

第 15 章 远 程 登 录

服务器是一些大家伙，需要专门的机房来存放。一些企业有自己的服务器机房，而更多的选择是把这些方盒子交给服务器托管商保管。无论是哪一种情况，没有一个网络管理员会选择把机房作为自己的办公室——那里应该是机器们的地盘。为此，管理员们总是使用“远程登录”的方式管理服务器。一个流行的说法是，最优秀的网络管理员应该让公司里的其他人不认识他。坐在自己的 PC 前，让远隔千里的服务器永远稳定地运行——这是每一个网络管理员的梦想和使命。

15.1 快速上手：关于搭建实验环境

本章主要是介绍如何使用客户端程序登录到远程服务器的，不过这首先需要有一台“远程服务器”才行。如果读者学习了本章后不能亲自动手，那么阅读本身就只是浪费时间了。因此本章的“快速上手”环节会有点特别——首先介绍如何搭建一个实验环境。尽管这意味着现在就要开始配置服务器了，但不必紧张，考虑到读者的实际情况，这里的“服务器配置”不会比安装软件复杂多少。当然，读者可以选择先跳过本节，等到需要的时候再回来。

15.1.1 物理网络还是虚拟机

如果读者所在的办公环境中就有一台现成的 Linux 服务器，并且管理员又愿意开放相应的权限，那么相信没有比这更好的事情了。但是又有多少人会这样幸运呢？大部分读者还是要自己搭建实验环境。不过这看起来并不糟糕，不能总是指望别人帮助自己完成所有的工作，Linux 用户也一样。

如果读者恰巧有两台（或者更多）联网的 PC，那么可以将一台作为服务器，另一台作为客户机。但若是服务器和客户机位于不同的房间里，那么读者可能会感觉总是进错了房间。毕竟，判断究竟是服务器还是客户机出了毛病并不是一件显而易见的事情。

最好的选择可能还是虚拟机。读者可以在 PC 上安装 Linux，然后在这个系统上安装 VMware Server（或者其他虚拟机产品），并在虚拟机中安装另一个 Linux；也可以同时开启两个安装了 Linux 的虚拟机。如果是第一种情况，那么可以将网络接口设置为 NAT 方式（使用宿主机的网络）；而后者则应该将网络接口设置为 Bridged 方式（直接使用物理网络），如图 15.1 所示。

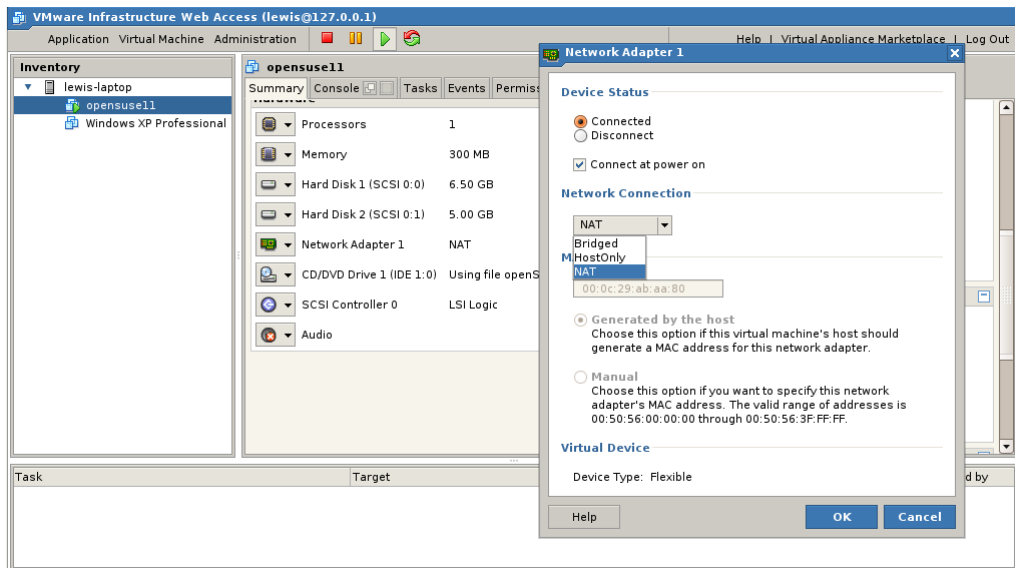


图 15.1 设置虚拟机的网络接口

关于 VMware Server 的安装和启动请参考 2.1 节。在虚拟机中安装 Linux 和在真实的硬件上安装完全相同。本章所有的示例就是在同一台主机上通过 VMware Server 实现的。

15.1.2 安装 OpenSSH

OpenSSH 是 Linux 下最常用的 SSH 服务器/客户端软件，在 15.2.1 节马上会用到它。所有的 Linux 发行版都附带了这个软件，可以简单地通过发行版的安装源（无论是光盘还是网络服务器）安装。Ubuntu 用户可以通过下面的命令安装 OpenSSH。

```
$ sudo apt-get install ssh                                ##获取并安装 OpenSSH
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
读取状态信息... 完成
将会安装下列额外的软件包:
  openssh-blacklist openssh-server
建议安装的软件包:
  molly-guard rssh
下列【新】软件包将被安装:
  openssh-blacklist openssh-server ssh
共升级了 0 个软件包，新安装了 3 个软件包，要卸载 0 个软件包，有 47 个软件未被升级。
需要下载 2398kB 的软件包。
操作完成后，会消耗掉 4948kB 的额外磁盘空间。
您希望继续执行吗？[Y/n]
...
正在设置 openssh-blacklist (0.1-1ubuntu0.8.04.1) ...
正在设置 openssh-server (1:4.7p1-8ubuntu1.2) ...
Creating SSH2 RSA key; this may take some time ...
Creating SSH2 DSA key; this may take some time ...
* Restarting OpenBSD Secure Shell server sshd                [ OK ]
```

可以看到，在安装完成后，系统会自动启动 SSH 服务器，同时设置为随系统启动（如

果不想让系统这样做，请参考 22 章）。如果发现服务器没有运行，那么可以手工执行带有 start 参数的 ssh 脚本，启动 SSH 服务器程序。

```
$ sudo /etc/init.d/ssh start
* Starting OpenBSD Secure Shell server sshd [ OK ]
```

15.1.3 安装 vnc4server

VNC 用于图形化的远程登录，将在 15.2.2 节详细介绍。绝大部分 Linux 发行版都附带了这个软件的服务器端（包括本书列举的 Ubuntu 和 openSUSE）。如果读者正在使用 Ubuntu，那么可以通过下面的命令安装这个软件（包括 vnc4-common 和 vnc4server 两个软件包）。

```
$ sudo apt-get install vnc4-common vnc4server ##获取并安装VNC的服务器端程序
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
读取状态信息... 完成
...
下列【新】软件包将被安装：
  vnc4-common vnc4server
共升级了 0 个软件包，新安装了 2 个软件包，要卸载 0 个软件包，有 47 个软件未被升级。
需要下载 1148kB 的软件包。
操作完成后，会消耗掉 2753kB 的额外磁盘空间。
...
选中了曾被取消选择的软件包 vnc4-common。
...
正在设置 vnc4-common (4.1.1+xorg1.0.2-0ubuntu7) ...
正在设置 vnc4server (4.1.1+xorg1.0.2-0ubuntu7) ...
```

完成安装后需要使用 vncserver 命令配置并启动 VNC 服务器。现在暂时不用理会，将在 15.2.2 节具体讨论。

15.1.4 SUSE 的防火墙设置

如果读者正在使用 Ubuntu Linux 的桌面版本，那么暂时防火墙不是一件需要考虑的事情，因为 Ubuntu Desktop 默认情况下是关闭防火墙的。然而 openSUSE 用户就要费些心思来设置防火墙规则了。这里介绍如何在 openSUSE 的 YAST2 管理工具中开启相应的端口，防火墙的命令行工具将在第 28 章详细讨论。

选择桌面左下角的“K 菜单”|“计算机”|“管理员设置”命令启动 YAST2 管理工具，为此需要首先提供 root 口令。YAST2 按功能划分了几个模块，选择“安全和用户”图标，如图 15.2 所示。

(1) 单击“防火墙”图标，打开防火墙配置工具，如图 15.3 所示。可以看到，当前防火墙处于启用状态，默认情况下 openSUSE 配置为拒绝一切服务请求。

(2) 选择“允许的服务”标签，在“要允许的服务”下拉列表框中选择相应的服务（这里选择安全 Shell 服务器和 VNC），并单击“添加”按钮。完成设置后如图 15.4 所示。

(3) 单击“下一步”按钮，YAST2 会给出当前设置的汇总信息，如图 15.5 所示。单

击“完成”按钮即可使配置生效。



图 15.2 YAST2 的“安全和用户”选项卡

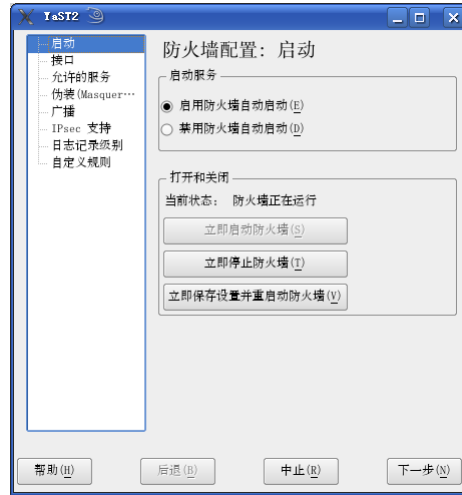


图 15.3 YAST2 的防火墙配置

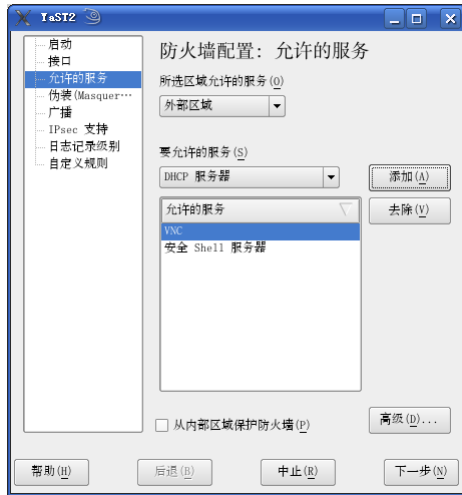


图 15.4 添加允许的服务

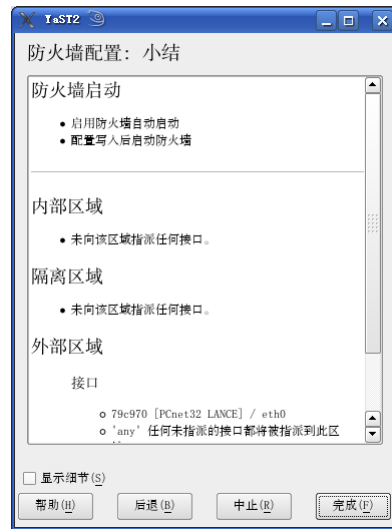


图 15.5 防火墙配置信息汇总

15.2 登录另一台 Linux 服务器

作为一款服务器操作系统，Linux 充分考虑了远程登录的问题。无论是从 Linux、Windows 还是其他一些操作系统登录到 Linux 都是非常方便的。支持多个用户同时登录对于服务器而言非常重要——这正是 Linux 擅长的。

有多种不同的协议可供选择，但 SSH 也许是其中“最好”的。这种协议提供了安全可靠的远程连接方式，SSH 将贯穿于本节的讨论中。

15.2.1 安全的 Shell: SSH

SSH 是 secure shell 的简写，意为“安全的 shell”。作为 rlogin、rcp、telnet 这些“古老”的远程登录工具的替代品，SSH 会对用户的身份进行验证，并加密两台主机之间的通信。SSH 在设计时充分考虑到了各种潜在的攻击，给出了有效的保护措施。尽管现在 SSH 已经转变为一款商业产品 SSH2，但开放源代码社区已经发布了 OpenSSH 软件作为回应。这款免费的开源软件由 FreeBSD 负责维护，并且实现了 SSH 协议的完整内容。

要从 Linux 下通过 SSH 登录另一台 Linux 服务器非常容易——前提是在远程服务器上拥有一个用户账号。打开 Shell 终端，执行 `ssh -l login_name hostname` 命令，应该把 `login_name` 替换成真实的用户账号，把 `hostname` 替换成服务器主机名（或者 IP 地址）。下面这条命令以 liu 用户的身份登录到 IP 地址为 10.71.84.145 的 Linux 服务器上。

```
$ ssh -l liu 10.71.84.145
```


如果是初次登录，SSH 可能会提示无法验证密钥的真实性，并询问是否继续建立连接，回答 `yes` 继续。用户口令验证通过后，SSH 会反馈上次登录情况并以一句“Have a lot of fun...”作为问候。

```
The authenticity of host '10.71.84.145 (10.71.84.145)' can't be established.  
RSA key fingerprint is c9:58:fd:e4:dc:4b:4a:bb:03:d7:9b:87:a3:bc:6a:b0.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '10.71.84.145' (RSA) to the list of known hosts.  
Password:  
Last login: Sun Nov  9 10:38:30 2008 from console  
Have a lot of fun...  
liu@linux-dqw4:~>
```

注意 Shell 提示符前的用户和主机名改变了，表示当前已经登录到这台名为 `linux-dqw4`（IP 地址 10.71.84.145）的服务器上。接下来的操作读者应该很熟悉了，例如用 `ls` 命令查看当前目录中的文件信息。

```
lewis@linux-dqw4:~> ls  
bin Desktop Documents public_html
```

时刻记住当前做的所有操作都发生在远程服务器上。当连接到几台不同的服务器时，管理员常常会在来回切换 Shell 的过程中搞糊涂。因此，尽量不要同时开启 3 个以上的远程 Shell。时刻注意 Shell 提示符前的主机名，并且在执行重要操作时保持警惕，是避免灾难的重要途径。

 **注意：**在任何时候直接使用 `root` 账号登录远程主机都不是一个好习惯。正确的做法应该是使用受限账号登录，然后在需要的时候通过 `su` 或者 `sudo` 命令临时取得 `root` 权限。

完成工作后，使用 `exit` 命令可以结束同远程主机的 SSH 连接，这将把用户带回到建立连接前的 Shell 中。

```
lewis@linux-dqw4:~> exit  
logout
```

```
Connection to 10.71.84.145 closed.
lewis@lewis-laptop:~/shell$
```

SSH 服务默认开启在 22 号端口，服务器的守护进程在 22 号端口监听来自客户端的请求。如果服务器端的 SSH 服务没有开启在 22 端口（这通常是为了防范居心不良端口扫描程序），那么可以通过 SSH 的 -p 选项指定要连接到的端口。下面这条命令指导 SSH 连接到远程服务器的 202 端口。

```
$ ssh -l liu -p 202 10.71.84.145
```

如果用户需要在远程主机上运行 X 应用程序，那么首先应该保证对方服务器开启了 X 窗口系统，然后使用带 -X 参数的 SSH 命令显式启动 X 转发功能。

```
lewis@lewis-laptop:~/shell$ ssh -X -l liu 10.71.84.145
Password:
Last login: Sun Nov 9 13:41:20 2008 from 10.71.84.18
Have a lot of fun...
```

下面这条命令在所登录到的服务器上运行 Firefox 浏览器，注意服务器会反馈一系列信息告诉用户此刻发生了什么。

```
liu@linux-dqw4:~> firefox
Launching a SCIM daemon with Socket FrontEnd...
Loading simple Config module ...
Creating backend ...
Reading pinyin phrase lib failed
Loading socket FrontEnd module ...
Starting SCIM as daemon ...
GTK Panel of SCIM 1.4.7
...
```

SSH 会把对方服务器上的 Firefox 界面完完整整地传输到本地，这样用户就可以在当前 PC 上使用远程服务器上的 Firefox 了。如果两台主机距离比较长，或者网络状况不太理想的话，那么传输一个 X 应用程序界面会比较慢，但最终应该能出现在本机的屏幕上。

15.2.2 登录 X 窗口系统：图形化的 VNC

读者已经看到，通过启用 SSH 的 X 转发功能可以在本地运行远程主机上的 X 应用程序，但有些时候用户可能希望更进一步，直接从 X 窗口登录服务器，就像操作本地的桌面一样。VNC（Virtual Network Computing，虚拟网络计算）实现了这一需求。

要使用 VNC 登录，首先要求服务器端运行有 X 窗口系统，且开启了相关服务和端口。在连接之前，要先在远程主机的用户目录下生成 VNC 的配置文件。使用 SSH 连接远程主机。

```
lewis@lewis-laptop:~/shell$ ssh -l liu 10.71.84.145
Password:
Last login: Sun Nov 9 14:13:41 2008 from console
Have a lot of fun...
```

运行 vncserver 脚本生成配置文件，配置过程中会要求用户输入远程访问密码。

```
liu@linux-dqw4:/home/lewis> vncserver
```

```

You will require a password to access your desktops.

Password:                                     ##设置远程访问密码
Warning: password truncated to the length of 8.
Verify:                                       ##再次输入密码
Would you like to enter a view-only password (y/n)? n
##是否输入一个只能查看的密码，选择否

New 'X' desktop is linux-dqw4:4
##配置文件的存放信息
Creating default startup script /home/liu/.vnc/xstartup
Starting applications specified in /home/liu/.vnc/xstartup
Log file is /home/liu/.vnc/linux-dqw4:4.log

```

服务器端的用户配置结束后，就可以从客户端登录了。有很多 VNC 的客户端工具可供使用，vncviewer 是一款跨平台的 VNC 客户端工具。在 Google 中使用关键字 vncviewer download 搜索，可以得到大量的下载地址。

完成安装后，就已经做好了登录远程主机的所有准备。下面在终端里执行 vncviewer 命令，将开启一个窗口，提示输入服务器地址。

```
$ vncviewer
```

输入 IP 地址连接指定的服务器 IP，如图 15.6 所示。注意这个 IP 地址后面跟了一个“:1”，这个数字指定了开启第 2 个 X 窗口会话（在 VNC 服务器上被设置，这也是默认的设置），单击 OK 按钮建立连接。成功连接后的界面如图 15.7 所示。

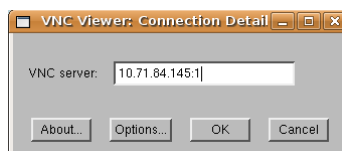


图 15.6 VNC Viewer 的连接界面

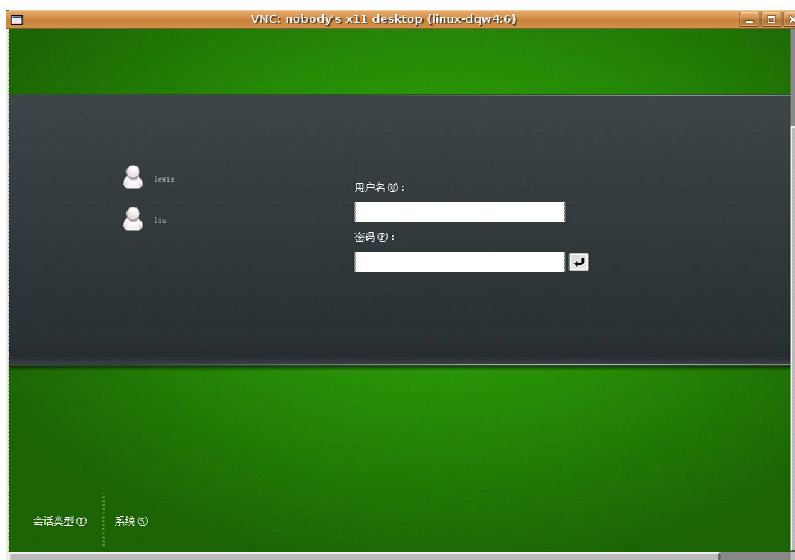


图 15.7 远程主机的登录界面

输入用户名和口令后，即可登录到 X 窗口并执行操作了，如图 15.8 所示。

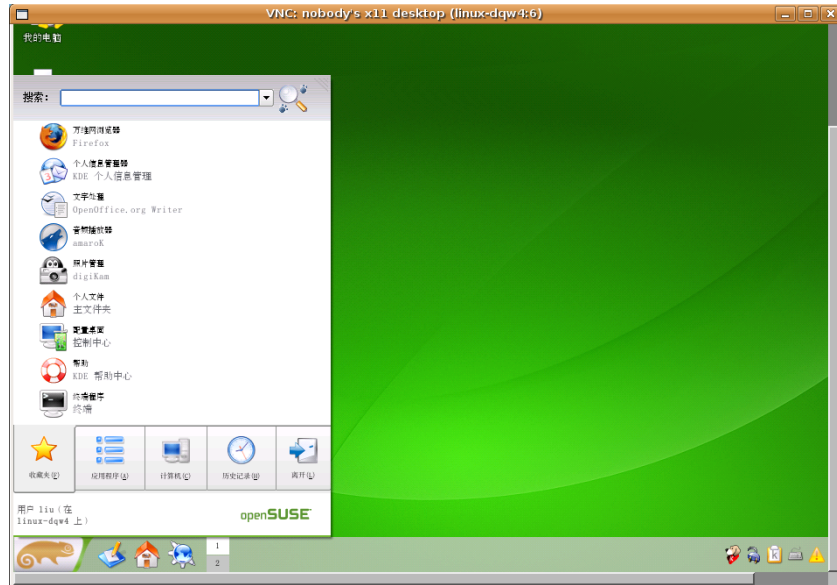


图 15.8 通过 VNC Viewer 控制远程主机

15.2.3 我想从 Windows 登录这台 Linux

管理员常常陷入这样的尴尬：公司的一些任务不得不在 Windows 下完成，而 Linux 作为一款优秀的服务器操作系统又被部署在机房中。在这种情况下，要么安装双系统，并且为了短暂的应用而不停地重启计算机；要么干脆从 Windows 登录到 Linux 服务器。幸运的是，经过开放源代码界的长期努力，这已经不是什么困难的事情了。

Windows 上有几种不同的 SSH 客户端，其中开放源代码的 PuTTY 是使用最为广泛、也是最受好评的一个。这是一个绿色软件，不需要安装。下载并运行其主程序 `putty.exe`，填写远程主机的主机名（或者 IP 地址）和登录端口，如图 15.9 所示。

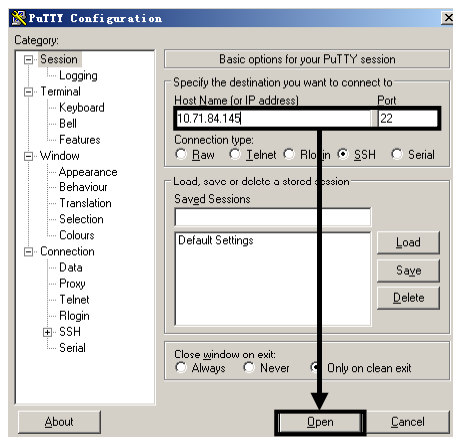


图 15.9 PuTTY 客户端的设置和登录界面

单击 Open 按钮，即可建立连接。如果是初次登录，会出现如图 15.10 所示的提示框，单击“是”按钮继续登录。PuTTY 将打开一个类似于 Shell 终端的命令行窗口，输入用户名和口令即可完成登录，接下来发生的事情就跟在 Linux 中一样了，如图 15.11 所示。

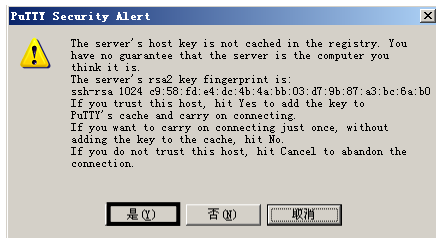


图 15.10 询问是否接受远程主机的密钥

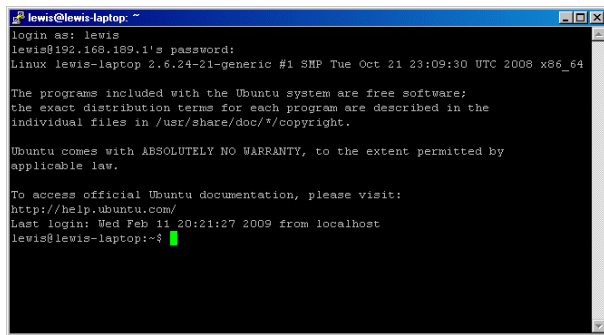


图 15.11 通过 PuTTY 连接到远程主机的 Shell

如果希望通过 VNC 从 Windows 登录到 Linux，那么老朋友 vncviewer 同样有 Windows 上的版本，读者可以从 www.realvnc.com/products/free/4.1/winvncviewer.html 上免费下载这款软件。安装和登录界面如图 15.12 和图 15.13 所示，其基本操作和 Linux 下的 vncviewer 基本一致。

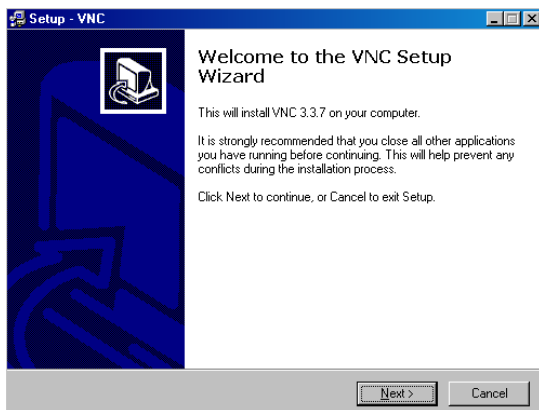


图 15.12 VNC for Windows 的安装界面

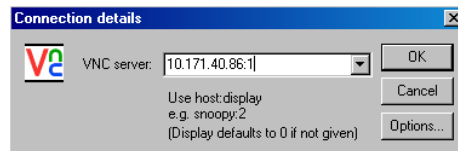


图 15.13 VNC Viewer for Windows 的登录界面

15.3 登录 Windows 服务器

本节将要从相反的方向讨论远程登录这个问题——从 Linux 登录到 Windows 服务器。通常来说，有两种比较常用的方法，一种是为 Windows 装上一个名叫 VNC Server 的软件，这样 Linux 就可以通过 VNC 登录到 Windows 服务器了。这是属于 Windows 服务器的配置问题，此处就不再赘述了。

另一种方法是借助 Linux 下已有的客户端软件，直接通过 RDP 协议连接到 Windows

服务器。当然，首先要求 Windows 服务器开启了远程登录功能，可以通过右击“我的电脑”，在弹出的快捷菜单中选择“属性”选项打开“系统属性”对话框，选择“远程”标签进入“远程”选项卡，在其中选中“允许用户远程连接到此计算机”复选框打钩开启这一功能。

下载命令行登录工具 `rdesktop` 并安装，开启 Shell 终端，通过下面这条命令即可连接到 Windows 服务器。

```
rdesktop -u username ip-address
```

例如，这里以用户 `liu` 的身份登录到一台 IP 地址为 `10.71.84.129` 的 Windows 服务器上。

```
$ rdesktop -u liu 10.71.84.129
```

同 Windows 服务器建立连接后，`rdesktop` 会打开一个窗口，显示熟悉的 Windows 登录界面，如图 15.14 所示。通过用户密码验证后，即可登录到这台远程 Windows 服务器。相信读者对于图 15.15 的界面应该非常熟悉了。

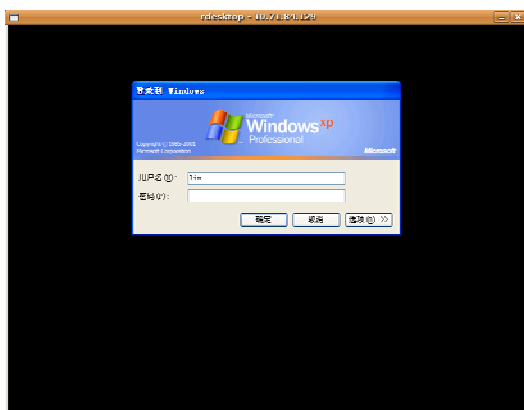


图 15.14 rdesktop 中的 Windows 登录界面



图 15.15 通过 rdesktop 远程控制 Windows

如果 Windows 服务器被配置为使用一个不同的端口，而不是 RDP 协议默认的 3389 端口，那么在使用 `rdesktop` 连接的时候应该在 IP 地址后加上冒号“:”和端口号，例如上面这条连接命令应该写成下面这种形式，其中 `6666` 应该被改成 Windows 远程桌面实际使用的端口号。

```
$ rdesktop -u liu 10.71.84.129:6666
```

15.4 为什么不使用 telnet

为什么不使用 `telnet`？答案很简单：为了安全。`telnet` 曾经是使用最广泛的远程登录工具，但是 `TELNET` 协议有一个致命的缺陷：使用明文口令。这意味着用户口令将以明文的形式在网络上传输，任何人都有机会通过“网络嗅探”工具直接获取该口令。`Linux` 已经不再包含 `TELNET` 服务器程序，并且也不推荐用户使用。与此类似的还有 `rlogin`、`rsh` 等远程登录工具，它们也因为同样的安全问题成了众矢之的。

15.5 进阶：使用 SSH 密钥

读者已经了解到如何使用 SSH 连接远程主机。SSH 利用加密算法来保证信息传输的安全性，在已经接触到的例子中，用户必须在远程主机上拥有一个账号，并提供口令。SSH 也提供了另外一些验证用户身份的方式——密钥对是其中的一种，也可能是最安全的一种。

15.5.1 为什么要使用密钥

对于管理有多台服务器的管理员而言，快速登录到某几台机器的 Shell 上是很重要的。每次都输入登录口令费时费力（很多口令长达 15 位甚至更多），并且还很容易分神。管理员的思维不得不在“找出问题”和“到达出问题的地方”之间来回切换，这种“思维体操”让绝大多数管理员不堪重负。

使用 SSH 密钥对可以有效解决这个问题，而且也足够安全。这种解决方案基于下面这些想法：

- ❑ 有一对互相匹配的密钥文件（公钥和私钥）；
- ❑ 管理员的 PC 上保存有私钥文件的副本；
- ❑ 与私钥文件匹配的公钥文件存放在服务器上；
- ❑ 建立 SSH 连接时检查密钥对的匹配性。

这样，管理员就不需要手动输入口令了，所有的一切都是自动完成的。这听上去很诱人，下面就来实践配置 SSH 密钥对的全过程。而对于管理员最关心的另一个安全性问题，将在本节的最后讨论。

15.5.2 生成密钥对

SSH 提供了 ssh-keygen 工具来生成密钥对，使用 -t 选项指定密钥类型。通常采用 SSH 的 rsa 密钥。

```
$ ssh-keygen -t rsa                                ##生成 SSH 密钥对
Generating public/private rsa key pair.
Enter file in which to save the key (/home/lewis/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/lewis/.ssh/id_rsa.
Your public key has been saved in /home/lewis/.ssh/id_rsa.pub.
The key fingerprint is:
d8:f9:24:15:c4:60:a7:3e:7a:2e:1a:64:c3:42:d1:76 lewis@lewis-laptop
```

该命令会在用户主目录下的 .ssh 目录中生成两个文件。其中，id_rsa 是私钥文件；对应的 id_rsa.pub 是公钥文件：

```
$ ls /home/lewis/.ssh/
id_rsa id_rsa.pub known_hosts
```

15.5.3 复制公有密钥至远程主机

下面只需将公有密钥文件复制到远程主机。假设远程主机的 IP 地址是 172.16.25.129，登录用户名为 liu，下面建立 SSH 连接。

```
lewis@lewis-laptop:~/.ssh$ ssh 172.16.25.129 -l liu
Password:
Last login: Mon Nov 10 10:44:16 2008 from 10.171.38.37
Have a lot of fun...
```

在远程主机用户 liu 的主目录下建立 .ssh 目录，并解除其他人对该文件的所有权限。

```
liu@linux-dqw4:~> mkdir .ssh
liu@linux-dqw4:~> chmod 700 .ssh/
liu@linux-dqw4:~> exit
logout
Connection to 172.16.25.129 closed.
```

最后，使用 scp 命令将公钥复制到远程主机的 /home/liu/.ssh 目录下，并重命名为 authorized_keys。

```
$ scp /home/lewis/.ssh/id_rsa.pub liu@172.16.25.129:/home/liu/.ssh/
authorized_keys
Password:
id_rsa.pub                                100% 400      0.4KB/s   00:00
```

15.5.4 测试配置

至此已经完成了 SSH 密钥对的配置。尝试以用户 liu 的身份登录该远程主机，可以看到 SSH 不再询问口令，而是直接允许用户登录到系统中。

```
lewis@lewis-laptop:~/.ssh$ ssh 172.16.25.129 -l liu
Last login: Tue Jan 13 00:57:18 2009 from 172.16.25.1
Have a lot of fun...
liu@linux-dqw4:~>
```

15.5.5 密钥的安全性

有些人认为使用公钥会显著增加潜在的安全风险，这种想法的确是有道理的。获取 SSH 密钥文件比获得 /etc/shadow 容易得多，并且公钥通常被管理员大量分发，为了快速登录到多台服务器，这就增加了其他人得到公钥的可能性。

不过仔细想一想，黑客得同时窃取到两份文件（一份公钥，一份私钥）才行。和 SSH 密钥带来的方便相比，管理员是否应该舍弃一些安全性？不同的人在不同的环境下会给出不同的回答。但无论如何，没有一定安全的“安全”措施。如果决定使用 SSH 密钥，就应注意保管好自己的私钥文件，并且只在需要的地方存放公钥；如果使用 SSH 口令，就应该保管好口令。管理员的警惕性是保证系统安全的最重要的武器。

15.6 小 结

- ❑ 读者可以使用虚拟机实现远程登录的实验环境。应该设置虚拟机使用合适的网络接口（NAT 或是 Bridged）。
- ❑ SSH 的服务器程序是 OpenSSH；图形化登录的 VNC 应该使用 vnc4server。
- ❑ 在必要的时候配置防火墙。openSUSE 可以使用 YAST2 管理工具。
- ❑ SSH 提供加密的远程通信通道。为此应该在远程主机上拥有一个用户账号。
- ❑ ssh -X 命令开启 SSH 的 X 图形系统转发功能。
- ❑ 使用 VNC 可以直接登录到远程主机的 X 窗口系统。
- ❑ 在 Windows 中可以使用 PuTTY 通过 SSH 远程登录到 Linux 主机。
- ❑ 在 Linux 中可以使用 rdesktop 通过 RDP 协议登录到 Windows 主机。
- ❑ telnet、rlogin 等远程登录工具使用明文口令，在安全性方面存在很大问题，应该避免使用。
- ❑ 使用 SSH 密钥对可以让管理员不提供口令即登录到远程主机。