mysql实验手册

1 部署数据库服务

2 数据导入

3 数据备份

4 部署mysql主从同步

5 部署读写分离

6 部署mysql高可用集群

7 配置mysql多实例

8 用户授权

9 安装图形管理工具

client http://192.168.4.253/test.html

|

HA

/ \

/ \

LVS\_88 LVS\_99

| |

----------------------------

|

web\_33 LB web\_44

| | monitor\_55 CA\_66

|----------| | |

| | |

-----------------------------------------------------

| | |

maxscale\_77 | SVN\_SERVER\_40

| |

-------------------------- |

| | |

mysql\_11-master mysql\_22-salve WEBstorage\_30(NFS)

| | /dev/vdb

DAS(LV) DAS(LV)

/dev/vdb /dev/vdb

/dev/vdc /dev/vdc

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。11

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-caa.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

1. **构建lamp，mysql**

**安装软件包**

停止mariadb服务

1. **[**root@localhost **~]**# systemctl stop mariadb

）删除/etc/my.cnf配置文件

此配置文件由RHEL自带的mariadb-libs库提供：

[root@localhost ~]# rm -rf /etc/my.cnf

3）删除数据

**[**root@localhost **~]**# rm **-**rf **/var**/lib/mysql/\*

4）卸载软件包（没有会显示未安装软件包）

1. **[**root@localhost **~]**# rpm **-**e **--**nodeps mariadb**-**server mariadb
2. 警告：/var/log**/**mariadb**/**mariadb**.**log 已另存为/var/log**/**mariadb**/**mariadb**.**log**.**rpmsave

**安装软件包**

1）安装mysql时可能会缺少某些依赖包，需提前单独安装

**[root@localhost ~]# yum -y install perl-Data-Dumper perl-JSON perl-Time-HiRes**

虚拟机192.168.4.1解压mysql-5.7.17.tar 整合包

**启动MySQL数据库服务并设置开机自启**

1. **[**root@localhost **~]**# systemctl start mysqld //启动mysql服务
2. **[**root@localhost **~]**# systemctl enable mysqld //设置开机自启
3. **[**root@localhost **~]**# systemctl status mysqld //查看mysql服务状态

**连接MySQL服务器，修改密码**

1. grep 'temporary password' /var/log**/**mysqld**.**log
2. **2017-04-**01T18**:10:42.**948679Z **1** **[**Note**]** A temporary password is generated **for** root@localhost**:** mtoa**>**Av**<**p6Yk //随机生成的管理密码为mtoa>Av<p6Yk

php和php-fpm(该软件包在lnmp\_soft中提供)

1. **[**root@proxy **~]**# yum **-**y install php php**-**mysql
2. **[**root@proxy **~]**# yum **-**y install php**-**fpm**-5.4.16-42.**el7**.**x86\_64**.**rpm

通过修改Nginx及php-fpm配置文件实现对PHP页面的支持。

注意，FastCGI的内存消耗问题，一个PHP-FPM解释器将消耗约25M的内存。

1. **keepalived+LVS**

使用Keepalived为LVS调度器提供高可用功能，防止调度器单点故障，为用户提供Web服务：

* LVS1调度器真实IP地址为192.168.4.5
* LVS2调度器真实IP地址为192.168.4.6
* 服务器VIP地址设置为192.168.4.15
* 真实Web服务器地址分别为192.168.4.100、192.168.4.200
* 使用加权轮询调度算法，真实web服务器权重不同

这里的子网掩码必须是32（也就是全255），网络地址与IP地址一样，广播地址与IP地址也一样

1. **[**root@web1 **~]**# cd **/**etc**/**sysconfig**/**network**-**scripts**/**
2. **[**root@web1 **~]**# cp ifcfg**-**lo**{,:0}**
3. **[**root@web1 **~]**# vim ifcfg**-**lo**:0**
4. DEVICE**=**lo**:0**
5. IPADDR**=192.168.4.15**
6. NETMASK**=255.255.255.255**
7. NETWORK**=192.168.4.15**
8. BROADCAST**=192.168.4.15**
9. ONBOOT**=**yes
10. NAME**=**lo**:0**

这里因为web1也配置与调度器一样的VIP地址，默认肯定会出现地址冲突。

写入这四行的主要目的就是访问192.168.4.15的数据包，只有调度器会响应，其他主机都不做任何响应。

1. **[**root@web1 **~]**# vim **/**etc**/**sysctl**.**conf
2. #手动写入如下**4**行内容
3. net**.**ipv4**.**conf**.**all**.**arp\_ignore **=** **1**
4. net**.**ipv4**.**conf**.**lo**.**arp\_ignore **=** **1**
5. net**.**ipv4**.**conf**.**lo**.**arp\_announce **=** **2**
6. net**.**ipv4**.**conf**.**all**.**arp\_announce **=** **2**
7. #当有arp广播问谁是**192.168.4.15**时，本机忽略该ARP广播，不做任何回应
8. #本机不要向外宣告自己的lo回环地址是**192.168.4.15**

重启网络服务，设置防火墙与SELinux

1. **[**root@web1 **~]**# systemctl restart network
2. **[**root@web1 **~]**# ifconfig
3. **[**root@web1 **~]**# systemctl stop firewalld
4. **[**root@web1 **~]**# setenforce **0**

* Web2 同上

**调度器安装Keepalived与ipvsadm软件**

***注意：两台LVS调度器执行相同的操作（如何已经安装软件，可用忽略此步骤）。***

安装软件

1. **[**root@proxy1 **~]**# yum install **-**y keepalived
2. **[**root@proxy1 **~]**# systemctl enable keepalived
3. **[**root@proxy1 **~]**# yum install **-**y ipvsadm
4. **[**root@proxy1 **~]**# ipvsadm **-**C
5. **[**root@proxy2 **~]**# yum install **-**y keepalived
6. **[**root@proxy2 **~]**# systemctl enable keepalived
7. **[**root@proxy2 **~]**# yum install **-**y ipvsadm
8. **[**root@proxy2 **~]**# ipvsadm **-**C

**部署Keepalived实现LVS-DR模式调度器的高可用**

1）LVS1调度器设置Keepalived，并启动服务

1. **[**root@proxy1 **~]**# vim **/**etc**/**keepalived**/**keepalived**.**conf
2. global\_defs **{**
3. notification\_email **{**
4. admin@tarena**.**com**.**cn                //设置报警收件人邮箱
5. **}**
6. notification\_email\_from ka@localhost    //设置发件人
7. smtp\_server **127.0.0.1**                //定义邮件服务器
8. smtp\_connect\_timeout **30**
9. router\_id lvs1                        //设置路由ID号(实验需要修改)
10. **}**
11. vrrp\_instance VI\_1 **{**
12. state MASTER                          //主服务器为MASTER（实验需要修改）
13. **interface** eth0                        //定义网络接口（实验需要修改）
14. virtual\_router\_id **50**                    //主辅VRID号必须一致（实验需要修改）
15. priority **100**                     //服务器优先级（实验需要修改）
16. advert\_int **1**
17. authentication **{**
18. auth\_type pass
19. auth\_pass forlvs                      //主辅服务器密码必须一致（实验需要修改）
20. **}**
21. virtual\_ipaddress **{** **192.168.4.15** **}** //配置VIP（实验需要修改）
22. **}**
23. virtual\_server **192.168.4.15** **80** **{**      //设置ipvsadm的VIP规则（实验需要修改）
24. delay\_loop **6**
25. lb\_algo wrr                 //设置LVS调度算法为WRR
26. lb\_kind DR                          //设置LVS的模式为DR
27. #persistence\_timeout **50**
28. #注意这样的作用是保持连接，开启后，客户端在一定时间内始终访问相同服务器
29. protocol TCP
30. real\_server **192.168.4.100** **80** **{** //设置后端web服务器真实IP（实验需要修改）
31. weight **1**                  //设置权重为1
32. TCP\_CHECK **{**
33. connect\_timeout **3**
34. nb\_get\_retry **3**
35. delay\_before\_retry **3**
36. **}**
37. **}**
38. real\_server **192.168.4.200** **80** **{** //设置后端web服务器真实IP（实验需要修改）
39. weight **2**                  //设置权重为2
40. TCP\_CHECK **{**
41. connect\_timeout **3**
42. nb\_get\_retry **3**
43. delay\_before\_retry **3**
44. **}**
45. **}**
46. **}**
47. **[**root@proxy1 **~]**# systemctl start keepalived
48. **[**root@proxy1 **~]**# ipvsadm **-**Ln #查看LVS规则
49. **[**root@proxy1 **~]**# ip a s #查看VIP配置

LVS2调度器设置Keepalived

1. **[**root@proxy2 **~]**# vim **/**etc**/**keepalived**/**keepalived**.**conf
2. global\_defs **{**
3. notification\_email **{**
4. admin@tarena**.**com**.**cn                //设置报警收件人邮箱
5. **}**
6. notification\_email\_from ka@localhost    //设置发件人
7. smtp\_server **127.0.0.1**                //定义邮件服务器
8. smtp\_connect\_timeout **30**
9. router\_id lvs2                        //设置路由ID号（实验需要修改）
10. **}**
11. vrrp\_instance VI\_1 **{**
12. state BACKUP                          //从服务器为BACKUP（实验需要修改）
13. **interface** eth0                        //定义网络接口（实验需要修改）
14. virtual\_router\_id **50**                    //主辅VRID号必须一致（实验需要修改）
15. priority **50**                      //服务器优先级（实验需要修改）
16. advert\_int **1**
17. authentication **{**
18. auth\_type pass
19. auth\_pass forlvs                      //主辅服务器密码必须一致（实验需要修改）
20. **}**
21. virtual\_ipaddress **{** **192.168.4.15** **}** //设置VIP（实验需要修改）
22. **}**
23. virtual\_server **192.168.4.15** **80** **{**      //自动设置LVS规则（实验需要修改）
24. delay\_loop **6**
25. lb\_algo wrr                 //设置LVS调度算法为WRR
26. lb\_kind DR                          //设置LVS的模式为DR
27. # persistence\_timeout **50**
28. #注意这样的作用是保持连接，开启后，客户端在一定时间内始终访问相同服务器
29. protocol TCP
30. real\_server **192.168.4.100** **80** **{** //设置后端web服务器的真实IP（实验需要修改）
31. weight **1**                  //设置权重为1
32. TCP\_CHECK **{**
33. connect\_timeout **3**
34. nb\_get\_retry **3**
35. delay\_before\_retry **3**
36. **}**
37. **}**
38. real\_server **192.168.4.200** **80** **{** //设置后端web服务器的真实IP（实验需要修改）
39. weight **2**                  //设置权重为2
40. TCP\_CHECK **{**
41. connect\_timeout **3**
42. nb\_get\_retry **3**
43. delay\_before\_retry **3**
44. **}**
45. **}**
46. **[**root@proxy2 **~]**# systemctl start keepalived
47. **[**root@proxy2 **~]**# ipvsadm **-**Ln #查看LVS规则
48. **[**root@proxy2 **~]**# ip a s #查看VIP设置
49. **构建数据库集群，实现读写分离**

**步骤一：搭建主从**

1）搭建一主一从结构，主库192.168.4.10上面操作

1. **[**root@master10 **~]**# vim **/**etc**/**my**.**cnf
2. **[**mysqld**]**
3. server\_id**=10**    //指定服务器ID号
4. log**-**bin**=**master10        //启用binlog日志，并指定文件名前缀
5. **...**
6. **[**root@master10 **~]**# systemctl restart mysqld        //重启mysqld

从库192.168.4.20上面操作

1. **[**mysqld**]**
2. server\_id**=20**            //指定服务器ID号，不要与Master的相同
3. log**-**bin**=**slave20        //启动SQL日志，并指定文件名前缀
4. read\_only**=1**

主库授权一个用户并查看master的状态

1. **[**root@master10 **~]**# mysql **-**u root **-**p123456
2. mysql**>** grant all on **\*.\*** to 'replicater'@'%' identified by '123456'**;**
3. Query OK**,** **0** rows affected**,** **1** warning **(0.00** sec**)**

从库通过CHANGE MASTER语句指定MASTER服务器的IP地址、同步用户名/密码、起始日志文件、偏移位置（参考MASTER上的状态输出）

1. **[**root@slave20 **~]**# mysql **-**u root **-**p123456
2. mysql**>** change master to master\_host**=**'192.168.4.10'**,**
3. **->** master\_user**=**'replicater'**,**
4. **->** master\_password**=**'123456'**,**
5. **->** master\_log\_file**=**'master10.000002'**,**
6. **->** master\_log\_pos**=738;**

主从同步建完

**步骤二：实现mysql读写分离**

配置数据读写分离服务器192.168.4.100

环境准备关闭防火墙和SElinux，保证yum源可以正常使用

1. **[**root@maxscale mysql**]**# rpm **-**ivh maxscale**-2.1.2-1.**rhel**.7.**x86\_64**.**rpm
2. //安装maxscale

配置maxscale

1. **[**root@maxscale mysql**]**# vim **/**etc**/**maxscale**.**cnf**.**template
2. **[**maxscale**]**
3. threads**=**auto            //运行的线程的数量
4. **[**server1**]**            //定义数据库服务器
5. type**=**server
6. address**=192.168.4.10**        //数据库服务器的ip
7. port**=3306**
8. protocol**=**MySQLBackend        //后端数据库
9. **[**server2**]**
10. type**=**server
11. address**=192.168.4.20**
12. port**=3306**
13. protocol**=**MySQLBackend
14. **[**MySQL Monitor**]**                //定义监控的数据库服务器
15. type**=**monitor
16. module**=**mysqlmon
17. servers**=**server1**,** server2        //监控的数据库列表，不能写ip
18. user**=**scalemon                    //监视数据库服务器时连接的用户名scalemon
19. passwd**=123456**                    //密码123456
20. monitor\_interval**=10000**        //监视的频率 单位为秒
21. #**[**Read**-**Only Service**]**        //不定义只读服务器
22. #type**=**service
23. #router**=**readconnroute
24. #servers**=**server1
25. #user**=**myuser
26. #passwd**=**mypwd
27. #router\_options**=**slave
28. **[**Read**-**Write Service**]**            //定义读写分离服务
29. type**=**service
30. router**=**readwritesplit
31. servers**=**server1**,** server2
32. user**=**maxscaled            //用户名 验证连接代理服务时访问数据库服务器的用户是否存在
33. passwd**=123456**                //密码
34. max\_slave\_connections**=100%**
35. **[**MaxAdmin Service**]**        //定义管理服务
36. type**=**service
37. router**=**cli
38. #**[**Read**-**Only Listener**]**        //不定义只读服务使用的端口号
39. #type**=**listener
40. #service**=**Read**-**Only Service
41. #protocol**=**MySQLClient
42. #port**=4008**
43. **[**Read**-**Write Listener**]**            //定义读写服务使用的端口号
44. type**=**listener
45. service**=**Read**-**Write Service
46. protocol**=**MySQLClient
47. port**=4006**
48. **[**MaxAdmin Listener**]**        //管理服务使用的端口号
49. type**=**listener
50. service**=**MaxAdmin Service
51. protocol**=**maxscaled
52. socket**=default**
53. port**=4099**     //手动添加，不指定时使用的是默认端口在启动服务以后可以知道默认端口是多少

根据配置文件的设置，在数据库服务器上添加授权用户（主库执行，从库查看）

1. mysql**>** grant replication slave**,**replication client on **\*.\*** to scalemon@'%' identified by "123456"**;**            //监控数据库服务器时，连接数据库服务器的用户
2. Query OK**,** **0** rows affected**,** **1** warning **(0.00** sec**)**
3. mysql**>** grant select on mysql**.\*** to maxscaled@"%" identified by "123456"**;**
4. //验证 访问数据时，连接数据库服务器使用的用户，是否在数据库服务器上存在的，连接用户

测试授权用户

1. **[**root@maxscale mysql**]**# mysql **-**h **192.168.4.10** **-**u scalemon **-**p123456
2. **[**root@maxscale mysql**]**# mysql **-**h **192.168.4.20** **-**u scalemon **-**p123456
3. **[**root@maxscale mysql**]**# mysql **-**h **192.168.4.10** **-**u maxscaled **-**p123456
4. **[**root@maxscale mysql**]**# mysql **-**h **192.168.4.20** **-**u maxscaled **-**p123456

启动服务

1. **[**root@maxscale **~]**# maxscale **-**f **/**etc**/**maxscale**.**cnf
2. **[**root@maxscale **~]**# ps **-**C maxscale        //查看进程
3. PID TTY TIME CMD
4. **17930** **?** **00:00:00** maxscale
5. **[**root@maxscale **~]**# netstat **-**antup **|** grep maxscale //查看端口
6. tcp6 **0** **0** **:::4099** **:::\*** LISTEN **17930/**maxscale
7. tcp6 **0** **0** **:::4006** **:::\*** LISTEN **17930/**maxscale

测试，在本机访问管理端口查看监控状态

maxadmin -P端口 -u用户名 -p密码

1. **[**root@maxscale **~]**# maxadmin **-**P4099 **-**uadmin **-**pmariadb
2. MaxScale**>**
3. MaxScale**>** list servers
4. Servers**.**
5. **-------------------+-----------------+-------+-------------+--------------------**
6. Server **|** Address **|** Port **|** Connections **|** Status
7. **-------------------+-----------------+-------+-------------+--------------------**
8. server1 **|** **192.168.4.10** **|** **3306** **|** **0** **|** Master**,** Running
9. server2 **|** **192.168.4.20** **|** **3306** **|** **0** **|** Slave**,** Running
10. **-------------------+-----------------+-------+-------------+----------**

7）在客户端访问读写分离服务器（没有mysql命令可以安装）

mysql -h读写分离服务ip -P4006 -u用户名 -p密码

1. **[**root@slave53 **~]**# mysql **-**h192**.168.4.100** **-**P4006 **-**ureplicater **-**p123456
2. mysql**>** select @@hostname**;**            //查看当前主机名
3. **+------------+**
4. **|** @@hostname **|**
5. **+------------+**
6. **|** slave20 **|**
7. **+------------+**
8. **1** row **in** set **(0.00** sec**)**
9. mysql**>** create table t2**(**id int**(4)** **);**
10. Query OK**,** **0** rows affected **(0.02** sec**)**
11. mysql**>** insert into aa**.**t2 values**(777);**
12. Query OK**,** **1** row affected **(0.01** sec**)**

在主库上面查看

1. mysql**>** use aa
2. mysql**>** select **\*** from t2**;**
3. **+------+**
4. **|** id **|**
5. **+------+**
6. **|** **777** **|**
7. **+------+**
8. **1** row **in** set **(0.00** sec**)**

从库（主库同步到从库）

1. mysql**>** use aa
2. mysql**>** select **\*** from t2**;**
3. **+------+**
4. **|** id **|**
5. **+------+**
6. **|** **777** **|**
7. **+------+**
8. **1** row **in** set **(0.00** sec**)**
9. **构建nfs**

**步骤一：配置NFS服务器，发布指定的共享**

1）确认服务端程序、准备共享目录

软件包nfs-utils用来提供NFS共享服务及相关工具，而软件包rpcbind用来提供RPC协议的支持，这两个包在RHEL7系统中一般都是默认安装的：

1. root@proxy **~]**# rpm **-**q nfs**-**utils rpcbind
2. nfs**-**utils**-1.3.0-0.48.**el7**.**x86\_64
3. rpcbind**-0.2.0-42.**el7**.**x86\_64

根据本例的要求，需要作为NFS共享发布的有/root、/usr/src这两个目录：

1. **[**root@proxy **~]**# ls **-**ld **/**root **/**usr**/**src**/**
2. dr**-**xr**-**x**---.** **35** root root **4096** **1**月 **15** **18:52** **/**root
3. drwxrwxr**-**x**+** **4** root root **4096** **1**月 **15** **17:35** **/**usr**/**src**/**

2）修改/etc/exports文件，添加共享目录设置

默认情况下，来自NFS客户端的root用户会被自动降权为普通用户，若要保留其root权限，注意应添加no\_root\_squash控制参数(没有该参数，默认root会被自动降级为普通账户)；另外，限制只读的参数为ro、可读可写为rw，相关配置操作如下所示：

1. **[**root@proxy **~]**# vim **/**etc**/**exports
2. **/**root **192.168.2.100(**rw**,**no\_root\_squash**)**
3. /usr/src **192.168.2.0/24(**ro**)**

3）启动NFS共享相关服务，确认共享列表

依次启动rpcbiind、nfs服务：

1. **[**root@proxy **~]**# systemctl restart rpcbind **;** systemctl enable rpcbind
2. **[**root@proxy **~]**# systemctl restart nfs **;** systemctl enable nfs

使用showmount命令查看本机发布的NFS共享列表：

1. **[**root@proxy **~]**# showmount **-**e localhost
2. Export list **for** localhost**:**
3. /usr/src **192.168.2.0/24**
4. **/**root **192.168.2.100**

**步骤二：从客户机访问NFS共享**

1）启用NFS共享支持服务

客户机访问NFS共享也需要rpcbind服务的支持，需确保此服务已开启：

1. **[**root@web1 **~]**# systemctl restart rpcbind **;** systemctl enable rpcbind

2）查看服务器提供的NFS共享列表

1. **[**root@web1 **~]**# showmount **-**e **192.168.2.5**
2. Export list **for** **192.168.2.5:**
3. /usr/src **192.168.2.0/24**
4. **/**root **192.168.2.100**

3）从客户机192.168.2.100访问两个NFS共享，并验证权限

将远程的NFS共享/root挂载到本地的/root5文件夹，并验证可读可写：

1. **[**root@web1 **~]**# mkdir **/**root5                          //建立挂载点
2. **[**root@web1 **~]**# mount **192.168.2.5:**/root /root5         //挂载NFS共享目录
3. **[**root@web1 **~]**# df **-**hT **/**root5                         //确认挂载结果
4. Filesystem Type Size Used Avail Use**%** Mounted on
5. **192.168.2.5:**/root nfs 50G 15G 33G 31% /root5
6. **[**root@web1 **~]**# cd **/**root5                             //切换到挂载点
7. **[**root@web1 root5**]**# echo "NFS Write Test" **>** test**.**txt     //测试写入文件
8. **[**root@web1 root5**]**# cat test**.**txt                      //测试查看文件
9. NFS Write Test

将远程的NFS共享/usr/src挂载到本地的/mnt/nfsdir，并验证只读：

1. **[**root@web1 **~]**# mkdir **/**mnt**/**nfsdir                     //建立挂载点
2. **[**root@web1 **~]**# mount **192.168.2.5:**/usr/src **/**mnt**/**nfsdir**/**     //挂载NFS共享目录
3. **[**root@web1 **~]**# df **-**hT **/**mnt**/**nfsdir**/**                         //确认挂载结果
4. Filesystem Type Size Used Avail Use**%** Mounted on
5. **192.168.2.5:**/usr/src nfs 50G 15G 33G **31%** /mnt/nfsdir
6. **[**root@web1 **~]**# cd **/**mnt**/**nfsdir**/**                         //切换到挂载点
7. **[**root@web1 nfsdir**]**# ls                                 //读取目录列表
8. debug install**.**log kernels test**.**txt
9. **[**root@web1 nfsdir**]**# echo "Write Test." **>** pc**.**txt //尝试写入文件失败
10. **-**bash**:** pc**.**txt**:** 只读文件系统

***如果从未授权的客户机访问NFS共享，将会被拒绝。比如从NFS服务器本机尝试访问自己发布的/root共享（只允许192.168.2.100访问），结果如下所示：***

1. **[**root@proxy **~]**# mkdir **/**root5
2. **[**root@proxy **~]**# mount **192.168.2.5:**/root /root5
3. mount**.**nfs**:** access denied by server **while** mounting **192.168.2.5:/**root

设置永久挂载

1. **[**root@web1 **~]**# vim **/**etc**/**fstab
2. **..** **..**
3. **192.168.2.5:**/usr/src nfsdir         nfs **default,**ro **0** **0**
4. **192.168.2.5:/**root root5         nfs **default** **0** **0**