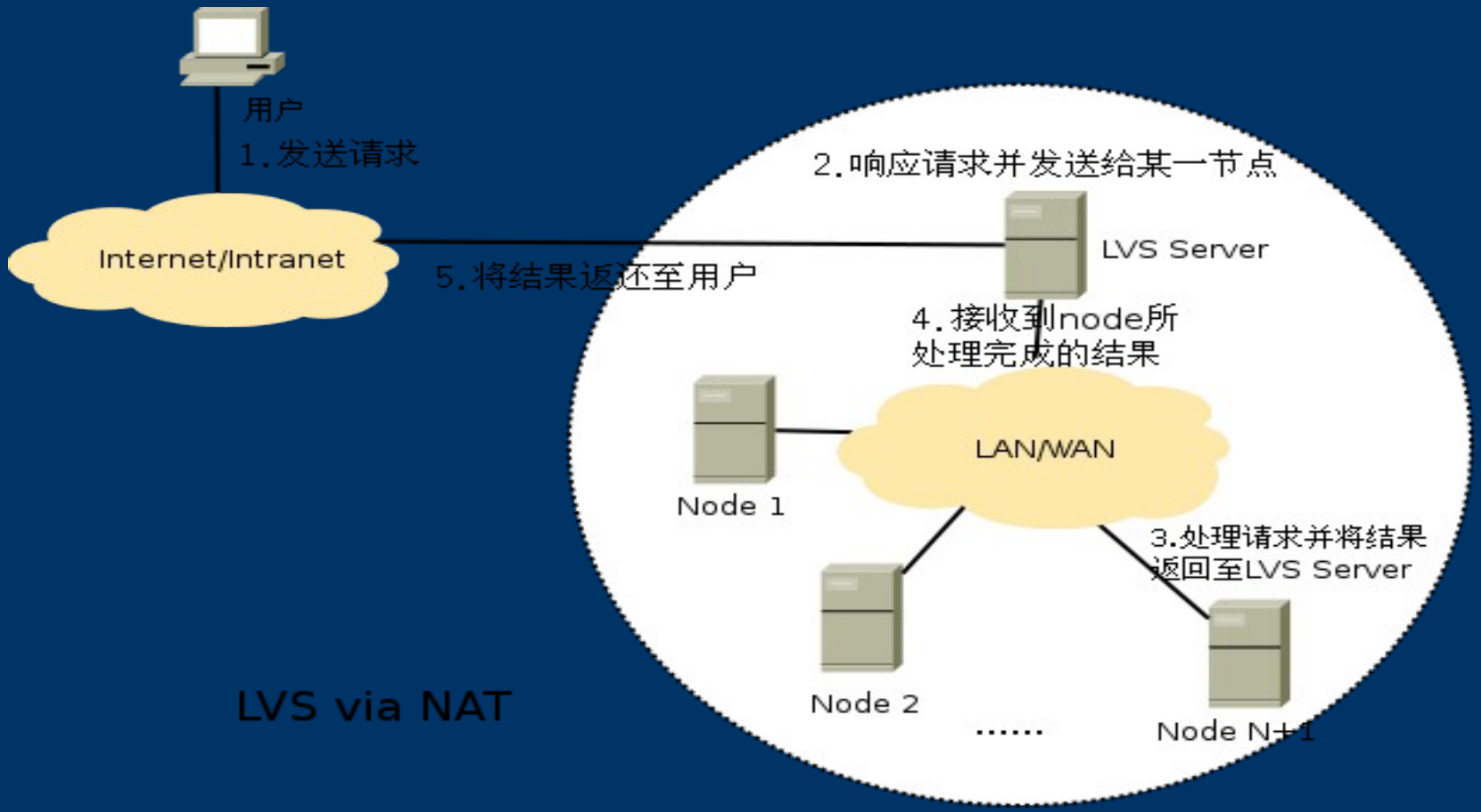
•(1)NAT

通过 NAT 网络地址转换方式实现负载均衡。

LVS 服务器同时充当一台 NAT 网关，拥有公有 IP ，同时负责将针对此公共 IP 的请求依据算法将请求转发给 LAN 中的某台真实服务器 (node) ， node 处理完成请求后将结果返回至 LVS Server, 在由 LVS Server 将结果返回给用户。

LVS 工作模式

- LVS via NAT(图)



LVS via NAT

LVS 工作模式

- LVS via NAT

NAT 方式能够很好的将真实服务器全部隐藏起来就像 NAT 一样。但因 LVS Server 全权负责请求与结果的转发这就需要 LVS Server 系统性能要求很高，同时 NAT 方式属于伪装方式中的一种因此需要所有的 node 都需要在同一个 LAN 中并且 node 中的默认网关都指向 LVS Server 的 IP 。

LVS 工作模式

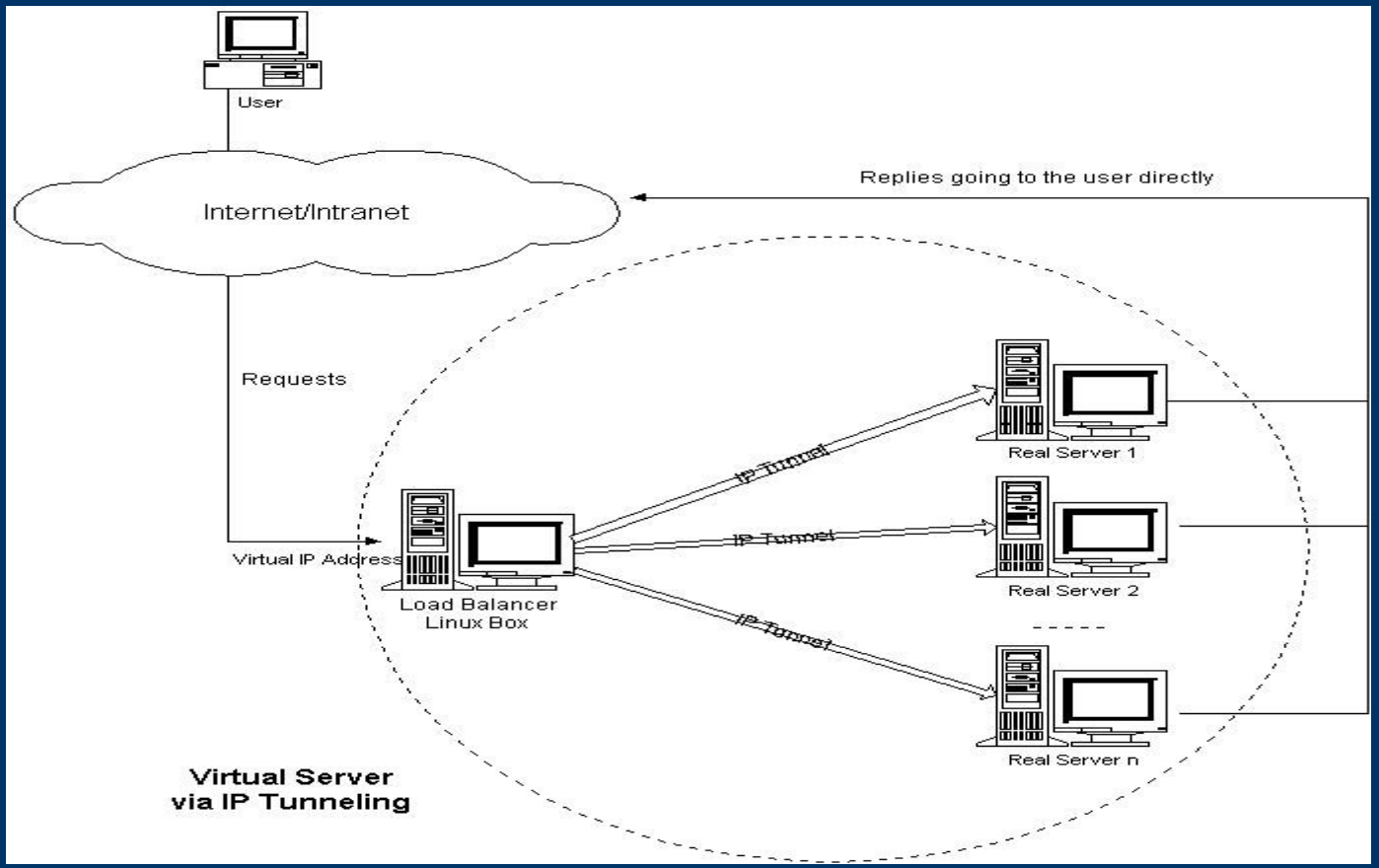
•(2)IP 隧道模式

用 IP 隧道技术实现虚拟服务器。这种方式可以让集群中的 node 可以在不同的网段中进行工作，是将 IP 包封装在其他网络流量中的工作方法。

考虑到安全因素，IP 隧道模式应该采用 VPN 隧道技术，也可使用专线方式

LVS 工作模式

- (2)IP 隧道模式 (图)



LVS 工作模式

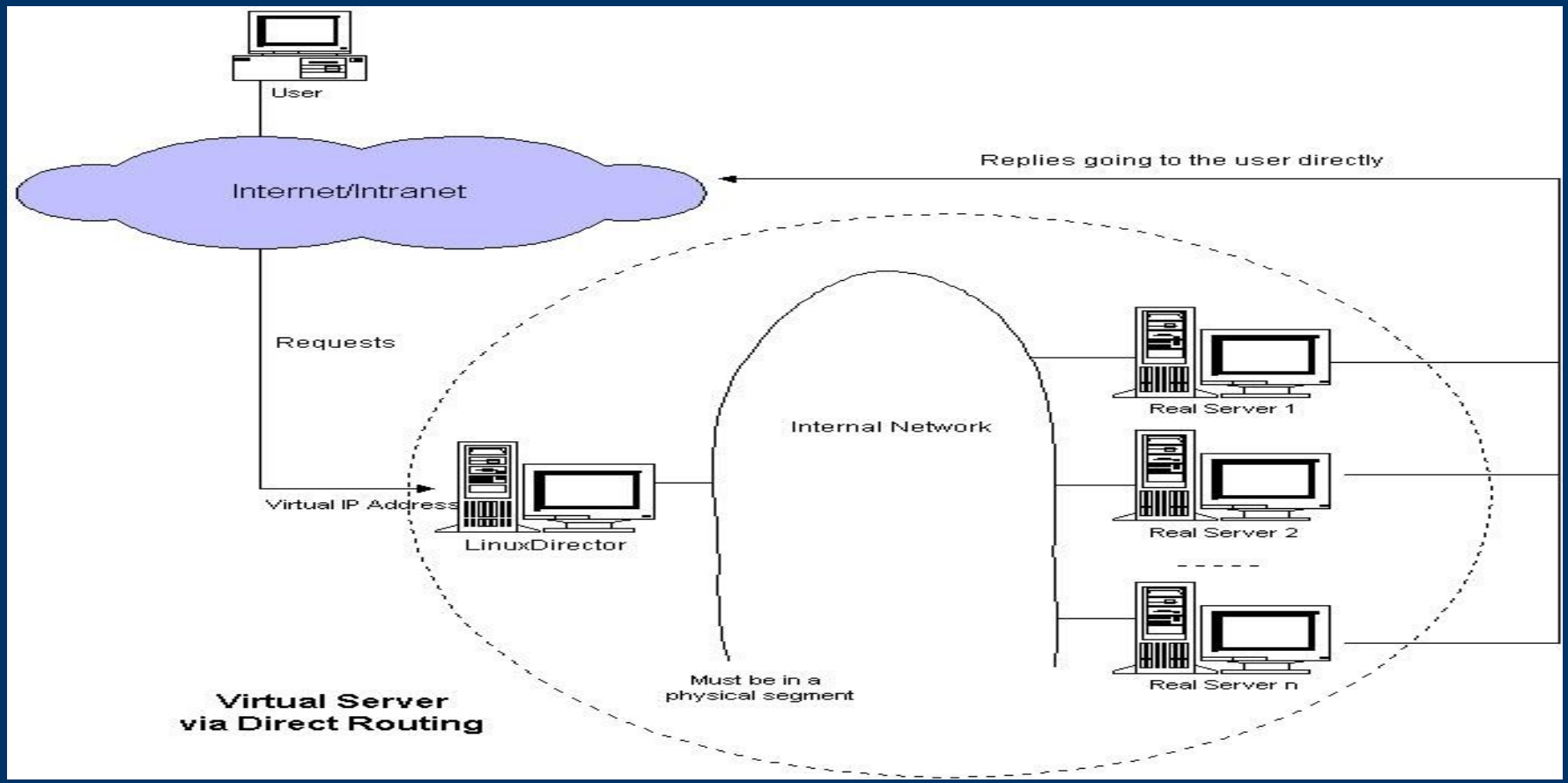
- (3) 直接路由方式 (DR)

当参与集群的计算机和作为控制管理的计算机在同一个网段时可以使用此种方法。

控制管理的计算机接收到请求包时直接送到参与集群的 node 上。当 node 处理完请求后将直接把结果返还至用户而不通过 LVS Server 返还。 DR 的优势在于速度快、开销少。

LVS 工作模式

- DR 工作模式 (图)



LVS 工作模式

- 三种模式比较（一）

	VS/NAT	VS/TUN	VS/DR
Server	any	Tunneling	Non-arp device
server network	private	LAN/WAN	LAN
server number	low (10~20)	High (100)	High (100)
server gateway	load balancer	own router	Own router

LVS 工作模式

- 从服务器连接访问方式看， NAT 支持任何访问方式，但 TUN 只能以隧道方式访问后台服务器。

- 从网络布局来看 ,NAT 与 DR 方式都要求所有的真实服务器与 LVS Server 在同一个网络中，而 TUN 可以位于同一网络或者不同网络，甚至可以存在于外部网络中

LVS 工作模式

- 从支持的服务器数量看， NAT 支持的 node 很少，而 TUN 与 DR 模式支持很多

- 从网关配置方式看 ,NAT 中 LVS Server 的 IP 地址必须是 node 中的默认网关。而 TUN 与 DR 则不受此限制

LVS 工作模式

- NAT 方式适用于在同一个 LAN 中实现小型 LB
- TUN 方式适用于 node 部分在 Internet 上
- DR 方式使用与在同一个 LAN 中实现较为大型 LB

LVS 工作模式

- 三种模式比较（二）

	网络地址转换	直接路由	IP 隧道
建立难易程度	易	有一定难度	
可扩展性	差	好	好
带宽	小	大	大
延迟	最大	小	大
支持的服务器数量	少	多	多
IP 包修改	双向修改数据包中的 IP 与端口	仅修改请求包中的 MAC 地址	仅对请求包进行 IP 包装
实际服务器 OS	可提供服务即可	多数	支持 IP Tunnel
网络连接要求	局域网		本地或远程
缺省路由	负载均衡服务器	不限	
实际服务器限制	无	Lo 设备不响应 ARP	Tunl 设备不响应 ARP
服务端口重映射	可以	不可	

LVS 算法

- 在将 LVS Server 把请求发送到 node 时除了考虑工作模式之外，也需要考虑在那种情况下把请求转发给最适合的 node 来完成请求的处理。依据不同的环境 LVS 在目前可以支持 8 种算法。

LVS 算法

- (1)rr(Round Robin— 轮询算法)

  rr 算法就是将外部请求顺序轮流分配到集群中的 node 上，但不考虑每台 node 的负载情况。

- (2)wrr(Weighted Round Robin- 加权轮询算法)

  wrr 算法在 rr 算法的基础上会考察每台 node 的负载情况，并尝试让负较轻的 node 承担更多请求。

LVS 算法

•(3)lc(Least Connections- 最少连接算法)

lc 算法可以让 LVS 尝试把新的请求交给当前连接数最少的 node ，直到此 node 连接数不再属于最少标准

•(4)wlc(Weighted Least Connections- 加权最少连接算法)

wlc 算法也由权重的干预。 LVS 会根据每台 node 的权重并综合连接数控制转发行为

LVS 算法

- (5)lblc(Locality-Based Least Connections)
- lblc---- 局部性最少连接算法

lblc 算法会加上针对源请求 IP 地址的路由估算，并尝试把请求发送到与源请求 IP 路由最近的 node 上。此种方法一般用于远程或者是大规模的集群组

LVS 算法

•(6)lblcr(Locality-Based Least Connections with Replication)
•lblcr— 带有复制的局部性最少连接算法

lblcr 算法是在 lblc 算法的基础上增加了一个 node 列表，先依据 lblc 算法计算出与源请求 IP 地址最近的一组 node ，然后在决定把请求发送到最近一组中的最近的一台 node 。若此 node 没有超载则将请求转发给这台 node, 如果超载则依据"最少连接"原则找到最少连接的 node 并将此 node 加入集群组中。并将请求转给此 node

LVS 算法

- (7)dh(Destination Hashing— 目标地址散列算法)

dh 算法是把请求的目标 IP 地址进行 hash ，并与 node 列表进行 hash 配对，如果 node 可用且并未超载则将请求发送到此 node 上，否则返回空值。属于一种类似于随机的方式分配请求。

LVS 算法

- (8)sh(Source Hashing— 源地址散列算法)

  sh 算法是将请求的源地址进行 hash, 然后在执行与 dh 算法类似的操作。

安装 LVS

- (1) 首先确认 Linux Kernel 已经支持 LVS
- (2)RedHat/CentOS 可以通过光盘直接安装或通过网络仓库进行安装
- (3)Debian/Ubuntu/Gentoo/ArchLinux 则可以通过网络仓库进行安装
-
- (4) 管理 LVS 的工具软件为 ipvsadm

安装 LVS

- CentOS

```
#yum  install  ipvsadm
```

- Debian/Ubuntu

```
#apt-get  install  ipvsadm
```

- ArchLinux

```
#pacman -Sy ipvsadm
#yaourt -Sy ipvsadm
```

安装 LVS

- ipvsadm 语法格式

#ipvsadm　一级指令　二级指令　二级指令参数　....

* 一级指令仅告之 ipvsadm 要执行的操作类别

* 二级指令则告知 ipvsadm 具体要执行哪些操作

* 直接使用 ipvsadm 指令将会输出当前虚拟服务的一些基本情况。

LVS 语法参数

- (1) 在完成 LVS 时首先需要定义和管理 LVS 的虚拟服务器。以下参数为管理虚拟服务器所用参数

- -A 为一级指令，告之 ipvsadm 需要在 kernel 的 LVS 列表中增加一个新的虚拟服务器记录。

- -s 指定虚拟服务所使用的算法

- -t|-u 虚拟服务器地址：端口

  指定虚拟服务器的 IP 地址， -t 为指定 TCP 端口， -u 为指定 UDP 端口

LVS 语法参数

- 例
- #ipvsadm -A -s rr -t 192.168.1.168:80
- 表示向 kernel 中的 LVS 列表增 (追) 加一个针对 TCP 端口 80 的虚拟服务器。此虚拟服务器地址为 192.168.1.168 ，所采用的 LVS 算法为 rr (轮询算法)

LVS 语法参数

- -E 编辑某个虚拟服务的参数，需要提供必需的替代参数

- 例

将 rr(轮询) 算法改为 lc(最少连接数) 算法

#ipvsadm -E -t 192.168.1.168:80 -s lc

LVS 语法参数

- -D 删除 LVS 列表中某个虚拟服务器记录
-
- 例
- #ipvsadm -D -t 192.168.1.168:80

LVS 语法参数

- 管理 LVS 真实服务器所需参数
- -a 添加完成虚拟服务器后需要写入某个请求由那些 node 进行处理
- -t|-u 指定协议类型及所用端口号
- -r 指定 node 地址与其监听的端口号

LVS 语法参数

- -g|-i|-m

  指定 LVS 虚拟服务器与此 node 所使用的工作模式。

-g 为 DR 模式
-i 为 TUN 模式
-m 为 NAT 模式

LVS 语法参数

- -w　指定每台服务器 (node) 权重，权重一般均为正整数。仅提供给需要权重的算法

- 例

```
#ipvsadm -a -t 192.168.1.168:80 -r 192.168.1.174:80 -m -w 2
```

- -e　修改指定的真实服务器 (node)

- -d　删除指定的真实服务器 (node)

LVS 语法参数

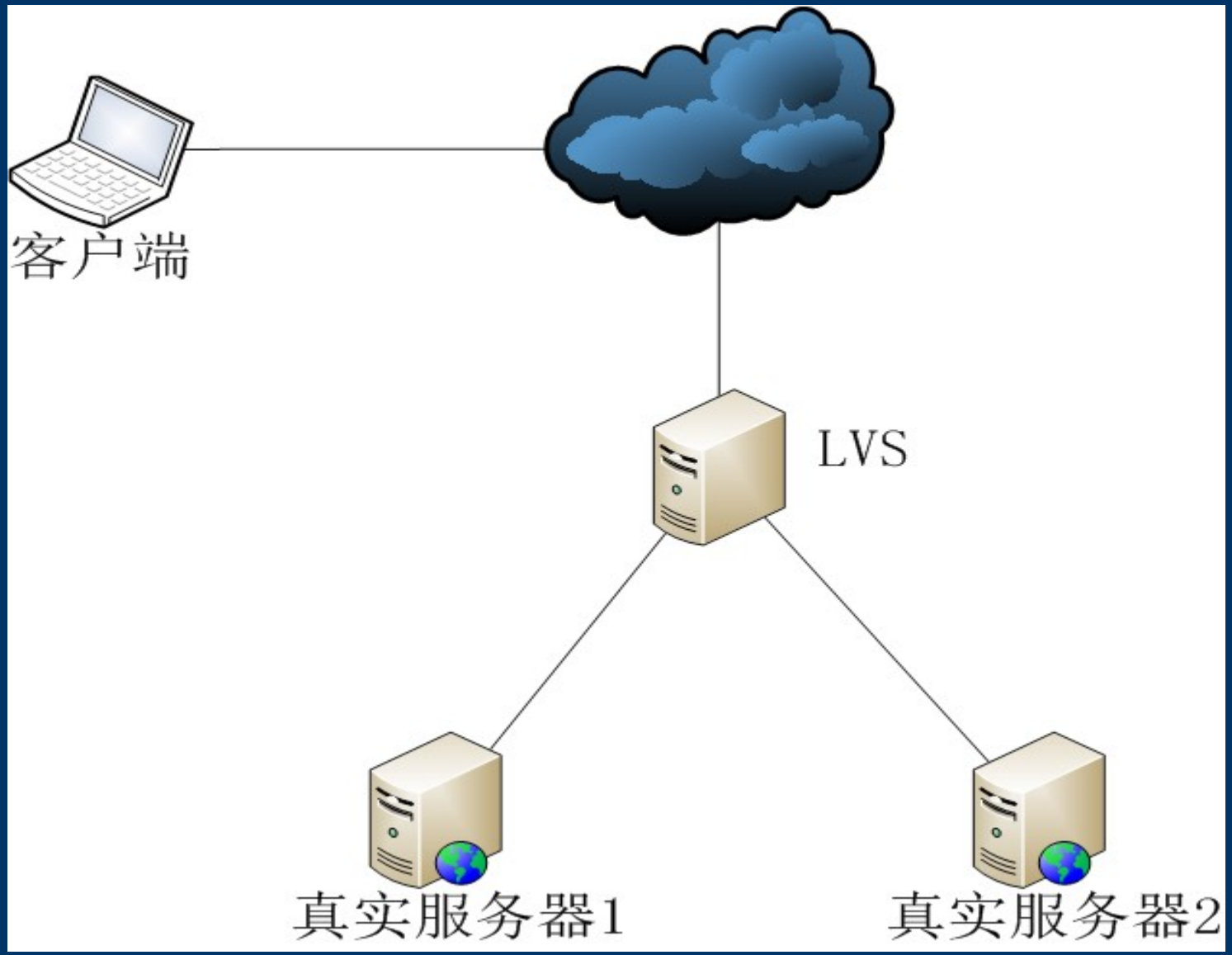
- -C　清除 LVS 列表中虚拟服务器与真实服务器 node 所有记录
- -S　保存当前的 LVS 记录为文件
- -R　将保存的文件读取到 Kernel 中
- -Z　将当前 LVS 连接计数器清零，对某些算法有可能将重新开始计算。

LVS 语法参数

- -L|-l
  显示当前内核中的 LVS 状态，其二级指令有

(1)-c 显示当前 LVS 的连接情况
(2)--stats 显示统计数据
(3)--rate 显示速率数据
(4)--sort 对输出进行排序
(5)-n 显示主机时直接输出 IP 地址，忽略主机名从而达到加快显示速度

实现 LVS 集群



LVS 集群实现 NAT 方式

- 注

  一般将 lvs-server 称为 Director Server(前端调度器)

  一般将 node 称为 real-server(真实服务器)

LVS 集群实现 NAT 方式

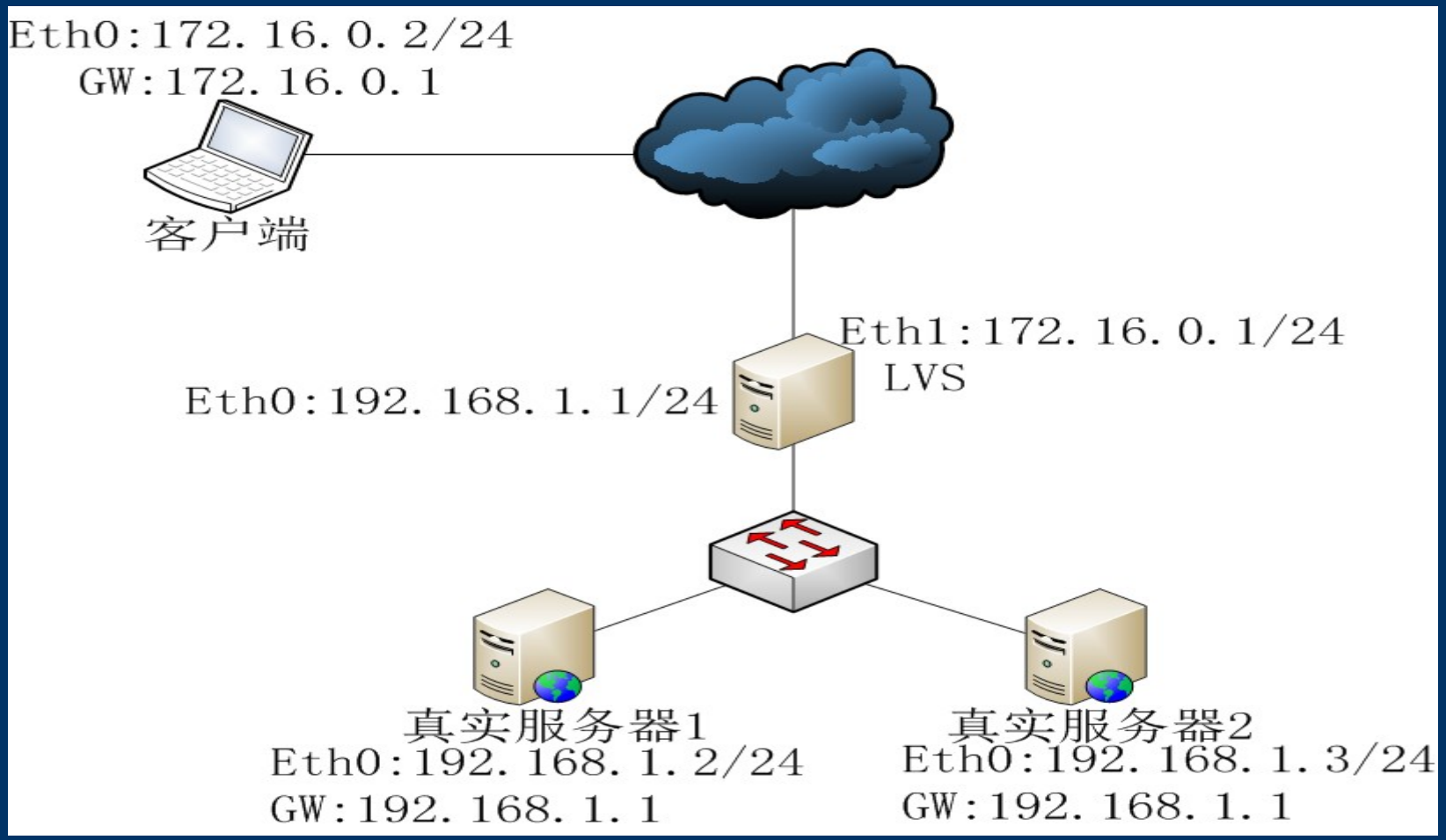
- (1) 验证 ipvsadm 是否安装正确

- #ipvsadm

```
[root@bogon ~]# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
```

LVS 集群实现 NAT 方式

- (2)NAT 试验拓扑



LVS 集群实现 NAT 方式

- (3) 依据网络拓扑为各服务器设置 IP 地址及网关
- (4) 启动 iptables 与 ipvsadm 服务
- (5) 启动真实服务器 Apache 服务
- (6) 在虚拟服务器端书写 LVS 的 NAT 模式的集群配置

LVS 集群实现 NAT 方式

•书写内容如下 (#vim lvs-nat.sh)

```
#! /bin/bash
# 默认所有 iptables 策略为空，默认策略为允许

iptables -t nat -A POSTROUTING -s 192.168.1.0/
24  -j MASQUERADE

echo 1 > /proc/sys/net/ipv4/ip_forwad
```

LVS 集群实现 NAT 方式

- (续上)

ipvsadm -C

ipvsadm -A -t 172.16.0.1:80 -s wrr

ipvsadm -a -t 172.16.0.1:80 -r 192.168.1.2:80 -m -w 2

ipvsadm -a -t 172.16.0.1:80 -r 192.168.1.3:80 -m -w 3

LVS 集群实现 NAT 方式

- 权重设置一般为正整数，数值越大越会被多分配到请求， 0 为不可用。
- (7) 使用客户端进行测试
- (8) 验证 LVS 集群 NAT 方式是否工作

LVS 集群实现 NAT 方式

```
[root@bogon ~]# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  172.16.0.1:http wrr
  -> 192.168.1.3:http             Masq    3      0          0
  -> 192.168.1.2:http             Masq    2      0          0
[root@bogon ~]# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  172.16.0.1:http wrr
  -> 192.168.1.3:http             Masq    3      0          1
  -> 192.168.1.2:http             Masq    2      0          0
[root@bogon ~]# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  172.16.0.1:http wrr
  -> 192.168.1.3:http             Masq    3      0          1
  -> 192.168.1.2:http             Masq    2      0          1
[root@bogon ~]#
```

LVS 集群实现 NAT 方式

- ActiveConn 为目前正在活动的连接数
- InActConn 为目前不活动的连接数
- 请注意如果流量很大，将会对 VS 性能造成严重影响

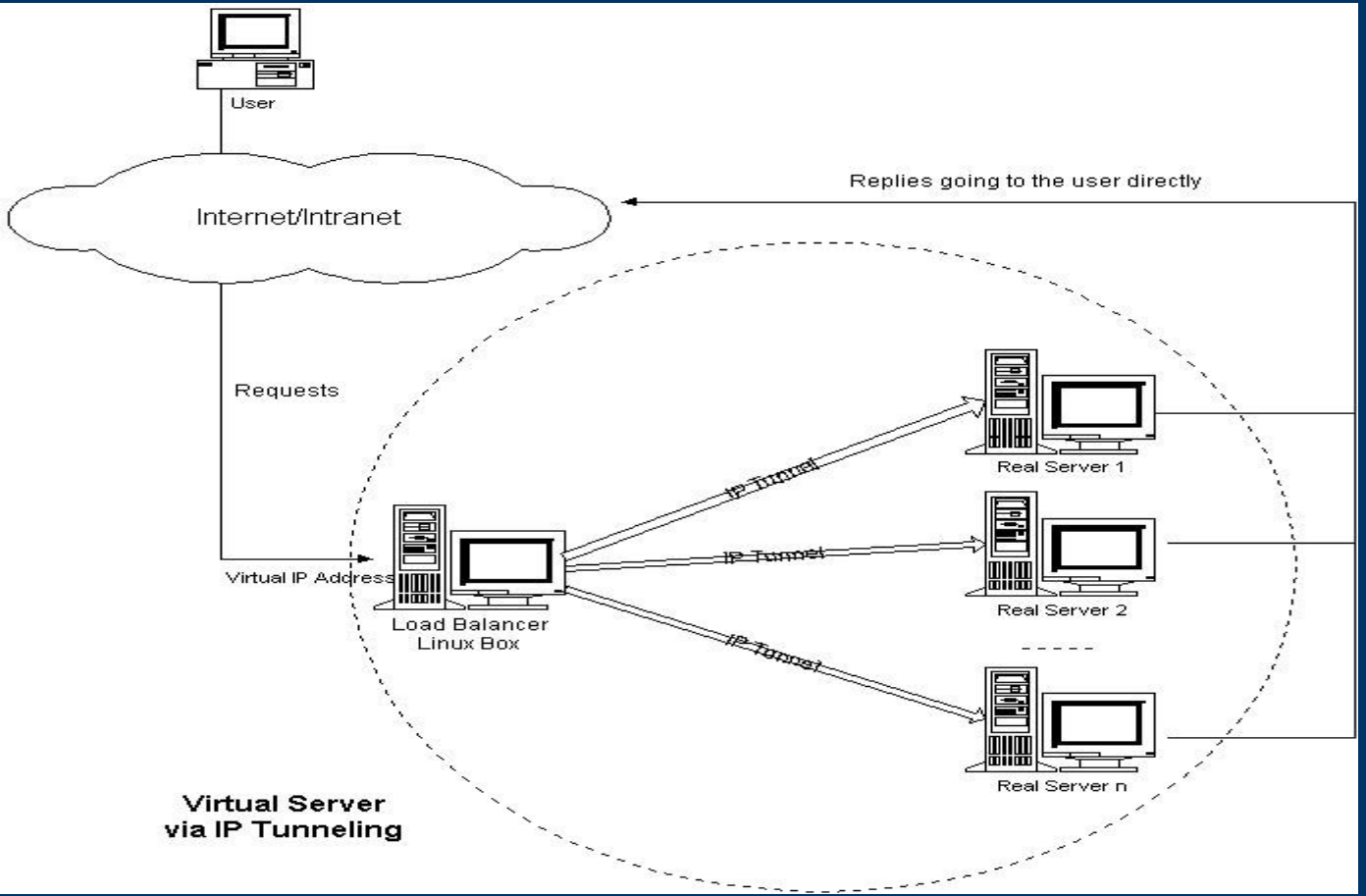
LVS 集群实现 NAT 方式

- 实现 LVS 中 NAT 方式的负载均衡
- 试验目的 : 掌握 NAT 方式的负载均衡
- 试验人员 : 个人
- 所需要计算机设备 : 至少 4 台计算机
- 试验时间 :30 分钟

LVS 集群实现之 TUN

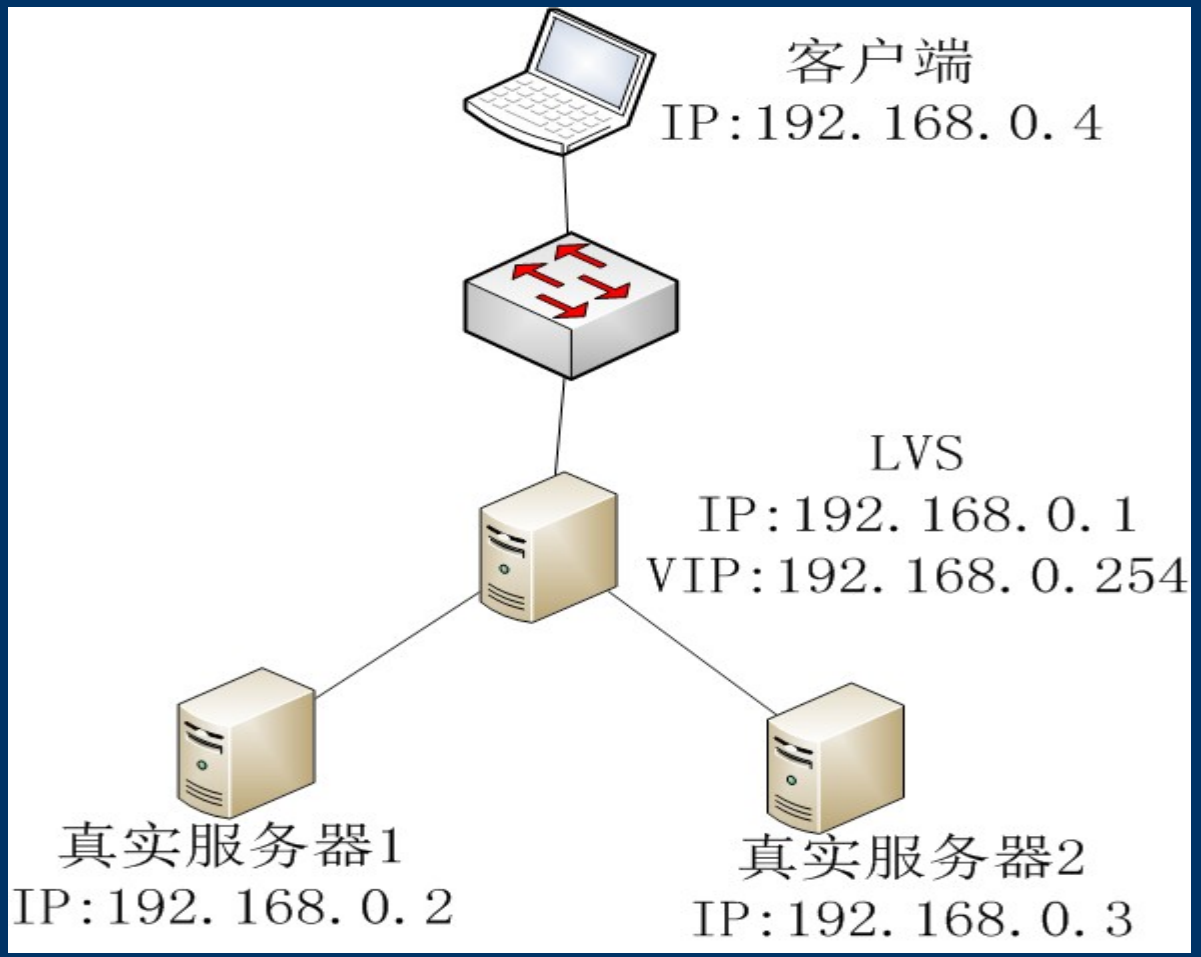
- TUN 指 IP Tunneling 。 其转发流程为
- (1)LVS 设备接收到外界请求，依据相应算法将其通过 IP 隧道发送到相应的 node.
- (2)node 处理完成后，将结果直接返还给客户

LVS 集群实现之 TUN



LVS 集群实现之 TUN

- (1) 试验拓扑图



LVS 集群实现之 TUN

- (1) 配置 LVS

#cat lvs-tun.sh

```
#!/bin/bash
    VIP=192.168.0.254
    RIP1=192.168.0.2
    RIP2=192.168.0.3
```

LVS 集群实现之 TUN

- (1) 配置 LVS

```
/sbin/ipvsadm -C
/sbin/ifconfig eth0:0 $VIP broadcast $VIP net mask 255.255.255.255 up
/sbin/route add -host $VIP dev eth0:0
/sbin/ipvsadm -A -t $VIP:80 -s wlc
/sbin/ipvsadm -a -t $VIP:80 -r $RIP1:80 -i -w 1
/sbin/ipvsadm -a -t $VIP:80 -r $RIP2:80 -i -w 1
```

LVS 集群实现之 TUN

- (2) 各 NODE 配置

#cat lvs-real-server.sh

```
#!/bin/bash
    VIP=192.168.0.2
    /sbin/ifconfig tunl0 down
    /sbin/ifconfig tunl0 up
    echo 1 > /proc/sys/net/ipv4/conf/tunl0/arp_ig
nore
```

LVS 集群实现之 TUN

- (2) 各 NODE 配置

echo 2 > /proc/sys/net/ipv4/conf/tunl0/arp_announce

echo 0 > /proc/sys/net/ipv4/conf/tunl0/rp_filter

echo 1 > /proc/sys/net/ipv4/conf/all/arp_ignore

echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce

LVS 集群实现之 TUN

- (2) 各 NODE 配置

  /sbin/ifconfig tunl0 $VIP broadcast $VIP netma
  sk 255.255.255.255 up

/sbin/route add -host $VIP dev tunl0

LVS 集群实现之 TUN

- (2) 各 NODE 配置

#cat lvs-real-server.sh

```
#!/bin/bash
    VIP=192.168.0.3
    /sbin/ifconfig tunl0 down
    /sbin/ifconfig tunl0 up
    echo 1 > /proc/sys/net/ipv4/conf/tunl0/arp_ig
nore
```

LVS 集群实现之 TUN

- (2) 各 NODE 配置

echo 2 > /proc/sys/net/ipv4/conf/tunl0/arp_announce

echo 0 > /proc/sys/net/ipv4/conf/tunl0/rp_filter

echo 1 > /proc/sys/net/ipv4/conf/all/arp_ignore

echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce

LVS 集群实现之 TUN

- (2) 各 NODE 配置

  /sbin/ifconfig tunl0 $VIP broadcast $VIP netma
  sk 255.255.255.255 up

/sbin/route add -host $VIP dev tunl0

LVS 集群实现之 TUN

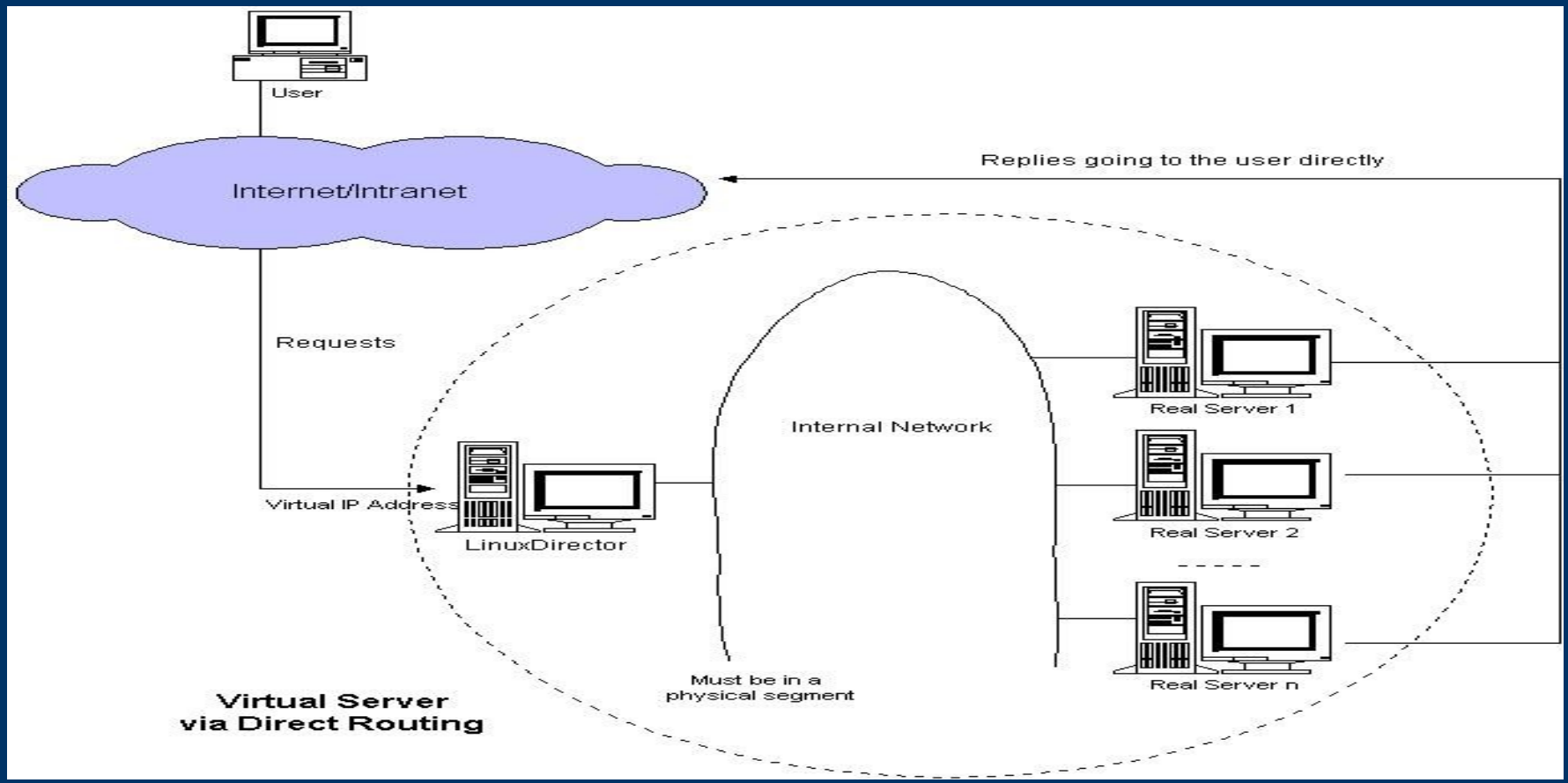
- 实现 LVS 中 TUN 方式的负载均衡
- 试验目的 : 掌握同网内 TUN 方式的负载均衡
- 试验人员 : 个人
- 所需要计算机设备 : 至少 4 台计算机
- 试验时间 :30 分钟

LVS 集群实现之 DR

- dr 指 Direct Routing. 其转发流程为 Director(LVS) 设备收到请求，依据指定算法发送到相应的真实 Node 上， Node 处理完成后直接将结果返还给客户端

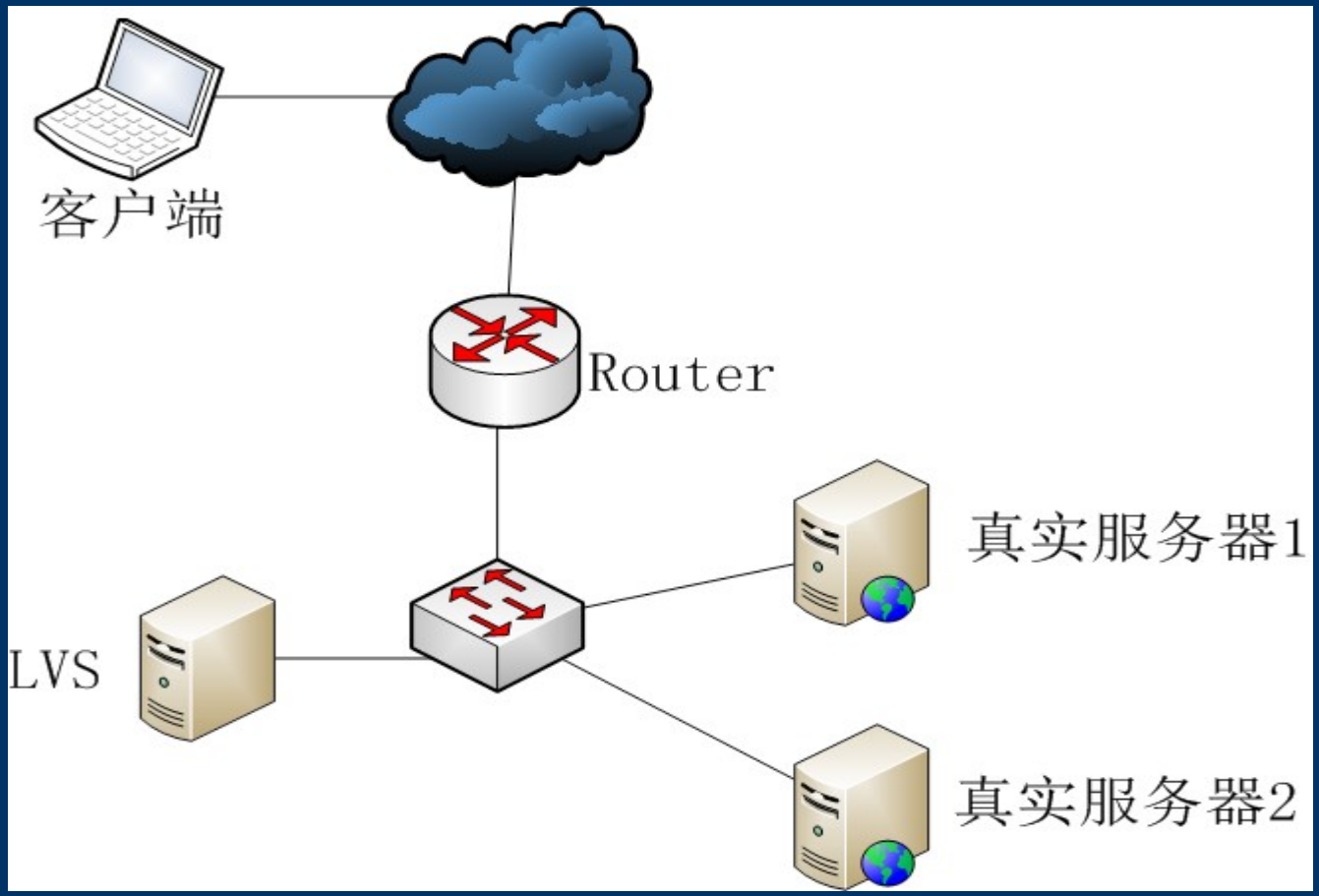
LVS 集群实现之 DR

- DR 流程示意图



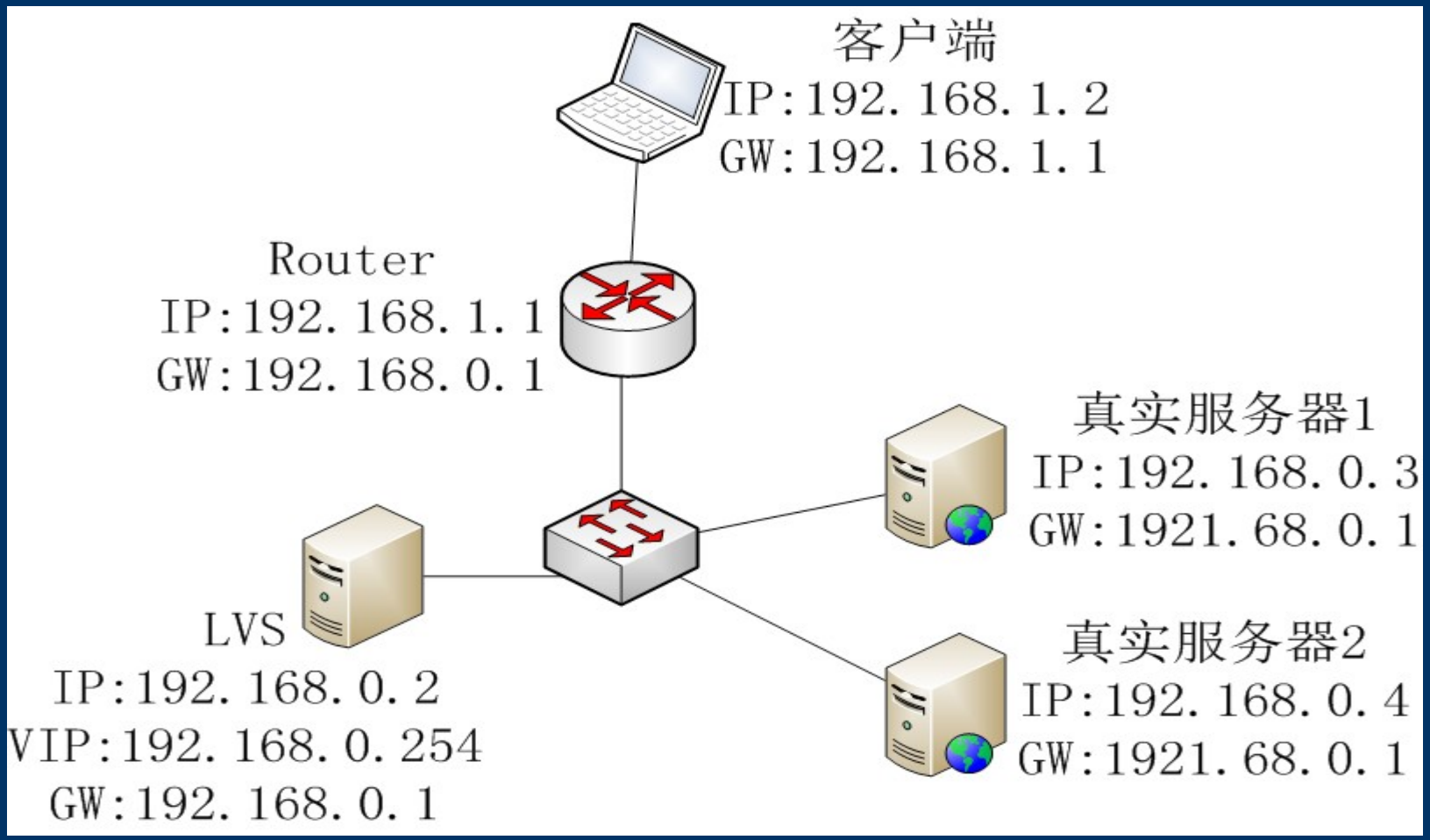
LVS 集群实现之 DR

- (1) 网络拓扑图



LVS 集群实现之 DR

- (2) 试验时逻辑关系图



LVS 集群实现之 DR

•(3) 配置 LVS/DR 环境— LVS 配置

#cat lvs-dr.sh

#!/bin/bash

VIP=192.168.0.254

RIP1=192.168.0.2

RIP2=192.168.0.3

/sbin/ipvsadm -C

LVS 集群实现之 DR

- (3) 配置 LVS/DR 环境— LVS 配置

```
/sbin/ifconfig eth0:0 $VIP broadcast $VIP netmask 255.255.255.255 up
/sbin/route add -host $VIP dev eth0:0
/sbin/ipvsadm -A -t $VIP:80 -s wlc
/sbin/ipvsadm -a -t $VIP:80 -r $RIP1:80 -g -w 1
/sbin/ipvsadm -a -t $VIP:80 -r $RIP2:80 -g -w 1
```

LVS 集群实现之 DR

- (4)node 配置

```
#cat  real-server.sh
------------------------------------
VIP=192.168.0.2
ifconfig lo:0 $VIP netmask 255.255.255.255 bro
   adcast $VIP
/sbin/route add -host $VIP dev lo:0
echo "1" >/proc/sys/net/ipv4/conf/lo/arp_ignor
   e
```

LVS 集群实现之 DR

- (4)node 配置

```
echo "2" >/proc/sys/net/ipv4/conf/lo/arp_anno
  unce

echo "1" >/proc/sys/net/ipv4/conf/all/arp_igno
  re

echo "2" >/proc/sys/net/ipv4/conf/all/arp_anno
  unce
```

LVS 集群实现 DR

- 实现 LVS 中 DR 方式的负载均衡
- 试验目的 : 掌握 DR 方式的负载均衡
- 试验人员 : 个人
- 所需要计算机设备 : 至少 5 台计算机
- 试验时间 :30 分钟

Linux 下的 Cluster 实现

结　　束

master.chuai@gmail.com

304630723

152990419