**计算机导论实验配置文档**

**1 实验环境预配置与知识准备**

（1）虚拟机中装好Ubuntu系统：需要安装两个虚拟系统，一个做服务器，一个做客户端，一台电脑中安装两个虚拟机对硬件要求较高，推荐安装无图形界面的Ubuntu Server版本，本说明文档安装的是Ubuntu Server 18.04 LTS；

（2）VMmare软件安装虚拟系统时默认网络连接是NAT，虚拟系统安装好后请确认网络连接方式为NAT；

（3）了解Linux系统命令行操作的基本知识，包括基本命令的使用（ls、cd、cp、apt、rm、chmod等），普通用户与超级用户的切换，vim文本编辑器的使用（实验中需要多次修改各种文件，本文使用vim编辑器，也可使用nano编辑器），更换下载源等；

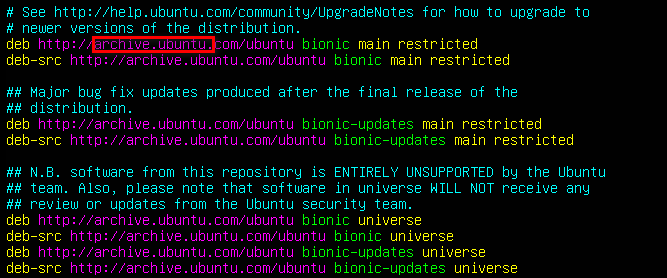
更换下载源相关操作：

备份原文件：

sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak

查看原有源的网址：

sudo vim /etc/apt/sources.list



将上图中所圈部分全部替换为阿里云源：

sudo sed -i “s/archive.ubuntu./mirrors.aliyun./g” /etc/apt/sources.list

更新：

sudo apt update

（4）在Ubuntu系统上装好实验所需软件（sudo apt-get install …），包括vim或nano文本编辑器、john、openssh-server、openssl、openvpn、easy-rsa（重要！最新版本的openvpn不提供easy-rsa，需用户自行安装）；

（5）利用ifconfig命令确认服务器以及客户端虚拟机的ip地址；

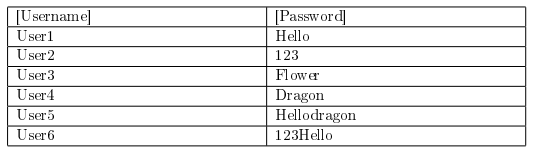
**2 密码破解软件的使用**

2.1 安装John the Ripper

sudo apt-get install john

2.2 添加其他用户

按下表添加用户并设定密码：



添加用户：

sudo useradd User1

设定其密码 ：

sudo passwd User1

在设定密码时，系统中的PAM模块可能会检查密码强度，若出现密码太弱等提示，禁用该模块：

sudo vim /etc/security/pwquality.conf

将文件中的enforcing=1修改为enforcing=0

查看密码文件内容与权限

sudo cat /etc/passwd

sudo cat /etc/shadow

sudo ls -l /etc/passwd

sudo ls -l /etc/shadow

2.3 破解密码

查找字典文件的路径：

sudo updated

sudo locate password.lst



运行JtR软件破解刚刚创建的用户密码：

sudo john -wordlist=/usr/share/john/password.lst /etc/shadow

运行过程中，可使用Ctrl-C停止过程

完成后，查看破解结果：

sudo john --show /etc/shadow

2.4 破解MD5密码（可选）

在另一台机器上，运行以下命令：

sudo authconfig --test | grep hashing

sudo authconfig --passalgo=md5 --update

sudo authconfig --test | grep hashing

重复步骤2.2与2.3

**3 基于公钥认证的SSH连接**

3.1 禁用 SELinux

在服务器与客户端上：

打开相关文件：sudo vim /etc/selinux/config

将enforcing 改为disabled

重启机器

若提示打开的文件为空，则代表机器没有使用该服务，可跳过此步骤

3.2 创建RSA密钥对

服务器和客户端都安装openSSH软件：

sudo apt-get install openssh-server

在服务器，打开sshd服务：

service sshd start

或 sudo service ssh start

在客户端，尝试过密码认证登录服务器：

ssh server\_name@server\_IPaddress

输入服务器账户的密码

若报错，检查ip地址、服务器22端口、sshd\_config文件等方面

登陆后，使用exit命令退出

在服务器上，打开/etc/ssh/sshd\_config文件，将PublickeyAuthentication no改为PublickeyAuthentication yes

每次配置文件修改后，都需要重启服务器端sshd服务才能使之前操作生效：

service sshd restart

在客户端，创建RSA密钥对

ssh-keygen -t rsa

出现以下提示信息：

Enter file in which to save the key (/home/user/.ssh/id\_rsa): 敲击回车键

Created directory /home/user/.ssh

Enter passphrase (empty for no passphrase): 敲击回车键

Enter same passphrase again: 敲击回车键

注意.ssh文件夹的路径，/home/user/…中，“user”为当前操作系统的用户名，在下面操作时牢记替换

查看公私钥：

cat /home/user/.ssh/id\_rsa.pub |more

cat /home/user/.ssh/id\_rsa |more

3.3 在服务器端配置公钥

在客户端，将公钥文件传输到服务器：

scp /home/user/.ssh/id\_rsa.pub server\_name@server\_IPaddress:

使用scp后，会出现文件传输100%的提示如下图，若提示未出现，代表文件没有传输成功



在服务器，登录想让客户端登录的账户

在最初配置时，authorized\_keys文件可能不存在，在服务器输入ssh-keygen命令，创建.ssh文件夹与authorized\_keys文件

将公钥内容复制到authorized\_keys文件（这里的user为当前服务器使用的用户名）：

cat /home/user/id\_rsa.pub >> /home/user/.ssh/authorized\_key

设置权限：

chmod 600 /home/user/.ssh/authorized\_keys

chmod 700 /home/user/.ssh/

删除公钥文件：

rm /home/user/id\_rsa.pub

重启服务器端sshd服务：

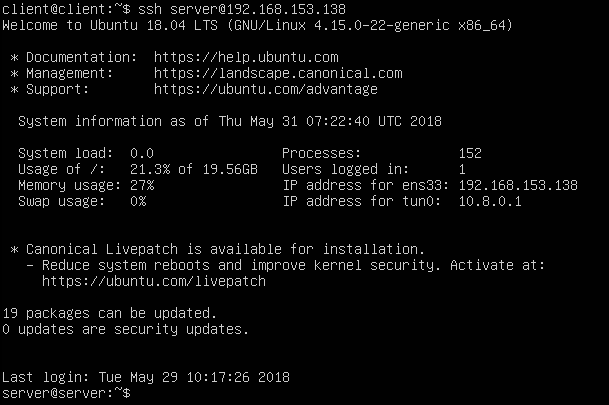
service sshd restart

3.4 测试认证

在客户端，登录服务器：

ssh server\_name@server\_IPaddress

若之前配置得当，则不需要输入密码就能登录到服务器上



使用ssh -vvv server\_name@server\_IPaddress 可以查看更多连接信息

3.5 使用DSA密钥认证

使用ssh-keygen -t dsa生成DSA密钥并重复上述步骤

**4 基于SSL的VPN设置**

在服务器和客户端，安装相应软件

sudo apt-get install openssl openvpn easy-rsa

4.1 创建CA

在服务器，进入root用户：

sudo su – root

在服务器

cp -r /usr/share/easy-rsa/ /etc/openvpn/

cd /etc/openvpn/easy-rsa/

执行下列命令，建立CA：

. ./vars

./clean-all

./build-ca

在CA建立过程中，唯一需要输入信息的是Common Name，可以将其命名为server，其余信息都可以直接敲击回车键略过

4.2 产生证书与密钥

在服务器，产生服务器的证书与密钥

./build-key-server server

同样只需要命名Common Name

在服务器，产生客户端的证书与密钥

./build-key client

在服务器，产生Diffie-Hellman参数

./build-dh

注意此时生成的dh文件是dh1024还是dh2048

将客户端所需文件ca.crt、client.crt以及client.key传输至客户端

可选用ssh服务：

cd /etc/openvpn/easy-rsa/keys/

scp ca.crt client\_name@client\_IP\_address:

scp client.crt client\_name @client\_IP\_address:

scp client.key client\_name @client\_IP\_address:

4.3 服务器配置

在服务器，将样例配置文件复制到/etc/openvpn中

gzip -d /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz

cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf /etc/openvpn

vim /etc/openvpn/server.conf

修改以下地方：

ca ca.crt

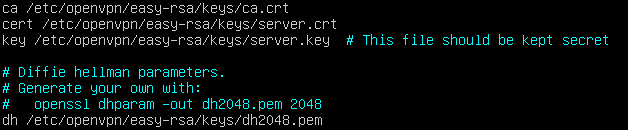
cert server.crt

key server.key # This file should be kept secret

…

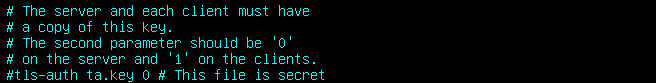
dh dh1024.pem

填入正确文件路径如下图



同时，禁用tls认证：

将tls-auth ta.key 0 一行用#号注释如下图



服务器打开VPN服务

openvpn /etc/openvpn/server.conf

看到Initialization Sequence Completed信息说明服务器设置正常：

4.4 客户端配置

在客户端，进入root用户：

sudo su – root

将之前接收到的文件复制到/etc/openvpn：

cp ca.crt /etc/openvpn

cp client.crt /etc/openvpn

cp client.key /etc/openvpn

将样例配置文件复制到/etc/openvpn中

cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf /etc/openvpn

vim /etc/openvpn/client.conf

修改以下地方：

ca ca.crt

cert client.crt

key client.key

填入正确文件路径如下图

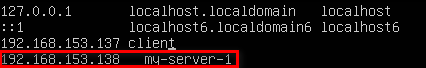


同样，禁用tls认证：



同时，注意服务器server.conf文件中cipher一行的加密算法应与客户端client.conf文件中cipher一行一致。

打开/etc/hosts文件，添加服务器的IP地址，如下图



客户端打开VPN服务

openvpn /etc/openvpn/client.conf

看到Initialization Sequence Completed信息说明客户端连接正常：

可使用ping命令查看VPN是否设置成功

**5 基于主机认证的SSH连接（可选）**

在实验前，在服务器上修改/etc/ssh/sshd\_config文件，将PublickeyAuthentication yes改为PublickeyAuthentication no，否则即使主机认证不成功，由于公钥认证的存在，SSH连接不需要密码仍会成功

在本说明文档中，客户端主机名为client，用户名为client，客户端主机IP地址为192.168.153.137；服务器主机名为server，用户名为server，服务器主机IP地址为192.168.153.138。

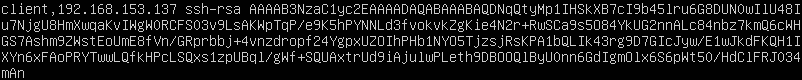
5.1 服务器配置

在服务器上，需要修改三个文件：

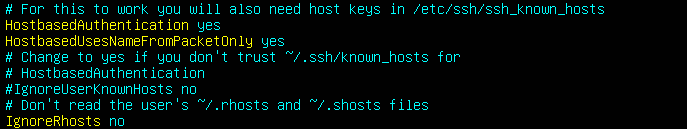
在/etc/ssh/shosts.equiv文件中，加入客户端主机名或客户端主机IP地址，如：



在/etc/ssh/ssh\_known\_hosts文件中，加入客户端主机公钥。每一行都需要三个数据，其格式为“客户端主机名或客户端主机IP地址 密钥类型 客户端主机公钥”，如：



在/etc/ssh/sshd\_config文件中，修改以下内容：



在服务器端添加一个客户端用户名相同的用户并设置密码

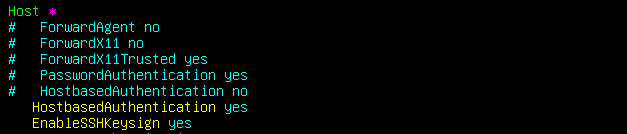
sudo adduser client

重启服务器端sshd服务：

service sshd restart

5.2 客户端配置

在/etc/ssh/ssh\_config文件中，修改以下内容：



在客户端，登录服务器在5.1中设置到的账户：

ssh client @192.168.153.138

若之前配置得当，则不需要输入密码就能登录到服务器上

