

配置 tftp 服务的步骤:

1. 若/etc/xinetd.d/下存在 tftp, 执行下列操作

1) 关闭 xinetd 服务

```
$sudo service xinetd stop
```

2) 删除 tftp 文件

```
$sudo rm /etc/xinetd.d/tftp
```

3) 启动 xinetd 服务

```
$sudo service xinetd start
```

2. 安装 tftp 客户端和服务端

```
$sudo apt-get install tftp-hpa
```

```
$sudo apt-get install tftpd-hpa
```

3. 修改 tftpd-hpa 配置文件

```
$vim /etc/default/tftpd-hpa
```

修改 “/var/lib/tftpboot” 为 “/tftpboot”

修改 “--secure” 为 “--secure -c” 允许上传新文件

4. 若/tftpboot 不存在, 创建该目录

```
$sudo mkdir /tftpboot
```

```
$sudo chmod 777 /tftpboot
```

5. 重启 tftpd-hpa 服务

```
$sudo service tftpd-hpa restart
```

若服务重启成功, 能查看到相应的进程

```
$ps -ef |grep in.tftpd
```

至此 tftp 服务已经安装完成了, 下面可以对其进行一下测试。(假设在当前目录下有一个测试文件 test.txt)

```
$tftp 127.0.0.1
```

```
tftp> put test.txt
```

```
Sent 1018 bytes in 0.0 seconds
```

```
tftp> get test.txt
```

```
Received 1018 bytes in 0.1 seconds
```

```
tftp> quit
```

```
$
```

通过 `get` 命令，可以把当前目录下的 `test.txt` 文件，通过 `tftp` 上传到它的服务文件目录。这时，在 `/tftpboot` 下面会出现 `test.txt` 文件。通过 `put` 命令，可以从 `/tftpboot` 下，下载 `test.txt` 文件。这样就验证了 `tftp` 服务配置的正确性。当文件上传与下载结束后，可以通过 `quit` 命令退出。

在 `ubuntu` 下安装、配置 `nfs` 服务的步骤如下：

### 1、安装 `nfs`

`Ubuntu` 上默认是没有安装 `nfs` 服务器的，因此我们首先安装 `nfs` 服务器端：

```
$sudo apt-get install nfs-kernel-server
```

### 2、配置 `/etc/exports`

`nfs` 允许挂载的目录及权限在文件 `/etc/exports` 中进行了定义。

例如，我们要将根目录下的 `rootfs` 目录共享出来，那么我们需要在 `/etc/exports` 文件末尾添加如下一行：`/rootfs *(rw,sync,no_root_squash)`

其中：`/rootfs` 是要共享的目录，`*`代表允许所有的网络段访问，`rw` 是可读写权限，`sync` 是资料同步写入内存和硬盘，`no_root_squash` 是 `nfs` 客户端分享目录使用者的权限，如果客户端使用的是 `root` 用户，那么对于该共享目录而言，该客户端就具有 `root` 权限。

其它 `nfs` 常用的参数有：

`ro` 只读访问

`rw` 读写访问 `sync` 所有数据在请求时写入共享

`async` `nfs` 在写入数据前可以响应请求

`secure` `nfs` 通过 1024 以下的安全 `TCP/IP` 端口发送

`insecure` `nfs` 通过 1024 以上的端口发送

`wdelay` 如果多个用户要写入 `nfs` 目录，则归组写入（默认）

`no_wdelay` 如果多个用户要写入 `nfs` 目录，则立即写入，当使用 `async` 时，无需此设置。

`hide` 在 `nfs` 共享目录中不共享其子目录

`no_hide` 共享 `nfs` 目录的子目录

`subtree_check` 如果共享 `/usr/bin` 之类的子目录时，强制 `nfs` 检查父目录的权限（默认）

`no_subtree_check` 和上面相对，不检查父目录权限

`all_squash` 共享文件的 `UID` 和 `GID` 映射匿名用户 `anonymous`，适合公用目录。

`no_all_squash` 保留共享文件的 `UID` 和 `GID`（默认）

`root_squash` `root` 用户的所有请求映射成如 `anonymous` 用户一样的权限（默认）

`no_root_squash` `root` 用户具有根目录的完全管理访问权限

`anonuid=xxx` 指定 `nfs` 服务器 `/etc/passwd` 文件中匿名用户的 `UID`

`anongid=xxx` 指定 `nfs` 服务器 `/etc/passwd` 文件中匿名用户的 `GID`

### 3、重启服务

```
$sudo /etc/init.d/portmap restart
```

```
$sudo /etc/init.d/nfs-kernel-server restart
```

### 4、测试 `nfs`

此时可以运行以下命令来显示一下共享出来的目录：

```
$showmount -e
```

或者可以使用以下命令把它挂载在本地磁盘上，例如将/rootfs 挂载到/mnt 下：

```
$ sudo mount -t nfs localhost:/rootfs /mnt
```

可以运行 df 命令查看是否挂载成功。查看后可以使用以下命令卸载：

```
$ sudo umount /mnt
```