ssh-keygen的使用方法

# 概述

## 就是为了让两个Linux机器之间使用ssh不需要用户名和密码。采用了数字签名RSA或者DSA来完成这个操作

## 模型分析

假设 A （192.168.20.59）为客户机器，B（192.168.20.60）为目标机；

要达到的目的：A机器ssh登录B机器无需输入密码；**加密方式选 rsa|dsa均可以，默认dsa**

# 具体实现流程

## 单向登陆的操作过程（能满足上边的目的）：

1、登录A机器

ssh-keygen -t [rsa|dsa]，**将会生成密钥文件和私钥文件** id\_rsa,id\_rsa.pub或id\_dsa,id\_dsa.pub

将 .pub 文件复制到B机器的 .ssh 目录， 并 cat id\_dsa.pub >> ~/.ssh/authorized\_keys

大功告成，从A机器登录B机器的目标账户，不再需要密码了；（直接运行 #ssh 192.168.20.60 ）

## 双向登陆的操作过程：

ssh-keygen做密码验证可以使在向对方机器上ssh ,scp不用使用密码.具体方法如下:

两个节点都执行操作：#ssh-keygen -t rsa

然后全部回车,采用默认值.

这样生成了一对密钥，存放在**用户目录的~/.ssh**下。

将公钥考到对方机器的用户目录下 ，并将其复制到~/.ssh/authorized\_keys中（操作命令：#cat id\_dsa.pub >> ~/.ssh/authorized\_keys ）。

## 设置文件和目录权限：

设置authorized\_keys权限  
$ chmod 600 authorized\_keys   
设置.ssh目录权限  
$ chmod 700 -R .ssh

## 要保证.ssh和authorized\_keys都只有用户自己有写权限。否则验证无效。（今天就是遇到这个问题，找了好久问题所在），其实仔细想想，这样做是为了不会出现系统漏洞。

# 总结

## 文件和目录的权限千万别设置成chmod 777.这个权限太大了，

不安全，数字签名也不支持。

## 生成的rsa/dsa签名的公钥是给对方机器使用的。这个公钥内容还要拷贝到authorized\_keys

## linux之间的访问直接 ssh 机器ip

## 某个机器生成自己的RSA或者DSA的数字签名，

将公钥给目标机器，然后目标机器接收后设定相关权限（公钥和authorized\_keys权限），这个目标机就能被生成数字签名的机器无密码访问了