SSH

早期的telnet是应用层的一种协议，但使用telnet的实现，发送数据都是明文发送，极不安全。Secure Shell简称ssh，是创建在应用层和传输层之间的安全协议，允许两台主机进行交换数据。基于ssh的通信的认证过程和传输过程都是加密的。一个 SSH 服务器，默认地，在 TCP 端口22进行监听。一个 SSH 客户端程序通常被用来建立一个远程连接到 sshd 守护进程。

OpenSSH是使用ssh协议实现的一套计算机程序。Linux使用的就是OpenSSH。

SSH的配置文件

```
#客户端配置文件
/etc/ssh/ssh_config
```

```
#服务器端配置文件
/etc/ssh/sshd_config
```

SSH的认证方式

> 基于密码的认证

客户端知道密码和账号就可以登陆主机，所有传输都是加密的，但有可能会被别的主机冒充服务器，无法避免被“中间人”攻击。服务器端主机发送一个公钥给客户端，该公钥作为客户端的私钥，用来加密数据，进行数据传输，从而实现主机密钥认证，确保数据的保密性。

服务器端发送给客户端的公钥通常保存在客户端家目录`~/.ssh/known_host`文件。

> 基于密钥的认证

相对基于密码的认证方式，基于密钥的认证方式更为安全。但需要客户端创建一对密钥，把公钥提供给服务器端。客户端和服务器端进行通信时，服务器端会把客户端发送过来的公钥和自身家目录下保存着客户端事先发过来的公钥进行比较，如果一致，则同意通信，这样，就避免的“中间人”攻击。客户端也保存一份服务器端的公钥，用来认证服务器端主机，这种方式的认证无须密码，也就是我们通常所说的免密登陆，但其安全性更高。

SSH的实现

ssh

```
#以当前主机用户名的身份类型登陆
ssh host_ip

#以指定用户身份登陆
ssh user_name@host_ip
ssh -l user_name host_ip

#不登陆主机，只将命令发送至远程主机并执行此命令
ssh user_name@host_ip 'command'
```

scp：跨主机安全复制工具

```
#命令格式。-r为递归复制选项。address:/path/to/file
scp -r s_address d_address

#如果地址是一个远程主机，命令格式为：
user_name@host_ip:/path/to/file
```

ssh-keygen：密钥生成器

```
#选项
-t #指定加密算法
-p #指定访问密钥密码
-f #指定密钥存放路径，默认保存位置~/.ssh/
```

SSH免密登陆

> 客户端生成密钥

```
ssh-keygen

[localhost@root ~]#ls -lh ~/.ssh/
total 16K
-rw------- 1 root root 1.7K Jan  8 22:21 id_rsa #私钥
-rw-r--r-- 1 root root  408 Jan  8 22:21 id_rsa.pub #公钥
```

> 将生成的公钥保存至服务器

```
#若远程服务器家目录没有.ssh目录，则手动创建，并将目录修改权限为700
#创建文件.ssh/authorized_keys或者.ssh/authorized_keys2
mkdir ~/.ssh
#手动将公钥复制到此文件即可
vim ~/.ssh/.ssh/authorized_keys
```

```
#使用scp复制公钥文件
scp /root/.ssh/authorized_keys root@host_ip:/root/.ssh/
```

```
#使用ssh-copy-id复制公钥。会自动在服务器端家目录生成.ssh目录，及公钥文件
ssh-copy-id -i ~/.ssh/id_rsa.pub user_name@host_ip
```