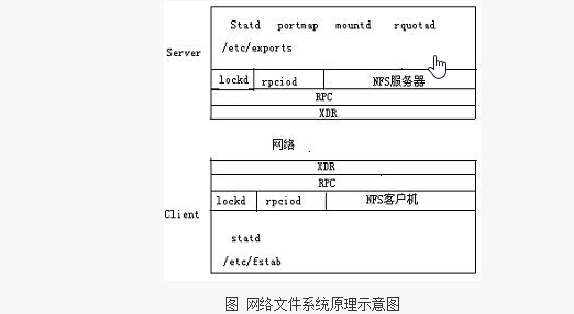
NFS服务器的安装与配置

NFS 是Network File System的缩写，即网络文件系统。一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 (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。

# NFS原理



NFS比较复杂，包括很多组件，通过特殊的协议进行交互。不同的组件在操作系统当中都使用不同的配置文件以及状态文件。下图说明了NFS的主要组件及配置文件。NFS分为服务器和客户机两部分，每个主机都有自己的内核级服务：外部数据表示(XDR，eXternal Data Representation)、远程过程调用(RPC，Remote Procedure Call)、I/O监控程序和锁监控程序。每个主机还有自己的用户级服务。内核级服务和用户级服务都依赖于主机的功能：NFS客户机或者是NFS服务器。当然，还要依赖于每个主机使用的不同功能的配置文件（如果是服务器，则用的是/etc/exports配置文件，如果是客户机，则用的是/etc/fstab配置文件）。如果一台主机既是服务器又是客户机，那么它需要运行两个部分的服务。

在服务器端，portmap(被rpcbind替代)、 mountd、 nfsd三个监控程序将在后台运行。portmap监控程序用来注册基于rpc的服务。当一个RPC的监控程序启动的时候，它告诉portmap监控程序它在哪一个端口进行侦听，并且它在进行什么样的RPC服务。当一个客户机向服务器提出一个RPC请求，那么它就会和portmap监控程序取得联系以确定RPC消息应该发往的端口号。而Mountd监控程序的功能是来读取服务器端的/etc/exportfs文件并且创建一个将服务器的本地文件系统导出的主机和网络列表，因而客户机的挂接(mount)请求都被定位到mountd监控程序(daemon)。当验证了服务器确实具有挂接所请求的文件系统的权限以后，mountd为请求的挂接点返回一个文件句柄。而nfsd监控程序则被服务器用来处理客户机端发过来的请求，由于服务器需要同时处理多个客户机的请求，所以在缺省情况下，在Linux当中将会自动启动八个nfsd线程。当然，如果NFS服务器特别忙的时候，系统有可能根据实际情况启动三十个线程。

# NFS安装

yum -y install nfs-utils\* rpcbind

chkconfig nfs on

chkconfig rpcbind on

# NFS配置和使用

## 服务器配置

NFS服务的配置文件为 /etc/exports，这个文件是NFS的主要配置文件，不过系统并没有默认值，所以这个文件不一定会存在，可能要使用vim手动建立，然后在文件里面写入配置内容。/etc/exports文件内容格式：

|  |
| --- |
| <共享目录> [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）] |

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：\*

NFS主要有3类选项：选项用来设置输出目录的访问权限、用户映射等。

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；

配置：home/k12 172.16.129.233(ro,sync,no\_root\_squash)

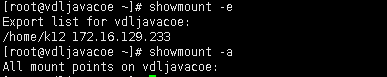
启动服务：service rpcbind start && service nfs start

# 默认查看自己共享的服务，前提是要DNS能解析自己，不然容易报错

showmount -e

#显示已经与客户端连接上的目录信息

showmount -a



## 客户机配置

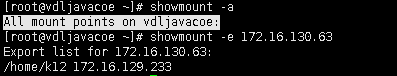
客户机配置相对简单，只需要使用下述命令mount NFS文件系统即可：

# mount -t nfs 172.16.130.63:/home/k12 /home/k12/hello

上述命令将远程的共享目录挂接到本地的/home/k12目录下，用户可以直接对该目录进行操作，从而获取远程的共享资源。

#命令查询NFS的共享状态

showmount -e 172.16.130.63



#自动挂载格式

<server>:</remote/export> </local/directory> nfs < options> 0 0

例如：172.16.130.63:/home/k12 /home/k12/hello nfs defaults 0 0

# 相关命令

## exportfs

如果我们在启动了NFS之后又修改了/etc/exports，是不是还要重新启动nfs呢？这个时候我们就可以用exportfs 命令来使改动立刻生效，该命令格式如下：

exportfs [-aruv]

* -a 全部挂载或卸载 /etc/exports中的内容
* -r 重新读取/etc/exports 中的信息 ，并同步更新/etc/exports、/var/lib/nfs/xtab
* -u 卸载单一目录（和-a一起使用为卸载所有/etc/exports文件中的目录）
* -v 在export的时候，将详细的信息输出到屏幕上。

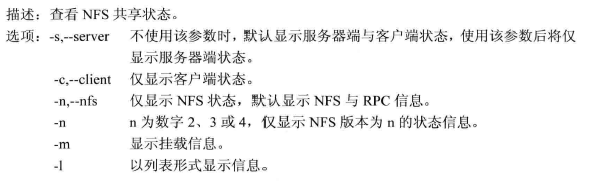
具体例子：

exportfs -au #卸载所有共享目录

exportfs -rv #重新共享所有目录并输出详细信息

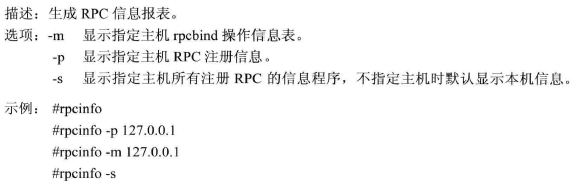
## nfsstat

查看NFS的运行状态，对于调整NFS的运行有很大帮助。



## rpcinfo

查看rpc执行信息，可以用于检测rpc运行情况的工具，利用rpcinfo -p 可以查看出RPC开启的端口所提供的程序有哪些。



## showmount

* -a 显示已经于客户端连接上的目录信息
* -e IP或者hostname 显示此IP地址分享出来的目录

## netstat

可以查看出nfs服务开启的端口，其中nfs 开启的是2049，portmap 开启的是111，其余则是rpc开启的。最后注意两点，虽然通过权限设置可以让普通用户访问，但是挂载的时候默认情况下只有root可以去挂载，普通用户可以执行sudo。NFS server 关机的时候一点要确保NFS服务关闭，没有客户端处于连接状态！通过showmount -a 可以查看，如果有的话用kill killall pkill 来结束，（-9 强制结束）