ssh_config 文件配置详解

ssh_config详解

配置"/etc/ssh/ssh_config"文件

"/etc/ssh/ssh_config" 文件是OpenSSH系统范围的配置文件，允许你通过设 置不同的选项来改变客户端程序的运行方式。这个文件的每一行包含"关键词－值"的匹配，其中"关键词"是忽略大小写的。下面列出来的是最重要的关键词，用 man命令查看帮助页（ssh (1)）可以得到详细的列表。

编辑"ssh_config"文件（vi /etc/ssh/ssh_config），添加或改变下面的参数：

```
# Site-wide defaults for various options
Host *
ForwardAgent no
ForwardX11 no
RhostsAuthentication no
RhostsRSAAuthentication no
RSAAuthentication yes
PasswordAuthentication yes
FallBackToRsh no
UseRsh no
BatchMode no
```

CheckHostIP yes
StrictHostKeyChecking no
IdentityFile ~/.ssh/identity
Port 22
Cipher blowfish
EscapeChar ~

下面逐行说明上面的选项设置：

Host *
选项“Host”只对能够匹配后面字串的计算机有效。“*”表示 所有的计算机。

ForwardAgent no
“ForwardAgent”设置连接是否经过验证代理（如果存在）转发给远程计 算机。

ForwardX11 no
“ForwardX11”设置X11连接是否被自动重定向到安全的通道和显示集 （DISPLAY set）。

RhostsAuthentication no
“RhostsAuthentication”设 置是否使用基于rhosts的安全验证。

RhostsRSAAuthentication no
“RhostsRSAAuthentication” 设置是否使用用RSA算法的基于rhosts的安全验证。

RSAAuthentication yes
“RSAAuthentication” 设置是否使用RSA算法进行安全验证。

PasswordAuthentication yes
“PasswordAuthentication” 设置是否使用口令验证。

FallBackToRsh no
“FallBackToRsh”设置如果用ssh连接出现错误是否自动 使用rsh。

UseRsh no
“UseRsh”设置是否在这台计算机上使用“rlogin/rsh”。

BatchMode no
“BatchMode”如果设为“yes”，passphrase/password（交互式输入口令）的提示将被禁止。当不能交互式输入 口令的时候，这个选项对脚本文件和批处理任务十分有用。

CheckHostIP yes
“CheckHostIP”设置ssh是 否查看连接到服务器的主机的IP地址以防止DNS欺骗。建议设置为“yes”。

StrictHostKeyChecking no
“StrictHostKeyChecking” 如果设置成“yes”，ssh就不会自动把计算机的密匙加入“$HOME/.ssh/known_hosts”文件，并且一旦计算机的密匙发生了变化，就 拒绝连接。

IdentityFile ~/.ssh/identity
“IdentityFile”设置从哪个文件读取用户的 RSA安全验证标识。

Port 22
“Port”设置连接到远程主机的端口。

Cipher blowfish
“Cipher” 设置加密用的密码。

EscapeChar ~
“EscapeChar”设置escape字符。

配置“/etc/ssh/sshd_config”文件

“/etc/ssh /sshd_config”是OpenSSH的配置文件，允许设置选项改变这个daemon的运行。这个文件的每一行包含“关键词－值”的 匹配，其中“关键词”是忽略大小写的。下面列出来的是最重要的关键词，用man命令查看帮助页（sshd (8)）可以得到详细的列表。

编 辑“sshd_config”文件（vi /etc/ssh/sshd_config），加入或改变下面的参数：

```
# This is ssh server systemwide configuration file.
Port 22
ListenAddress 192.168.1.1
HostKey /etc/ssh/ssh_host_key
ServerKeyBits 1024
LoginGraceTime 600
KeyRegenerationInterval 3600
PermitRootLogin no
IgnoreRhosts yes
IgnoreUserKnownHosts yes
StrictModes yes
X11Forwarding no
PrintMotd yes
```

SyslogFacility AUTH
LogLevel INFO
RhostsAuthentication no
RhostsRSAAuthentication no
RSAAuthentication yes
PasswordAuthentication yes
PermitEmptyPasswords no
AllowUsers admin

下面逐行说明上面的选项设置：

Port 22
“Port”设置 sshd监听的端口号。

ListenAddress 192.168.1.1
“ListenAddress”设置sshd服务器 绑定的IP地址。

HostKey /etc/ssh/ssh_host_key

“HostKey”设置包含计算机私人密 匙的文件。

ServerKeyBits 1024
“ServerKeyBits”定义服务器密匙的位数。

LoginGraceTime 600
“LoginGraceTime”设置如果用户不能成功登录，在切断连接之前服务器需要等待的时间（以秒为单位）。

KeyRegenerationInterval 3600
“KeyRegenerationInterval”设置在多少秒之后自动重新生成服务器的密匙（如果使用密匙）。重新生成密匙是为了 防止用盗用的密匙解密被截获的信息。

PermitRootLogin no
“PermitRootLogin”设置root能不 能用ssh登录。这个选项一定不要设成“yes”。

IgnoreRhosts yes
“IgnoreRhosts”设置验证的时 候是否使用“rhosts”和“shosts”文件。

IgnoreUserKnownHosts yes
“IgnoreUserKnownHosts” 设置ssh daemon是否在进行RhostsRSAAuthentication安全验证的时候忽略用户的“$HOME/.ssh/known_hosts”

StrictModes yes
“StrictModes”设置ssh在接收登录请求之前是否检查用户家目录和rhosts文件的权限和所有权。这通常是必要的，因为新 手经常会把自己的目录和文件设成任何人都有写权限。

X11Forwarding no
“X11Forwarding”设置是否允 许X11转发。

PrintMotd yes
“PrintMotd”设置sshd是否在用户登录的时候显示“/etc/motd” 中的信息。

SyslogFacility AUTH
“SyslogFacility”设置在记录来自sshd的消息的时候，是否 给出“facility code”。

LogLevel INFO
“LogLevel”设置记录sshd日志消息的层次。 INFO是一个好的选择。查看sshd的man帮助页，已获取更多的信息。

RhostsAuthentication no
“RhostsAuthentication” 设置只用rhosts或“/etc/hosts.equiv”进行安全验证是否已经足够了。

RhostsRSAAuthentication no
“RhostsRSA”设置是否允许用rhosts或“/etc/hosts.equiv”加上RSA进行安全验证。

RSAAuthentication yes
“RSAAuthentication”设置是否允许只有RSA安全验证。

PasswordAuthentication yes
“PasswordAuthentication”设置是否允许口令验证。

PermitEmptyPasswords no
“PermitEmptyPasswords”设置是否允许用口令为空的帐号登录。

AllowUsers admin
“AllowUsers” 的后面可以跟着任意的数量的用户名的匹配串（patterns）或user@host这样的匹配串，这些字符串 用空格隔开。主机名可以是DNS名或IP地址。
使用SFTP代替FTP传输文件
FTP(文件传输协议)是一种使用非常 广泛的在网络中传输文件的方式，但是，它也同样存在被网络窃听的危险，因为它也是以明文传送用户认证信息。其实在SSH软件包中，已经包含了一个叫作 SFTP(Secure FTP)的安全文件传输子系统，SFTP本身没有单独的守护进程，它必须使用sshd守护进程（端口号默认是22）来完成相应的连接操作，所以从某种意义 上来说，SFTP并不像一个服务器程序，而更像是一个客户端程序。SFTP同样是使用加密传输认证信息和传输的数据，所以，使用SFTP是非常安全的。但 是，由于这种传输方式使用了加密/解密技术，所

以传输效率比普通的FTP要低得多，如果您对网络安全性要求更高时，可以使用SFTP代替FTP。若要开启 SFTP功能可以修改sshd2_config文件的下列内容：

subsystem-sftp sftp-server

去掉行首的“#”，然后重新启动SSH服务器，这样在进行SSH连接时就可以同时使用SFTP传输文件。

关于客户端设置

以上是对服务器的设置，其实在SSH服务器 中已经包含了一些客户端工具（如SSH,SFTP工具）。但是，更多的客户端用户使用Windows系统，下面就对Windows上的客户端系统设置加以 说明。

首先从上文给出的网址下载“SSHSecureShellClient-3.2.3.exe”文件并安装。安装完成后，在桌面上会产成两个 快捷方式，一个是“SSH Secure Shell Client”，用于远程管理，另一个是“SSH Secure File Transfer Client”，用于和服务器进行文件传输。在工具栏中点击“quick connnect”，输入正确的主机名和用户名，然后在弹出的对话框中输入密码完成登录，即可开始执行命令或者传输文件。在使用SFTP时，默认只能显示 用户的宿主目录的内容和非隐藏文件。但是，有时候您可能还要查看其它目录或者隐藏文件，这时只需要在菜单“eidt->setting-> file transfer”的选项中选中“show root directory”和“show hidden file”两个选项即可。

使 普通用户仅使用SFTP而没有使用Shell的权限

默认情况下管理员给系统添加的账号将同时具有SFTP和SSH的权限。让普通用户使用 shell执行命令也是有很大的安全隐患的，如果能够禁止用户使用shell执行命令而仅使用SFTP传输文件，就能消除这种安全隐患，完全实现FTP的 功能，

正如上文所述，SFTP没有单独的守护进程，只能借助于sshd守护进程，所以我们仍然需要使用SSH服务器，要保证sshd守护进程处于 运行状态。具体实现方法如下：

首先，在编译安装时，编译中一定要有“--enable-static” 选项。安装成功后，在安装目录下的bin目录中执行下面的命令：

[root@localhost bin]# ls -l ssh-dummy-shell* sftp-server2*

将看到下列输出内容：

-rwxr-xr-x 1 root root 1350417 Apr 28 16:30 sftp-server2
-rwxr-xr-x 1 root root 3566890 Apr 28 16:30 sftp-server2.static
-rwxr-xr-x 1 root root 72388 Apr 28 16:30 ssh-dummy-shell
-rwxr-xr-x 1 root root 1813412 Apr 28 16:30 ssh-dummy-shell.static

其中带“static”后缀名，且比较大的两个文件就是加上“--enable-static”选 项后生成的，后面我们将用到这里两个文件。

下面以添加普通账号test为例讲述具体操作步骤。

1．在“/home”目录（或者将要存放普 通用户宿主目录的目录）下创建“bin”子目录，并将两个static文件复制到此目录下（复制后改名去掉static后缀），执行如下命令：

[root@localhost bin]# cd /usr/local/ssh3.2/bin
[root@localhost bin]#cp ssh-dummy-shell.static /home/bin/ssh-dummy-shell
[root@localhost bin]# cp sftp-server2.static /home/bin/sftp-server
[root@localhost bin]#chown -R root.root /home/bin
[root@localhost bin]#chmod -R 755 /home/bin

2．添加一个组，使以后所有禁止使用shell的用户都属于这个组，这样便于管理更多的用户：

[root@localhost bin]#groupadd template

3．在添加系统账号时使用如下命令：

[root@localhost root]#useradd -s /bin/ssh-dummy-shell -g template test
[root@localhost root]#passwd test
[root@localhost root]#mkdir /home/test/bin

[root@localhost root]#cd /home/test/bin
[root@localhost bin]#ln /home/bin/ssh-dummy-shell ssh-dummy-shell
[root@localhost bin]#ln /home/bin/sftp-server sftp-server
[root@localhost bin]#chown -R root.root /home/test/bin
[root@localhost bin]#chmod -R 755 /home/test/bin
3．用户添加成功后，还需要修改/etc/ssh2/sshd2_config文件，将下列内容：
#ChRootGroups sftp,guest
改为：
ChRootGroups sftp,guest,template
修改上面这行内容，主要是为了 禁止普通用户查看系统的其它目录，把其权限限制在自己的主目录下。重新启动SSH服务器程序，在客户端使用SSH Secure File Transfer Client登录，即使选择显示根目录，普通用户也看不到其它的任何目录，而是把自己的主目录当作根目录。注意，这里使用的是按用户所属组限制，这样可以 使包含在template组内的所有用户都可以实现此功能。若您只要限制个别用户的话，可以修改下面的内容：
#ChRootUsers anonymous,ftp,guest

事实证明SSH是一种非常好的网络安全解决方案，但是，目前仍有很多管理员使用Telnet或 FTP这种非常不安全的工具，希望尽快转移到SSH上来，以减少网络安全隐患。

原文地址 http://blog.chinaunix.net/u1/35016/showart_494442.html

发表于： 2010-04-12 ，修改于： 2010-04-12 10:57，已浏览155次，有评论0条 推荐 投诉

网友评论

发表评论