|  |
| --- |
| 文章标题  ssh远程管理服务 |
| **作者：**  **归档：学习笔记** |
| **快捷键：**  Ctrl + 1 标题1  Ctrl + 2 标题2  Ctrl + 3 标题3  Ctrl + 4 实例  Ctrl + 5 程序代码  Ctrl + 6 正文 |
| **格式说明：**  蓝色字体：注释  黄色背景：重要  绿色背景：注意 |

****

**老男孩教育教学核心思想6重：重目标、重思路、重方法、重实践、重习惯、重总结**

**学无止境，老男孩教育成就你人生的起点！**

# SSH远程管理服务：

### SSH远程管理服务是什么？

是提供类似telnet远程联机服务器的服务，即上面提到的SSH服务（具有数据加密功能）；

是类似FTP服务的sftp-server，借助SSH协议来传输数据的，提供更安全的SFTP服务(vsftp,proftp)。

### 测试服务端口有没有开启方法

# 01：telnet 10.0.0.41 22

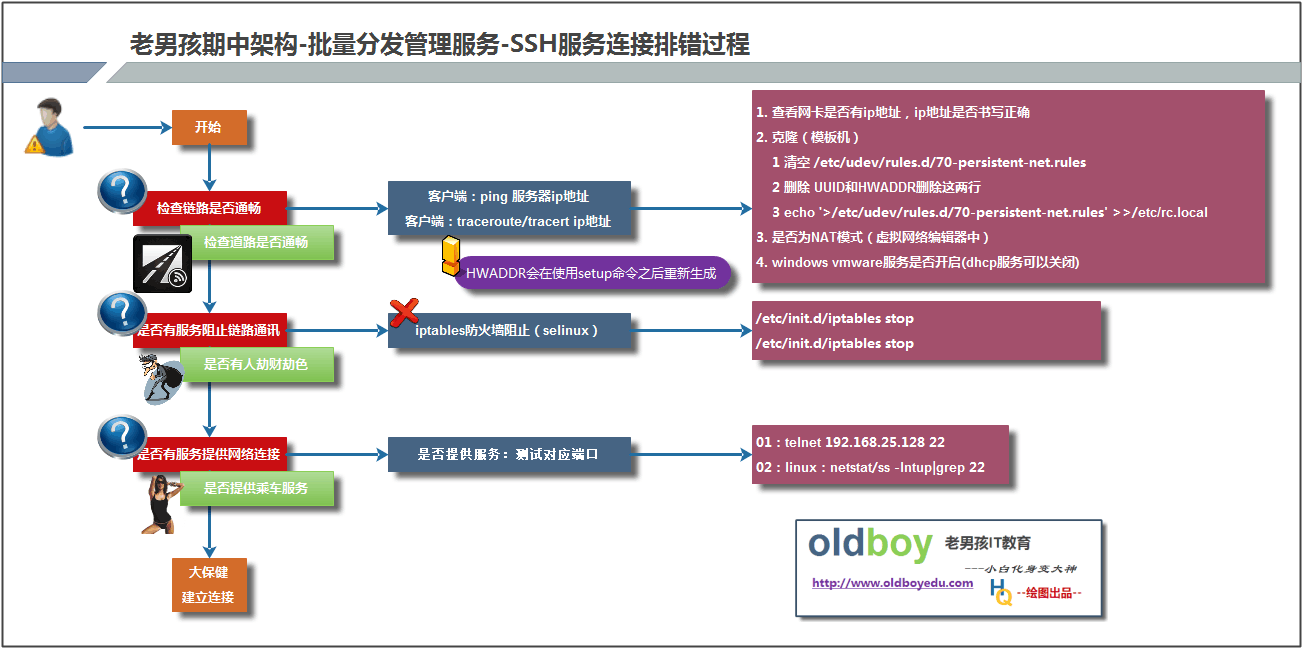
# 02. ss -lntup|grep 22

# 03. netstat -lntup

# 04，nmap -p 22 10.0.0.41

# 05. nc 10.0.0.41 22

#### ssh 排错



### 配置telnet远程连接服务：

#### 第一个里程碑：安装telnet服务软件

yum install telnet telnet-server

#### 第二个里程碑：配置xinetd服务目录中的telnet文件，让telnet服务允许被xinetd服务管理

[root@backup ~]# cat /etc/xinetd.d/telnet

disable = no

#### 第三个里程碑：启动xinetd服务

/etc/init.d/xinetd start

#### 第四个里程碑：进行远程登录测试

[d:\~]$ telnet 10.0.0.31

Connecting to 10.0.0.31:23...

Connection established.

To escape to local shell, press 'Ctrl+Alt+]'.

CentOS release 6.9 (Final)

Kernel 2.6.32-696.el6.x86\_64 on an x86\_64

nfs01 login: yyc

Password:

[yyc@nfs01 ~]$

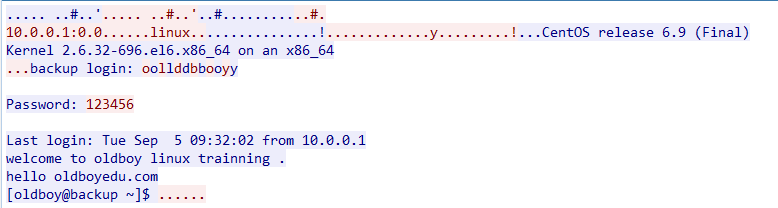
说明：不能利用root远程登录

### ssh远程连接服务与Telnet服务区别：

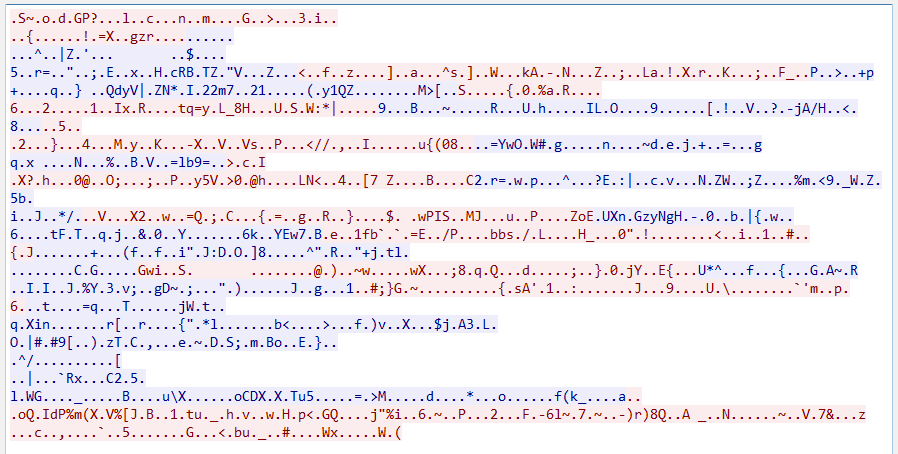
ssh服务是加密服务协议，telnet服务是非加密服务协议

ssh服务默认支持root用户登录，Telnet服务默认不支持root用户登录

#### telnet 远程连接抓包（非加密）



#### ssh 连接抓包（加密）



### SSH知识要点总结

01. SSH是安全的加密协议，用于远程连接linux服务器。

02. SSH默认端口是22，安全协议版本SSHv2，除了2之外还有SSHv1（有漏洞）。

03. SSH服务端主要包含两个服务功能SSH远程连接和SFTP服务。

04. Linux SSH客户端包含ssh远程连接命令，以及远程拷贝scp命令等。

## ssh软件服务命令知识

### openssh-clients软件信息：

[root@backup ~]# rpm -ql openssh-clients

/etc/ssh/ssh\_config --- ssh客户端配置文件

/usr/bin/.ssh.hmac

/usr/bin/scp --- 远程复制命令

/usr/bin/sftp --- 远程文件传输服务

/usr/bin/slogin --- 远程登录命令

/usr/bin/ssh --- ssh远程登录管理主机命令

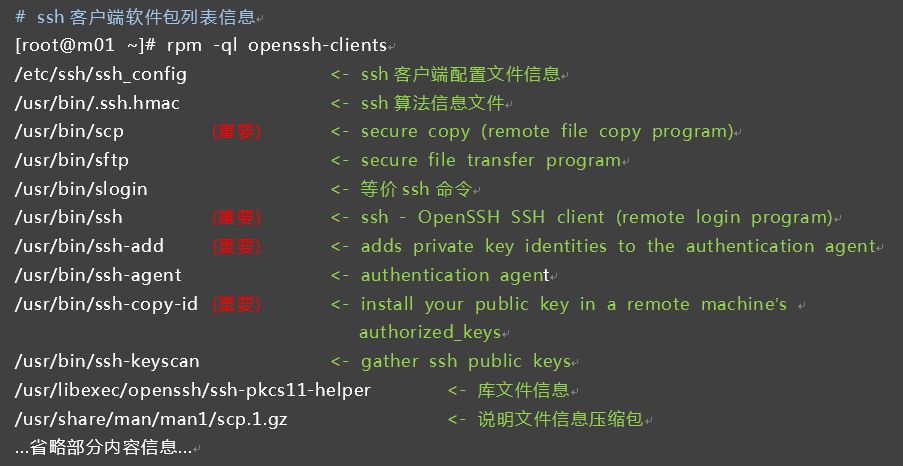
/usr/bin/ssh-add

/usr/bin/ssh-agent

/usr/bin/ssh-copy-id --- ssh服务分发公钥命令？

/usr/bin/ssh-keyscan

ssh客户端包含ssh以及像scp（远程拷贝）、slogin（远程登陆）、sftp（安全FTP文件传输）等应用程序



openssh-server软件信息：

/etc/pam.d/ssh-keycat

/etc/pam.d/sshd

/etc/rc.d/init.d/sshd --- ssh服务启动脚本文件

/etc/ssh/sshd\_config --- ssh服务配置文件

/etc/sysconfig/sshd

/usr/libexec/openssh/sftp-server

/usr/libexec/openssh/ssh-keycat

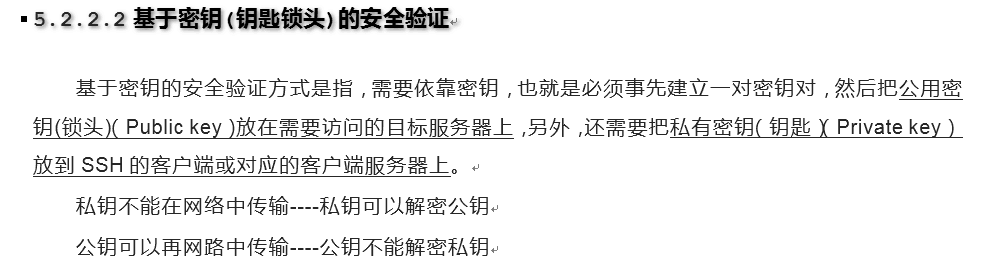
/usr/sbin/.sshd.hmac

/usr/sbin/sshd --- ssh服务进程启动命令（采用绝对路径进行启动）

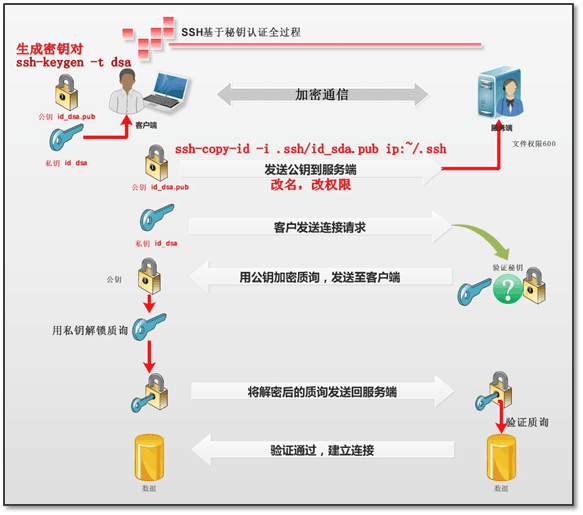
### ssh服务实现连接安全连接建立,利用钥匙和锁头；

钥匙==私钥 锁头==公钥，私钥可以解密公钥（钥匙可以开启锁头）

公钥可以在网络中传输，私钥在本地主机保存

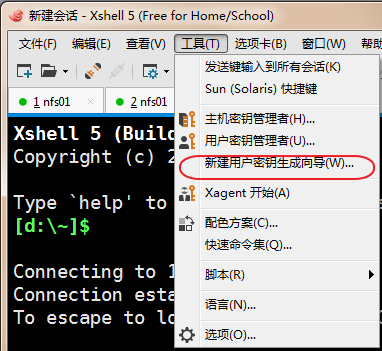


#### ssh 基于密钥认证全过程

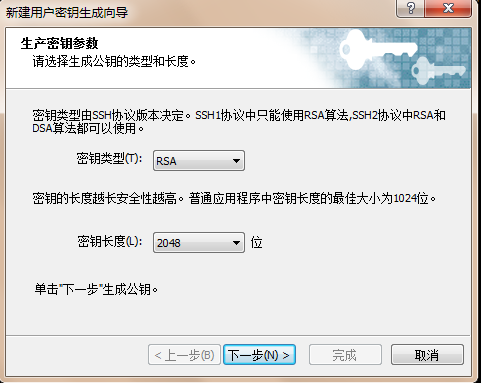


### 使用密钥登录xshell

#### 在xshell 生成一对密钥

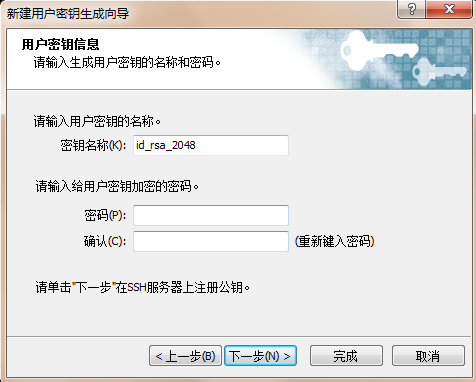


#### 根据向导进行





#### 设置密码



#### 复制公钥，粘贴到远程主机上



#### 将公钥粘贴到远程主机，在家目录创建.ssh 目录，创建authorized\_keys文件

[root@nfs01 .ssh]# cd ~

[root@nfs01 ~]# cd .ssh

[root@nfs01 .ssh]# chmod 600 authorized\_keys 修改文件权限

[root@nfs01 .ssh]# cat authorized\_keys 将公钥粘贴到文件中

ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAyIidCio848yABvT7gxhbNazGc9VYwv+QWzBpF8LiAK++yWG9R+4HFbFnEkgXeFnfg7V0GFFBm9Kx0vExZeetAvrlVXEtYk4fmjSwrLv2rvGIiNnommi1xGRfFQfUJearhPgWFe0pp6V5caERBEXEsLinaCtLzPOZwRPvRRQls/BdSpI0BWqf5Z4kUnyB+ApKhF7o1P/2xXDNIHuEW5o/33VhscL8wuqUYTC/hOvwoOJinFYYr0gSQgbaalp3+sDV5xSX1UKG2Ur7tIB2TV6qkUDXcJ3YkpXgu6pQ0t76MMavjoyx8bp68pponPNGewMpilXRLPVXO8btYh70V3b8KQ==

#### 进行连接



### SSH服务认证类型

基于密码的认证类型

基于密钥(钥匙锁头)的安全验证

### ssh配置文件修改-默认登录参数

# ssh配置文件默认路径信息

/etc/ssh/sshd\_config --- sshd服务的配置文件

井号注释的参数信息，表示默认配置

Port 52113 --- 表示修改ssh服务端口号

ListenAddress 10.0.0.41 --- 表示指定监听地址信息

PermitRootLogin no --- root超级用户黑客都知道，建议禁止它远程登陆能力。

PermitEmptyPasswords no --- 禁止空密码登陆

UseDNS no --- 不使用DNS 172.16.1.41（ip----域名 主机名）

GSSAPIAuthentication no --- http://oldboy.blog.51cto.com/2561410/1300964

## ssh服务相关命令用法：

### ssh远程登录命令

ssh -p22 oldboy@10.0.0.150 [命令]

#→SSH连接远程主机命令的基本语法;

#→-p（小写）接端口，默认22端口时可以省略-p22;

#→“@”前面为用户名，如果用当前用户连接，可以不指定用户。

#→“@”后面为要连接的服务器的IP. 更多用法，请man ssh;

### scp远程复制数据命令

#### 推：PUSH

scp -P22 -rp /tmp/oldboy oldboy@10.0.0.143:/tmp

<- -P（大写，注意和ssh命令的不同）接端口，默认22端口时可以省略-P22;

<- -r递归，表示拷贝目录;

<- -p表示在拷贝前后保持文件或目录属性;

<- -l limit 限制速度。 rsync限速 --bwlimit

<- /tmp/oldboy为本地的目录。“@”前为用户名，“@”后为要连接的服务器的IP。

IP后的:/tmp目录，为远端的目标目录。

说明：以上命令作用是把本地/tmp/oldboy拷贝到远端服务器10.0.0.143的/tmp目录;

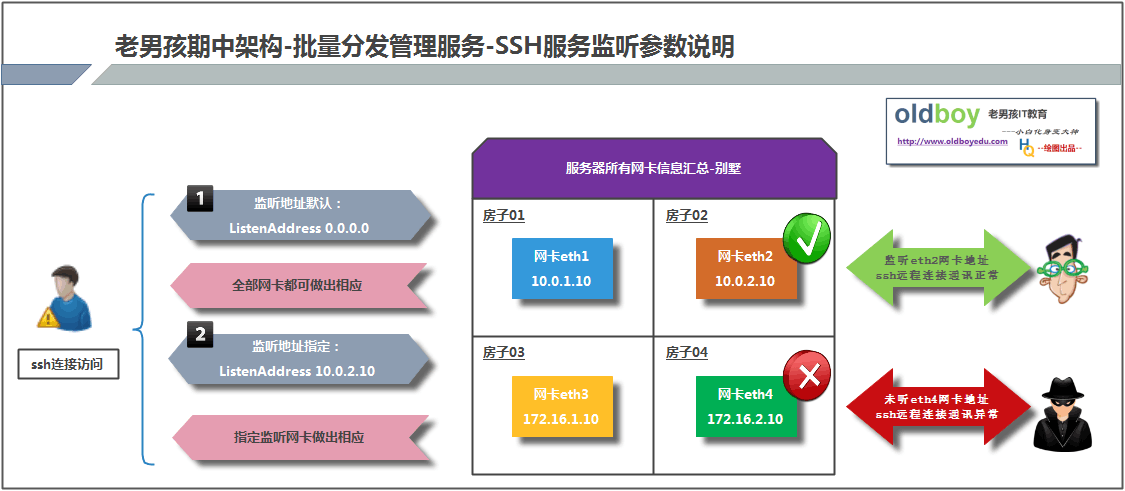
#### 拉：PULL

scp -P22 -rp root@10.0.0.7:/tmp/oldboy /opt/

说明：还可以把远端目录抓到本地

结论：scp为远程拷贝文件或目录的命令，更多用法，请man scp；拷贝权限为连接的用户对应的权限。

#### 修改ssh 服务监听参数



13 #Port 22

14 #AddressFamily any

15 #ListenAddress 0.0.0.0 监听默认地址 ，当系统默认时全部地址都可以访问连接，当修改为某一个具体地址时，只能指定地址访问连接，只能设置这两种。网段不可设定

16 #ListenAddress ::

#### 修改ssh 端口

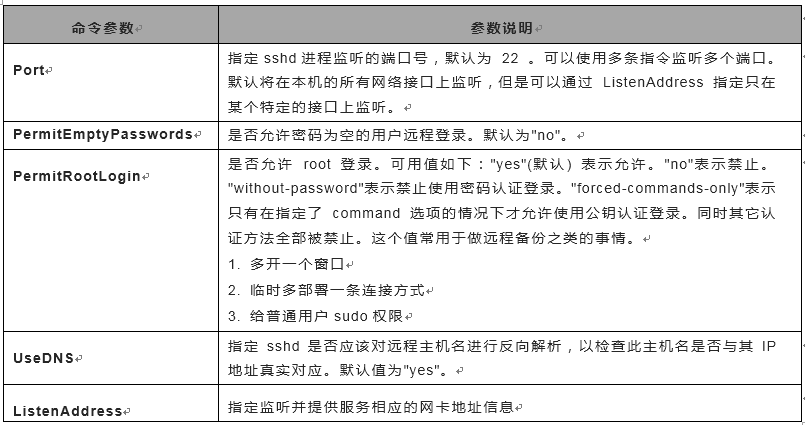
13 #Port 22

14 #AddressFamily any

15 #ListenAddress 0.0.0.0

16 #ListenAddress ::

17



### sftp数据传输命令

sftp -oPort=52113 root@10.0.0.41 <- 特殊端口的sftp连接命令

help --- Display this help text

显示帮助信息

pwd --- Display remote working directory

显示远程工作目录

ls [-1aflnrSt] [path] --- Display remote directory listing

显示远端主机目录中的文件列表信息

lpwd --- Print local working directory

lls [ls-options [path]] --- Display local directory listing

cd path --- Change remote directory to 'path'

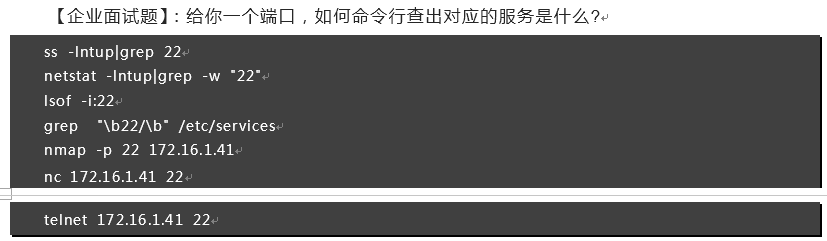
lcd path --- Change local directory to 'path'

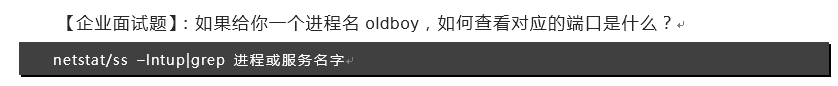
put [-P] local-path [remote-path] --- Upload file

rm path --- Delete remote file

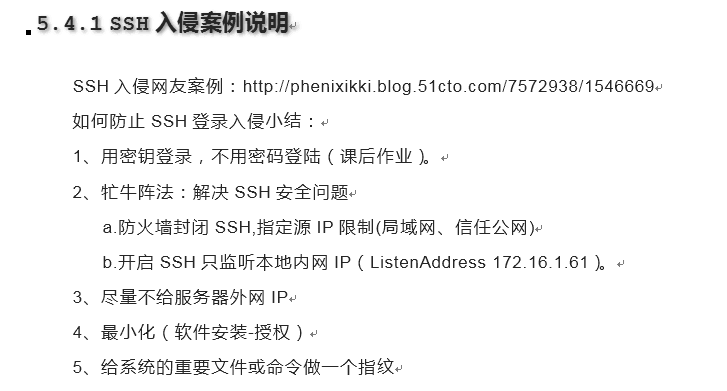
get [-P] remote-path [local-path] --- Download file

### 企业面试题





### ssh 入侵案例



# SSH key基于秘钥的远程登录方式

## 部署过程：

### 第一个里程碑：在管理服务器上创建秘钥对（公钥和私钥）：

[root@web01 .ssh]# ssh-keygen -t dsa

Generating public/private dsa key pair.

Enter file in which to save the key (/root/.ssh/id\_dsa): <-- 表示确认私钥创建后保存路径位置

Enter passphrase (empty for no passphrase): <-- 表示确认私钥需不需要进行加密，设置密码，测试环境设置空

Enter same passphrase again: <-- 表示确认私钥需不需要进行加密，设置密码，测试环境设置空

Your identification has been saved in /root/.ssh/id\_dsa.

Your public key has been saved in /root/.ssh/id\_dsa.pub.

The key fingerprint is:

d1:96:4a:8c:bc:97:49:c3:cd:3b:9b:ec:ed:02:82:61 root@web01

The key's randomart image is:

+--[ DSA 1024]----+

| |

| . + + . |

| o B \* |

| E + B . |

| . o. S o |

| . .... + |

| . .+ |

| ... |

| .oo |

+-----------------+

[root@web01 .ssh]# ll

total 8

-rw------- 1 root root 668 Sep 6 15:14 id\_dsa <-- 创建生成私钥（钥匙）文件

-rw-r--r-- 1 root root 600 Sep 6 15:14 id\_dsa.pub <-- 创建生成公钥（锁头）文件

### 第二个里程碑，分发公钥

#### 密钥分发执行公式

ssh-copy-id - install your public key in a remote machine’s authorized\_keys

ssh-copy-id [-i [identity\_file]] [user@]machine

-i ---指定要分发的公钥文件以及路径信息

[user@] ---以什么用户身份进行分发公钥（root）,如果不输入，表示以当前系统用户身份分发公钥

machine ---将公钥分发到哪台主机上，填写远程主机IP地址

#### 分发公钥给客户端

[root@web01 .ssh]# ssh-copy-id -i /root/.ssh/id\_dsa.pub root@172.16.1.41

The authenticity of host '172.16.1.41 (172.16.1.41)' can't be established.

RSA key fingerprint is e0:4d:55:05:80:02:15:d6:63:23:0c:7f:5c:13:d3:e2.

Are you sure you want to continue connecting (yes/no)? yes 继续选择yes

Warning: Permanently added '172.16.1.41' (RSA) to the list of known hosts.

root@172.16.1.41's password: 需要输入密码

Now try logging into the machine, with "ssh 'root@172.16.1.41'", and check in:

.ssh/authorized\_keys

to make sure we haven't added extra keys that you weren't expecting.

### 第三个里程碑，基于密钥登录测试

[root@web01 .ssh]# ssh 172.16.1.41

Last login: Wed Sep 6 13:49:34 2017 from 10.0.0.1

[root@backup ~]# exit

logout

Connection to 172.16.1.41 closed.

## 批量管理主机，远程执行命令

### 远程操纵主机执行命令

[root@m01 scripts]# ssh 172.16.1.31 "date"

Wed Sep 6 15:41:48 CST 2017

### 批量管理主机

[root@m01 scripts]# sh ssh.sh

Wed Sep 6 15:47:46 CST 2017

Wed Sep 6 15:47:46 CST 2017

Wed Sep 6 15:47:46 CST 2017

## 端口不是22条件下怎么执行

[root@m01 scripts]# ssh-copy-id -i /root/.ssh/id\_dsa.pub root@172.16.1.41

ssh: connect to host 172.16.1.41 port 22: Connection refused

操作失败原因，客户端端口不是默认的22

### 查看ssh-copy-id 命令存放位置

[root@m01 scripts]# which ssh-copy-id

/usr/bin/ssh-copy-id

[root@m01 scripts]# file /usr/bin/ssh-copy-id

/usr/bin/ssh-copy-id: POSIX shell script text executable

ssh $1 "exec sh -c 'cd; umask 077; test -d .ssh || mkdir .ssh ; cat >> .ssh/authorized\_keys && (te

st -x /sbin/restorecon && /sbin/restorecon .ssh .ssh/authorized\_keys >/dev/null 2>&1 || true)'" || exit 1

说明：分发公钥的实质，就是通过ssh远程执行分发公钥命令

01. 切换到用户家目录下，临时设置umask值

02. 判断客户端相应用户家目录中有没有.ssh目录，如果没有.ssh目录就进行创建

03. 将管理端公钥文件中的内容添加到客户端~/.ssh/authorized\_keys,默认authorized\_keys不存在，需要创建，文件权限（600）

### 第一种解决方法

修改ssh-copy-id文件中内容：

ssh -p52113 $1 "exec sh -c 'cd; umask 077; test -d .ssh || mkdir .ssh ; cat >> .ssh/authorized\_keys && (te

st -x /sbin/restorecon && /sbin/restorecon .ssh .ssh/authorized\_keys >/dev/null 2>&1 || true)'" || exit 1

再次执行分发公钥命令即可

### 第二种解决方法

修改ssh-copy-id命令中传参的参数信息：

[root@m01 scripts]# ssh-copy-id -i /root/.ssh/id\_dsa.pub "-p52113 [root@172.16.1.41](mailto:root@172.16.1.41)" -p 指定端口，将端口与接收端的IP用双引号引起，代表一个整体。"-p52113 [root@172.16.1.41" 会默认为$1](mailto:root@172.16.1.41%22%20会默认为$1)变量，涉及shift将传参的参数向前递进

## until语句中的shift用法

### until 语句，只要条件为真就不执行，为假就一直执行

[root@m01 scripts]# vim shift.sh

#!/bin/bash

until [ $# -eq 0 ]

do

echo $\*

shift

done

### shift将传参的参数向前递进（表示执行一次就把第一个参数删除，直到最后一个）

[root@m01 scripts]# sh shift.sh 1 2 3 4 5 6

1 2 3 4 5 6

2 3 4 5 6

3 4 5 6

4 5 6

5 6

6

## while语句

只要条件为真就一直执行，为假就不执行

[root@m01 scripts]# cat shift.sh

#!/bin/bash

while [ $# -gt 0 ]

do

echo $\*

shift

done

[root@m01 scripts]# sh shift.sh 1 2 3 4 5 6

1 2 3 4 5 6

2 3 4 5 6

3 4 5 6

4 5 6

5 6

6

## 阻碍脚本运行的因素：

### 创建秘钥对时，需要进行交互

a. 指定创建后的私钥文件保存路径

b. 指定私钥文件加密的密码信息

#### 解决创建秘钥对时，需要进行交互问题

ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P "" 选择默认目录，密码设置为空

[root@m01 .ssh]# ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P ""

Generating public/private dsa key pair.

Your identification has been saved in /root/.ssh/id\_dsa.

Your public key has been saved in /root/.ssh/id\_dsa.pub.

The key fingerprint is:

6d:fa:d7:dc:ed:1e:cb:10:5a:eb:2c:25:ae:17:02:3b root@m01

The key's randomart image is:

+--[ DSA 1024]----+

| |

| |

| |

| . . |

| S o o |

| E + oo.o |

| o o.+= o.|

| . ++ = =|

| .+..o =o|

+-----------------+

ssh-keygen -t dsa -f /root/.ssh/id\_dsa -N ""

ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P "" &>/dev/null

ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P "" -q

#### ssh-keygen命令参数：

-f filename 遇到指定密钥文件的文件名时，直接跳过

Specifies the filename of the key file.

-P passphrase

Provides the (old) passphrase.

-N new\_passphrase 跳过设置密码过程

Provides the new passphrase.

-q Silence ssh-keygen. Used by /etc/rc when creating a new key. 不显示执行过程

#### 交互问题解决

[root@m01 .ssh]# ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P "" -q

[root@m01 .ssh]#

### 分发公钥时，需要进行交互问题

#### 需要有yes/no确认交互问题

ssh-copy-id -i /root/.ssh/id\_dsa.pub "root@172.16.1.41" == ssh $1 = -o StrictHostKeyChecking

ssh-copy-id -i /root/.ssh/id\_dsa.pub "-o StrictHostKeyChecking=no root@172.16.1.41" 参数表示不进行交互

#### 第一次分发公钥需要输入密码

首先需要安装sshpass软件

yum install -y sshpass

安装完成再执行

sshpass -p123456 ssh-copy-id -i /root/.ssh/id\_dsa.pub "-o StrictHostKeyChecking=no [root@172.16.1.31](mailto:root@172.16.1.31)" sshpass -p 指定登录密码

#### 交互问题解决

[root@m01 .ssh]# sshpass -p123456 ssh-copy-id -i /root/.ssh/id\_dsa.pub "-o StrictHostKeyChecking=no root@172.16.1.31"

Now try logging into the machine, with "ssh '-o StrictHostKeyChecking=no root@172.16.1.31'", and check in:

.ssh/authorized\_keys

to make sure we haven't added extra keys that you weren't expecting.

## 编写批量分发公钥脚本

#!/bin/bash

# create key pair

\rm -f /root/.ssh/id\_dsa\* 删除曾经执行生成的私钥公钥

ssh-keygen -t dsa -f /root/.ssh/id\_dsa -P "" -q

# fenfa key file

for ip in 8 31 41

do

echo "==== fenfa key to host 172.16.1.$ip======================"

sshpass -p123456 ssh-copy-id -i /root/.ssh/id\_dsa.pub "-o StrictHostKeyChecking=no root@172.16.1.$ip"

echo "=================== fenfa end ==========================="

echo ""

done

### 编写批量管理脚本

#!/bin/bash

CMD="$1"

for ip in 8 31 41

do

echo "====host 172.16.1.$ip info===="

ssh 172.16.1.$ip "$CMD"

echo "=============================="

echo ""

done

至此：ssh key批量分发与批量管理功能就实现了

补充：基于秘钥登录方式，分发公钥文件会识别用户信息

## 补充：：ssh 语法检查方法

[root@m01 ansible-playbook]# /usr/sbin/sshd -t /etc/ssh/sshd\_config

Extra argument /etc/ssh/sshd\_config