## **Tutorial Install and Configure NFS on Ubuntu**

### **Step 1: Installing the NFS Server**

For the first step, you should start with setting up the NFS server. Before that, updating the system packages is reasonable. Then, you are ready to install the NFS server. So, run:

**$ sudo apt update**

**$ sudo apt install –y nfs-kernel-server**

### **Step 2: Create a Shared NFS Directory**

To Configure the NFS server, you need to make a directory to be shared by all the client systems. Here we name it ‘’nfs-share’’.

**$ sudo mkdir -p /mnt/nfs\_share**

### **Step 3: Configure Directory Permissions**

When your directory is created, all the client machines should be able to access it easily. So, set the permissions of it by running the following command:

**$ sudo chown -R nobody:nogroup /mnt/nfs\_share/**

### **Step 4: Establish File Permissions**

Configure NFS on Ubuntu is where you can set the file permissions to the ‘’nfs-share’’ directory file.

**$ sudo chmod 777 /mnt/nfs\_share/**

In this way, read, write, and execute permissions will be allocated.

### **Step 5: Grant NFS Access**

To access the NFS server, you need to grant access to the client system. Open “**/etc/exports**” in your favorite editor.

**$ sudo vim /etc/exports**

Then, add the required line depending on whether you prefer to grant access to the entire subnet, single, or multiple clients.

**/mnt/nfs\_share EntireSubnetIP(rw,sync,no\_subtree\_check)**

The functions of the above command are:

* Clients are given read-and-write access to server directories using the “**rw**” option.
* To reply to the client, “**sync**” causes NFS to write modifications. By using this option, clients will always see the host’s actual condition.
* Subtree checking is disabled by “**no subtree check**” When users rename files, the subtree procedure could lead to issues.

Use the command below to grant access to a single client.

**/mnt/nfs\_share client\_IP\_1 (rw,sync,no\_subtree\_check)**

If you prefer multiple clients, you need to specify each client on a separate file as you see below:

**/mnt/nfs\_share client\_IP\_1 (rw,sync,no\_subtree\_check) /mnt/nfs\_share client\_IP\_2 (re,sync,no\_subtree\_check)**

### **Step 6: Export the NFS Share Directory**

As the edits are made on */etc/exports*, you can export all shared directories you registered in that file. To do this, run:

**$ sudo exportfs -a**

Next, type the following command to **restart the NFS Kernel Server** to apply the changes to the configuration.

**$ sudo systemctl restart nfs-kernel-server**

### **Step 7: Grant NFS Access through the Firewall [only if you are using UFW]**

In this step, you are going to allow clients to access the server if you are using UFW. In this way, accessing and mounting the shared directory will be possible.

**$ sudo ufw allow from [clientIP or clientSubnetIP] to any port nfs**

Also, if the firewall is turned off, enable it:

**$ sudo ufw enable**

Then, you can use the command below to check the changes:

**$ sudo ufw status**

When you see that the traffic in the port 2049 (default file share) is allowed in your output, it means you’re all set.

### **Step 8: Set Up the NFS Clients on Ubuntu**

So far, you installed the NFS server and exported the shares. Let’s configure the clients and mount the remote file system. Run the commands below to enable NFS on client machines and install the NFS common package.

To install the NFS client on **Debian** and **Ubuntu**:

**$ sudo apt update**

**$ sudo apt install nfs-common**

To install the NFS client on [**CentOS**](https://operavps.com/blog/what-is-centos/) and [**Fedora**](https://operavps.com/blog/what-is-fedora/):

**$ sudo yum install nfs-utils**

### **Step 9: Create a Mount Point on the NFS Client System**

Since the client machine needs a mount point for the shared directory exported by the server, use the following command to create a directory.

**$ sudo mkdir -p /mnt/nfsdir\_client**

Next, run the command below to mount the shared directory on the client system.

**$ sudo mount server\_IP:/mnt/nfs\_share /mnt/nfsdir\_client**

Verify if the mounting folder is done successfully by using the **df -h** command.

Run the command below to unmount the shared folder when you no longer need it.

**$ sudo umount /mnt/nfsdir\_client**

### **Step 10: Test the NFS Share on the Client System**

You can create a few files in the NFS share directory located on the server to check that your NFS setup is working correctly.

**$ cd /mnt/nfs\_share/$ touch file1.txt file2.txt file3.txt**

To check whether the files exist in the directory or not, head back to the NFS client system:

**$ ls -l /mnt/nfsdir\_client/**

On your Output, check if you can access the files you created on the NFS server or not. If files are successfully accessed on the client via the NFS server you’re all set.

### **How to Mount NFS Shared Directories on OS Boot**

You can run the command below to let the folders stay mounted after you restart the machine. Just add them to the */etc/fstab* file.

**$ sudo vim /etc/fstab**

Next, copy the following line to the bottom of the file, replacing “host\_IP” with the actual IP address of the host:  
 *host\_IP:/mnt/nfsdir /mnt/nfsdir\_client nfs auto,nofail,noatime,nolock,intr,tcp,actimeo=1800 0 0*

In this way, all folders on the list will be mounted automatically on every boot. You can add the line for any folder you need.