## 文件说明

两个文件夹，SSHServer里面是server的代码和证书、私钥文件

SSHClient是client的代码和证书、私钥文件

以.crt为结尾的都是证书，以.key为结尾的文件都是私钥。以.py结尾的是python源代码。

两个文件夹里面都有一个ca.crt文件，这个是根证书，server.crt和client.crt都是用它来签发的。

文件夹里面的东西一个不要删，少一个都不能运行。

## 运行方法

运行务必要有python环境 （python3以上！）

代码里没有用第三方包，理论上讲是不需要额外下载包的。

把SSHServer文件夹放到Server机器上 把SSHClient文件夹放到Client机器上

一定记得**先启动Server，让他等待监听端口。**然后再启动Client。

启动Server的命令：

1.通过域名（domain）+端口(port)启动

python3 ./SSHserver.py --domain Server的域名 --port Server监听的端口

或者

也可以通过ip地址+端口号的方式启动

python3 ./SSHserver.py -- domain Server的ip地址 --port Server监听的端口

启动Client的命令

python3 ./SSHclient.py --domain Server的域名 --port Server监听的端口

或者

python3 ./SSHclient.py -- domain Server的ip地址 --port Server监听的端口

## 参数解释：

--domain ：

作业要求里说在remote上测试，比如拿remote1当Server，那么这里的domain域名参数就是remote01.cs.binghamton.edu。

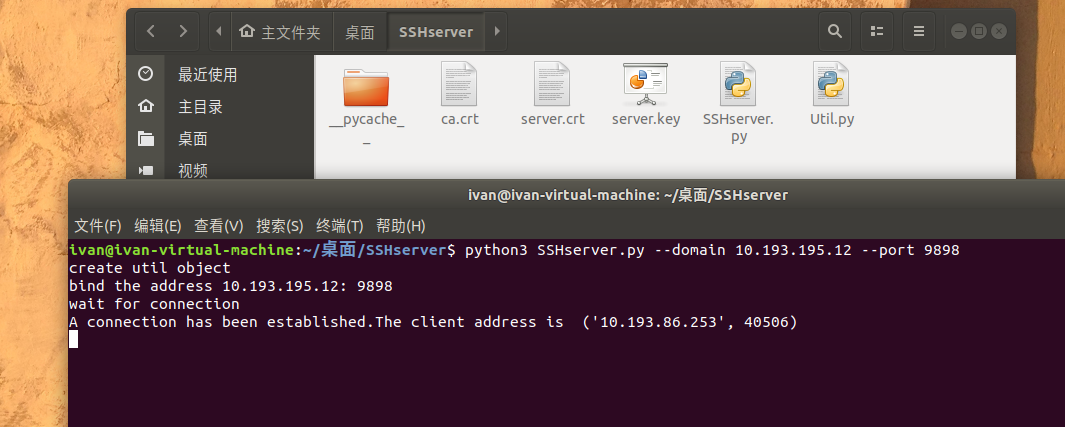
因为我这边无法用你们学校的remote环境测试，所以我自己在两台同一个局域网的机器上测试的。

如果域名无效，无法解析，那么用机器的ip地址代替，（用ifconfig可以查询机器的ip地址）

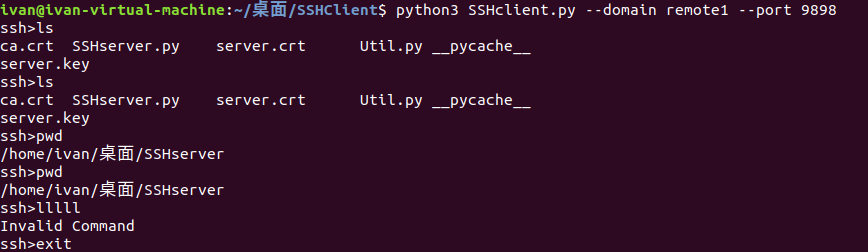
--port Server的监听端口，自己随便选。老师在作业里建议用学号的一部分作为端口？

## 我自己的测试结果

Server:



Client: 三个命令的测试（ls pwd 和 exit）



注：命令行参数中的 –domain remote1 ，这个是我自己的测试机器

## 关于证书文件

因为代码需要用到SSL，所以必须有证书，在Client和Server的文件夹里面都有一个ca.crt文件，这是X509格式的根证书文件，Client和Server各自的证书文件client.crt和server.crt都是由ca.crt签发的，签发证书时用到的私钥文件 client.key和server.key 以及所有的证书文件全部是用OpenSSL工具生成的。

证书直接拿来用就行，如果你想重新签发，那么需要先安装openssl工具，并用以下步骤生成证书：

1. 先用openSSL工具生成一个CA证书。步骤如下：

**利用openssl工具 制作CA**

1. **创建CA私钥文件ca.key**

**openssl genrsa -out ca.key 1024**

**(2)使用openssl 自颁发CA证书**

**命令：openssl req -new -x509 -days 3650 -key ca.key -out ca.crt**

**签发时会让你输入各种信息，地址，名字联系方式等等。**

1. 用openSSL分别生成server和client的RSA私钥文件，然后通过openSSL工具签发证书，步骤如下：
2. **利用openssl工具生成server.key 和client.key两个私钥文件**

**openssl genrsa -out server.key 1024**

**openssl genrsa -out client.key 1024**

**（2）生成csr请求文件**

**openssl req -new -key server.key -out server.csr**

**openssl req -new -key client.key -out client.csr**

**（3）使用CA签发证书 得到Server和Client的证书文件**

**openssl x509 -req -days 365 -sha1 -extensions v3\_req -CA ca.crt -CAkey ca.key -CAserial ca.srl -CAcreateserial -in server.csr -out server.crt**

**openssl x509 -req -days 365 -sha1 -extensions v3\_req -CA ca.crt -CAkey ca.key -CAserial ca.srl -CAcreateserial -in client.csr -out client.crt**

**验证签发正确性：**

openssl verify -CAfile ca.crt server.crt

openssl verify -CAfile ca.crt client.crt