ssh

openssh套件中的客户端连接工具

补充说明

ssh命令 是openssh套件中的客户端连接工具，可以给予ssh加密协议实现安全的远程登录服务器。

语法

```
ssh(选项)(参数)
```

选项

```
-1：强制使用ssh协议版本1；
-2：强制使用ssh协议版本2；
-4：强制使用IPv4地址；
-6：强制使用IPv6地址；
-A：开启认证代理连接转发功能；
-a：关闭认证代理连接转发功能；
-b：使用本机指定地址作为对应连接的源ip地址；
-C：请求压缩所有数据；
-F：指定ssh指令的配置文件；
-f：后台执行ssh指令；
-g：允许远程主机连接主机的转发端口；
-i：指定身份文件；
-l：指定连接远程服务器登录用户名；
-N：不执行远程指令；
-o：指定配置选项；
-p：指定远程服务器上的端口；
-q：静默模式；
-X：开启X11转发功能；
-x：关闭X11转发功能；
-y：开启信任X11转发功能。
```

参数

- 远程主机：指定要连接的远程ssh服务器；
- 指令：要在远程ssh服务器上执行的指令。

实例

```
# ssh 用户名@远程服务器地址
ssh user1@172.24.210.101
# 指定端口
ssh -p 2211 root@140.206.185.170

# ssh 大家族
ssh user@ip -p22 # 默认用户名为当前用户名，默认端口为 22
ssh-keygen # 为当前用户生成 ssh 公钥 + 私钥
ssh-keygen -f keyfile -i -m key_format -e -m key_format # key_format:
```

```
RFC4716/SSH2(default) PKCS8 PEM
ssh-copy-id user@ip:port # 将当前用户的公钥复制到需要 ssh 的服务器的
~/.ssh/authorized_keys之后可以免密登录
```

背后故事

> 英文Tatu Ylonen
> 编译Linux中国/kenxx
> 来源：https://linux.cn/article-8476-1.html

为什么 SSH安全终端）的端口号是 22 呢，这不是一个巧合，这其中有个我Tatu YlonenSSH 协议的设计者）未曾诉说的故事。

将 SSH 协议端口号设为 22 的故事

1995 年春我编写了 SSH 协议的最初版本，那时候 telnet 和 FTP 正被广泛使用。

当时我设计 SSH 协议想着是为了替代 telnet端口 23）和 ftp 的端口中间的数字。我觉得端口号虽然是个小事但似乎又存在着某种信念。但我到底要怎么拿到那个端口号呢？我未曾拥有过任何一个端口号，但我却认识几个拥有端口号的人！

在那时取得端口号的事情其实说来挺简单的。毕竟当时的因特网Internet并不是很大，是因特网爆炸的早期。端口号分配的活儿是 IANAInternet Assigned Numbers Authority互联网数字分配机构）干的。在那时这机构可相当于是因特网先驱 Jon Postel 和 Joyce K. Reynolds 一般的存在Jon 参与编写了多项主要的协议标准，例如 IPRFC 791ICMPRFC 792和 TCPRFC 793等一些你应该早有耳闻的协议。

我可以说是敬畏 Jon 先生的，他参与编写了几乎所有主要的因特网标准文档Internet RFC

1995 年 7 月，就在我发布 ssh-1.0 前，我发送了一封邮件给 IANA

> From ylo Mon Jul 10 11:45:48 +0300 1995
> From: Tatu Ylonen
> To: Internet Assigned Numbers Authority
> Subject: 请求取得一个端口号
> Organization: 芬兰赫尔辛基理工大学
>
> 亲爱的机构成员：
>
> 我写了个可以在不安全的网络环境中安全地从一台机器登录到另一台机器的程序。它主要是对现有的 telnet 协议以及 rlogin 协议的功能性提升和安全性改进。说的具体些，就是可以防御 IPDNS > 或路由等欺骗行为。我打算将我的软件免费地发布在因特网上，以得到广泛地使用。
>
> 我希望为该软件注册一个特权端口号，要是这个端口号在 1 到 255 > 之间就更好了，这样它就可以用在名字服务器的 WKS 字段中了。
>
> 我在附件中附上了协议标准的草案。这个软件已经在本地运行了几个月了，我已准备在获得端口号后就发布。如果端口号分配一事安排的及时，我希望这周就将要发布的软件准备好。我目前在 beta 版测试时使用的端口号是 > 22，如果要是能够分配到这个端口，我就不用做什么更改了（目前这个端口在列表中还是空闲的）。
>
> 软件中服务的名称叫 sshsecure Shell 的缩写）。

> 您最真诚的，
> Tatu Ylonen

LCTT 译注DNS 协议中的 WKS 记录类型意即“ 众所周知的业务描述 ”， 是类似于 AMX 这样的 DNS 记录类型，用于描述某个 IP 所提供的服务，目前鲜见使用。参见：https://docs.oracle.com/cd/E19683-01/806-4077/dnsintro-154/index.html

第二天，我就收到了 Joyce 发来的邮件：

> Date: Mon, 10 Jul 1995 15:35:33 -0700
> From: jkrey@ISI.EDU
> To: ylo@cs.hut.fi
> Subject: 回复：请求取得一个端口号
> Cc: iana@ISI.EDU
> Tatu,
> 我们将端口号 22 分配给 ssh 服务了，你目前是该服务的主要联系人。
> Joyce

这就搞定了SSH 的端口正式使用 22！！！

1995 年 7 月 12 日上午 2 点 21 分，我给我在赫尔辛基理工大学的测试者们宣布了 SSH 的最后 beta 版本。当日下午 5 点 23 分，我给测试者们宣布了 ssh-1.0.0 版本。1995 年 7 月 12 日，下午 5 点 51 分，我将一份 SSH安全终端）的宣告发给了 cypherpunks@toad.com 的邮件列表，此外我还将其发给了一些新闻组、邮件列表和一些在因特网上讨论相关话题的人们。

如何更改 SSH 服务的端口号

SSH 服务器是默认运行在 22 号端口上的。然而，由于某些原因需要，它也可以运行在别的端口上。比如为了方便测试使用，又比如在同一个宿主机上运行多个不同的配置。当然，极少情况下，不使用 root 权限运行它也可以，比如某些必须运行在非特权的端口的情况（端口号大于等于 1024）。

端口号可以在配置文件 /etc/ssh/sshd_config 中将 Port 22 更改。也可以使用 -p 选项运行 sshdSSH 客户端和 sftp 程序也可以使用 -p 选项。

配置 SSH 协议穿越防火墙

SSH 是少数通常被许可穿越防火墙的协议之一。通常的做法是不限制出站的 SSH 连接，尤其常见于一些较小的或者比较技术型的组织中，而入站的 SSH 连接通常会限制到一台或者是少数几台服务器上。

出站的 SSH 连接

在防火墙中配置出站的 SSH 连接十分简单。如果完全限制了外发连接，那么只需要创建一个允许 TCP 端口 22 可以外发的规则即可。如果你想限制目标地址，你可以限制该规则仅允许访问你的组织放在云端的外部服务器或保护该云端的跳板服务器即可。

反向通道是有风险的

其实不限制出站的 SSH 连接虽然是可以的，但是是存在风险的SSH 协议是支持 通道访问 的。最初的想法是在外部服务器搭建一个 SSH 服务监听来自各处的连接，将进入的连接转发到组织，并让这个连接可以访问某个内部服务器。

在某些场景下这当然非常的方便。开发者和系统管理员经常使用它打开一个通道以便于他们可以远程访问，比如在家里或者在旅行中使用笔记本电脑等场景。

然而通常来讲这些做法是违背安全策略的，跳过了防火墙管理员和安全团队保护的控制无疑是违背安全策略的，比如这些 PCI、HIPAA、NIST SP 800-53 等。它可以被黑客和外国情报机构用来在组织内留下后门。

CryptoAuditor 是一款可以控制通道穿过防火墙或者一组云端服务器入口的产品。该款产品可以配合 通用 SSH 密钥管理器（Universal SSH Key Manager） 来获得对 主机密钥（host keys）的访问，以在启用防火墙并阻挡未授权转发的场景中解密 SSH 会话。

入站的 SSH 访问

对于入站访问而言，这里有几点需要说一下：

配置防火墙，并转发所有去往 22 端口的连接只能流向到一个特定的内部网络 IP 地址或者一个 DMZ 主机。在该 IP 上运行 CryptoAuditor 或者跳板机来控制和审查所有访问该组织的连接。 在防火墙上使用不同的端口访问不同的服务器。 只允许使用 IPsec 协议这样的 VPN（虚拟专用网）登录后连接 SSH 服务。

通过 iptables 服务限制 SSH 访问

iptables 是一款内建在 Linux 内核的宿主防火墙。通常配置用于保护服务器以防止被访问那些未明确开启的端口。

如果服务器上启用了 iptables，使用下面的命令将可以允许进入的 SSH 访问，当然命令需要以 root 身份运行。

```
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -
j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -m conntrack --ctstate ESTABLISHED -j
ACCEPT
```

如果你想将上述命令创建的规则持久地保存，在某些系统版本中，可使用如下命令：

```
service iptables save
```