

# 构建嵌入式开发环境

# 课程指数

---

■ 重要程度：

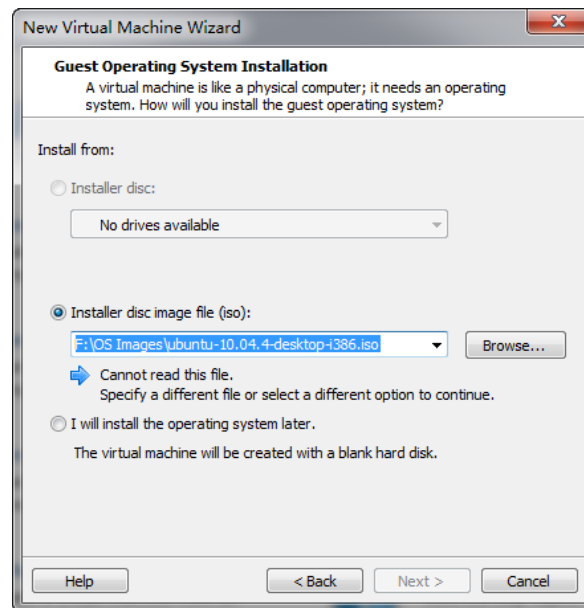


■ 掌握程度：

了解 掌握 熟练 精通

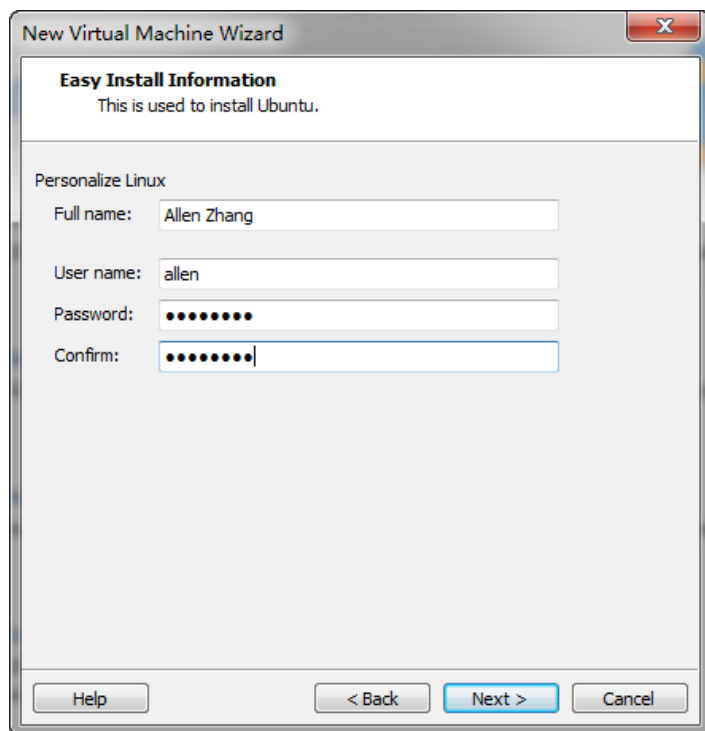
# Ubuntu 的安装

- 在虚拟机中安装Ubuntu
  - 选择使用映像文件安装



# Ubuntu 的安装

- 输入用户名及密码，更改虚拟机文件的安装位置



New Virtual Machine Wizard

**Easy Install Information**  
This is used to install Ubuntu.

Personalize Linux

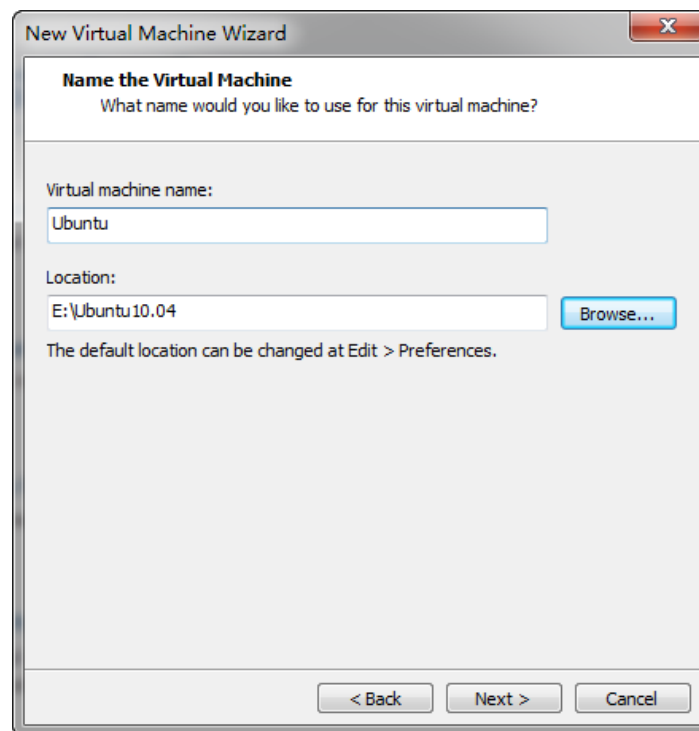
Full name: Allen Zhang

User name: allen

Password: ●●●●●●

Confirm: ●●●●●●

Help < Back Next > Cancel



New Virtual Machine Wizard

**Name the Virtual Machine**  
What name would you like to use for this virtual machine?

Virtual machine name: Ubuntu

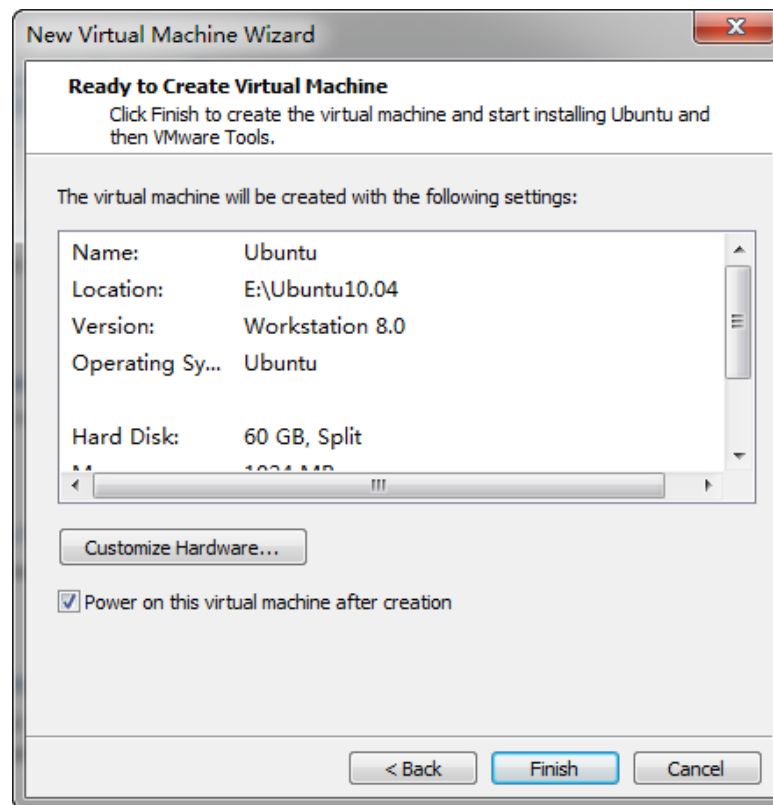
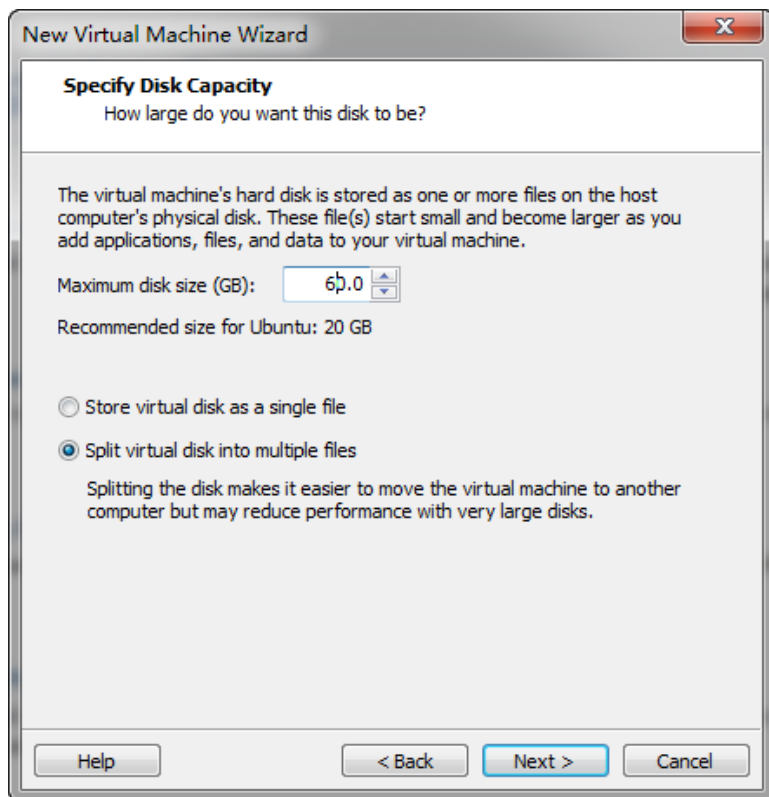
Location: E:\Ubuntu10.04 Browse...

The default location can be changed at Edit > Preferences.

< Back Next > Cancel

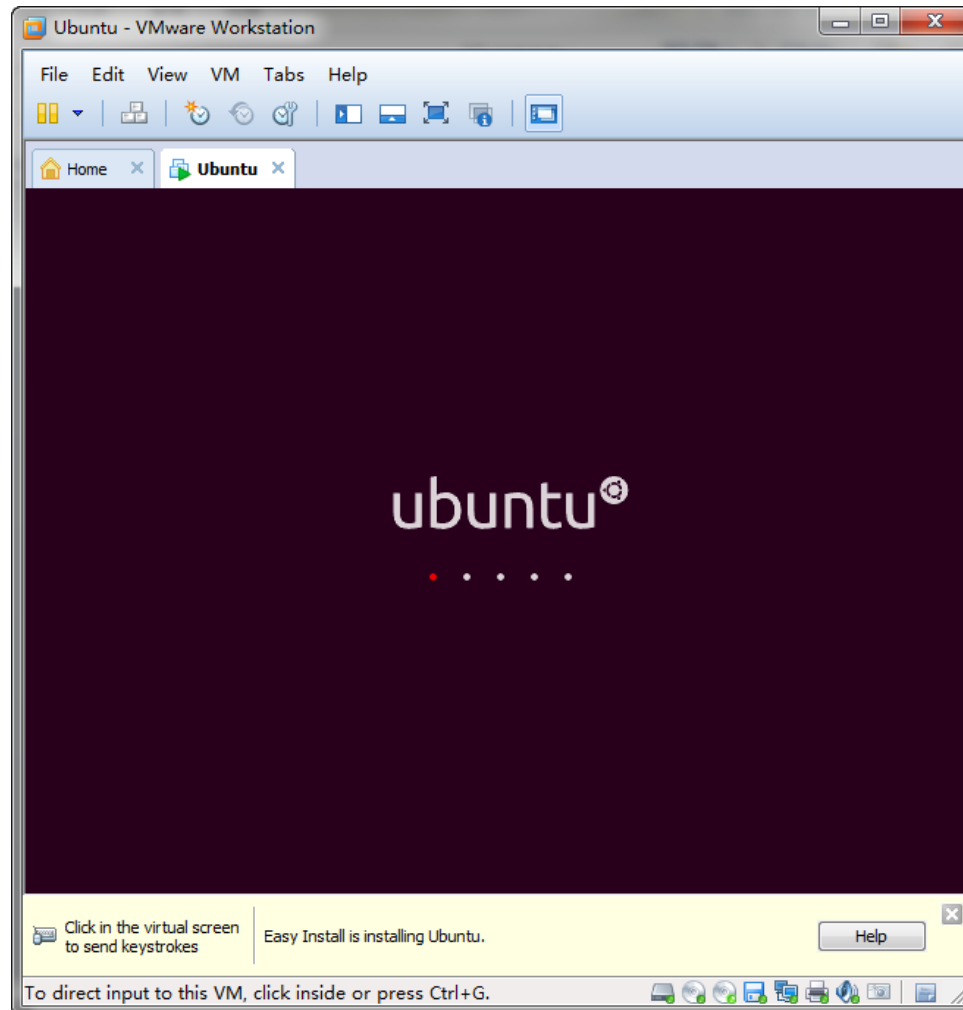
# Ubuntu 的安装

## ■ 指定硬盘容量大小



# Ubuntu 的安装

## ■ 开始安装



# 安装嵌入式开发工具链

- 开启虚拟机的共享文件夹功能，将工具链复制到虚拟机的ubuntu系统中，解压安装

- 为工具链建立工作目录

lh@ubuntu:~\$ mkdir CrossTools

- 在该目录下建立source和tools目录

lh@ubuntu:~/CrossTools\$ mkdir tools source

- 将工具链复制到source目录下

lh@ubuntu:~/CrossTools\$ cp /mnt/hgfs/share/arm-linux-gcc-4.4.3.tar.gz source/

- 进入source目录，将工具链解压缩到tools目录下

lh@ubuntu:~/CrossTools\$ cd source/

lh@ubuntu:~/CrossTools/source\$ tar xzvf arm-linux-gcc-4.4.3.tar.gz -C ../tools/

# 安装嵌入式开发工具链

---

## ■ 配置环境变量

- 修改~/.bashrc文件，加入相应的环境变量

```
lh@ubuntu:~/CrossTools$ vi /home/lh/.bashrc
```

- 在该文件的最后加入下面的一行

```
export PATH=$PATH:/home/lh/CrossTools/tools/opt/FriendlyARM/toolschain/4.4.3/bin
```

- 使用source命令在shell中执行~/.bashrc文件中的内容

```
lh@ubuntu:~/CrossTools$ source ~/.bashrc
```

- 在终端中输入arm双击Tab键，查看是否能找到arm工具链



- 编译一个嵌入式版本的HelloWorld程序
  - 编辑一个HelloWorld的C程序，使用arm-linux-gcc进行编译

```
lh@ubuntu:~/WorkSpace$ arm-linux-gcc hello.c -o hello
```

```
lh@ubuntu:~/WorkSpace$ file hello
```

```
hello: ELF 32-bit LSB executable, ARM, version 1 (SYSV),  
dynamically linked (uses shared libs), for GNU/Linux  
2.6.32, not stripped
```

# 安装NFS服务

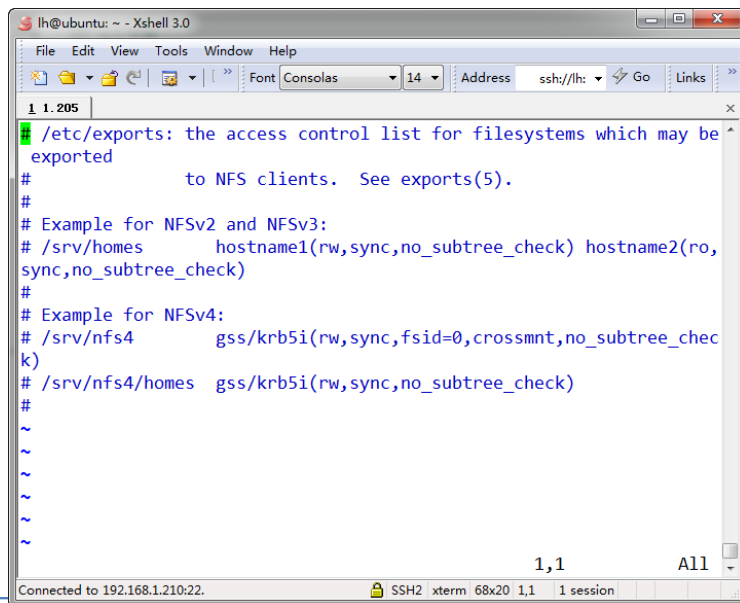
## ■ 什么是NFS

□ NFS ( Network File System ) 网络文件系统，是文件系统中的一种。NFS允许一个系统在网络上与他人共享目录和文件。通过使用NFS，用户和程序可以像访问本地文件一样访问远端系统上的文件。

## □ 安装NFS服务

```
lh@ubuntu:~$ sudo apt-get install nfs-kernel-server
```

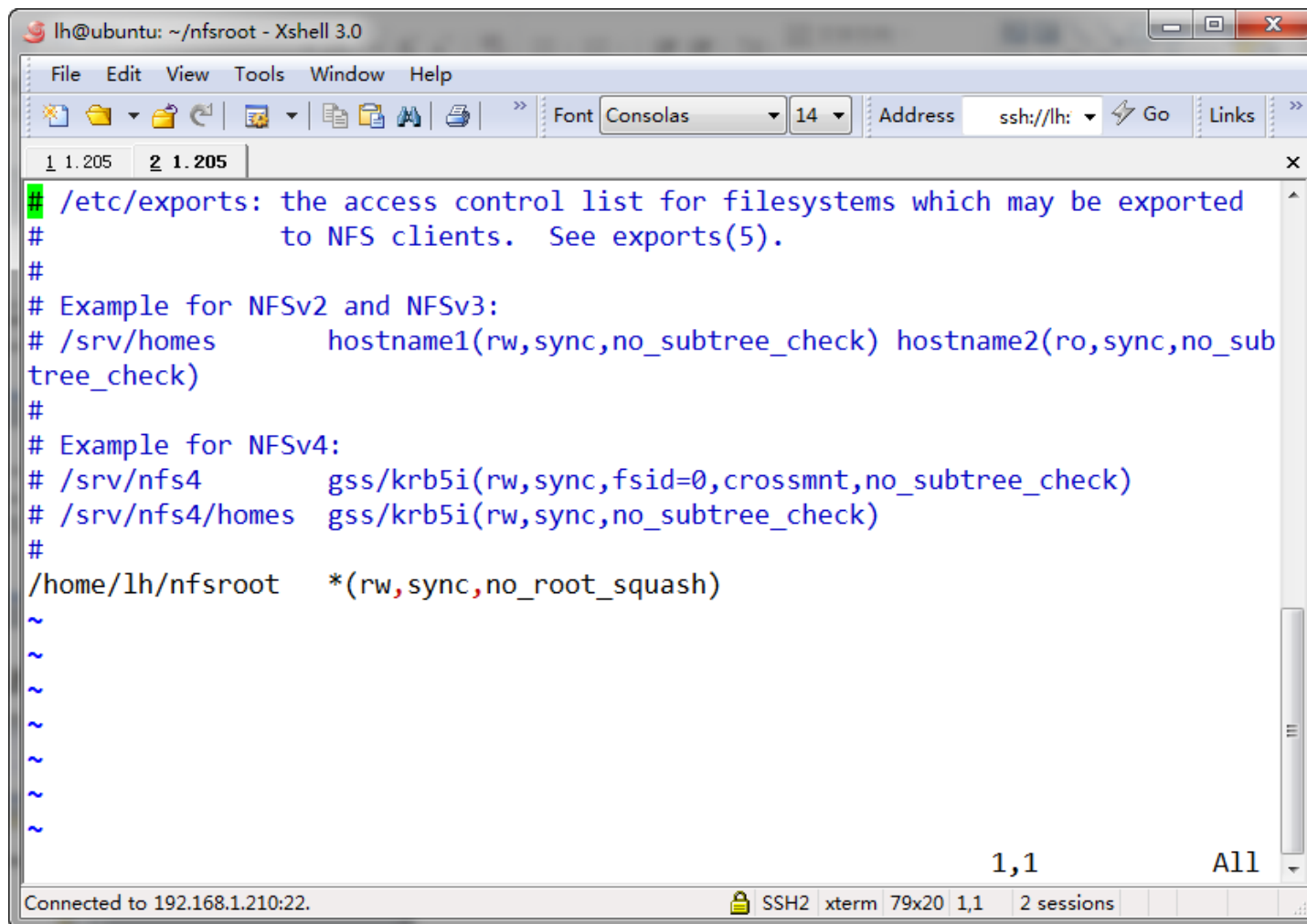
□ 配置NFS，  
修改/etc/exports配置文件



```
lh@ubuntu: ~ - Xshell 3.0
File Edit View Tools Window Help
Font Consolas 14 Address ssh://lh: Go Links
1 1.205
/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_che
k)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
~
~
~
~
~
~
1,1 All
Connected to 192.168.1.210:22. SSH2 xterm 68x20 1,1 1 session
```

# 安装NFS服务

- 在/etc/exports文件中添加NFS服务目录以及相应的权限



The screenshot shows an Xshell 3.0 terminal window with the title bar 'lh@ubuntu: ~/nfsroot - Xshell 3.0'. The window contains the content of the /etc/exports file. The text is as follows:

```
# /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_sub
tree_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)
#
/home/lh/nfsroot    *(rw,sync,no_root_squash)
~
~
~
~
~
~
~
```

At the bottom of the terminal window, the status bar indicates 'Connected to 192.168.1.210:22.' and 'SSH2 xterm 79x20 1,1 2 sessions'.

## ■ NFS配置中的权限说明

- **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\_squas** 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

# 安装NFS服务

## ■ 启动NFS服务

```
lh@ubuntu:~/nfsroot$ sudo /etc/init.d/nfs-kernel-server start
[sudo] password for lh:
* Exporting directories for NFS kernel daemon...
exportfs: /etc/exports [1]: Neither 'subtree_check' or 'no_subtree_check' speci
fied for export "*/home/lh/nfsroot".
    Assuming default behaviour ('no_subtree_check').
    NOTE: this default has changed since nfs-utils version 1.0.x
```

```
[ OK ]
[ OK ]
```

```
* Starting NFS kernel daemon
lh@ubuntu:~/nfsroot$
```

## ■ 关闭NFS服务

```
lh@ubuntu:~/nfsroot$ sudo /etc/init.d/nfs-kernel-server stop
* Stopping NFS kernel daemon
* Unexporting directories for NFS kernel daemon...
lh@ubuntu:~/nfsroot$
```

```
[ OK ]
[ OK ]
```

## ■ 重启NFS服务

```
lh@ubuntu:~/nfsroot$ sudo /etc/init.d/nfs-kernel-server restart
* Stopping NFS kernel daemon
* Unexporting directories for NFS kernel daemon...
* Exporting directories for NFS kernel daemon...
exportfs: /etc/exports [1]: Neither 'subtree_check' or 'no_subtree_check' speci
fied for export "*/home/lh/nfsroot".
    Assuming default behaviour ('no_subtree_check').
    NOTE: this default has changed since nfs-utils version 1.0.x
```

```
[ OK ]
[ OK ]
```

```
* Starting NFS kernel daemon
lh@ubuntu:~/nfsroot$
```

# 安装NFS服务

- 测试NFS服务是否可以正常使用，通过mount命令挂载

- 在/mnt目录下创建nfs挂载目录

```
lh@ubuntu:~$ sudo mkdir /mnt/nfs
```

- 使用mount命令挂载

```
lh@ubuntu:~$ sudo mount -t nfs localhost:/home/lh/nfsroot/ /mnt/nfs/
```

- 卸载命令

```
lh@ubuntu:~$ umount /mnt/nfs/
```

- 测试：在~/nfsroot目录下建立文件或目录后，再到/mnt/nfs目录下查看该文件或目录是否存在，修改文件或目录名，查看两个目录内的内容是否一致？

## ■ 什么是TFTP服务

□ TFTP ( Trivial File Transfer Protocol,简单文件传输协议 ) 是TCP/IP协议族中的一个用来在客户机与服务器之间进行简单文件传输的协议，它基于UDP协议实现。提供不复杂、开销不大的文件传输服务。端口号为69。

## □ 安装TFTP服务器与客户端

```
lh@ubuntu:~$ sudo apt-get install tftp-hpa tftpd-hpa xinetd
```

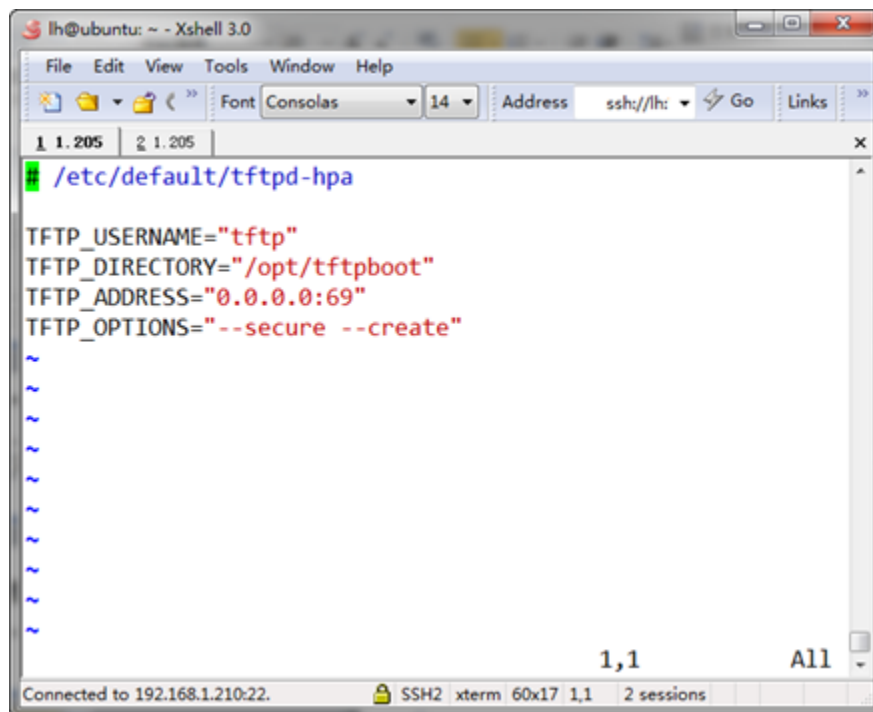
# 安装TFTP服务

## ■ 创建TFTP服务目录，并修改访问权限

```
lh@ubuntu:~$ sudo mkdir /opt/tftpboot
```

```
lh@ubuntu:~$ sudo chmod 777 /opt/tftpboot/
```

## ■ 修改/etc/default/tftpd-hpa配置文件



The screenshot shows an Xshell 3.0 terminal window with a menu bar (File, Edit, View, Tools, Window, Help) and a toolbar. The address bar shows 'ssh://lh:'. The terminal content shows the file path `/etc/default/tftpd-hpa` and its configuration:

```
TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/opt/tftpboot"
TFTP_ADDRESS="0.0.0.0:69"
TFTP_OPTIONS="--secure --create"
```

At the bottom of the terminal, it says "Connected to 192.168.1.210:22." and "SSH2 xterm 60x17 1,1 2 sessions".



# 安装TFTP服务

---

## ■ 重新启动TFTP服务

```
lh@ubuntu:~$ sudo service tftpd-hpa restart
```

## ■ tftp命令测试TFTP服务

```
lh@ubuntu:~/WorkSpace$ ls
hello.c
lh@ubuntu:~/WorkSpace$ tftp localhost
tftp> get hello
tftp> q
lh@ubuntu:~/WorkSpace$ ls
hello  hello.c
```

注意：如果不出现错误，或长时间无反应，则表示成功



# 联航精英训练营

United Elite Training Camp

## Thank you