**老男孩linux运维实战培训**

**老男孩教育教学核心思想6重：重目标、重思路、重方法、重实践、重习惯、重总结**

**学无止境，老男孩教育成就你人生的起点！**

**版权声明：**

本文作者为《老男孩linux运维实战培训》学生—**程军**

本文的所有内容均来自老男孩培训**命令总结**，未经本人及老男孩培训许可，禁止私自转发及使用。

QQ: **你的**XXXX

E-mail: **你的**XXXX

**联系方式:**

|  |  |  |  |
| --- | --- | --- | --- |
| 网站运维交流群： | | | |
| **Linux QQ交流群 385168604** | | **架构师 QQ交流群 390642196** | |
| **Python QQ交流群 29215534** | | **大数据 QQ交流群 421358633** | |
| =================================================================================== | | | |
| **老男孩linux实训联系方式** |  | |  |
| **咨询QQ** | **41117397（丹丹）** | | **70271111（歪歪）** |
|  | **80042789（飞雪）** | | **390320151（小雨）** |
|  | **41117483（冰冰）** | |  |
| **电话：** | **158-1059-0206（丹丹）** | | **189-1171-8229（歪歪）** |
|  | **135-5261-2571（飞雪）** | | **186-0046-2391（小雨）** |
| **网站:** | <http://www.etiantian.org> | | <http://www.oldboyedu.com> |
| **博客:** | http://oldboy.blog.51cto.com | | <http://blog.oldboyedu.com> |

**快捷方式说明:**

**ctrl + 1 一级标题**

**ctrl + 2 二级标题**

**ctrl + 3 三级标题**

**ctrl + 5 程序代码**

**ctrl + 6 正文**

**格式约定：**

蓝色字体：内容注释

目 录

[集群之NFS课程总结 1](#_Toc510338399)

[第1章 NFS共享存储 1](#_Toc510338400)

[1.1 NFS 介绍 1](#_Toc510338401)

[1.1.1 NFS企业应用场景 1](#_Toc510338402)

[1.1.2 NFS原理 2](#_Toc510338403)

[第2章 NFS环境搭建 5](#_Toc510338404)

[第3章 NFS SERVER 端的设置 6](#_Toc510338405)

[3.1 NFS 软件列表 6](#_Toc510338406)

[3.1.1 检查NFS软件包 6](#_Toc510338407)

[3.1.2 安装nfs及rpc程序 7](#_Toc510338408)

[3.1.3 NFS服务端配置语法及配置实战 9](#_Toc510338409)

[3.2 小结：NFS配置过程 13](#_Toc510338410)

[第4章 NFS客户端挂载 15](#_Toc510338411)

[4.1 NFS客户端挂载优化 16](#_Toc510338412)

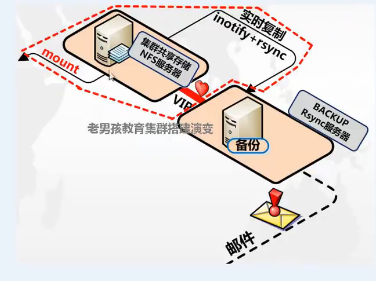
[4.2 NFS企业级内核优化及总结 16](#_Toc510338413)

[4.3 企业生产场景NFS共享存储化小结 17](#_Toc510338414)

[4.4 NFS系统应用的优缺点 17](#_Toc510338415)

集群之NFS课程总结

# NFS共享存储



## NFS 介绍

NFS是Network File System 缩写，网络文件系统，主要功能是通过网络让不同的主机系统之间可以共享文件或目录。NFS客户端可以通过挂在方式将NFS服务器端共享的数据目录挂载到NFS客户端本地系统中。从客户端看NFS服务器端共享的目录就像是客户端自己的磁盘分区或目录一样。

NFS是第一个构建于IP协议之上的网络文件系统。

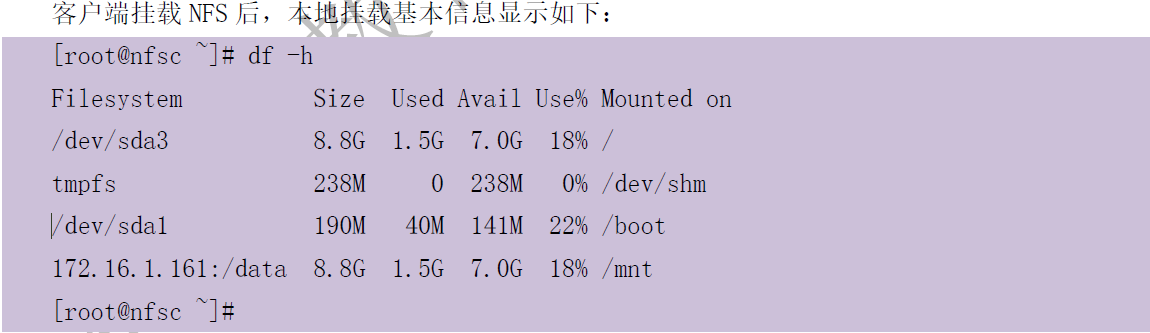
### NFS企业应用场景

存储共享视频、图片、附件等静态资源文件，通常网站用户上传的文件都会放到NFS共享中；一般存放网站用户上传的图片、附件、头像等 注意网站的程序代码不要放在NFS中。

大网站高并发选用Moosefs、GlusterFS、FastDFS

### NFS原理





从挂载信息来看，和本地的磁盘分区几乎没有差别，NFS文件系统对应开头是以IP地址形式的

NFS是通过网络涉及到一些端口-变化

是通过RPC协议服务来实现





NFS工作流程图





1、启动RPC服务

2、启动NFS服务，向RPC注册

3、客户端向RPC申请资源端口

4、客户端使用申请到的资源访问NFS服务器

[1]先启动RPC再启动NFS，NFS向RPC注册

[2]客户端NFS服务访问客户端RPC服务，客户端RPC和服务器PRC打交道，并返回都端口数据

[3]客户端使用RPC向服务端RPC申请的端口访问NFS服务端，NFS服务端返回客户端请求

# NFS环境搭建

***服务器规划表***

|  |  |  |
| --- | --- | --- |
| 服务器说明 | 数量 | 描述 |
| 负载均衡器 | 两台 | 对访问网站的流量进行分流，减少流量对莫台服务器的压力 |
| Web服务器 | 两台 | 处理用户页面访问请求ngixn\apache（） |
| NFS存储 | 一台 | 存储图片、附件、头像等静态数据 兼批量分发和管理 |
| 备份服务器（rsync） | 一台 | 对全网服务器数据进行实时与定时备份 |
| 数据库服务器（Mysql） | 一台 | 对动态变化数据进行存储 |
| 管理服务器 | 一台 | 1. 作为yum仓库、提供全网服务器的软件下载 2. 跳板机、操作审计 3. Vpn(pptp) 4. 监控（nagios、zabbix） |

***主机规划表***

|  |  |  |  |
| --- | --- | --- | --- |
| **服务器名称** | **外网IP（NAT）** | **内网IP（NAT）** | **主机名规划** |
| **A1-nginx负载均衡器01** | **10.0.0.5/24** | **172.16.1.5/24** | **lb01** |
| **A1-nginx负载均衡器01** | **10.0.0.6/24** | **172.16.1.6/24** | **lb02** |
| **B1-apache web 服务器** | **10.0.0.7** | **172.16.1.7/24** | **web02** |
| **B2-nginx web 服务器** | **10.0.0.8** | **172.16.1.8/24** | **web01** |
| **C3-mysql 数据库服务器** | **10.0.0.51** | **172.16.1.51/24** | **db01** |
| **C1-NFS 存储服务器** | **10.0.0.31** | **172.16.1.31/24** | **nfs01** |
| **C2-rsync 存储服务器** | **10.0.0.41** | **172.16.1.41/24** | **backup** |
| **X-管理服务器** | **10.0.0.61/24** | **172.16.1.61/24** | **m01** |

提示：

1. 与老师保持高度一致
2. 灰色IP是临时使用，企业场景中可以没有
3. 负载均衡器的VIP10.0.0.3/24
4. 带外网IP的服务器的内网IP不配网关和DNS

# NFS SERVER 端的设置

## NFS 软件列表

Nfs-utils NFS服务主程序，包括rpc.nfsd、rpc.mountd这两个daemons和相关文件说明，及执行命令文件

Rpcbind RPC主程序。NFS可以视为RPC程序，在启动如何一个RPC程序之前，需要做好端口和g功能的对应映射工作，由rpcbind完成。因此在提供NFS服务前必须先启动rpcbind服务。

### 检查NFS软件包

安装基础软件：yum install -y tree dos2unix nc nmap nfs-utils rpcbind

[root@nfs01 ~]# rpm -qa nfs-utils rpcbind

配置YUM源

[root@nfs01 ~]# vim /etc/yum.conf

[main]

cachedir=/var/cache/yum/$basearch/$releasever

keepcache=1 🡨修改成1，yum完后自动保存

debuglevel=2

logfile=/var/log/yum.log

exactarch=1

obsoletes=1

gpgcheck=1

plugins=1

installonly\_limit=5

bugtracker\_url=http://bugs.centos.org/set\_project.php?project\_id=19&ref=http://bugs.centos.org/bug\_report\_page.php?category=yum

distroverpkg=centos-release

配置wage下载源：

yum install -y tree dos2unix nc nmap nfs-utils rpcbind

查看已经下载保持的包

[root@backup ~]# tree /var/cache/yum/x86\_64/6/|grep "rpm"

│   │   ├── keyutils-1.4-5.el6.x86\_64.rpm

│   │   ├── libevent-1.4.13-4.el6.x86\_64.rpm

│   │   ├── libgssglue-0.1-11.el6.x86\_64.rpm

│   │   ├── nfs-utils-lib-1.1.5-13.el6.x86\_64.rpm

│   │   └── python-argparse-1.2.1-2.1.el6.noarch.rpm

│   ├── libtirpc-0.2.1-13.el6\_9.x86\_64.rpm

│   ├── nfs-utils-1.2.3-75.el6\_9.x86\_64.rpm

│   └── rpcbind-0.2.0-13.el6\_9.1.x86\_64.rpm

**备份已经下载的包**

[root@nfs01 backup]# tar -zcvf /backup/nfs-rpm.tar.gz /var/cache/yum/x86\_64/6/base/packages/

### 安装nfs及rpc程序

yum install nfs-utils rpcbind -y

#### 启动RPC及检查

[root@nfs01 ~]# LING=en

[root@nfs01 ~]# /etc/init.d/rpcbind start

正在启动 rpcbind： [确定]

[root@nfs01 ~]# netstat -lntup|grep rpc

tcp 0 0 0.0.0.0:111 0.0.0.0:\* LISTEN 1889/rpcbind

tcp 0 0 :::111 :::\* LISTEN 1889/rpcbind

udp 0 0 0.0.0.0:792 0.0.0.0:\* 1889/rpcbind

udp 0 0 0.0.0.0:111 0.0.0.0:\* 1889/rpcbind

udp 0 0 :::792 :::\* 1889/rpcbind

udp 0 0 :::111 :::\*

检查NFS资源（房源）

[root@nfs01 ~]# rpcinfo -p localhost

program vers proto port service

100000 4 tcp 111 portmapper

100000 3 tcp 111 portmapper

100000 2 tcp 111 portmapper

100000 4 udp 111 portmapper

100000 3 udp 111 portmapper

100000 2 udp 111 portmapper

#### 启动NFS及检查

[root@nfs01 ~]# /etc/init.d/nfs start

启动 NFS 服务： [确定]

关掉 NFS 配额： [确定]

启动 NFS mountd： [确定]

启动 NFS 守护进程： [确定]

正在启动 RPC idmapd： [确定]

[root@nfs01 ~]# rpcinfo -p localhost

program vers proto port service

100011 1 udp 875 rquotad

100011 2 udp 875 rquotad

100011 1 tcp 875 rquotad

100011 2 tcp 875 rquotad

100005 1 udp 54233 mountd

100005 1 tcp 58592 mountd

100005 2 udp 51578 mountd

100005 2 tcp 47480 mountd

100005 3 udp 32845 mountd

100005 3 tcp 39288 mountd

#### 检查NFS用户

[root@nfs01 ~]# id nfsnobody

uid=65534(nfsnobody) gid=65534(nfsnobody) 组=65534(nfsnobody)

注意一定要先启动RPC服务再启动NFS服务

#### 设置开机启动

[root@nfs01 ~]# chkconfig nfs on

[root@nfs01 ~]# chkconfig rpcbind on

[root@nfs01 ~]# ls /etc/rc.d/rc3.d/|grep -E "nfs|rpc"

K61nfs-rdma

K69rpcsvcgssd

S13rpcbind

S14nfslock

S19rpcgssd

S30nfs

#### NFS多进程功能介绍

[root@nfs01 ~]# ps -ef|egrep "rpc|nfs"

rpc 1889 1 0 10:48 ? 00:00:00 rpcbind 🡨rpc主进程

root 1932 2 0 10:50 ? 00:00:00 [rpciod/0]

root 1941 1 0 10:50 ? 00:00:00 rpc.rquotad 🡨磁盘配额进程

root 1946 1 0 10:50 ? 00:00:00 rpc.mountd 🡨权限管理验证

root 1953 2 0 10:50 ? 00:00:00 [nfsd4]

root 1954 2 0 10:50 ? 00:00:00 [nfsd4\_callbacks]

root 1955 2 0 10:50 ? 00:00:00 [nfsd] 🡨主NFS进程

root 1993 1 0 10:50 ? 00:00:00 rpc.idmapd 🡨名字映射后台进程

### NFS服务端配置语法及配置实战

NFS服务器的默认配置文件路径为/etc/exports 且默认为空

/etc/exports文件配置格式为：

NFS共享的目录 NFS客户端地址1 （参数1，参数2） 客户端地址2

#share /data by cj for zhangkun at 20180329

/data 172.16.1.0/24(rw,sync)

NFS共享的目录:是NFS服务端要共享的实际目录，要用绝对路径。如（/data）.注意共享目录的本地权限，如果需要读写共享，一定要让本地目录可以被NFS客户端的用户nfsnobody读写

NFS客户端地址：为NFS服务器端授权的可以访问共享目录的NFS客户端地址，可以单独的IP地址或主机名、域名等，也可以为整个网段地址，还可以用“\*”来匹配所有客户端服务器

#### 配置实例

准备工作：

[root@nfs01 ~]# id nfsnobody

uid=65534(nfsnobody) gid=65534(nfsnobody) 组=65534(nfsnobody)

[root@nfs01 ~]# ll -ld /data

drwxr-xr-x 2 root root 4096 3月 29 14:25 /data

[root@nfs01 ~]# chown -R nfsnobody.nfsnobody /data

[root@nfs01 ~]# ll -ld /data

drwxr-xr-x 2 nfsnobody nfsnobody 4096 3月 29 14:25 /data

检查RPC及NFS是否启动；注意先启动RPC再启动NFS

[root@nfs01 ~]# /etc/init.d/rpcbind status

rpcbind (pid 1889) 正在运行...

[root@nfs01 ~]# /etc/init.d/nfs status

rpc.svcgssd 已停

rpc.mountd (pid 1946) 正在运行...

nfsd (pid 1962 1961 1960 1959 1958 1957 1956 1955) 正在运行...

rpc.rquotad (pid 1941) 正在运行...

***启动或重启一定要平滑启动***：reload

***检查是否正确注册***

[root@nfs01 ~]# rpcinfo -p localhost

***配置实例一：***

[root@nfs01 ~]# cat /etc/exports

#share /data by cj for zhangkun at 20180329

/data 172.16.1.0/24(rw,sync)

[root@nfs01 ~]# exportfs -rv 🡨加载配置文件也可以检查错误

exporting 172.16.1.0/24:/data

[root@nfs01 ~]# cat /var/lib/nfs/etab 🡨检查NFS服务端配置文件的参数

/data 172.16.1.0/24(rw,sync,wdelay,hide,nocrossmnt,secure,root\_squash,no\_all\_squash,no\_subtree\_check,secure\_locks,acl,anonuid=65534,anongid=65534,sec=sys,rw,root\_squash,no\_all\_squash)

***检查是否生效：***

[root@nfs01 ~]# showmount -e 172.16.1.31

Export list for 172.16.1.31:

/data 172.16.1.0/24

#### NFS客户端配置

客户端只需要rpcbind服务

[root@web01 ~]# /etc/init.d/rpcbind start 🡨启动rpcbind服务

正在启动 rpcbind： [确定]

[root@web01 ~]# /etc/init.d/rpcbind status

rpcbind (pid 2199) 正在运行...

[root@web01 ~]# chkconfig rpcbind on 🡨设置开机启动

[root@web01 ~]# showmount -e 172.16.1.31 🡨查看共享的目录

Export list for 172.16.1.31:

/data 172.16.1.0/24

[root@web01 ~]# mount -t nfs 172.16.1.31:/data /mnt

[root@web01 ~]# df -HP

Filesystem Size Used Avail Use% Mounted on

/dev/sda3 8.7G 1.6G 6.7G 19% /

tmpfs 515M 0 515M 0% /dev/shm

/dev/sda1 199M 37M 152M 20% /boot

172.16.1.31:/data 8.7G 1.6G 6.7G 19% /mnt

***另外一台挂载练习***

[root@backup ~]# /etc/init.d/rpcbind start

正在启动 rpcbind： [确定]

[root@backup ~]# /etc/init.d/rpcbind status

rpcbind (pid 2121) 正在运行...

[root@backup ~]# showmount -e 172.16.1.31

Export list for 172.16.1.31:

/data 172.16.1.0/24

[root@backup ~]# mount -t nfs 172.16.1.31:/data /opt/

[root@backup ~]# df -HP

Filesystem Size Used Avail Use% Mounted on

/dev/sda3 8.7G 1.6G 6.7G 19% /

tmpfs 515M 0 515M 0% /dev/shm

/dev/sda1 199M 37M 152M 20% /boot

172.16.1.31:/data 8.7G 1.6G 6.7G 19% /opt

小结：web01、backup往/opt|/mnt中写入文件数据，服务端【nfs01-server】/data下也可以看到-相当于都是在同一个目录下操作(增减修改)

#### 设置rpcbind及NFS开机启动

NFS服务端：

[root@nfs01 data]# echo "/etc/init.d/rpcbind start" >>/etc/rc.local

[root@nfs01 data]# echo "/etc/init.d/nfs start" >>/etc/rc.local

[root@nfs01 data]# cat /etc/rc.local

>/etc/udev/rules.d/70-persistent-net.rules

/etc/init.d/rpcbind start

/etc/init.d/nfs start

Nfs客户端：

[root@backup opt]# echo "/etc/init.d/rpcbind start" >>/etc/rc.local

[root@backup opt]# cat /etc/rc.local

>/etc/udev/rules.d/70-persistent-net.rules

/usr/bin/rsync --daemon

/etc/init.d/rpcbind start

[root@web01 mnt]# echo "/etc/init.d/rpcbind start" >>/etc/rc.local

[root@web01 mnt]# cat /etc/rc.local

>/etc/udev/rules.d/70-persistent-net.rules

/etc/init.d/rpcbind start

#### 开机自动挂载

写入/etc/ec.local

[root@web01 ~]# echo "/bin/mount -t nfs 172.16.1.31:/data /mnt/" >>/etc/rc.local

[root@web01 ~]# cat /etc/rc.local

>/etc/udev/rules.d/70-persistent-net.rules

/etc/init.d/rpcbind start

/bin/mount -t nfs 172.16.1.31:/data /mnt

#### 多NFS用户共享配置实例

多NFS客户端用户挂载共享目录的方式与上面相同，仅仅是客户端用户UID及GID要与服务端相同（新建用户（客户端与服务器）指定UID及GID）

服务器部分：

[root@nfs01 ~]# groupadd zuma -g 888

[root@nfs01 ~]# useradd zuma -u 888 -g zuma

[root@nfs01 ~]# id zuma

uid=888(zuma) gid=888(zuma) 组=888(zuma)

[root@nfs01 ~]# mkdir /oldboy -p

[root@nfs01 ~]# chown -R zuma.zuma /oldboy/

[root@nfs01 ~]## ll -ld /oldboy/

drwxr-xr-x 2 zuma zuma 4096 3月 29 15:57 /oldboy/

[root@nfs01 ~]# echo "/oldboy 172.16.1.31/24(rw,sync,all\_squash,anonuid=888,anongid=888)" >>/etc/exports

[root@nfs01 ~]# /etc/init.d/nfs reload

[root@nfs01 ~]# showmount -e 172.16.1.31

Export list for 172.16.1.31:

/oldboy 172.16.1.31/24

/data 172.16.1.0/24

客户端部分：

[root@web01 ~]# groupadd zuma -g 888

[root@web01 ~]# useradd zuma -u 888 -g zuma

[root@web01 ~]# id zuma

uid=888(zuma) gid=888(zuma) 组=888(zuma)

[root@web01 ~]# showmount -e 172.16.1.31

Export list for 172.16.1.31:

/oldboy 172.16.1.31/24

/data 172.16.1.0/24

[root@web01 ~]# mount -t nfs 172.16.1.31:/oldboy /home

[root@web01 ~]# df -HP

Filesystem Size Used Avail Use% Mounted on

/dev/sda3 8.7G 1.6G 6.7G 19% /

tmpfs 515M 0 515M 0% /dev/shm

/dev/sda1 199M 37M 152M 20% /boot

172.16.1.31:/data 8.7G 1.6G 6.7G 19% /mnt

172.16.1.31:/oldboy 8.7G 1.6G 6.7G 19% /home

[root@web01 ~]# echo "/bin/mount -t nfs 172.16.1.31:/oldboy /home/" >>/etc/rc.local

[root@web01 ~]# cat /etc/rc.local

>/etc/udev/rules.d/70-persistent-net.rules

/etc/init.d/rpcbind start

/bin/mount -t nfs 172.16.1.31:/data /mnt/

/bin/mount -t nfs 172.16.1.31:/oldboy /home/

#### 取消nfs共享挂载

[root@web01 ~]# umount -lf /home

## 小结：NFS配置过程

服务端：

1）安装软件

yum install nfd-utils rpcbind -y

2）启动服务（注意先后顺序）

/etc/init.d/rpcbind start

rpcinfo -p localhost

/etc/init.d/nfs start

rpcinfo -p localhost

3）设置开机自启动

chkconfig nfs on

Chkconfig rpcbind on

4）配置nfs服务

echo “/data 172.16.1.0/24(rw,sync)”>>/etc/exports

mkdir -p /data

chown -R nfsnobody.nfsnobody /data

（查看nfs默认使用的用户以及共享的参数cat /var/lib/nfs/etab）

5)重新挂载服务（优雅重启）

/etc/init.d/nfs reload ====exportfs -r

6）检查或测试挂载

showmount -e localhost

mount -t nfs 172.16.1.31:/data /mnt

7) 取消NFS共享

umount -lf /home

客户端：

1）安装软件

yum install nfs-utils rpcbind -y

2）启动rpcbind

/etc/init.d/rpcbind start

3）配置开机自启动

chkconfig rpcbind on

4）测试服务端共享情况

showmount -e server\_ip:/data /mnt

5）挂载

mount -t nfs server\_ip:/data /mnt

6）测试读写

7）开机自启动

echo “/etc/init.d/rpcbind start”>>/etc/rc.local

echo “/bin/mount -t nfs 172.16.1.31:/data /mnt”>>/etc/rc.local

写项目方案思路以及流程

提出问题

分析问题

解决问题

# NFS客户端挂载

NFS 客户端mount参数

|  |  |  |
| --- | --- | --- |
| fg | 前台挂载 |  |
| bg | 后台挂载 |  |
| Soft | NFS客户端以soft挂载时出现问题造成数据无法传输，则客户端一直会尝试，直到超时后显示错误才停止 |  |
| hard | 持续挂载，一直到解除挂载 |  |
| Intr | 配合head参数中断挂载请求 |  |
| Rsize wsize | 读出于写入的区块大小，这个值邮箱客户端与服务器传输数据的缓冲存储 | C6:  Rsize:131072  Wsize:131072 |
| Proto=udp | 使用UDP协定传输资料 |  |

可以通过nfs查看上述参数信息，追求极致如下：

[root@web01 ~]# mount -t nfs -o fg,hard,intr,rsize=131072,wsize=131072 172.16.1.31:/data /mnt

|  |  |  |
| --- | --- | --- |
| 参数 | 参数意思 | 默认值 |
| suid nosuid | 当挂载的文件系统上有任务SUID的程序时，只要使用nosuid就能够取消设置SUID的功能 | Suid |
| Rw  Ro | 指定文件系统只读或可写 | Rw |
| Dev  nodev | 是否保留装置文件的特殊功能 |  |
| Exec  noexec | 是否具有执行文件的权限 | Exec |
| User  Nouser | 是否允许用户拥有文件的挂载与卸载功能，如果要保护文件系统，最好不要为用户提供挂载与卸载功能 | Nouser |
| Auto  noauto | Auto是指“mount -a”时会不会被挂载的项目，如果不需要这个分区时被挂载，可以设置为noauto | auto |

## NFS客户端挂载优化

[root@web01 ~]# mount -t nfs -o noatime,nodiratime,nosuid,noexec,nodev,rsize=131072,wsize=131072 172.16.1.31:/data /mnt

实际测试:默认挂载配置效果也不错

[root@web01 ~]# mount -t nfs 172.16.1.31:/data /mnt

## NFS企业级内核优化及总结

/proc/sys/net/core/rmem\_default 指定接受套接字缓冲区大小的默认值

/proc/sys/net/core/rmem\_max 指定接受套接字缓冲区大小的最大值

/proc/sys/net/core/wmem\_default 指定发送套接字缓冲区大小的默认值

/proc/sys/net/core/wmem\_max 指定发送套接字缓冲区大小的最大值

快捷优化命令：

[root@nfs01 ~]# cat /etc/sysctl.conf <<EOF

net.core.wmem\_default = 8388608

net.core.rmem\_default = 8388608

net.core.wmem\_max = 16777216

net.core.rmem\_max = 16777216

EOF

Sysctl -p

## 企业生产场景NFS共享存储化小结

* 硬件：sas/ssd磁盘，买多块，raid5/raid10 网卡吞吐量要大，至少千兆 多块（bond）
* NFS服务器配置：/data 10.0.0.0/24(rw,sysnc,all\_squash,anonuid=65534,anongid=65534)
* NFS客户端挂载：mount -t nfs -o noatime,nodiratime,nosuid,noexec,nodev,rsize=131072,wsize=131072 172.16.1.31:/data /mnt
* NFS服务器内核的优化：

cat /etc/sysctl.conf <<EOF

net.core.wmem\_default = 8388608

net.core.rmem\_default = 8388608

net.core.wmem\_max = 16777216

net.core.rmem\_max = 16777216

EOF

* 如果卸载的时候提示“umount: /mnt: device is busy”,需要退出挂载目录再进行卸载，如果是NFS server 宕机，则需要强制卸载，可以执行 umount -lf /mnt
* 大型网站NFS网络问卷系统的替代软件为分布式文件系统Moosefs(mfs)、GlusterFS、FastDFS

## NFS系统应用的特性

优点：

* 简单，容易上手，容易掌握
* NFS文件系统内数据是在文件系统之上，即数据是能看得见的
* 部署快速，维护简单方便，且可控
* 可靠，软件层面看，数据可靠性高，经久耐用
* 服务非常稳定

局限：

* 存在单点故障，NFSserver宕机了，所有客户端都不能访问。这可以使用负载均衡弥补
* 在大数据高并发场合，NFS效率、性能有限
* 客户端认证是给予IP和主机名，权限要根据ID识别安全性一般（内网问题不大）
* NFS数据明文，本身不对数据完整性进行验证
* 多台客户端挂载一个NFS服务器，连接管理维护麻烦耦合度高。
* 涉及了同步（实时等待）和异步解耦的概念，NFS服务器端和客户端相对来说就是耦合度有些高

Showmount -e 172.16.1.31 查看共享的文件目录

exportfs -rv 等同不重启nfs使nfs配置文件生效

用户上传的数据（图片、资料）需要做到实时备份

1. 企业项目案例：

1、集权分治权限管理及日志审计

2、全网数据备份解决方案

***3、共享存储实时备份（共享存储单点解决方案）***

第一个里程碑：

a.实现从nfs客户端到rsync服务端的rsync服务部署

第二个里程碑

b.实现从nfs客户端对nfs目录文件系统事件的实时监控

第三个里程碑

c.监控到nfs目录文件系统事件变化后，触发rsync推送变化的文件（脚本实现）

## NFS存储实时复制备份原理



准备rsync服务器

[root@backup ~]# ps -ef|grep rsync &&lsof -i :873

root 1455 1 0 09:49 ? 00:00:00 /usr/bin/rsync --daemon

root 1511 1489 0 09:58 pts/0 00:00:00 grep rsync

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

rsync 1455 root 4u IPv4 11092 0t0 TCP \*:rsync (LISTEN)

rsync 1455 root 5u IPv6 11093 0t0 TCP \*:rsync (LISTEN)

[root@backup ~]# vim /etc/rsyncd.conf

[nfsbackup] 🡨添加nfsbackup模块

path = /nfsbackup

ignore errors

read only = false

list = false

hosts allow = 172.16.1.0/24

#host deny = 0.0.0.0/32

##auth users = rsync\_backup

##secrets file = /etc/rsync.password

[root@backup ~]# pkill rsync

[root@backup ~]# mkdir -p /nfsbackup

[root@backup ~]# cd /nfsbackup/

[root@backup nfsbackup]# ps -ef|grep rsync &&lsof -i :873

root 1525 1489 0 10:00 pts/0 00:00:00 grep rsync

[root@backup nfsbackup]# rsync --deamon

rsync: --deamon: unknown option

rsync error: syntax or usage error (code 1) at main.c(1422) [client=3.0.6]

[root@backup nfsbackup]# rsync --daemon

[root@backup nfsbackup]# ps -ef|grep rsync &&lsof -i :873

root 1544 1 0 10:01 ? 00:00:00 rsync --daemon

root 1546 1489 0 10:01 pts/0 00:00:00 grep rsync

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

rsync 1544 root 4u IPv4 11789 0t0 TCP \*:rsync (LISTEN)

rsync 1544 root 5u IPv6 11790 0t0 TCP \*:rsync (LISTEN)

[root@backup nfsbackup]# chown -R rsync.rsync /nfsbackup/

***Nfs客户端创建nfsbackup目录，写入测试 被推送到backup服务器***

[root@nfs01 ~]# mkdir -p /nfsbackup

[root@nfs01 ~]# cd /nfsbackup/

[root@nfs01 nfsbackup]# touch a{1..5}.log

[root@nfs01 /]# rsync -avz /nfsbackup/ rsync\_backup@172.16.1.41::nfsbackup/--password-file=/etc/rsync.password

sending incremental file list

./

a1.log

a2.log

a3.log

a4.log

a5.log

sent 264 bytes received 106 bytes 740.00 bytes/sec

total size is 0 speedup is 0.00