

如果效果达不到，就是环境不一样

拓扑结构及涉及思路:存储层 服务层 缓存层 负载均衡层 配置高可用层

```
master11
lsblk
vgcreate master /dev/vdb /dev/vdc
vgs
lvcreate -n mysql -L 3G master
lvs
mkfs.ext4 /dev/master/mysql
mkdir /lvm
echo "/dev/master/mysql /lvm ext4 defaults 0 0" >> /etc/fstab
mount -a
df -h----->DAS
tar -xf mysql-5.7.17.tar
yum -y reinstall mysql*.rpm
vim /etc/my.cnf
    server_id=11
    log_bin=master11
    binlog_format="mixed"
systemctl start mysqld
grep password /var/log/mysqld.log
mysql -hlocalhost -uroot -p"fZDzPwE.(10;"
mysql> alter user root@"localhost" identified by "123qqq...A";
mysql> grant replication slave on *.* to repl@"192.168.4.%" identified by "123qqq...A";
mysql> show master status;---->mysql 主从
【mysql> grant all on *.* to user@"192.168.4.111" identified by 123qqq...A;----->授权客户端下载 mariadb-server 测试】
mysql> grant replication slave,replication client on *.* to scalemon@"%" identified by "123qqq...A";
---创建监控用户
mysql> grant select on mysql.* to maxscale@"%" identified by "123qqq...A"; ----->创建路由用户
mysql> grant all on *.* to student@"%" identified by "123qqq...A"; ----->创建访问数据用户
mysql> select user,host from mysql.user where user in ("scalemon","maxscale","student");
----->查询读写分离授权
yum -y install iscsi-initiator-utils
echo "InitiatorName=iqn.2018-06.com.example:desktop0" >> /etc/iscsi/initiatorname.iscsi
systemctl restart iscsid
iscsiadm --mode discoverydb --type sendtargets --portal 192.168.4.10 --discover ----man iscsiadm (/example)
sed -i 's/manual/automatic/' /var/lib/iscsi/nodes/
                                iqn.2018-06.com.example:server0/192.168.4.10\,3260\,1/default
systemctl restart iscsi
lsblk -----发现并连接网络磁盘
vgextend master /dev/sda && vgs
lvextend -L 5.5G /dev/master/mysql && lvs && df -h
resize2fs /dev/master/mysql && df -h -----拓展逻辑卷
mv /var/lib/mysql /mnt/
mkdir /var/lib/mysql
vim /etc/fstab 【/dev/master/mysql /var/lib/mysql ext4 defaults 0 0】
```

```
mount -a
mv /mnt/mysql/* /var/lib/mysql
chown -R mysql:mysql /var/lib/mysql
systemctl start mysqld
```

slave22

```
lsblk
vgcreate master /dev/vdb /dev/vdc
vgs
lvcreate -n mysql -L 3G master
lvs
mkfs.ext4 /dev/master/mysql
mkdir /lvm
echo "/dev/master/mysql /lvm ext4 defaults 0 0" >> /etc/fstab
mount -a
df -h----->DAS
tar -xf mysql-5.7.17.tar
yum -y reinstall mysql*.rpm
vim /etc/my.cnf
server_id=22
systemctl start mysqld
grep password /var/log/mysqld.log
mysql -hlocalhost -uroot -p"jaejwpy/A1)?"
mysql> alter user root@"localhost" identified by "123qqq...A";
mysql> change master to
-> master_host="192.168.4.11",
-> master_user="repl",
-> master_password="123qqq...A",
-> master_log_file="lly.000001",
-> master_log_pos=447;
mysql> start slave;
mysql> show slave status\G;---->mysql 主从
```

maxscale 服务器 77:

```
rpm -ivh maxscale-2.1.2-1.rhel.7.x86_64.rpm
vim /etc/maxscale.cnf
[maxscale] -----在主库授权
threads=auto
```

```
[server1]
type=server
address=192.168.4.11
port=3306
protocol=MySQLBackend
```

```
[server2]
type=server
address=192.168.4.22
port=3306
protocol=MySQLBackend
```

```
[MySQL Monitor]
type=monitor
module=mysqlmon
servers=server1,server2
user=scalemon
passwd=123qqq...A
monitor_interval=10000
```

```
#[Read-Only Service]
```

```
[Read-Write Service]
type=service
router=readwritesplit
servers=server1,server2
user=maxscale
passwd=123qqq...A
max_slave_connections=100%
```

```
#[Read-Only Listener]
```

```
[MaxAdmin Listener]
port=4099          ---->添加一行端口
```

```
maxscale --config=/etc/maxscale.cnf  -----启动
ps -C maxscale netstat -antupl | grep maxscale -----查看进程及端口
maxscale 主机登陆: maxadmin -uadmin -pmariadb -P4099---> MaxScale> list servers
登陆测试主从库的两种授权用户
客户端登陆测试
```

Inmp33

```
yum -y install pcre-devel gcc openssl-devel
./configure --prefix=/usr/local/nginx --user=nginx --group=nginx --with-http_ssl_module
make && make install
cd /etc/sysconfig/network-scripts/
cp ifcfg-lo{, :0} && vim ifcfg-lo{, :0}
vim /etc/sysctl.conf
systemctl restart network
sysctl -p
yum -y install php php-fpm-5.4.16-42.el7.x86_64.rpm
systemctl start php-fpm
cd /usr/local/nginx/
vim conf/nginx.conf
echo "<?php
    phpinfo()
?> " > /usr/local/nginx/html/test.php
./sbin/nginx -s reload          -----php 解析
echo "192.168.4.66:/nfs          /usr/local/nginx/html    nfs    defaults 0 0" >> /etc/fstab
mount -a                        -----将页面存放在第三节点上
yum -y install php-mysql        ----php 的 mysql 扩展包
测试: 连接数据库的动态页面
```

#### Inmp44

```
yum -y install pcre-devel gcc openssl-devel
./configure --prefix=/usr/local/nginx --user=nginx --group=nginx --with-http_ssl_module
make && make install
cd /etc/sysconfig/network-scripts/
cp ifcfg-lo{, :0} && vim ifcfg-lo{, :0}-----> 设置 VIP
vim /etc/sysctl.conf -----防止地址冲突
systemctl restart network
sysctl -p
yum -y install php php-fpm-5.4.16-42.el7.x86_64.rpm
echo "<?php
    phpinfo()
    ?> " > /usr/local/nginx/html/test.php
cd /usr/local/nginx/
vim conf/nginx.conf
./sbin/nginx -s reload
systemctl start php-fpm
echo "192.168.4.66:/nfs      /usr/local/nginx/html      nfs      defaults 0 0" >> /etc/fstab
mount -a
```

#### lvs88

```
yum -y install keepalived ipvsadm
vim /etc/keepalived/keepalived.conf ----->lvs-DR 模式需要服务器设置 VIP 及防止地址冲突
    router_id LVS1
    #vrrp_strict
    state MASTER
    virtual_router_id 51
    priority 100
    auth_pass 1111
    virtual_ipaddress {192.168.4.253}
    virtual_server 192.168.4.253 80 {
        lb_algo rr
        lb_kind DR
        #persistence_timeout 5
        real_server 192.168.4.33 80 {
            TCP_CHECK {connect_timeout 3      nb_get_retry 3      delay_before_retry 3} }
    }
ip a s
iptables -F
ipvsadm -Ln -----> 都有 ipvsadm 规则
```

#### lvs99

```
yum -y install keepalived ipvsadm
vim /etc/keepalived/keepalived.conf
    router_id LVS2
    #vrrp_strict
    state BACKUP
    virtual_router_id 51
```

```

priority 50
auth_pass 1111
virtual_ipaddress {192.168.4.253}
virtual_server 192.168.4.253 80 {
lb_algo rr
lb_kind DR
#persistence_timeout 5
real_server 192.168.4.33 80 {
TCP_CHECK {connect_timeout 3      nb_get_retry 3      delay_before_retry 3} }
ip a s
iptables -F
ipvsadm -Ln

```

```

ca66
openssl genrsa > cert.key
openssl req -new -x509 -key cert.key > cert.pem
mv cert.key cert.pem /var/www/html ----->提供证书下载
mkfs.ext4 /dev/vdb
mount /dev/vdb /nfs/
df -h
yum -y install nfs-utils
echo "/nfs 192.168.4.0/24(rw)" >> /etc/exports
chmod o+w /nfs/
systemctl start nfs-server ----->将磁盘共享出去

```

```

iscsi10
分区: fdisk /dev/vdb d 删除 n 新增 p 查找 q 不保存退出 w 保存退出 刷新分区表: partprobe
yum -y install targetcli
targetcli
/> backstores/block create iscsi_store /dev/vdb
/> iscsi/ create iqn.2018-06.com.example:server0
/> iscsi/iqn.2018-06.com.example:server0/tpg1/acls create iqn.2018-06.com.example:desktop0
/> iscsi/iqn.2018-06.com.example:server0/tpg1/luns create /backstores/block/iscsi_store
/> iscsi/iqn.2018-01.cn.tedu:server1/tpg1/portals/ create 0.0.0.0
systemctl restart target

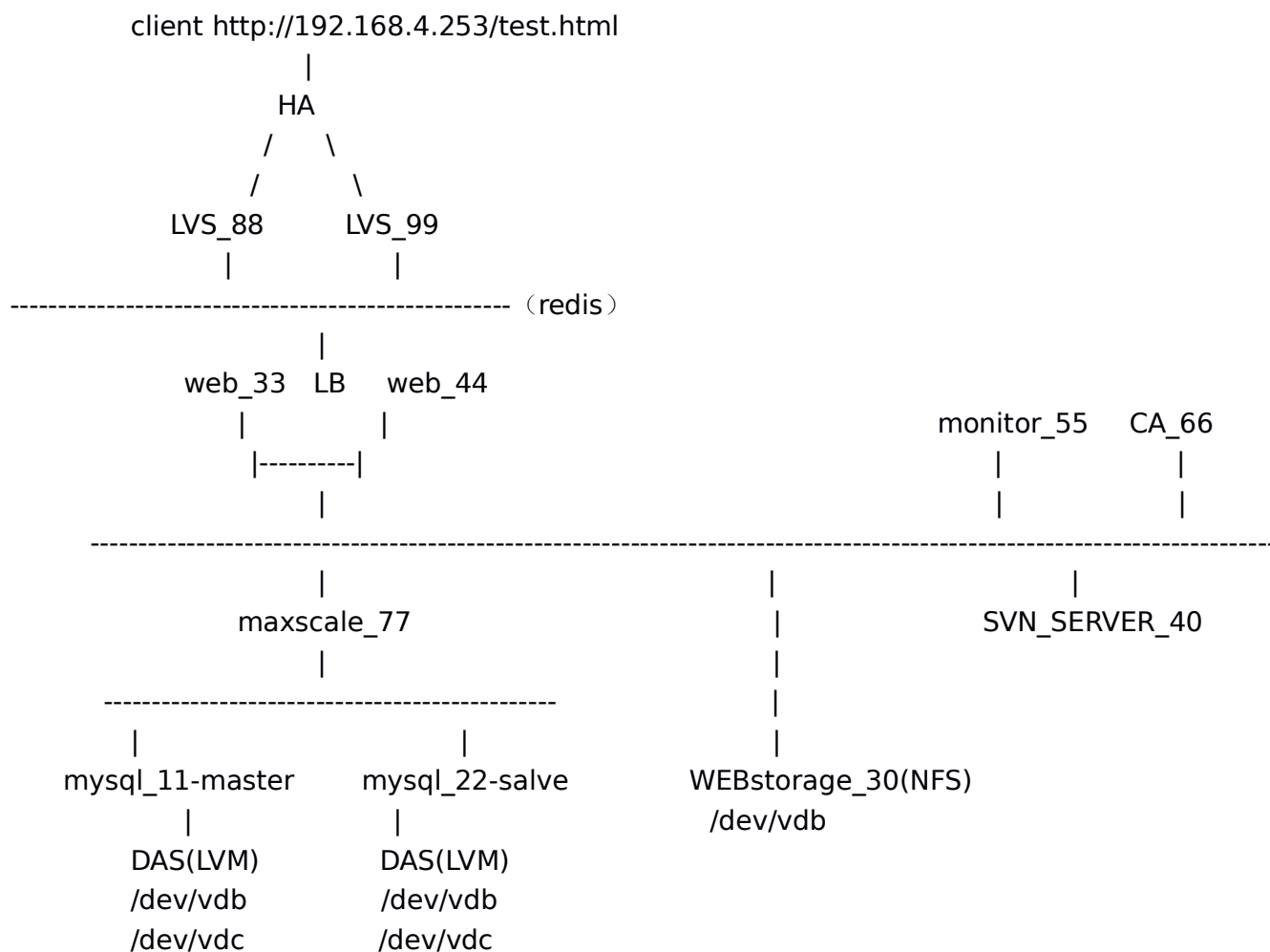
```

```

svn40
yum -y install subversion && rpm -q subversion
mkdir /var/svn
svnadmin create /var/svn/project && ls /var/svn
cd /usr/lib/systemd/system/
svn import . file:///var/svn/project/ -m "Init Data"
vim /var/svn/project/conf/svnserve.conf 【19 20 27 34】 passwd authz
svnserve -d -r /var/svn/project
客户端测试: svn --username harry --password 123456 co svn://192.168.4.10/ code
svn (tab)

```

## 综合项目拓扑



### 一、IP 规划:

MySQL 数据库服务器 192.168.4.11

MySQL 数据库服务器 192.168.4.22

网站服务器(LAMP) 192.168.4.33(RHEL6)

网站服务器(LMAP) 192.168.4.44(RHEL6)

Zabbix 监控服务器 192.168.4.55

CA 服务器 192.168.4.66

mysql 读写分离服务器 192.168.4.77

NAS 存储服务器 192.168.4.30

iscsi 存储服务器 192.168.4.10/20

lvs 服务器 192.168.4.88/99 (RHEL6)

lvs 服务 VIP 192.168.4.253

+++++

### 一、架设网站服务器

1.1 需求: 在主机 33 和 44 上部署 Lamp 网站运行平台

### 二、部署负载均衡集群

2.1 需求: ip 地址是 88 的主机做分发器模式是 LVS/DR, 把自己接收到访问网站的请求平均的分发给主机 33 和 44, LVS 虚拟服务的 vip 地址是 192.168.4.253

### 三、部署分发器的高可用集群

3.1 需求: 把 ip 地址是 99 的主机设置为备用的 LVS/DR 模式的分发器。

#### 四、架设数据库服务器

4.1 需求：在主机 11 和 22 上运行 MySQL 数据库服务器（mysql5.7）

4.2 需求：把数据库服务器 22 配置为 11 主机的从库

#### 五、部署 MySQL 读写分离

5.1 需求：在主机 77 上运行 maxscale 服务。

5.2 需求：把接收到的查询请求分发给主机 22，接收到的写请求分发给主机 22 。

#### 六、配置网站共享存储服务器（NAS）

6.1 需求：主机 30 把自己本机的磁盘/dev/vdb 使用 NFS 服务共享给前端的 2 台网站服务器 33 和 44，存储网站的网页文件。

#### 七、配置数据库共享存储服务器（SAN）

7.1 主机 10 把自己本机磁盘/dev/sdb 共享前端数据库服务器 11

主机 20 把自己本机磁盘/dev/sdb 共享前端数据库服务器 22

#### 八、搭建监控服务器

需求：

8.1 在主机 55 上运作 zabbix\_server 服务。

8.2 在所有数据库服务器和网站服务器主机上运行 zabbix\_agent 服务。

8.2 监控所有数据库服务器和网站服务器主机上的如下资源：

主机状态、系统运行情况、系统资源使用情况、应用的运行状态、设置使用本机的邮件服务发送报警消息给 admin@localhost 邮箱账号

#### 九、搭建 CA 证书服务器

9.1 部署 60 主机可以给网站服务器签发数字证书，ca 服务器根证书的文件名叫 my-cao.crt ,并通过本机的网站服务共享给客户端主机。

#### 十、部署网站服务数据加密

10.1 配置网站服务器 33 和 44 把自己接收到访问 80 端口请求转给自己本机的 443 端口