

四川大学计算机学院、软件学院

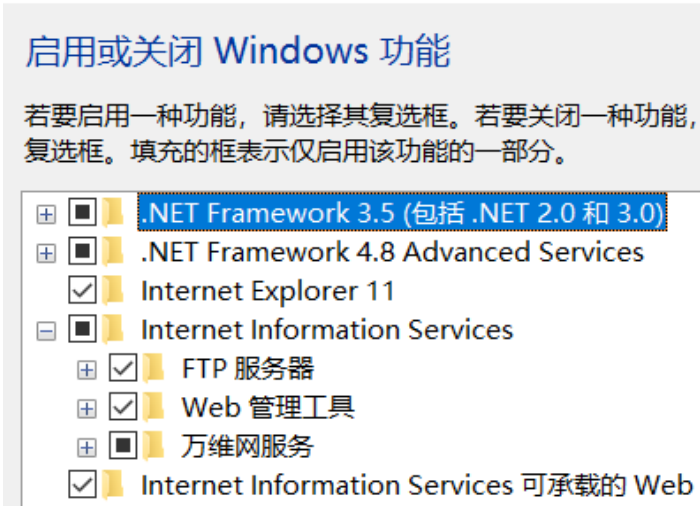
# 实验报告

学号: 2017141051019 姓名: 王崇智 专业: 计算机科学与技术 班级: 173040104 第 7 周

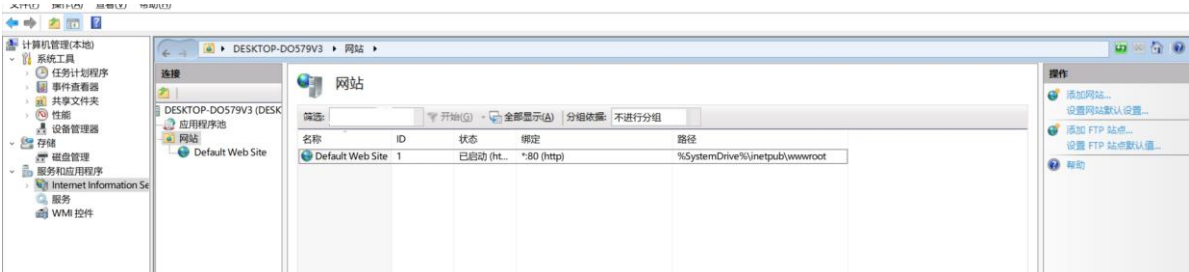
课程名称	计算机网络	实验课时	2
实验项目	FTP 服务器的配置与分析	实验时间	2019/10/15
实验目的	本实验通过配置 FTP 服务器(安装软件不限), 完成对 FTP 协议及相关技术的掌握。实验过程中, 需要完成对以下知识点的掌握:  1) 掌握 FTP 服务器的配置方法;  2) 了解 FTP 服务器的原理。		
实验环境	WIN10, linux vps centos7		

一、 FTP 服务器的安装和配置

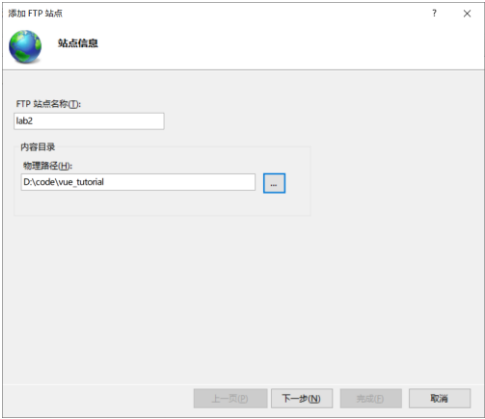
在程序和功能中确认打开 ftp 服务

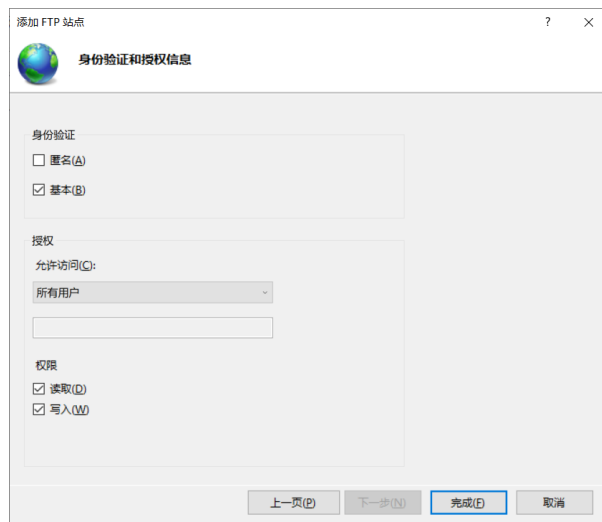
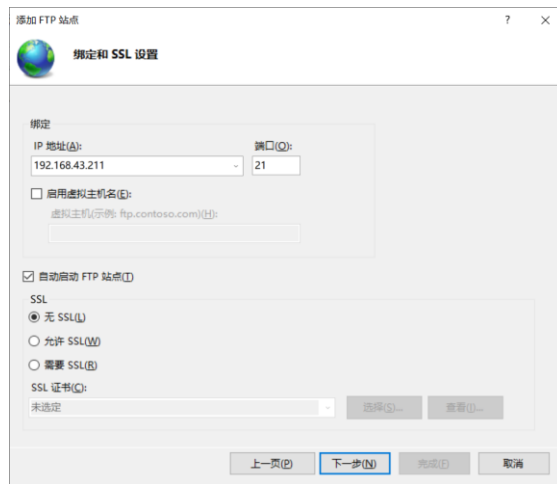


之后进入 IIS 配置网页



右键网站，添加 FTP 站点，或者点击右侧的添加 FTP 站点操作即可





点击完成后，在浏览器中输入 `ftp://192.168.43.211` 进行测试



但此时应该还未注册账户，返回界面继续寻求解决方案；

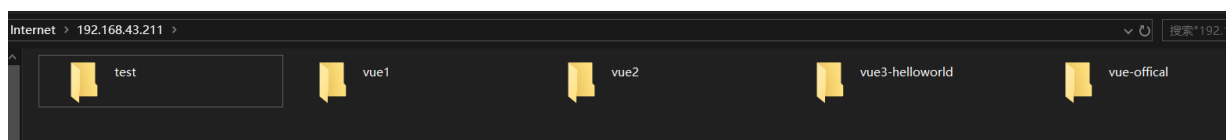
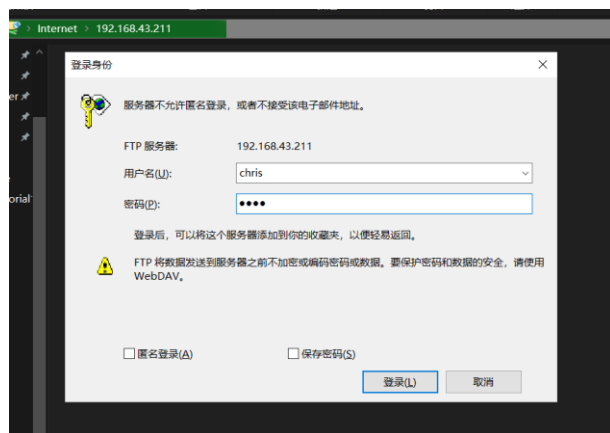
查询资料后发现，因为之前身份验证与授权那一步的时候，选择的身份验证方式为：基本。大致可以理解为允许在本机注册过的 user 进行验证访问。即 win 最基本的登陆账户即可满足需求。

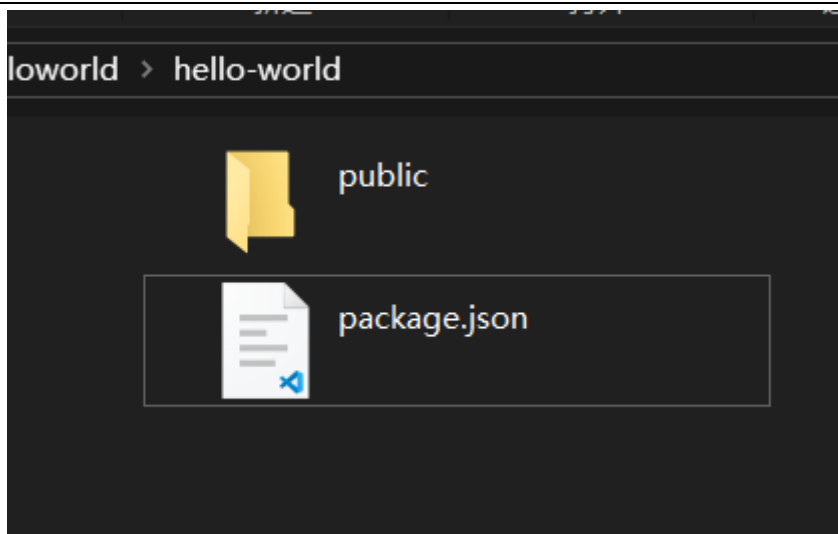


如图所示, ftp 基本配置成功。可以通过 chrome 浏览器打开文件夹在服务器中所在位置。

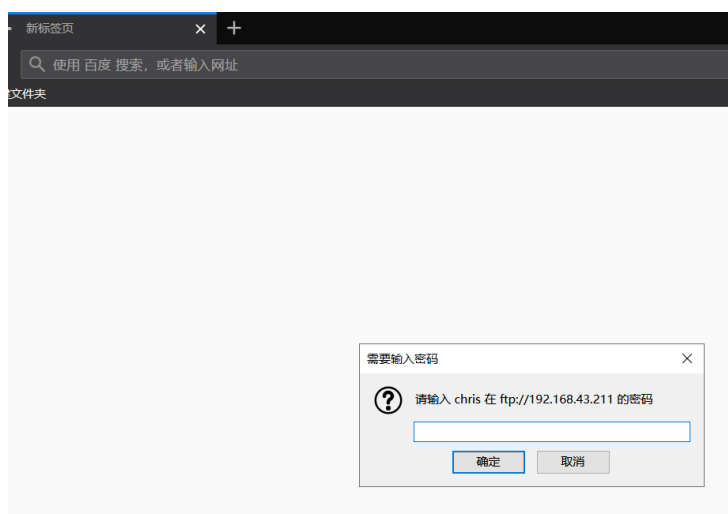
## 二、进行文件下载尝试

下利用文件浏览器登陆服务器

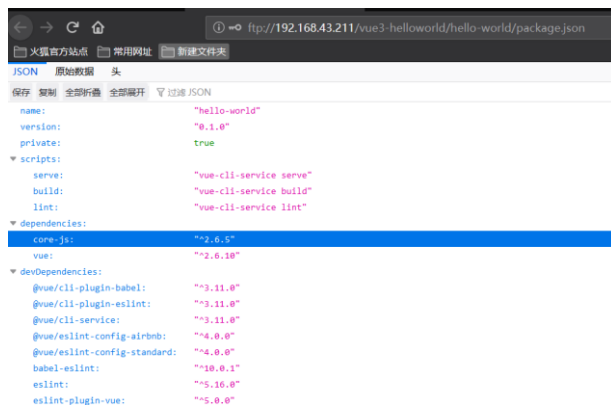




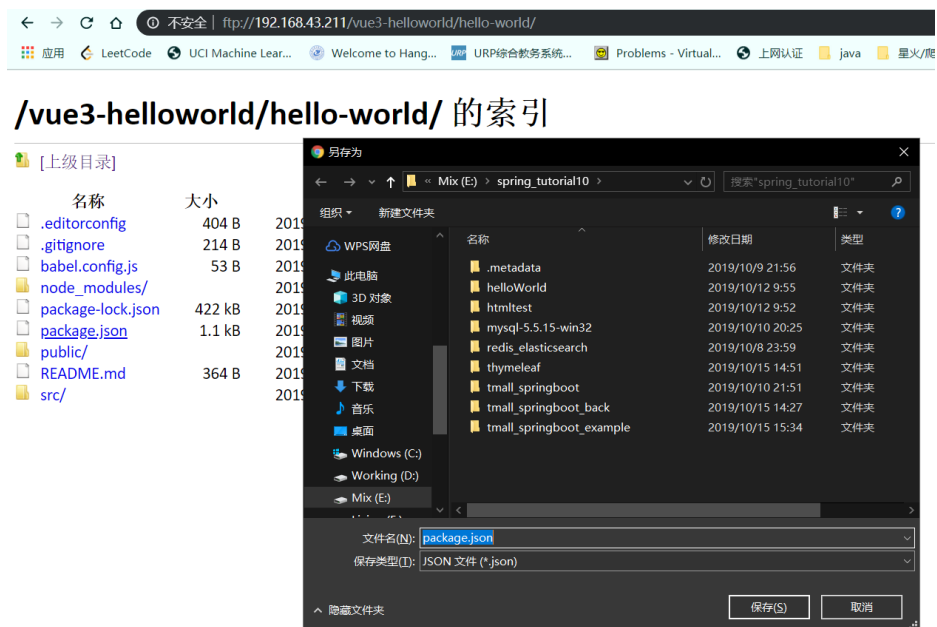
双击文件，



选择以火狐浏览器方式打开后呈现如上页面，

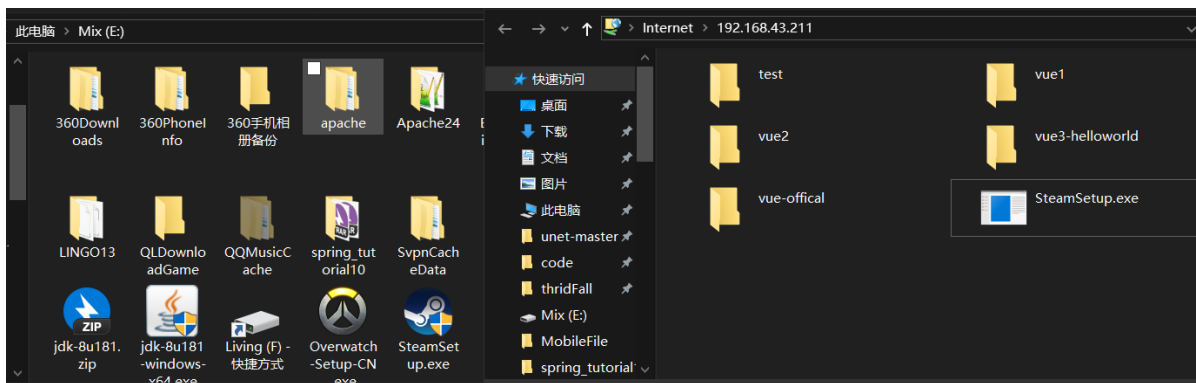


成功，即代表文件下载并展示成功。另一方面，如果用浏览器进行服务器的登陆



点击同一文件后，即跳出下载请求，文件下载功能得到成功实现。

现尝试文件上传



打开本地文件夹与 ftp 服务器位置，尝试将 .exe 文件拖拽至服务器内，因为本身主机用户，即默认 user 并未切换，所以权限许可已经得到满足。当然，作为测试的例子，上传的文件大小有限，如果需要上传至服务器的文件过大，则最好预先进行压缩工作以加快上传速度。

由于本次实验需要对不同用户配置不同权限以体现服务器特性的需求，现尝试在 pc 中注册新的用户。参考 <https://blog.csdn.net/pengpengpeng85/article/details/84977436>

发现 win10 系统的计算机管理中并没有本地用户和组

Mazda\_GU

Microsoft 代理 | 审阅人

回复日期 2018/07/31

您好，

了解到您的问题。

Windows 10 家庭版需要升级到专业版后才会计算机管理中有本地用户和组项目的。

微软账户和本地账户是可以关联使用的，关联后使用微软账户登录系统也是具有管理员权限的。

方法如下：

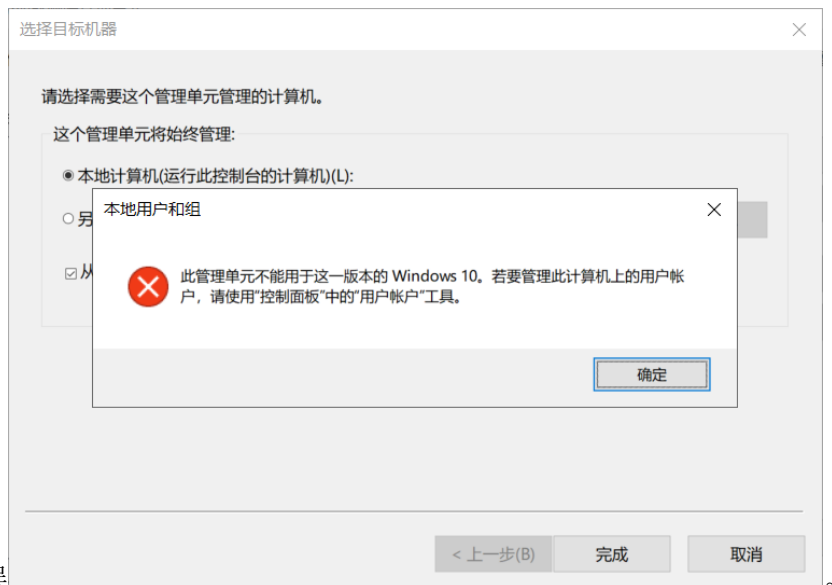
使用本地管理员登录>>WIN+X+N打开设置>>账户>>账户信息（旧版本可能为你的账户）>>选择“改用 Microsoft 帐户登录”  
然后根据系统提示操作即可。

希望以上信息可以帮助到您。



，继续寻求新的解决方案。

<https://jingyan.baidu.com/article/f3ad7d0f55fd4309c3345bb5.html>但是按照该教程的方



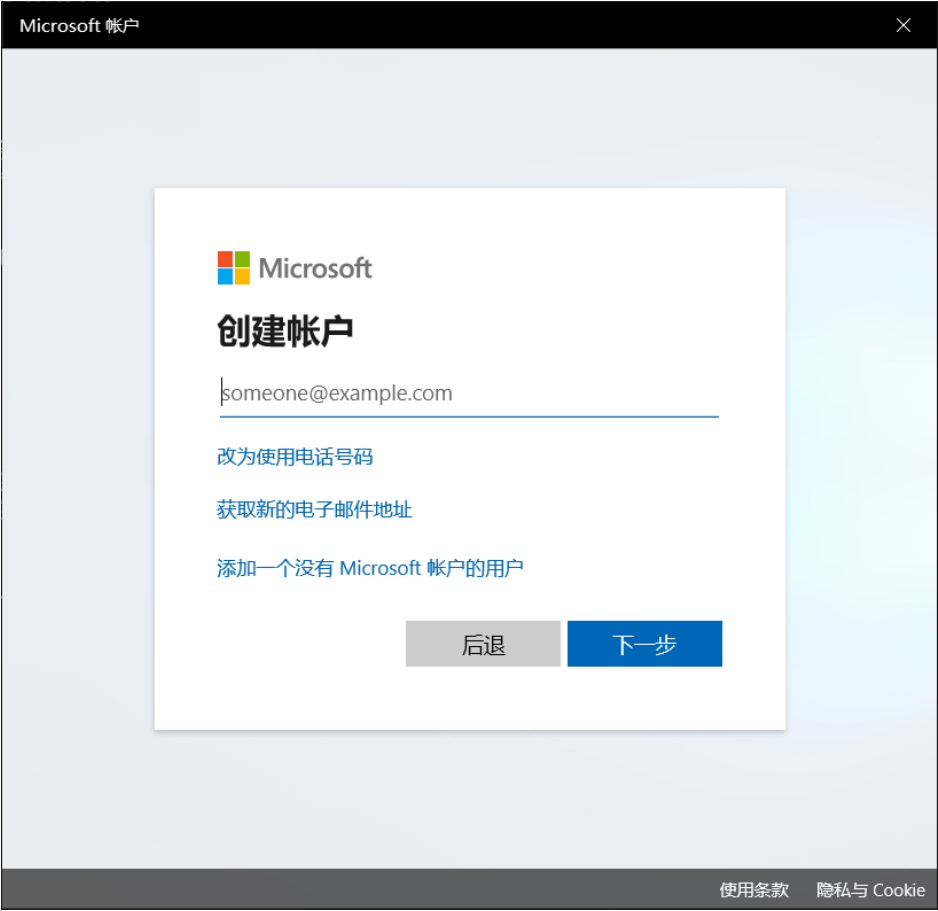
法仍然会报出错误

于是向控制面板寻求帮助



点击其他用户下方的加号





点击添加一个没有 ms 帐户的用户

### 为这台电脑创建一个帐户

如果你想使用密码，请选择自己易于记住但别人很难猜到的内容。

谁将会使用这台电脑？

ftproot

确保密码安全。


•••••

•••••

如果你忘记了密码

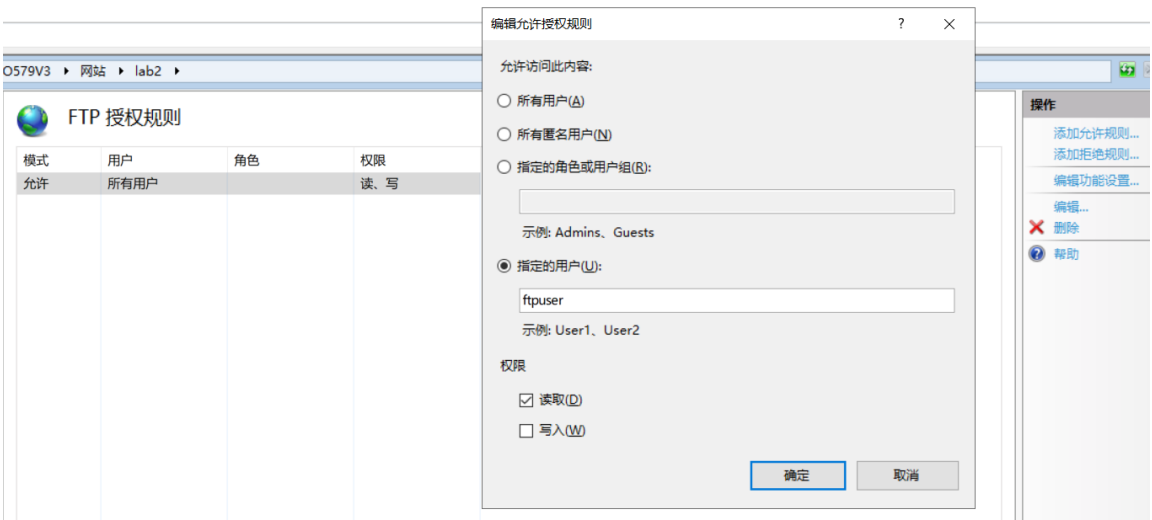
你出生城市的名称是什么？

允许不是你家庭一员的用户使用自己的帐户来登录。这样不会将其添加到你的家庭中。

 将其他人添加到这台电脑

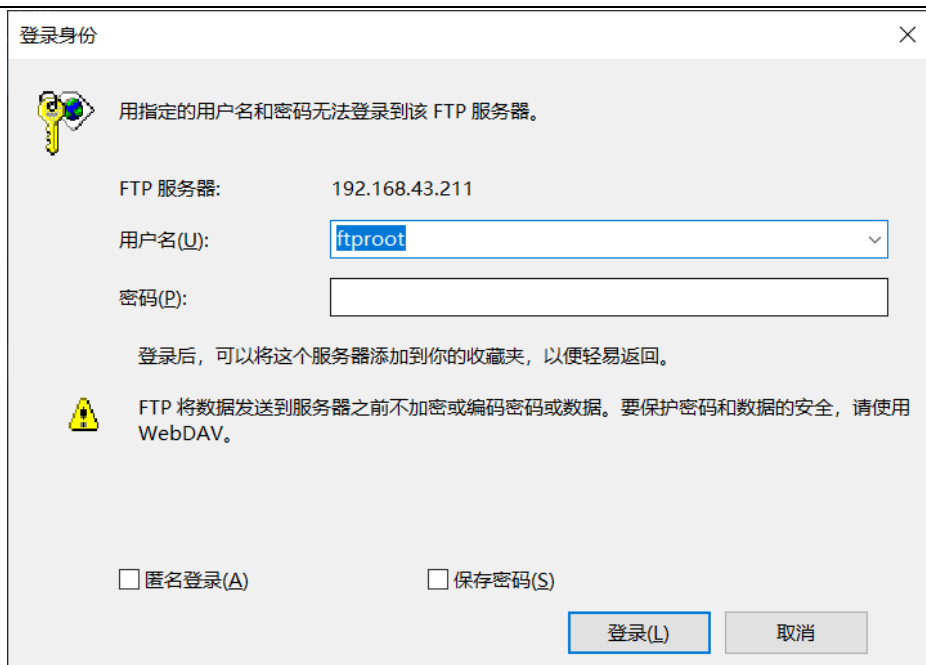
 **ftproot**  
本地帐户

效果如下，此时已将用户添加至本地。下面即可利用新用户并配合 ftp 授权规则修改进行验证。

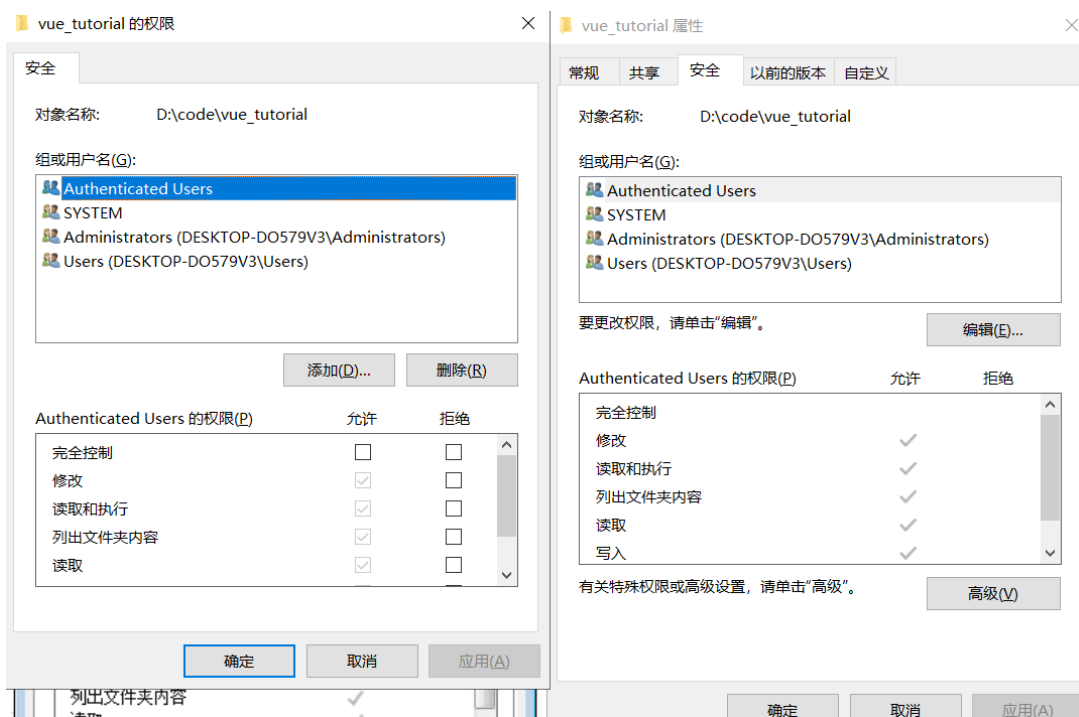


将该用户对原服务器的权限限制在读取上

之后再用浏览器登陆服务器并输入原有账户时，提示“用指定的用户名和密码无法登录到该 FTP 服务器”，而利用新注册的账户登陆时



仍然报错, 无法登陆。试图寻找新的解决方案。



找到 ftp 服务器配置的物理路径, 添加用户权限

选择用户或组

选择此对象类型(S):

用户、组或内置安全主体

对象类型(Q)...

查找位置(F):

DESKTOP-DO579V3

位置(L)...

输入对象名称来选择(示例)(E):

ftproot

检查名称(C)

高级(A)...

确定

取消

对象名称: D:\code\vue\_tutorial

组或用户名(G):

Authenticated Users

SYSTEM

Administrators (DESKTOP-DO579V3\Administrators)

ftproot (DESKTOP-DO579V3\ftproot)

Users (DESKTOP-DO579V3\Users)

添加(D)...

删除(R)

ftproot 的权限(P)

	允许	拒绝
完全控制	<input checked="" type="checkbox"/>	<input type="checkbox"/>
修改	<input checked="" type="checkbox"/>	<input type="checkbox"/>
读取和执行	<input checked="" type="checkbox"/>	<input type="checkbox"/>
列出文件夹内容	<input checked="" type="checkbox"/>	<input type="checkbox"/>
读取	<input checked="" type="checkbox"/>	<input type="checkbox"/>

确定

取消

应用(A)

vue\_tutorial 的权限

安全

对象名称: D:\code\vue\_tutorial

组或用户名(G):

Authenticated Users

SYSTEM

Administrators (DESKTOP-DO579V3\Administrators)

ftproot (DESKTOP-DO579V3\ftproot)

Windows 安全

设置安全信息:

D:\code\vue\_tutorial\test\node\_modules\ajv\lib\cache.js

取消

列出文件夹内容

读取

☒☐

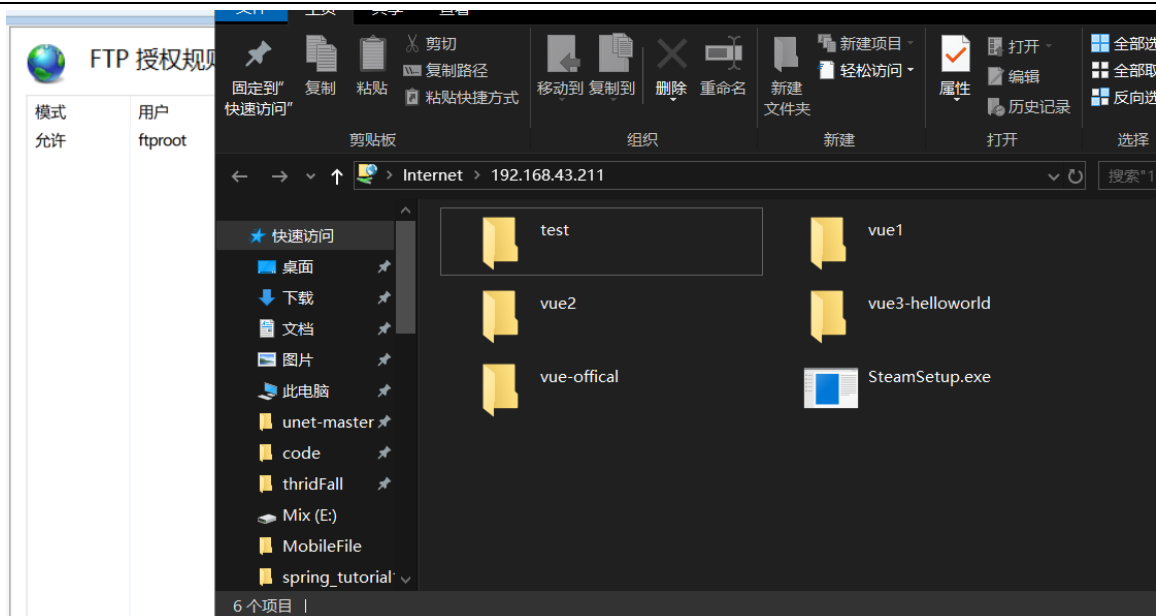
☒☐

确定

取消

应用(A)

之后可以成功进入



但是！折腾了半天才发现

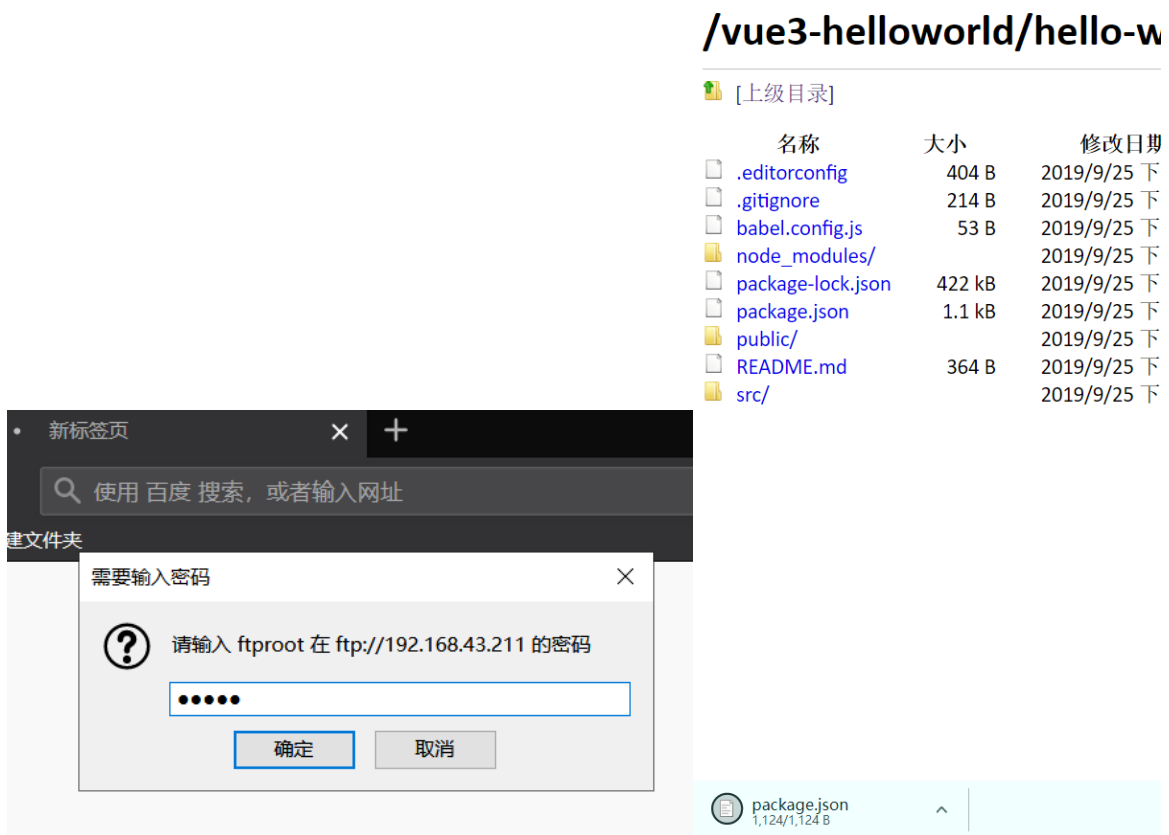


## FTP 授权规则

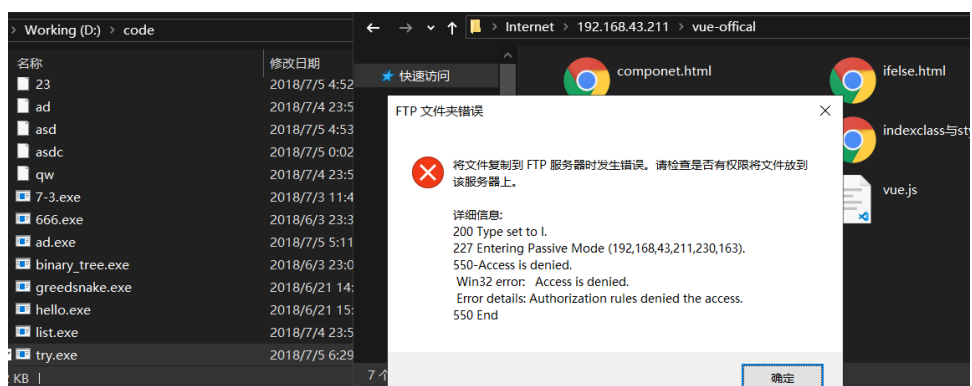
模式	用户	角色	权限
允许	ftproot		读取

自己最初再定义授权规则的时候，将本地用户名 ftproot 在唯一的授权规则中错误的写成了 ftpuser，所以之前的很多操作都是可以避免的。

现在在只允许新用户有读取权限的情况下测试其写入功能。



发现仍然可以实现下载功能，



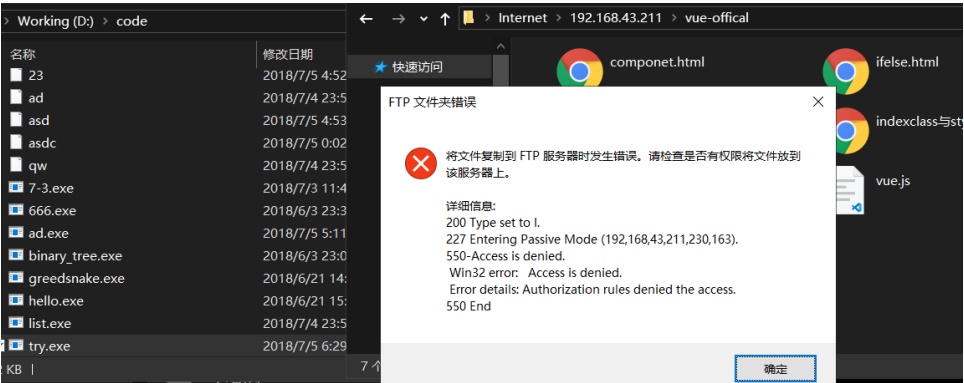
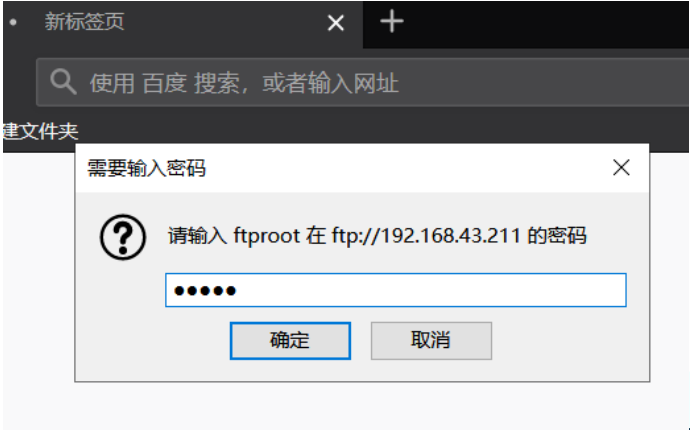
但是测试其上传功能，如上传 `try.exe` 至 `ftp` 根目录时报错，即说明该用户对 `ftp` 服务器缺少特定的缺陷，即之前对 `ftp` 的用户配置服务实现。

之后实验的时候因为切换网络环境，`ftp` 绑定的 `ip` 需要做出相应的更改

# /vue3-helloworld/hello-w

[上级目录]

名称	大小	修改日期
.editorconfig	404 B	2019/9/25 下
.gitignore	214 B	2019/9/25 下
babel.config.js	53 B	2019/9/25 下
node_modules/		2019/9/25 下
package-lock.json	422 kB	2019/9/25 下
package.json	1.1 kB	2019/9/25 下
public/		2019/9/25 下
README.md	364 B	2019/9/25 下
src/		2019/9/25 下



编辑网站绑定

类型(I): ftp IP 地址(I): 192.168.43.211 端口(O): 21

主机名(H): www.ftptutorial.com

示例: www.contoso.com 或 marketing.contoso.com

确定 取消

ftp://192.168.43.211

需要授权

ftp://192.168.43.211 要求您输入用户名和密码。

用户名: chris

密码: ●●●●

确定 取消

但实际上发现，无法进行正常的登陆功能，进入 logs 文件目录



```

9-10-15 11:17:25 192.168.43.211 - 192.168.43.211 21 ControlChannelClosed - - 0 0 0130d9a4-6b30-4077-ae07-364da3106cc7 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 ControlChannelOpened - - 0 0 a4a3f8e8-34cd-4c66-8db7-84ed8698f726 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 USER anonymous 530 11001 37 a4a3f8e8-34cd-4c66-8db7-84ed8698f726 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 ControlChannelClosed - - 0 0 a4a3f8e8-34cd-4c66-8db7-84ed8698f726 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 ControlChannelOpened - - 0 0 bbe35ef7-fa7e-4fae-a93c-29eb82f24b77 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 USER anonymous 530 11001 37 bbe35ef7-fa7e-4fae-a93c-29eb82f24b77 -
9-10-15 11:17:43 192.168.43.211 - 192.168.43.211 21 ControlChannelClosed - - 0 0 bbe35ef7-fa7e-4fae-a93c-29eb82f24b77 -
9-10-15 11:17:50 192.168.43.211 - 192.168.43.211 21 ControlChannelOpened - - 0 0 b2c2205f-a7f7-474b-a9da-ec1270ee0a1d -
9-10-15 11:17:50 192.168.43.211 - 192.168.43.211 21 USER anonymous 530 11001 37 b2c2205f-a7f7-474b-a9da-ec1270ee0a1d -
9-10-15 11:17:50 192.168.43.211 - 192.168.43.211 21 ControlChannelClosed - - 0 0 b2c2205f-a7f7-474b-a9da-ec1270ee0a1d -

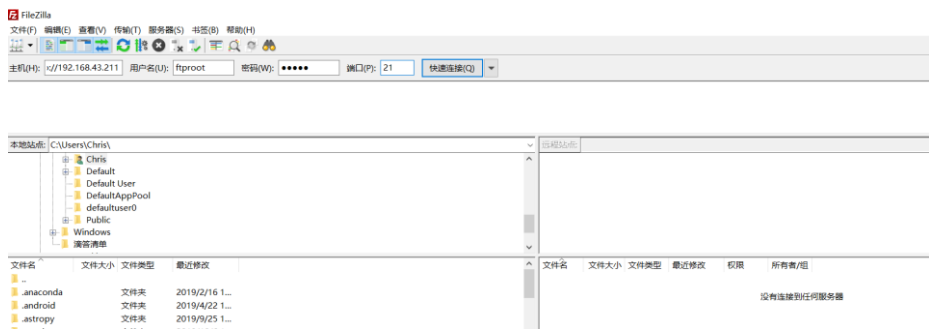
```

从 log 中发现是匿名用户出现问题，重新配置 ftp 后得到恢复，注意细节。同时留一到，如果利用 chrome 浏览器进行访问的时候，有的设备上会提示当前连接为非私密连接，这很有可能是因 ftp 没有 ssl 的原因，切换浏览器或者换用文件浏览器就可以解决这个问题。

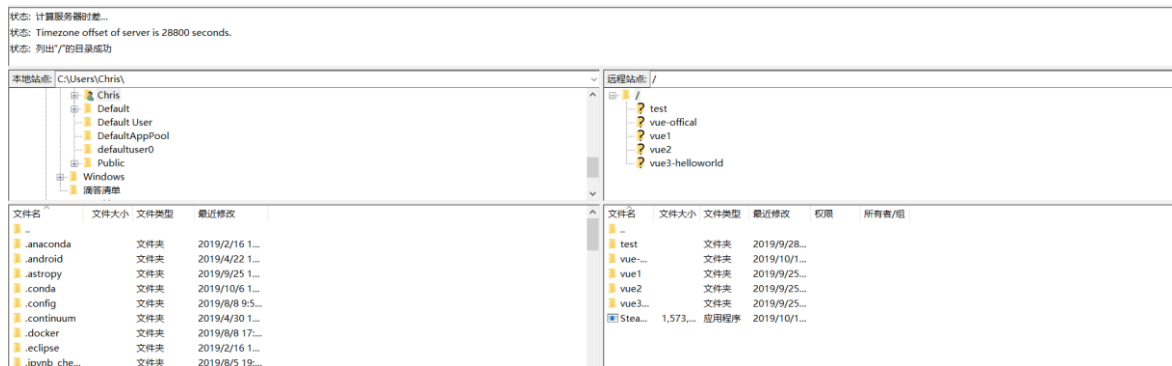
### 三、 扩展加分点

#### 1. 阐述 ftp 工作原理及为什么使用 tcp、为什么使用两个端口号

FTP 即 file transfer protocol 文件传输协议，它在 TCP/IP 协议中属于应用层协议，而其运行于 TCP 协议之上是一种可靠的传输协议，主要功能用于实现用户间文件分发共享，以及网络管理者在进行设备版本升级、日志下载和配置保存等业务操作使，均会利用到 FTP 功能。利用 filezilla 平台进行连接操作，以实现第三方软件访问。



确认登陆时会收到警告，用户名与密码会按明文方式发送，这也是本次配置时的一个缺陷



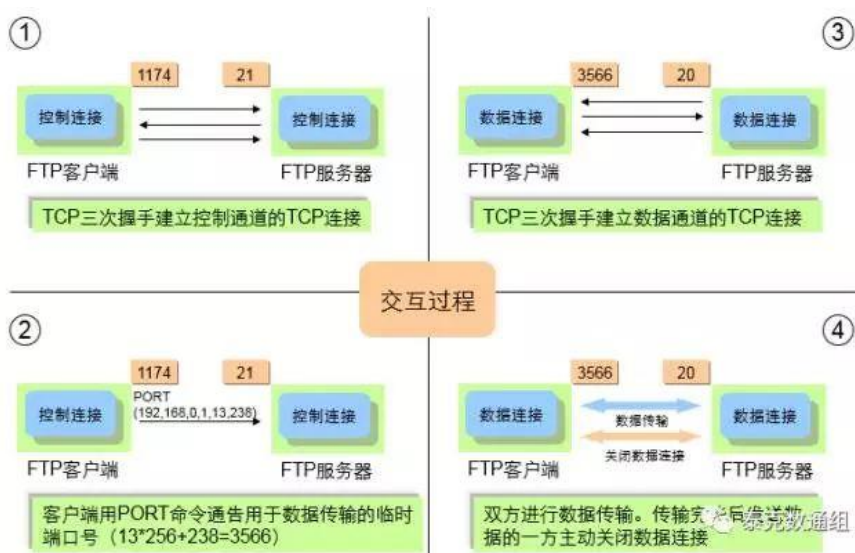
## 关于 FTP 的两个 TCP 连接

<https://zhuanlan.zhihu.com/p/34109504>

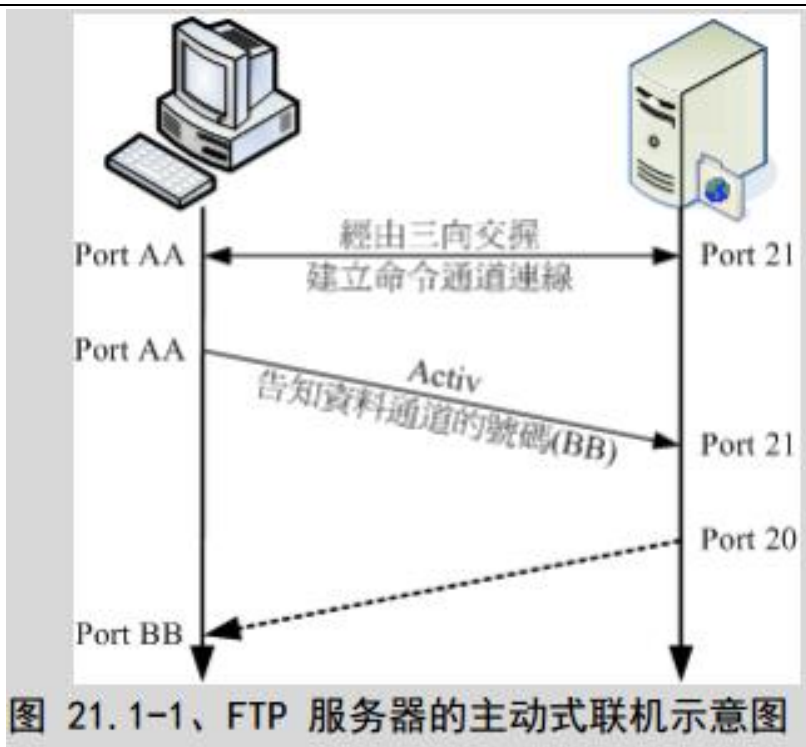
- 控制连接,通常为 TCP21 端口。其主要用于实现 FTP 协议控制信令的传送,服务器端打开 TCP21 端口被动等待客户端发起连接命令; 客户端则通过主动方式打开随机端口, 再发起连接的建立请求
- 数据连接,通常为 TCP20 端口, 主要用于实现应用数据传送, 服务器的数据连接端使用端口 20。服务器主动执行打开数据连接, 通常也执行主动关闭数据。

## 又需要认识 FTP 建立连接方式的两种区别

- 主动方式
  - 主动方式也称为 port 方式, 是 FTP 协议最初定义的数据传输链接方式
  - FTP 客户端通过向 FTP 服务器发送 PORT 命令, 告诉服务器该客户端用于传输数据的临时端口号当需要传送数据时, 服务器通过 TCP 端口 20 号与客户端的临时端口建立数据传输通道, 完成数据传输。再建立数据链接的过程中, 服务器主动发起连接, 所以被称为主动方式。

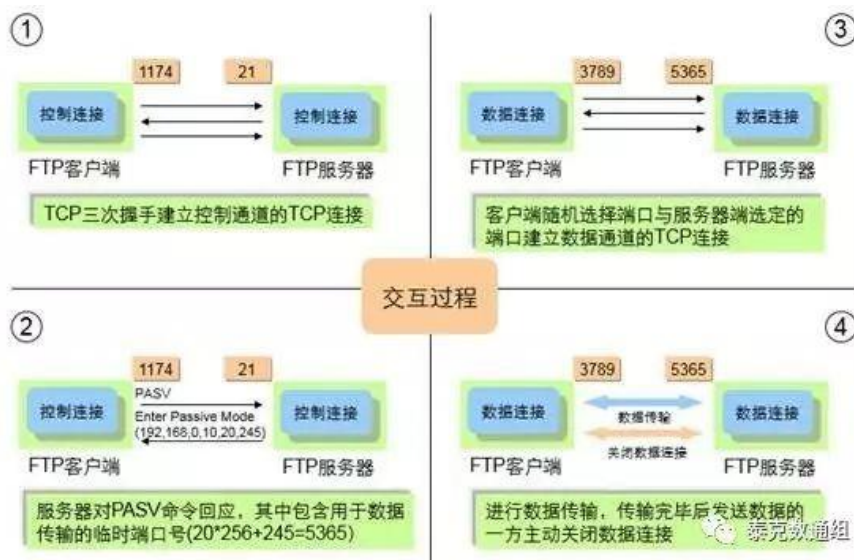


- 这里用到从知乎上查到的资料图。

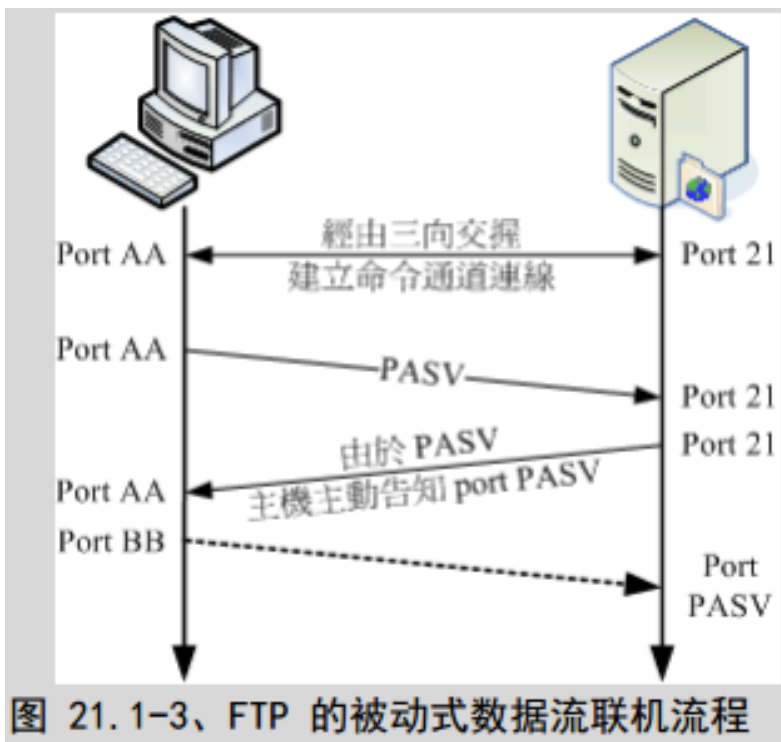


- 被动方式

- 被动方式也称为 PASV，被动方式的主要特点
- FTP 客户端通过向 FTP 服务器发送 PASV 命令，告诉服务器其进入被动方式。服务器选择临时端口号并告知客户端当需要传送数据时，客户端主动与服务器的临时端口号建立数据传输通道，完成数据传输。再整个过程中，由于服务器总是被动接收客户端的数据链接，因此其就被叫做被动方式。



- 图片同样来源于知乎参考。



- 如何进行方式的选择
  - 当 FTP 服务器与 FTP 客户端均处于同一局域网内，即两者之间互相访问不存在防火墙或其他安全设备时，两种方式均可以实现 FTP 文件的分发与分享
  - 如果 FTP 服务器位于路由器，防火墙或其他 NAT 设备中，通常选择使用被动模式。主要因为再主动模式下传输数据时，由 FTP 发起的数据传输一般会被客户端网关的防火墙阻断。

## 2. 配置 FTP 的被动模式

由上一问对 FTP 的被动模式流程已经有了初步的了解，接下来就开始尝试对客户端与服务器进行配置。

查阅资料发现 win10 的 iis 在 ftp 搭建完成后默认为主动式，即由客户端 n 端口发起，连接到服务器端 21 端口。传输数据时，服务器端打开随机高位 X 端口通知客户端，客户端通过 n+1 端口连接 X 端口，传输数据时，服务器等待客户端连接。这样由于连接是由客户端发起的，在被动方式 FTP 中，命令连接和数据连接都由客户端发起，这样就可以解决从服务器到客户端的数据端口的入方向连接被防火墙过滤掉的问题。，但是这样不便于服务器的管理。



在服务器配置防火墙时发现端口范围为灰色无法更改需要执行相关指令来添加防火墙策略，

```
管理员: Windows PowerShell
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6
PS C:\WINDOWS\system32> netsh advfirewall firewall add rule name="FTP Service" action=allow service=ftpsvc proto
col=TCP dir=in system32> 确定。
```

```
PS C:\WINDOWS\system32> netsh advfirewall set global StatefulFTP disable
确定。
```

以管理员身份在 powershell 中输入指令

```
netsh advfirewall firewall add rule name="FTP Service" action=allow service=ftpsvc proto
protocol=TCP dir=in
```

```
netsh advfirewall set global StatefulFTP disable
```

之后可以打开 FTP 防火墙支持，

在输入框中输入允许 FTP 服务器同时使用的 100 个端口号范围，比如 41000-41099，再输入建立连接时会经过防火墙的 ipv4 地址

点击 apply 以执行新的配置，此时再使用被动连接即可成功。



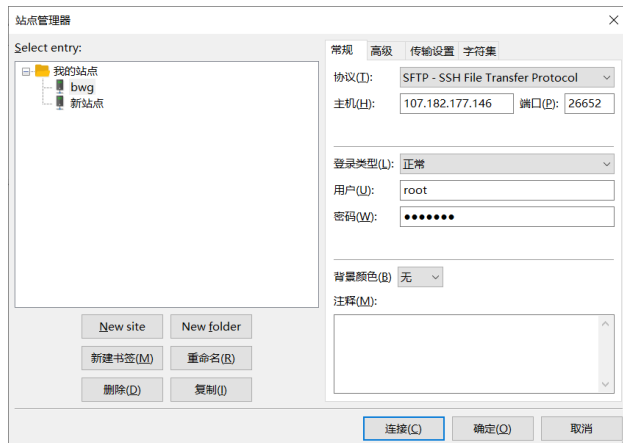
这里用 fileZilla client 建立连接，并且传输模式设置为被动也可实现配置的要求。

### 3. 使用 ssh 远程登陆 linux 并使用 sftp 下载文件

首先利用 sftp 工具进行尝试，本次即采用 filezilla；  
本次的实验对象为

localhost [KVMV2-10G] KVM			
Physical Location:	US, California	Node ID: v83	VPS ID: 105
Public IP address:	107.182.1.1		
SSH Port:	2601		
Status:	Running, LA: 0.00 0.00 0.00 2/116 25570		
Actions:	<button>start</button> <button>stop</button> <button>reset</button> <button>force stop</button>		
RAM:	308.04/512 MB		
SWAP:	1.46/132 MB		
Disk usage (/):	2.53/11 GB		
Bandwidth usage:	10.14/183 GB – USCA_8 Premium Bandwidth Multiplier:		
Resets: 2019-11-03	0.33x		
Operating system:	Centos 7 x86_64 bbr		
Hostname:	localhost.localdomain <a href="#">change</a>		

装有 centos7 的 vps 服务器



以 SFTP 协议进行站点的配置后点击连接

状态: 读取"/"的目录列表...

状态: Listing directory /

状态: 列出"/"的目录成功



文件名	文件大小	文件类型	最近修改	权限	所有者/组
..					
bin		文件夹	2019/10/7...	lrwxrwx...	root root
boot		文件夹	2019/10/7...	dr-xr-xr...	root root
dev		文件夹	2018/4/11...	drwxr-x...	root root
etc		文件夹	2019/10/7...	drwxr-x...	root root
home		文件夹	2018/4/11...	drwxr-x...	root root
lib		文件夹	2019/10/7...	lrwxrwx...	root root
lib64		文件夹	2019/10/7...	lrwxrwx...	root root
lost+...		文件夹	2017/3/14...	drwx---...	root root
media		文件夹	2018/4/11...	drwxr-x...	root root
mnt		文件夹	2018/4/11...	drwxr-x...	root root
opt		文件夹	2018/4/11...	drwxr-x...	root root
proc		文件夹	2018/4/11...	dr-xr-xr...	root root
root		文件夹	2019/10/7...	dr-xr-x---	root root
run		文件夹	2019/10/1...	drwxr-x...	root root

文件读取成功

find	199,304	文件	2018/10/3...	-rwxr-x...	root root
find-r...	4,371	文件	2019/8/9 ...	-rwxr-x...	root root

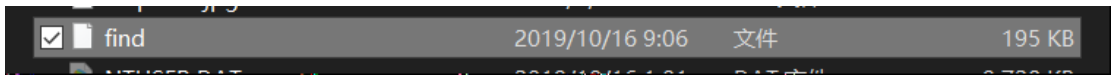
选择了 1 个文件。大小总共: 199,304 字节

**传输完成**  
所有文件都已经成功传输  
FileZilla

服务器/本地文件	方向	远程文件	大小	优先级	时间
 sftp://root@107.18...					
 C:\Users\Chris\find	< <--	/usr/bin/find	199,304	正常	2019/10/16 9:06:...

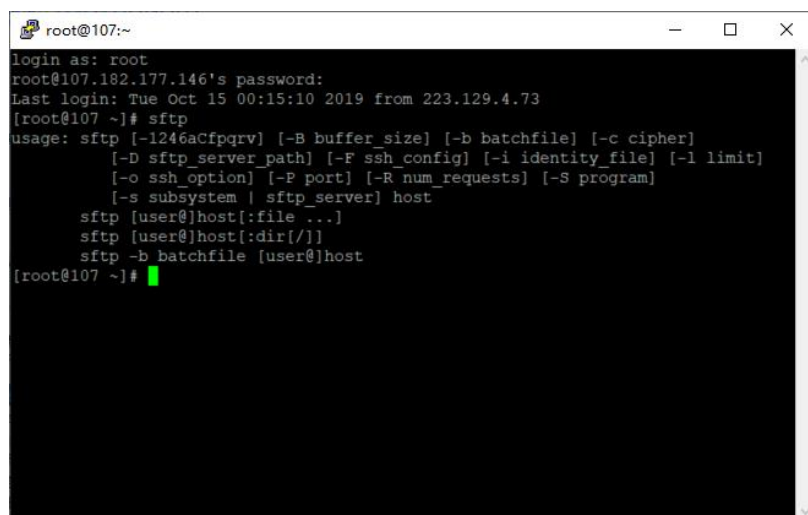
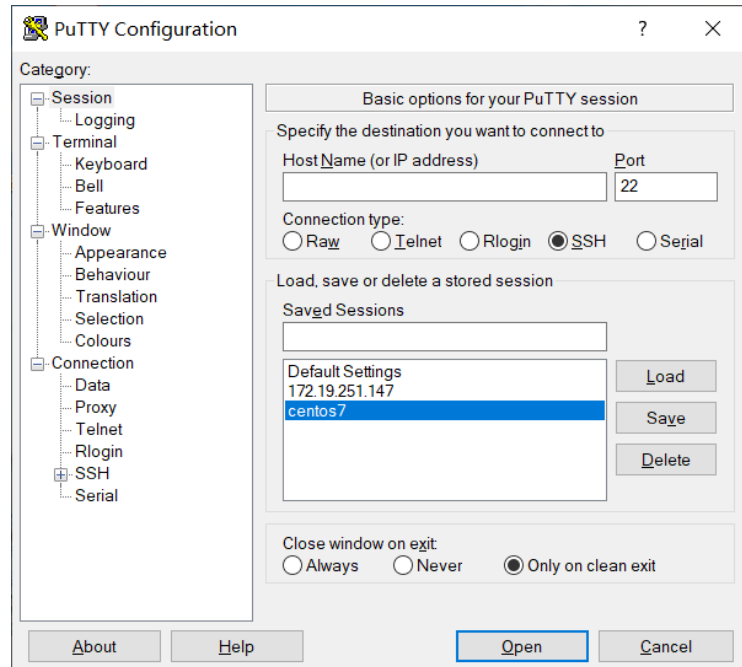


以一个任意文件为例进行下载工作



成功在本地找到从远程下载得到的文件；

下尝试采用 putty，即命令行的形式实现 vps 的 ssh 访问



先查看服务器是否已有 sftp 工具；但做到这一步发现自己的理解有错，不能用 linux 登陆之后在远程访问 linux。切回 windows 的 cmd 中 windows 的 cmd 中

```
C:\Users\Chris>sftp
usage: sftp [-46aCfpqrvt] [-B buffer_size] [-b batchfile] [-c cipher]
          [-D sftp_server_path] [-F ssh_config] [-i identity_file] [-l limit]
          [-o ssh_option] [-P port] [-R num_requests] [-S program]
          [-s subsystem | sftp_server] destination
```

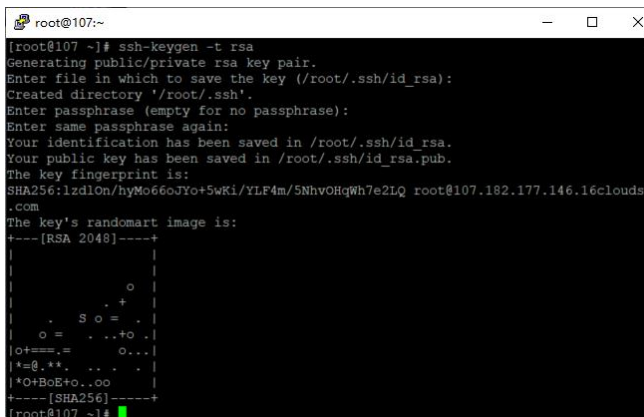
调用相同指令

```
C:\Users\Chris>sftp root@107.182.177.146
ssh: connect to host 107.182.177.146 port 22: Connection refused
Connection closed
```

```
C:\Users\Chris>ssh root@107.182.177.146
ssh: connect to host 107.182.177.146 port 22: Connection refused
```

之后发现自己的 vps 还未配置公钥

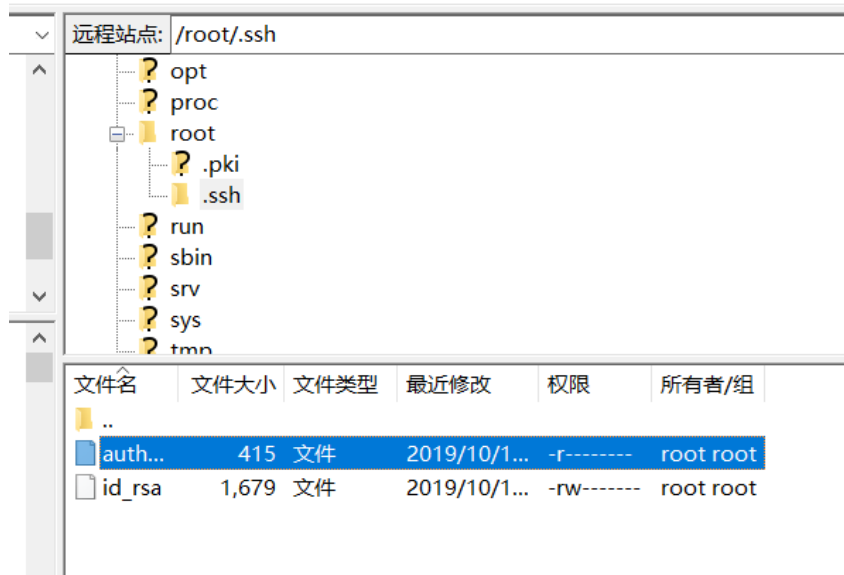
```
[root@107 ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
```



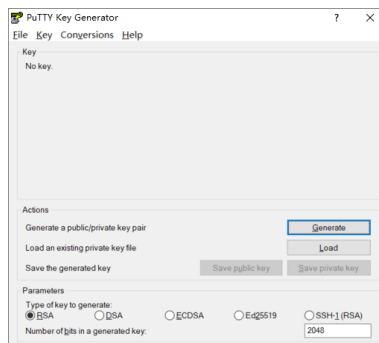
```
root@107:~
[root@107 ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:1zd1On/hyMo66oJYo+5wKi/YLF4m/5NhvOHqWh7e2LQ root@107.182.177.146.16clouds
.com
The key's randomart image is:
+---[RSA 2048]-----+
|
|   o
|  . S o = .
| o = . . .+o .
|O+==.= . O...
|*=@.*. . . .
|*O+BoE+o..oo
+---[SHA256]-----+
[root@107 ~]#
```

```
[root@107 ~]# cd .ssh
[root@107 .ssh]# dir
id_rsa id_rsa.pub
[root@107 .ssh]#
```

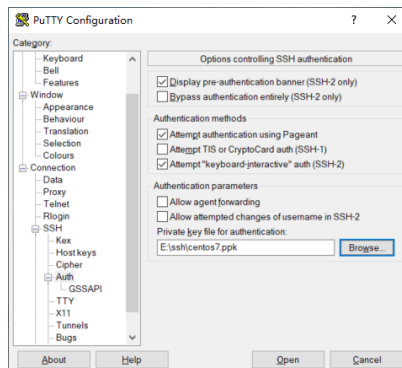
进入目录找到文件说明生成完毕；现在需要把公钥放在 server 上；



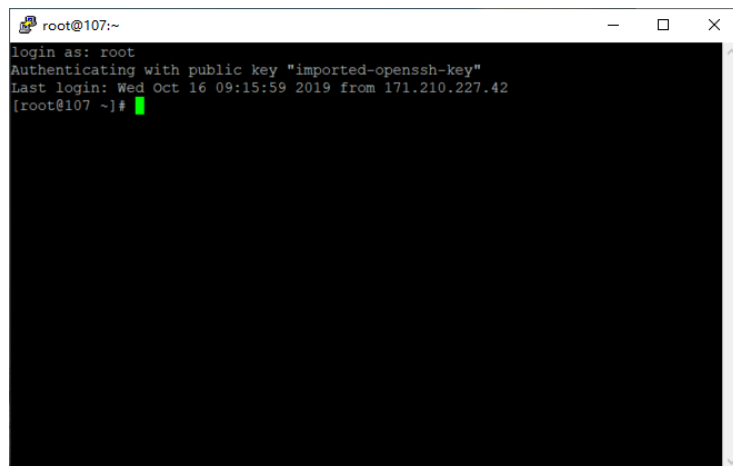
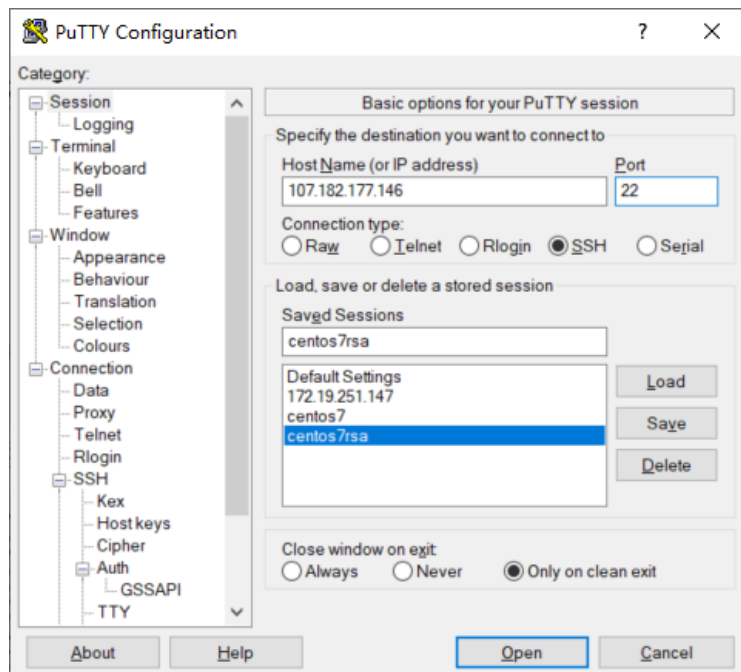
在进行公钥私钥生成之后通过客户端找到服务器端产生的私钥 id\_rsa



利用 puttygen 工具进行私钥格式的转换



成功之后在 putty 相关路径下导入经过转换得到的 ppk 私钥文件



发现只需要输入用户名，无需输入密码即可实现访问，即完成了公钥私钥的配对，密码的输入不再必要。这点操作虽然看似没有用，但是可以为其他很多操作节省时间。

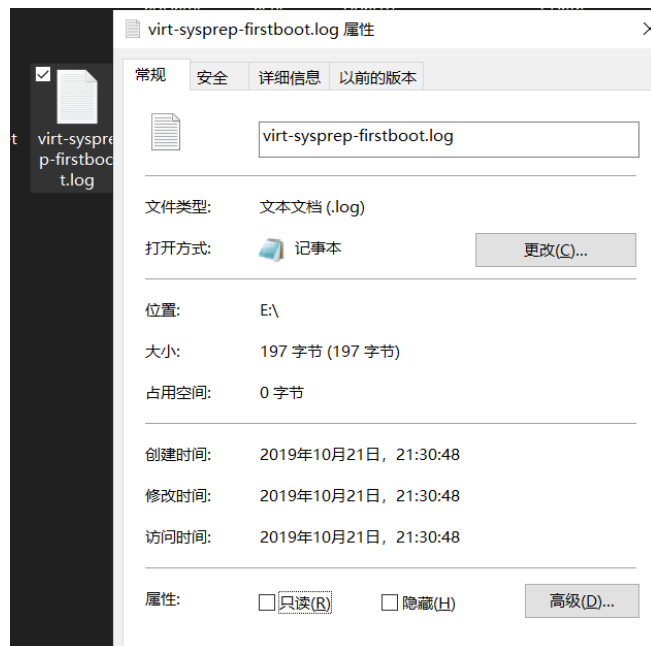
已知通过 ssh 访问 vps 成功，且有很多可以作为操作测试的例子。下用 cmd 通过 sftp 连接 vps

```
C:\Users\Chris>sftp root@107.182.177.146 -p 26652
usage: sftp [-46aCfpgrv] [-B buffer_size] [-b batchfile] [-c cipher]
          [-D sftp_server_path] [-F ssh_config] [-i identity_file] [-l limit]
          [-o ssh_option] [-P port] [-R num_requests] [-S program]
          [-s subsystem | sftp_server] destination

C:\Users\Chris>sftp -P 26652 root@107.182.177.146
root@107.182.177.146's password:
Connected to root@107.182.177.146.
sftp>
```

```
sftp> ls
virt-sysprep-firstboot.log
sftp> ls -l
-rw-r--r--  1 root    root      197 Oct  7 18:27 virt-sysprep-firstboot.log
sftp> get virt-sysprep-firstboot.log e:/
Fetching /root/virt-sysprep-firstboot.log to e:/virt-sysprep-firstboot.log
/root/virt-sysprep-firstboot.log                               100% 197    0.4KB/s   00:00
sftp>
```

在本地进行实际的查看



文件确实已经从远程服务器下载至本地。将改文件重命名后尝试上传，调用命令 put

```
sftp> put e:/refresh.log /
Uploading e:/refresh.log to /refresh.log
e:/refresh.log                               100% 197    0.2KB/s
sftp>
```

重新用 ssh 登陆服务器，

```
find: ./refresh.log: No such file or directory
[root@107 ~]# find / name refresh.log
/refresh.log
```

找到目标文件，即代表上传成功。至此下载与上传任务均完成

区分 sftp 与 ftp

## sftp 原理

	<ul style="list-style-type: none"> <li>sftp 协议能够实现“远程文件获取，文件内容传输，文件管理”等操作它的控制信号和数据信号的传输通过安全数据通道进行。通常，这个安全数据通道由 ssh 连接提供，即 ssh 连接隧道作为安全数据通道。其协议工作与 OSI 七层体系的应用层中。</li> <li>sftp 是 ssh 内包含的协议，只要 sshd 服务器启动了 sftp 就可以工作，本身不需要启动 ftp 服务器。</li> <li>SFTP 与 FTP 同样分为两个端，一个为 sftp server，一个为 sftp client。两者之间的控制信号和文件内容的传输得通过安全数据通道进行安全数据通道可由“transport layer security”或者 ssh 等协议提供，通常遇到得均为 ssh 协议提供，但是，由于这种传输方式使用了加密/解密技术，所以传输效率比普通的 FTP 要低得多，如果您对网络安全性要求更高时，可以使用 SFTP 代替 FTP。</li> </ul> <p>4. ftp 下载可以多线程，上传可能会受到限制</p> <ol style="list-style-type: none"> <li>其支持断点续传</li> <li>其理论上可以提高下载速度于特定场景。如带宽为 100Mb/s，下载一个 500MB 的大文件，正常情况下需要下载五秒，但是如果条件允许时创建五个线程同时进行下载，可能只需要 1 秒左右的时间，实际上加快了下载速度。但是会占用更多的内存空间。虽然可能不会提升每个线程各自的下载速度，但是会提高下载整个文件的速度。但要保证原始文件可以被分为几个更小的临时文件而不会受到影响。</li> <li>通常解决下载速度的最好方法是增加带宽。如果下载的任务均具有相同的瓶颈，那么他们都会在同一事务上排队，只有当其他线程产生瓶颈的时候其线程才会得到一个“look-in”</li> <li>对 FTP 来说，如果使用 comms 的线程，只有在必须写入磁盘或发生上下文切换时才会产生该资源。与通信产生的开销相比，ftp 在磁盘上的活动影响不大。</li> <li>因为有些文件必须连载，加上有时候因为上下文切换的额外开销，多线程不一定会比单线程更快。</li> </ol>
小结	<p>本次实验基础任务的任务量适中，但是需要在扩展加分点花费一些功夫来解决困难。同时本次实验课的内容在理论课上并没有过多的讲解，所以很多知识和资料需要自己额外的查阅很多内容才能掌握。</p> <p>这次实验细节很重要，有些勾勾如果没注意就会影响很多配置项，有不少需要填写 ip 地址或者端口的地方，如果稍不留神就会出现错误。同时通过本次实验，增强了自己使用 putty, puttygen, filezilla 等的熟练度，也一定程度上增强了自己配置环境，站点的能力。感觉在这次实验中最重要收获除了了解到了“主动与被动”两种 ftp 传输方式，更重要的是进一步了解了 ssh 的机制和一些使用规范，实践了公钥私钥生成配对的过程，进一步体会到了生成密钥的好处。虽然过程指令不多，但是还是需要留意细节才能最终完成目标。</p>

	学习就是一个共同进步，共同前进的过程，与大家讨论让我发现自己没有注意到的地方，同时也能针对他人的问题提供自己已有解决方案，感觉很棒。
指导老师 老师 评 议	<div>成绩评定：</div> <div>指导教师签名：</div>

## 实验报告说明

专业实验中心

**实验名称** 要用最简练的语言反映实验的内容。如验证某程序、定律、算法，可写成“验证×××”；分析×××。

**实验目的** 目的要明确，要抓住重点，可以从理论和实践两个方面考虑。在理论上，验证定理、公式、算法，并使实验者获得深刻和系统的理解，在实践上，掌握使用实验设备的技能技巧和程序的调试方法。一般需说明是验证型实验还是设计型实验，是创新型实验还是综合型实验。

**实验环境** 实验用的软硬件环境（配置）。

**实验内容（算法、程序、步骤和方法）** 这是实验报告极其重要的内容。这部分要写明依据何种原理、定律算法、或操作方法进行实验，要写明经过哪几个步骤。还应该画出流程图（实验装置的结构示意图），再配以相应的文字说明，这样既可以节省许多文字说明，又能使实验报告简明扼要，清楚明白。

**数据记录和计算** 指从实验中测出的数据以及计算结果。

**结论（结果）** 即根据实验过程中所见到的现象和测得的数据，作出结论。

**小结** 对本次实验的体会、思考和建议。

**备注或说明** 可写上实验成功或失败的原因，实验后的心得体会、建议等。

**注意：**

- 实验报告将记入实验成绩；
- 每次实验开始时，交上一次的实验报告，否则将扣除此次实验成绩。