joe@nfss:~$ sudo apt-get install nfs-kernel-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  keyutils libgssglue1 libnfsidmap2 libtirpc1 nfs-common rpcbind
Suggested packages:
  open-iscsi watchdog
The following NEW packages will be installed:
  keyutils libgssglue1 libnfsidmap2 libtirpc1 nfs-common nfs-kernel-server rpcbind
0 upgraded, 7 newly installed, 0 to remove and 172 not upgraded.
Need to get 460 kB of archives.
After this operation, 2,049 kB of additional disk space will be used.
Do you want to continue? [Y/n] [Y]
joe@nfss:~$ echo "Is it running?"
Is it running?
joe@nfss:~$
joe@nfss:~$ ps aux | egrep 'nfs|lock|rpc|portmap'
root   20  0.0  0.0     0    0    ?   S<     11:43     0:00 [kblockd]
root  2427  0.0  0.2   23424   1168    ?   Ss   13:08     0:00 rpcbind -w
statd  2660  0.0  0.2   21544   1368    ?   Ss   13:08     0:00 rpc.statd -L
root  2734  0.0  0.0     0    0    ?   S<   13:08     0:00 [rpciod]
root  2735  0.0  0.0     0    0    ?   S<   13:08     0:00 [nfsiod]
root  2744  0.0  0.0   23480   420    ?   Ss   13:08     0:00 rpc.idmapd
joe  2984  0.0  0.1   8168   888     pts/0  S+   13:09     0:00 egrep --color=auto nfs

joe@nfss:~$
try to start the service
$ sudo service nfs-kernel-server start
  * Not starting NFS kernel daemon: no exports.

$ echo "we need to define what to export"
we need to define what to export

$
joe@nfss:/etc

joe@nfss:/etc$ sudo vi exports
# /etc/exports: the access control list for filesystems which may be exported to NFS clients. See exports(5).

# Example for NFSv2 and NFSv3:
#/srv/homes   hostname1(rw, sync, no_subtree_check) hostname2(ro, sync, no_subtree_check)

# Example for NFSv4:
#/srv/nfs4    gss/krb5i(rw, sync, fsid=0, crossmnt, no_subtree_check)
#/srv/nfs4/homes  gss/krb5i(rw, sync, no_subtree_check)

#/data/images will be the directory that we want to share
# 144.xx.yy.zz is the host that we allow to share it
# followed by various options
/data/images  144.38.220.211(rw, sync, no_subtree_check)
joe@nfss:/etc$ echo "make the directory"
make the directory
joe@nfss:/etc$
joe@nfss:/etc$ sudo mkdir -p /data/images
joe@nfss:/etc$ echo "Now lets see if it will start"
Now lets see if it will start
joe@nfss:/etc$
joe@nfss:/etc$ sudo service nfs-kernel-server start
* Exporting directories for NFS kernel daemon... [ OK ]
* Starting NFS kernel daemon [ OK ]
joe@nfss:/etc$ echo "YAY!"
YAY!
joe@nfss:/etc$
If it doesn't start, go back and fix something
joe@nfss:/var/lib/nfs$ cd
joe@nfss:~$ cat /var/lib/nfs/etab
/data/images 144.38.220.211(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)
joe@nfss:~$
cd /data/images/
ls -la

made_on_server

ls -la

Made_on_server

echo "We just created a file on the server"
we are now on the client

Install the client nfs utilities

sudo apt-get install nfs-common
joe@web:~$ ps aux | egrep 'rpc|nfs'
root  4258  0.0  0.2  23424  1168 ?   Ss   13:16  0:00  rpcbind -w
statd  4491  0.0  0.2  21544  1364 ?   Ss   13:16  0:00  rpc.statd -L
root  4564  0.0  0.0   0     0  ?      S<   13:16  0:00  [rpciod]
root  4565  0.0  0.0   0     0  ?      S<   13:16  0:00  [nfsiod]
root  4582  0.0  0.0  23480   416 ?   Ss   13:16  0:00  rpc.idmapd
joe   4604  0.0  0.1  8164    888 pts/1  S+   13:17  0:00  egrep --color=auto rpc

joe@web:~$
Create a local mount point

joe@web:~$ echo "Create a local mount point"
Create a local mount point
joe@nfss:/data/images

joe@web:~$ sudo mkdir -p /remote/images
joe@web:~$
joe@web:~$ df -h
Filesystem    Size  Used  Avail  Use% Mounted on
/dev/sda1     12G    1.3G  9.4G    12% /
none          4.0K     0  4.0K     0% /sys/fs/cgroup
udev          235M    4.0K 235M     1% /dev
tmpfs         50M    380K  49M     1% /run
none           5.0M     0  5.0M     0% /run/lock
none          246M     0 246M     0% /run/shm
none          100M     0 100M     0% /run/user
joe@web:~$
Now for the magic!

```
joe@web:~$ echo "Now for the magic!"
Now for the magic!
joe@web:~$ sudo mount -t nfs 144.38.220.200:/data/images /remote/images/
joe@web:~$
```
<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Size</th>
<th>Used</th>
<th>Avail</th>
<th>Use%</th>
<th>Mounted on</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda1</td>
<td>12G</td>
<td>1.3G</td>
<td>9.4G</td>
<td>12%</td>
<td>/</td>
</tr>
<tr>
<td>none</td>
<td>4.0K</td>
<td>0</td>
<td>4.0K</td>
<td>0%</td>
<td>/sys/fs/cgroup</td>
</tr>
<tr>
<td>udev</td>
<td>235M</td>
<td>4.0K</td>
<td>235M</td>
<td>1%</td>
<td>/dev</td>
</tr>
<tr>
<td>tmpfs</td>
<td>50M</td>
<td>380K</td>
<td>49M</td>
<td>1%</td>
<td>/run</td>
</tr>
<tr>
<td>none</td>
<td>5.0M</td>
<td>0</td>
<td>5.0M</td>
<td>0%</td>
<td>/run/lock</td>
</tr>
<tr>
<td>none</td>
<td>246M</td>
<td>0</td>
<td>246M</td>
<td>0%</td>
<td>/run/shm</td>
</tr>
<tr>
<td>none</td>
<td>100M</td>
<td>0</td>
<td>100M</td>
<td>0%</td>
<td>/run/user</td>
</tr>
<tr>
<td>144.38.220.200:/data/images</td>
<td>12G</td>
<td>1.3G</td>
<td>9.4G</td>
<td>12%</td>
<td>/remote/images</td>
</tr>
</tbody>
</table>
joe@web:~$ cd /remote/images/
joe@web:/remote/images$ ls -l
total 0
-rw-r--r-- 1 root root 0 Jan 20 13:15 MADE_ON_SERVER
joe@web:/remote/images$
joe@web:/remote/images$ ls
MADE_ON_SERVER
joe@web:/remote/images$ sudo touch MADE_FROM_CLIENT
  touch: cannot touch ‘MADE_FROM_CLIENT’: Permission denied
joe@web:/remote/images$ echo "OUCH!"
OUCH!
joe@web:/remote/images$ echo "Why can't we write to the shared dir?"
Why can't we write to the shared dir?
joe@web:/remote/images$
We are writing as root, and root does not have privileges on the remote server. They are squashed.
joe@nfss:/data/images$ echo "Back on server"
Back on server
joe@nfss:/data/images$ sudo vi /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
to NFS clients. See exports(5).

# Example for NFSv2 and NFSv3:
# /srv/home  hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)

# Example for NFSv4:
# /srv/nfs4  gsskrb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gsskrb5i(rw,sync,no_subtree_check)

#/data/images will be the directory that we want to share
# 144.xx.yy.zz is the host that we allow to share it
# followed by various options
/data/images 144.38.220.211(rw,sync,no_subtree_check,no_root_squash)
joe@nfss:/data/images$ sudo service nfs-kernel-server reload
* Re-exporting directories for NFS kernel daemon... [ OK ]

joe@nfss:/data/images$
joe@web:/remote/images$ echo "Back on client"
Back on client
joe@web:/remote/images$
joe@web:/remote/images$ sudo touch MADE_FROM_CLIENT
joe@web:/remote/images$ ls -la
total 8
drwxr-xr-x 3 root root 4096 Jan 20 13:18 ..
-rw-r--r-- 1 root root 0 Jan 20 13:24 MADE_FROM_CLIENT
-rw-r--r-- 1 root root 0 Jan 20 13:15 MADE_ON_SERVER
joe@web:/remote/images$ echo "YAY"
YAY
joe@web:/remote/images$
How do we unmount that directory?
joe@web:/remote/images$ echo "How do we unmount that directory?"
How do we unmount that directory?
joe@web:/remote/images$ sudo umount /remote/images
umount.nfs: /remote/images: device is busy
umount.nfs: /remote/images: device is busy
joe@web:/remote/images$ echo "Why is device busy?"
Why is device busy?
joe@web:/remote/images$
joe@web:/remote/images$
joe@web:/remote/images$
joe@web:/remote/images$
joe@web:/remote/images$ echo "Because we are IN that directory!"
Because we are IN that directory!
joe@web:/remote/images$
joe@web:~$ sudo umount /remote/images
joe@web:~$ echo "YAY"
YAY
joe@web:~$ df -h
Filesystem  Size  Used  Avail  Use%  Mounted on
/dev/sda1  12G  1.3G  9.4G   12%   /
none      4.0K   0   4.0K   0%  /sys/fs/cgroup
udev     235M   4.0K 235M    1%   /dev
tmpfs    50M 380K   49M    1%   /run
none      5.0M   0   5.0M   0%  /run/lock
none     246M   0   246M  0%   /run/shm
none     100M   0   100M  0%   /run/user
joe@web:~$
Now, make it persistent everytime we boot
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>  <type> <options>       <dump>  <pass>
#/ was on /dev/sda1 during installation
UUID=16c336d9-038e-40a8-a65f-737585b4ab8c  /     ext4    errors=remount-ro  0 1
# swap was on /dev/sda5 during installation
UUID=500baf4a-4ec1-4900-b4d5-e1e1a700e0f1  none        swap  sw  0 0
/dev/fd0  /media/floppy0  auto    rw,user,noauto,exec,utf8 0 0
144.38.220.200:/data/images /remote/images nfs  bg,rw,tcp 0 0
```bash
ejoe@web:~$ sudo mount /remote/images/

ejoe@web:~$ df -h

<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Size</th>
<th>Used</th>
<th>Avail</th>
<th>Use%</th>
<th>Mounted on</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda1</td>
<td>12G</td>
<td>1.3G</td>
<td>9.4G</td>
<td>12%</td>
<td>/</td>
</tr>
<tr>
<td>none</td>
<td>4.0K</td>
<td>0</td>
<td>4.0K</td>
<td>0%</td>
<td>/sys/fs/cgroup</td>
</tr>
<tr>
<td>udev</td>
<td>235M</td>
<td>4.0K</td>
<td>231M</td>
<td>1%</td>
<td>/dev</td>
</tr>
<tr>
<td>tmpfs</td>
<td>50M</td>
<td>380K</td>
<td>49M</td>
<td>1%</td>
<td>/run</td>
</tr>
<tr>
<td>none</td>
<td>5.0M</td>
<td>0</td>
<td>5.0M</td>
<td>0%</td>
<td>/run/lock</td>
</tr>
<tr>
<td>none</td>
<td>246M</td>
<td>0</td>
<td>246M</td>
<td>0%</td>
<td>/run/shm</td>
</tr>
<tr>
<td>none</td>
<td>100M</td>
<td>0</td>
<td>100M</td>
<td>0%</td>
<td>/run/user</td>
</tr>
<tr>
<td>144.38.220.200:/data/images</td>
<td>12G</td>
<td>1.3G</td>
<td>9.4G</td>
<td>12%</td>
<td>/remote/images</td>
</tr>
</tbody>
</table>
```

If we want normal users to have access we need UID synchronization
joe@nfss:/data/images$ sudo addgroup --gid 2000 nacho
Adding group 'nacho' (GID 2000) ...
Done.

joe@nfss:/data/images$ sudo adduser --uid 2000 --gid 2000 nacho
Adding user 'nacho' ...
Adding new user 'nacho' (2000) with group 'nacho' ...
Creating home directory '/home/nacho' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
password: password updated successfully
Changing the user information for nacho
Enter the new value, or press ENTER for the default
   Full Name []:
   Room Number []:
   Work Phone []:
   Home Phone []:
   Other []:
Is the information correct? [Y/n] Y
joe@nfss:/data/images$
Repeat on client system

Adding group `nacho' (GID 2000) ...
Done.

Adding user `nacho' ...
Adding new user `nacho' (2000) with group `nacho' ...
Creating home directory `/home/nacho' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
pwd: password updated successfully
Changing the user information for nacho
Enter the new value, or press ENTER for the default
   Full Name []:
   Room Number []:
   Work Phone []:
   Home Phone []:
   Other []:
Is the information correct? [Y/n] Y
joe@nfss:/data/images$ ls
MADE_FROM_CLIENT  MADE_ON_SERVER
joe@nfss:/data/images$ sudo mkdir lucha
joe@nfss:/data/images$ sudo chown nacho:nacho lucha
joe@nfss:/data/images$ ls -la
total 12
drwxr-xr-x 3 root  root  4096 Jan 20 13:12 ..
drwxr-xr-x 2 nacho nacho  4096 Jan 20 13:30 lucha
-rw-r--r-- 1 root  root  0 Jan 20 13:24 MADE_FROM_CLIENT
-rw-r--r-- 1 root  root  0 Jan 20 13:15 MADE_ON_SERVER
joe@nfss:/data/images

nacho@web:~$ sudo su - nacho
nacho@web:~$ cd /remote/images/
nacho@web:/remote/images$ ls
lucha  MADE_FROM_CLIENT MADE_ON_SERVER
nacho@web:/remote/images$ cd lucha/
nacho@web:/remote/images/lucha$ touch eskeleto.jpg
nacho@web:/remote/images/lucha$ ls -la
 total 8
 drwxr-xr-x 2 nacho nacho  4096 Jan 20 13:32 .
 drwxr-xr-x 3 root  root  4096 Jan 20 13:30 ..
-rw-r--r-- 1 nacho nacho  0 Jan 20 13:32 eskeleto.jpg
nacho@web:/remote/images/lucha$