

XI`AN TECHNOLOGICAL UNIVERSITY

**部署NFS网络文件系统**

专 业： 信息与计算科学

班 级： 17100101

姓 名： 李佳钰 葛伟鹏 孙文成

学 号： 17100101104 17100101103

17100101108

实验学时： 2学时

指导教师： 刘建伟

成 绩：

2019 年 12 月 15 日

**摘要**

NFS是Network File System的缩写,即网络文件系统。-种使用于分散式文件系统的协定，由Sun公司开发，于1984年向外公布。功能是通过网络让不同的机器不同的操作系统能够彼此分享个别的数据，让应用程序在客户端通过网络访问位于服务器磁盘中的数据，是在类Unix系统间实现磁盘文件共享的一种方法。

NFS的基本原则是‘容许不同的客户端及服务端通过一组RPC分享相同的文件系统”,  它是独立于操作系统，容许不同硬件及操作系统的系统共同进行文件的分享。  
 NFS在文件传送或信息传送过程中依赖于RPC协议。RPC,远程过程调用(RemoteProcedure Call) 是能使客户端执行其他系统中程序的一-种机制。NFS本身是没有提供信息传输的协议和功能的，但NFS却能让我们通过网络进行资料的分享，这是因为NFS使用了一些其它的传输协议。而这些传输协议用到这个RPC功能的。可以说NFS本身就是使用RPC的一-个程序。或者说NFS也是一-个RPC SERVER。所以只要用到NFS的地方都要启动RPC服务，不论是NFS SERVER或者NFS CLIENT。这样SERVER和CLIENT才能通过RPC来实现PROGRAM PORT 的对应。可以这么理解RPC和NFS的关系: NFS是一-个文件系统，而RPC是负责负责信息的传输。

**关键字：**NFS网络文件系统 文件传输 NFS服务端部署 NFS客户端部署

**目录**

[NFS简介 4](#_Toc6932)

[NFS运行原理 5](#_Toc3014)

[通信过程 7](#_Toc7018)

[NFS和RPC的关系 8](#_Toc28521)

[服务器端文件分享权限分配方式 9](#_Toc10797)

[NFS服务器配置 11](#_Toc8845)

[NFS客户端下载安装 16](#_Toc3605)

[实验过程 18](#_Toc8598)

[总结 19](#_Toc486)

[参考文献 20](#_Toc20012)

# NFS简介

NFS (Network FileSystem,网络文件系统)是由SUN公司发展，并于1984年推出的技术,用于在不同机器，不同操作系统之间通过网络互相分享各自的文件。NFS设计之初就是为了在不同的系统间使用，所以它的通讯协议设计与主机及操作系统无关。  
 NFS分服务器和客户机，当使用远端文件时只要用mount命令就可把远端NFS服务器上的文件系统挂载在本地文件系统之下，操作远程文件与操作本地文件没有不同。NFS服务器所共享文件或目录记录在/etc/exports文件中。嵌入式Linux开发中，会经常使用NFS，目标系统通常作为NFS客户机使用，Linux主机作为NFS服务器。在目标系统\_上通过NFS，将服务器的NFS共享目录挂载到本地,可以直接运行服务器.上的文件。在调试系统驱动模块以及应用程序，NFS都是十分必要的，并且Linux还支持NFS根文件系统,能直接从远程NFS root启动系统，这对嵌入式Linux根文件系统裁剪和集成也是十分有必要的。

NFS的优点：

1、本地工作站可以使用更少的磁盘空间，因为常用的数据可以保存你在一台机器上，并让网络上的其他机器可以访问他。

2、不需要为用户在每台网络机器上放一个用户目录，用户目录可以在NFS服务器上设置使其在整个网络上使用。

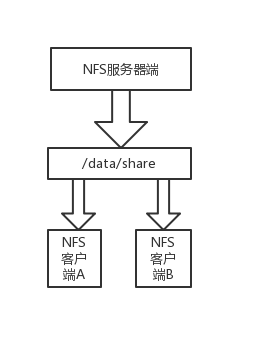
3、存储设备可以被网络上其他机器使用，可以减少网络上移动设备的数量。

# **NFS运行原理**

NFS有好几个版本，最新版是NFSv4。而NFSv 1到NFSv4都是透过RPC来运作，所谓的RPC (Remote Procedure Call) 是远端程序呼叫。当跑--个程式时,该程式呼叫另外一个程式去做事情然后把结果传回来称为Procedure Call,所以当一个程式呼叫在不同主机的程式去处理事情并传回结果，我们就称为Remote Procedure Call。对于呼叫别人的程式来说，它会感觉到被呼叫的程式也是在本地执行的。所以RPC有个好处就是在写程式的时候，只要管好呼叫的部份，我们不必实际去处理网络封包等细节，这些细节就给RPC系统处理即可，这样程式也会好写很多。故RPC程式要处理远端传输的问题,只要写好procedurecall来呼叫远端主机的程序读取或写入硬盘即可，这大大简化coding的难度。

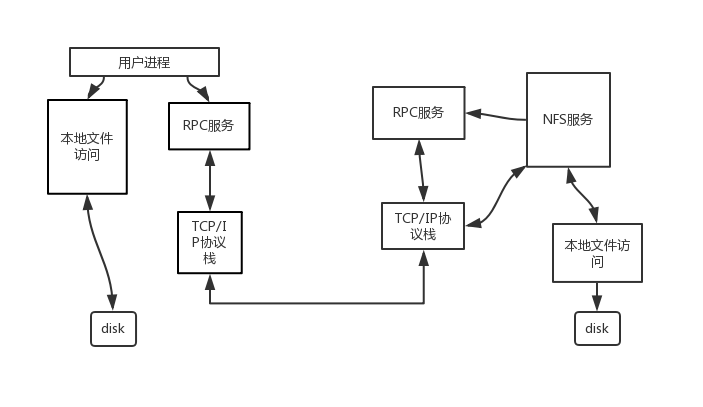
NFS的实作就是靠RPC了，所以NFS是不处理网络封包的问题，它只从Client呼叫.Server\_上的NFS procedure ，并且告诉该remote procedure想要干嘛，而封包实际的传输就透过标准的RPC协定来达成。故在NFS系统上, Client端和Server端都要安装RPC系统,而RPC在Linux核心里已经有内建。

挂载结构如图：当在NFS服务器设置好一个共享目录/data/share后，其他的有权限访问NFS服务器的NFS客户端就可以将这个目录挂载到自己的本地，并且可以看到服务器端/data/share下的所有数据。



NFS客户端/NFS服务端、RPC关系图：

NFS客户端 NFS服务器端



# **通信过程**

1、首先服务器端启动RPC服务，并开启端口

2、服务器端启动NFS服务，并向RPC注册端口信息

3、客户端启动RPC服务，向服务端的RPC服务请求服务端的NFS端口

4、服务端的RPC服务反馈NFS端口信息给客户端

5、客户端通过获取的NFS端口来建立和服务端的NFS连接并进行数据的传输

# **NFS和RPC的关系**

NFS在文件传送或信息传送过程中依赖于RPC协议。RPC是远程过程调用 (Remote Procedure Call) 是能使客户端执行其他系统中程序的一种机制。NFS本身是没有提供信息传输的协议和功能的，但NFS却能让我们通过网络进行资料的分享，这是因为NFS使用了一些其它的传输协议。而这些传输协议用到这个RPC功能，可以理解为NFS本身就是使用RPC的一个程序。

# **服务器端文件分享权限分配方式**

1、no\_root\_squash:若客户端账号在服务端上也有的话，那么客户端操作该分享目录时，会有该账号在服务器端上的权限。

2、root\_squash:同no\_root\_squash的功能基本一样，不同点在于当客户端用root去操作时，就会变成匿名nobaby权限，相对于no\_root\_squash来说更加安全，客户端也无法大规模的破坏。通常会配合NIS一起使用。

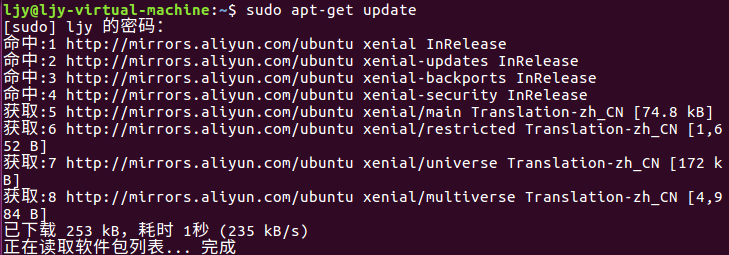
3、all\_squash:无论客户端用什么账号，在该分享目录的权限都只有nobaby。在开放给所有人使用的匿名NFS Server通常会用这样的设定。

4、anonuid/anongid：当服务器端安装后，不管用什么账号，在该目录的权限只有anonuid指定的账号与anongid制定的群组。

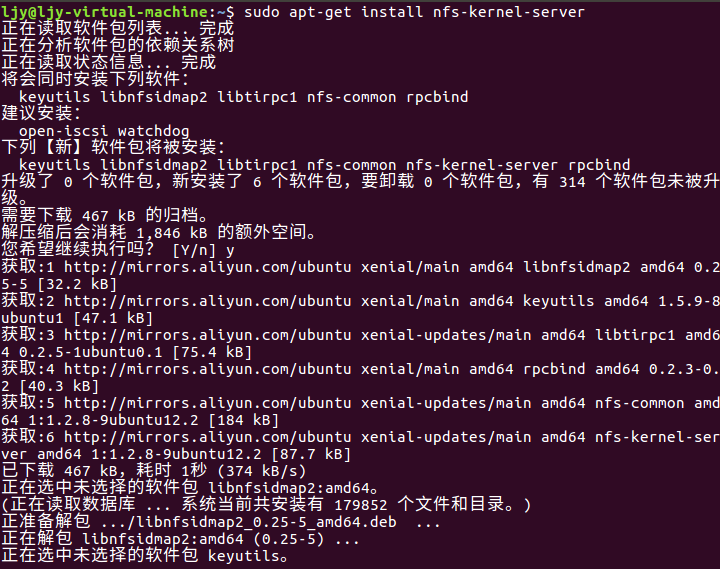
**软件安装**

服务器端需要安装 nfs-kernel-server 软件包：

命令:sudo apt-get update



命令:sudo apt-get install nfs-kernel-server



# **NFS服务器配置**

在服务器端创建共享目录

命令:

mkdir -p /tmp

mkdir -p /data

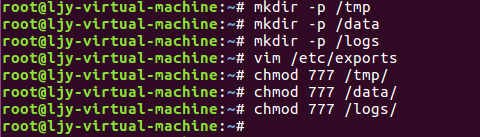
mkdir -p /logs

修改权限：

chmod 777 /tmp

chmod 777 /data

chmod 777 /logs

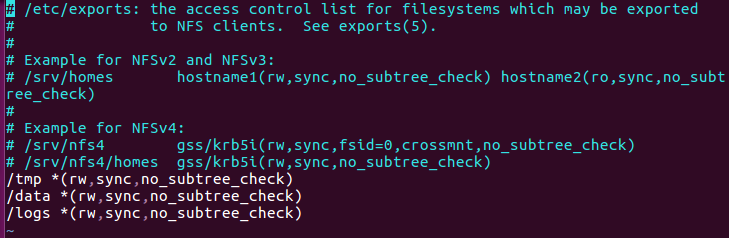


修改 exports 文件

命令:

sudo vim /etc/exports





<输出目录> [客户端1 选项（访问权限,用户映射,其他）] [客户端2 选项（访问权限,用户映射,其他）]

a. 输出目录：

输出目录是指NFS系统中需要共享给客户机使用的目录；

b. 客户端：

客户端是指网络中可以访问这个NFS输出目录的计算机

客户端常用的指定方式

指定ip地址的主机：192.168.0.200

指定子网中的所有主机：192.168.0.0/24 192.168.0.0/255.255.255.0

指定域名的主机：david.bsmart.cn

指定域中的所有主机：\*.bsmart.cn

所有主机：\*

c. 选项：

选项用来设置输出目录的访问权限、用户映射等。

NFS主要有3类选项：

访问权限选项

设置输出目录只读：ro

设置输出目录读写：rw

用户映射选项

all\_squash：将远程访问的所有普通用户及所属组都映射为匿名用户或用户组（nfsnobody）；

no\_all\_squash：与all\_squash取反（默认设置）；

root\_squash：将root用户及所属组都映射为匿名用户或用户组（默认设置）；

no\_root\_squash：与rootsquash取反；

anonuid=xxx：将远程访问的所有用户都映射为匿名用户，并指定该用户为本地用户（UID=xxx）；

anongid=xxx：将远程访问的所有用户组都映射为匿名用户组账户，并指定该匿名用户组账户为本地用户组账户（GID=xxx）；

其它选项

secure：限制客户端只能从小于1024的tcp/ip端口连接nfs服务器（默认设置）；

insecure：允许客户端从大于1024的tcp/ip端口连接服务器；

sync：将数据同步写入内存缓冲区与磁盘中，效率低，但可以保证数据的一致性；

async：将数据先保存在内存缓冲区中，必要时才写入磁盘；

wdelay：检查是否有相关的写操作，如果有则将这些写操作一起执行，这样可以提高效率（默认设置）；

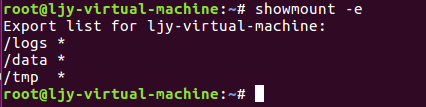
no\_wdelay：若有写操作则立即执行，应与sync配合使用；

subtree：若输出目录是一个子目录，则nfs服务器将检查其父目录的权限(默认设置)；

no\_subtree：即使输出目录是一个子目录，nfs服务器也不检查其父目录的权限，这样可以提高效率；

查看是否共享出去文件：

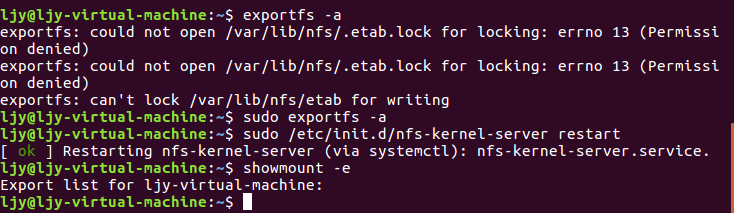
命令：showmount -e



查看服务器ip地址

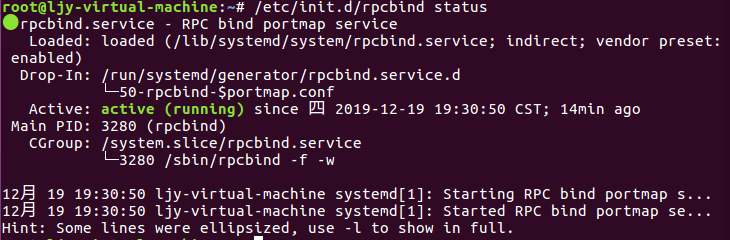


更新exports重新启动nfs



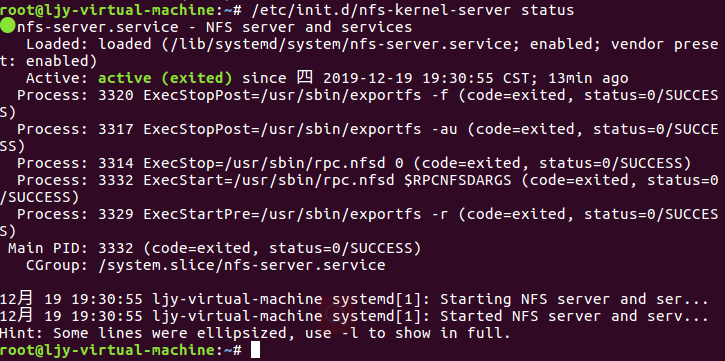
查看RPC的服务状态：

命令：/etc/init.d/rpcbind status



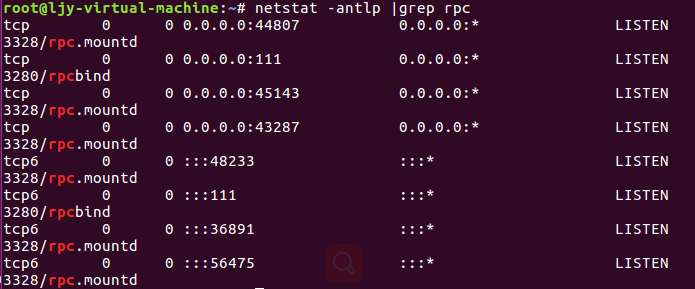
查看nfs服务的状态：

命令:/etc/init.d/nfs-kernel-server status



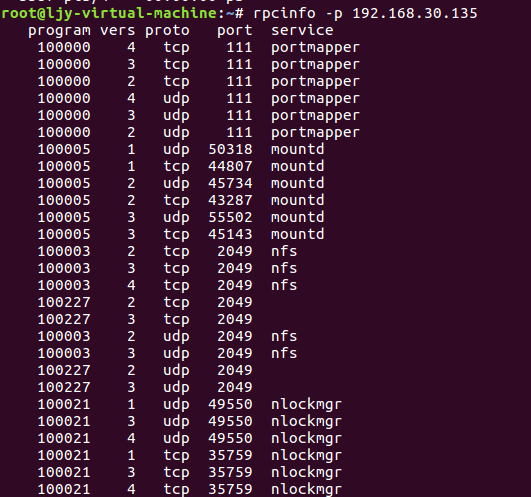
查看rpcbind服务的端口：

命令：netstat -antlp|grep rpc



查看此时RPC服务上是否有端口注册：

命令：rpcinfo -p 192.168.30.135

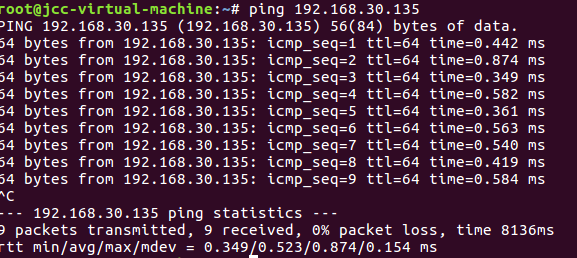


**NFS客户端下载安装**

命令:sudo apt install nfs-common



查看是否能ping通服务器



查看客户端ip地址

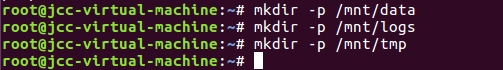


创建本地挂载目录:

命令:mkdir -p /mnt/data

命令:mkdir -p /mnt/logs

命令:mkdir -p /mnt/tmp

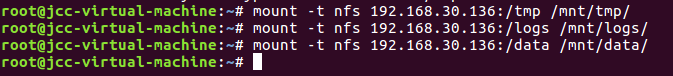


挂载共享目录:

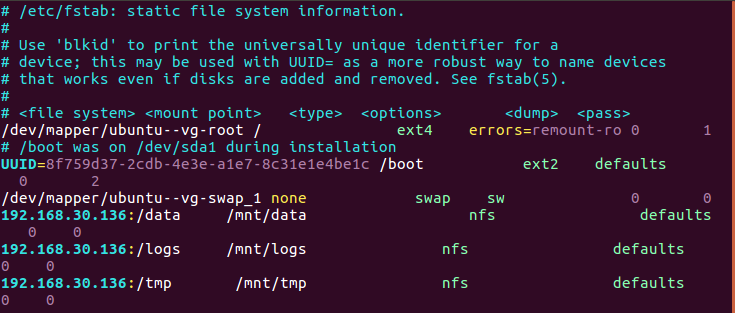
命令:sudo mount -t nfs 192.168.30.136:/data /mnt/data

命令:sudo mount -t nfs 192.168.30.136:/logs /mnt/logs

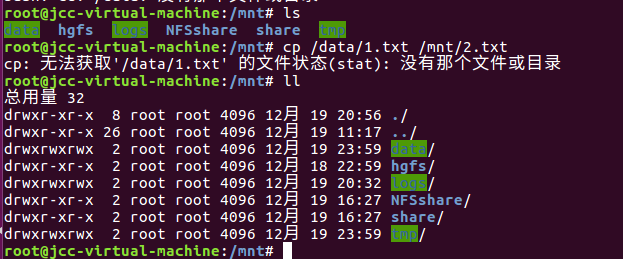
命令:sudo mount -t nfs 192.168.30.136:/tmp /mnt/tmp



配置/etc/fstab

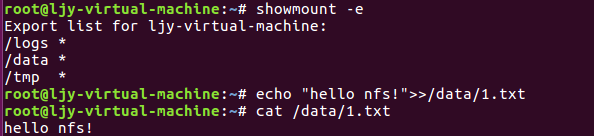


挂载成功：

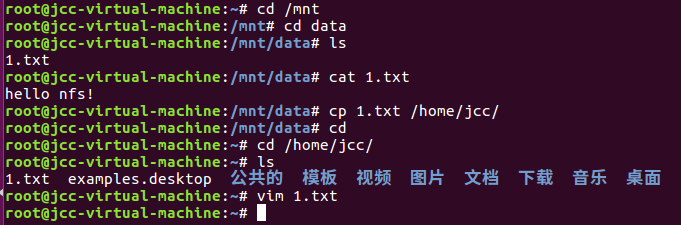


**实验过程**

在服务端/data下新建一个1.txt文件，写入hello nfs！



在客户端访问/mnt/data/1.txt并拷贝到本地的/home/jcc



# 总结

在本次Ubuntu16.04.6下部署NFS网络文件系统的过程中，我学习到了很多东西。在部署过程中也遇到了诸多问题，首先是下载安装NFS的时候遇到了版本不一致的问题，

在配置/etc/exports文件的过程中，没有将文件的路径写对，没有给到客户端的ip地址导致的错误。在配置/etc/exports好exports后忘记重启NFS导致客户端无法访问。没有关闭防火墙导致客户端ping不到服务器的问题。

在客户端安装nfs-common时候版本不一致的错误，以及ping不通服务端ip的问题，还有挂载服务端文件的时候/etc/fstab文件无法访问的问题。

在我们一一解决后，我们成功将服务端的文件系统分享到客户端，实现了NFS文件系统，可以对服务器文件进行拷贝下载，和对本地文件的写入修改，以及上传到服务端等操作。

本次部署NFS没有实现开机自动启动NFS，以及没有绑定服务端静态ip，导致每次重启都需要重新挂载服务端。后续我们会对服务端进行静态绑定，实现开机自动启动NFS网络文件系统等其他功能。

**参考文献**

[1]NFS部署实战及原理介绍\_百度百科

<https://www.linuxidc.com/Linux/2019-07/159479.htm>

2019-12-15

[2]Ubuntu16.04搭建NFS 文件共享服务器的方法\_CSDN

<https://www.jb51.net/article/138141.htm>

2019-12-15

# Ubuntu下的NFS服务配置总结

# <https://blog.csdn.net/jin970505/article/details/78847028>

2019-12-15

[4]明哲.Linux系统构架与运维实战,电子工业的出版社.2011-1

[5]姜启源,谢金星,叶俊.数学建模(第四版),高等教育出版社.2011-1

[6]鸟哥的私房菜,清华大学出版社.2012-9