

### 学习目标

- 能够在 Linux 中配置主机名及 IP 地址
- 能够掌握 Linux 中常见的网络配置文件及内容
- 能够使用如 ping、lsof 等命令进行网络信息检测
- 能够配置 Telnet 服务
- 能够配置 SSH 服务
- 能够配置 FTP 服务
- 能够使用 VNC 远程接入网络

## 5.1 常见的网络配置文件

Linux 网络配置包括主机名称、网卡安装、协议管理和 IP 地址的建立等,要了解一些相关文件的作用和位置,例如/etc/hosts、/etc/sysconfig/network、/etc/protocols、/etc/services、ifcfg-eth0 和/etc/resolve.conf 等。

### 1. /etc/hosts 文件

该文件提供了主机名与 IP 之间的映射,当以主机名访问一台主机时,系统检查/etc/hosts 文件,根据文件将主机名称转换为 IP 地址。文件的内容为:

IP 地址	主机名	别名
127.0.0.1	localhost	localhost.localdomain... localhost4.localdomain4
::1	localhost	localhost.localdomain... localhost6.localdomain6

注意: hosts 文件用于指明本地主机名与 IP 地址间的对应关系。例如本机的 IP 地址为 10.1.1.1/24,希望通过主机名访问本机,主机名为 mylinux,则可在/etc/hosts 文件中加一行记录: 10.1.1.1 mylinux。

### 2. /etc/sysconfig/network 文件

网络配置信息,完成网络域名与网络地址(网络 ID)的映射,参考内容如下:

```
NETWORKING = yes          # 是否使用网络
HOSTNAME = localhost.localdomain # 主机名
GateWay = 172.17.31.254      # 网关
```

如果要更改主机名或网关,可更改文件 gedit/etc/sysconfig/network。

### 3. /etc/protocols 文件

该文件提供一个 TCP/IP 系统支持列表,文件的每一行描述一个协议,包括协议名、协议编号、协议别名和注释,部分内容如下。

```
# /etc/protocols:  
# $ Id: protocols,v 1.9 2009/09/29 15:11:55 ovasik Exp $  
# Internet (IP) protocols  
# from: @(#)protocols 5.1 (Berkeley) 4/17/89  
# Updated for NetBSD based on RFC 1340, Assigned Numbers (July 1992).  
# Last IANA update included dated 2009 - 06 - 18  
# See also http://www.iana.org/assignments/protocol-numbers  
ip    0   IP          # internet protocol, pseudo protocol number  
hopopt 0   HOPOPT     # hop - by - hop options for ipv6  
icmp   1   ICMP       # internet control message protocol  
igmp   2   IGMP       # internet group management protocol  
ggp    3   GGP         # gateway - gateway protocol  
ipencap 4   IP - ENCAP # IP encapsulated in IP (officially "IP'')  
st    5   ST          # ST datagram mode  
tcp   6   TCP         # transmission control protocol  
cbt   7   CBT         # CBT, Tony Ballardie <A.Ballardie@cs.ucl.ac.uk>  
egp   8   EGP         # exterior gateway protocol  
igp   9   IGP         # any private interior gateway (Cisco: for IGRP)  
... ...
```

### 4. /etc/services 文件

文件的每一行提供一个服务名,所提供的信息的部分内容有:

# 服务名称	端口号/协议名	别名	说明
tcpmux	1/tcp		# TCP port service multiplexer
tcpmux	1/udp		# TCP port service multiplexer
rje	5/tcp		# Remote Job Entry
rje	5/udp		# Remote Job Entry
echo	7/tcp		
echo	7/udp		
discard	9/tcp	sink null	
discard	9/udp	sink null	
systat	11/tcp	users	
systat	11/udp	users	
daytime	13/tcp		
daytime	13/udp		
qotd	17/tcp	quote	
qotd	17/udp	quote	
msp	18/tcp		# message send protocol
msp	18/udp		# message send protocol
chargen	19/tcp	ttytst source	
... ...			

### 5. /etc/sysconfig/network-scripts/ifcfg-eth0 文件

网络配置文件 ifcfg-eth0 所在目录是/etc/sysconfig/network-scripts/,这个文件保存了

网络设备 eth0 的配置信息,主要内容如下:

```
DEVICE = eth0          # 网卡设备名(接口名)
HWADDR = 00:0c:29:81:71:1e   # MAC 地址
ONBOOT = no            # 系统启动时网络接口是否自动加载
IPADDR = 172.17.31.1    # IP 地址
BOOTPROTO = none        # 启动时不使用任何协议(static:静态协议,bootp,DHCP 协议)
NETMASK = 255.255.255.0  # 子网掩码
TYPE = Ethernet         # 网卡类型
GATEWAY = 172.17.31.254 # 网关地址
DNS1 = 172.17.3.8      # DNS 地址
```

## 6. /etc/resolv.conf 文件

该文件是域名服务器客户端的配置文件,用于指定域名服务器的位置,参考内容如下:

```
# Generated by NetworkManager
nameserver 172.17.3.8          # 域名服务器的地址
search 172.17.3.8             # 搜索的 DNS 地址
```

## 5.2 常用的网络配置命令

在 Linux 中掌握网络配置的方法是非常重要的,通过命令可以对网络参数进行全方位的设置,例如 IP 地址、主机名等。

### 1. 设置网络配置参数

执行命令:

```
[root@localhost ~]# setup
```

系统会弹出配置界面,如图 5-1 所示,可以在此界面中配置防火墙、键盘、系统服务等内容。选择“网络配置”选项,出现网络配置界面,如图 5-2 所示(此界面也可以使用 system-config-network 命令调出)。



图 5-1 系统的配置界面

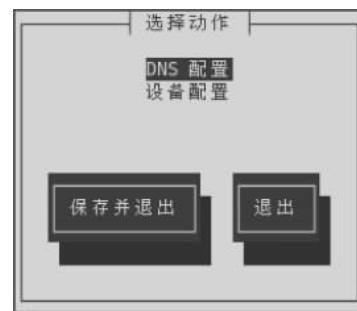


图 5-2 网络配置界面

在图 5-2 中选择“设备配置”选项，出现本地识别出的网络设备，如图 5-3 所示，选中设备 eth0 后回车，出现本地的网络配置界面，如图 5-4 所示。



图 5-3 本地网卡设备名称

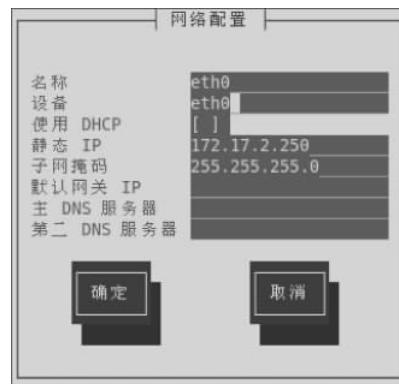


图 5-4 本地网络配置界面

配置好网络参数后，单击“确定”按钮，依次返回上一层界面，直至退出。此时所做的配置信息会被写入到 /etc/sysconfig/network-scripts/ifcfg-eth0 文件中。注意在 Linux 中默认的网卡名称为 eth0，参数配置完成后网卡不会立即被激活，需要使用 ifup eth0 命令进行激活后，设置的 IP 地址等参数才会生效。

## 2. ifup 命令

**例 5-1** 激活网卡连接。

```
[root@localhost ~]# ifup eth0
```

## 3. ifdown 命令

**例 5-2** 断开网卡连接。

```
[root@localhost ~]# ifdown eth0
```

## 4. ifconfig 网络配置命令的使用方法

该命令可以查看系统的网络参数，也可以增加新的 IP 地址。命令格式为：

```
ifconfig [ interface ] [ type options|address ]
```

其中，interface 是网络设备名，可以是 eth0、eth1 或 lo(回路设备)；type 选项如下。

- (1) up：打开网络接口设备。
- (2) down：关闭网络接口设备。
- (3) netmask：设置子网掩码。
- (4) broadcast：设置广播地址。

**例 5-3** 显示所有网络接口。

```
[root@localhost ~]# ifconfig
```

**例 5-4** 显示 eth0 的配置参数。

```
[root@localhost ~]# ifconfig eth0
```

**例 5-5** 修改 eth0 的 IP 地址。

```
[root@localhost ~]# ifconfig eth0 192.168.1.100
```

**例 5-6** 设置 eth0 的网络掩码和广播地址。

```
[root@localhost ~]# ifconfig eth0 netmask 255.255.255.0 broadcast 192.168.1.255
```

**例 5-7** 增加一个 IP 地址 192.168.1.120, 掩码为 255.255.255.0。

```
[root@localhost ~]# ifconfig eth0:1 192.168.1.120 netmask 255.255.255.0
```

如果想在开机时就建议这个 IP, 可以将这条命令加入到开机启动文件中, 即

```
[root@localhost ~]# echo "ifconfig eth0:1 192.168.1.120 netmask 255.255.255.0">>>/etc/rc.d/rc.local
```

利用此方法可以在开机时建立多个 IP。

**例 5-8** 关闭网卡。

```
[root@localhost ~]# ifconfig eth0 down
```

**例 5-9** 加载网卡。

```
[root@localhost ~]# ifconfig eth0 up
```

## 5. route 命令

route 命令用于设置本地的路由信息。在 Linux 中可以使用 route 命令查看本机的路由表信息, 添加、删除路由记录, 设置默认网关等, 其语法格式为:

```
route add/del -net/host/default 网络/主机地址 netmask 子网掩码 [dev 网络设备名][gw 网关]
```

**例 5-10** 查看路由表。

```
[root@localhost ~]# route
```

路由表中会出现如下信息。

- (1) Destination: 目标网络 IP 地址, 可以是一个网络地址, 也可以是一个主机地址。
- (2) Gateway: 网关地址, 即该路由条目中下一跳的路由器 IP 地址。
- (3) Genmask: 路由项的子网掩码, 与 Destination 信息进行与操作得出目标地址。
- (4) Flags: 路由标志。其中, U 表示路由项是活动的; H 表示目标是单个主机; G 表

示使用网关；R 表示对动态路由进行复位；D 表示路由项是动态安装的；M 表示动态修改路由；！表示拒绝路由。

- (5) Metric：路由开销值,用来衡量路径的代价。
- (6) Ref：依赖于本路由的其他路由条目。
- (7) Use：该路由项被使用的次数。
- (8) Iface：该路由项发送数据包使用的网络接口。

#### 举例 5-11 设置默认网关。

```
[root@localhost ~]# route add default gw 192.168.1.2
```

#### 例 5-12 删除默认网关。

```
[root@localhost ~]# route del default gw 192.168.1.2
```

#### 例 5-13 添加到达 172.17.2.0/24 的路由,经由 eth0 接口,并由 172.17.2.254 转发。

```
[root@localhost ~]# route add - net 172.17.2.0 netmask 255.255.255.0 gw 172.17.2.254  
dev eth0
```

#### 例 5-14 删除到过 172.17.2.0/24 网络的路由。

```
[root@localhost ~]# route del - net 172.17.2.0 netmask 255.255.255.0
```

## 6. ping 命令

ping 命令用于检测主机。其语法为：

```
ping [ -dfnqrRv ] [ -c <完成次数> ] [ -i <间隔秒数> ] [ -I <网络界面> ] [ -l <前置载入> ] [ -p <范本  
样式> ] [ -s <数据包大小> ] [ -t <存活数值> ] [ 主机名称或 IP 地址 ]
```

执行 ping 命令会使用 ICMP 传输协议,发出要求回应的信息,若远端主机的网络功能没有问题,就会回应该信息,从而得知该主机运作正常。

ping 命令的参数如下。

- (1) -d：使用 Socket 的 SO\_DEBUG 功能。
- (2) -c <完成次数>：设置完成要求回应的次数。
- (3) -f：极限检测。
- (4) -i <间隔秒数>：指定收发信息的间隔时间。
- (5) -I <网络界面>：使用指定的网络界面送出数据包。
- (6) -l <前置载入>：设置在送出要求信息之前,先行发出的数据包。
- (7) -n：只输出数值。
- (8) -p <范本样式>：设置填满数据包的范本样式。
- (9) -q：不显示命令执行过程,开头和结尾的相关信息除外。
- (10) -r：忽略普通的 Routing Table,直接将数据包送到远端主机上。
- (11) -R：记录路由过程。

- (12) -s <数据包大小>：设置数据包的大小。
- (13) -t <存活数值>：设置存活数值 TTL 的大小。
- (14) -v：详细显示命令的执行过程。

**例 5-15** 向 127.0.0.1 发 3 个 ICMP 数据包。

```
[root@localhost ~]# Ping 127.0.0.1 -c 3
```

## 7. hostname 命令

hostname 命令用于设置本机名称。在网络中，每台主机都有一个只属于自己的名字，hostname 命令用于显示或临时设置当前系统主机名称。此命令不会将信息写入/etc/sysconfig/network 文件，当系统重启后，此设置失效。

**例 5-16** 显示当前系统的主机名。

```
[root@localhost ~]# hostname
```

**例 5-17** 临时设置系统主机名为 test。

```
[root@localhost ~]# hostname test
```

## 8. service 命令

service 命令用于设置服务状态。常见的状态有 3 种，分别为 start、restart 和 stop。以网络服务为例，该命令的使用方法为：

```
[root@localhost ~]# service network restart
```

或者

```
[root@localhost ~]# /etc/rc.d/init.d/network restart.
```

## 9. traceroute 命令

traceroute 命令用于实现路由跟踪。traceroute 的基本原理就是发出 TTL 字段为 1~n 的 IP 包，然后等待路由器的 ICMP 超时回复，进而记录下来经过的路由器。traceroute 可以在 IP 包中放 3 种数据：UDP 包（默认选项是-U）、TCP 包（选项是-T）、ICMP 包（选项是-I），而且每个包 traceroute 都发 3 次。

该命令输出的每一行代表一个段，利用它可以跟踪从当前主机到达目标主机所经过的路径。常用参数如下。

- (1) -i：指定网络接口，对于多个网络接口有用。例如-i eth1 或-i ppp1 等。
- (2) -m：把在外发探测包中所用的最大生存期设置为 max-ttl 次转发，默认值为 30 次。
- (3) -n：显示 IP 地址，不查主机名。当 DNS 不起作用时常用到这个参数。
- (4) -p port：探测包使用的基本 UDP 端口设置为 port，默认值是 33 434。
- (5) -q n：在每次设置生存期时，把探测包的个数设置为值 n，默认时为 3。
- (6) -r：绕过正常的路由表，直接发送到网络相连的主机。

(7) -w n: 把对外发探测包的等待响应时间设置为 n 秒,默认值为 3 秒。

**例 5-18** 显示从本地到 mylinux.net 的路由信息,跳数为 10。

```
[root@localhost ~]# traceroute -m 10 mylinux.net
```

**例 5-19** 显示 IP 地址,不查主机名。

```
[root@localhost ~]# traceroute -n mylinux.net
```

**例 5-20** 使用 UDP 端口 6688 进行探测。

```
[root@localhost ~]# traceroute -p 6688 mylinux.net
```

**例 5-21** 设置探测包的数值为 4。

```
[root@localhost ~]# traceroute -q 4 mylinux.net
```

**例 5-22** 设置对外发探测包的等待时间为 5 秒。

```
[root@localhost ~]# traceroute -w 5 mylinux.net
```

## 10. netstat 命令

netstat 命令用于查看网络的连接状态。此命令的网络连接状态只对 TCP 协议有效。常见的连接状态有: ESTABLISHED(已建立连接)、SYN SENT(发起连接)、SYN RECV(接受发起的连接)、TIME WAIT(等待时间)、LISTENING(监听)。

**例 5-23** 显示网络接口状态信息。

```
[root@localhost ~]# netstat -i
```

**例 5-24** 显示核心路由表信息。

```
[root@localhost ~]# netstat -nr
```

**例 5-25** 显示 TCP 协议的连接状态。

```
[root@localhost ~]# netstat -t
```

## 11. arp 命令

arp 命令用于查看或配置系统的 MAC 地址与 IP 地址的映射关系。

**例 5-26** 查看 arp 缓存。

```
[root@localhost ~]# arp
```

**例 5-27** 添加 IP 地址 172.17.2.230 到 MAC 地址 00:11:12:DE:EF:12 的映射。

```
[root@localhost ~]# arp -s 172.17.2.230 00:11:12:DE:EF:12
```

**例 5-28** 删除 IP 地址与 MAC 地址的映射。

```
[root@localhost ~]# arp -d 172.17.2.230
```

## 12. lsof 命令

lsof 命令用于列出当前系统中打开的文件,需要以 root 身份执行。在不加任何参数的情况下部分输出结果如下:

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE NAME
init	1	root	cwd	DIR	253,0	4096	2 /
init	1	root	rtd	DIR	253,0	4096	2 /

输出各列信息含义如表 5-1 所示。

表 5-1 lsof 命令输出各项含义

字 段 名	含 义	字 段 名	含 义
COMMAND	进程的名称	PID	进程标识符
USER	进程所有者	FD	文件描述符,应用程序通过
TYPE	文件类型	DEVICE	文件描述符识别该文件
SIZE	文件的大小	NODE	磁盘的名称
NAME	打开文件的名称		索引节点

**例 5-29** 显示打开指定文件的所有进程。

```
[root@localhost ~]# lsof filename
```

**例 5-30** 显示 COMMAND 列中包含指定字符的进程所有打开的文件。

```
[root@localhost ~]# lsof -c string
```

**例 5-31** 显示属于指定用户打开的文件。

```
[root@localhost ~]# lsof -u username
```

**例 5-32** 显示归属 gid 的进程情况。

```
[root@localhost ~]# lsof -g gid
```

**例 5-33** 显示在/etc/目录下被进程打开的文件。

```
[root@localhost ~]# lsof +d /etc
```

**例 5-34** 显示/etc/下被进程打开的文件,包含子目录。

```
[root@localhost ~]# lsof +D /etc
```

**例 5-35** 显示指定文件描述符的进程。

```
[root@localhost ~]# lsof -d FD
```

**例 5-36** 显示符合条件的进程情况。

```
[root@localhost ~]# lsof -i
```

**例 5-37** 查看 22 端口的运行情况。

```
[root@localhost ~]# lsof -i:22
```

## 5.3 远程登录

远程登录是指在本地通过网络访问其他计算机就像用户在现场操作一样。一旦进入主机，用户可以操作主机允许的任何事情，例如读文件、编辑文件或删除文件等。常见的远程登录方式有 Telnet、SSH、远程桌面。由于 Telnet 是以明文传输密码的，所以使用并不是很安全；SSH 以密文传输，应用较广泛；远程桌面功能实现了以桌面形式远程控制其他计算机，常用的工具软件为 VNC(Virtual Network Computing，虚拟网络计算)。

### 5.3.1 Telnet 配置

#### 1. 什么是 Telnet

Telnet 是远程登录的一种服务，属于应用层的协议，但它的底层协议是 TCP/IP，所用到的端口是 23。使用 Telnet 可以在本地登录远程的计算机，并且可以对远程计算机进行修改和操作，所用的界面是 DOS 界面，而不是图形界面。

#### 2. 远程登录的工作过程

使用 Telnet 协议进行远程登录时需要满足以下条件：

- (1) 在本地计算机上必须装有包含 Telnet 协议的客户程序。
- (2) 必须知道远程主机的 IP 地址或域名。
- (3) 必须知道登录标识与密码。

Telnet 远程登录服务分为以下 4 个过程：

- (1) 本地与远程主机建立连接。该过程实际上是建立一个 TCP 连接，用户必须知道远程主机的 IP 地址或域名。
- (2) 将本地终端上输入的用户名和密码及以后输入的任何命令或字符以 NVT(Net Virtual Terminal)格式传输到远程主机。该过程实际上是从本地主机向远程主机发送一个 IP 数据报。
- (3) 将远程主机输出的 NVT 格式的数据转化为本地所接受的格式送回本地终端，包括输入命令回显和命令执行结果。

(4) 最后,本地终端对远程主机进行撤销连接。该过程是撤销一个 TCP 连接。

### 3. 与 Telnet 服务相关的文件

Telnet 服务使用未加密的用户名/密码组进行认证,依附于 xinetd 服务,与 Telnet 服务相关为/etc/xinetd.d/telnet,其文件内容如下:

```
# default: on
# description: The telnet server serves telnet sessions; it uses\
# unencrypted username/password pairs for authentication.
service telnet
{
    flags          = REUSE
    socket_type   = stream
    wait           = no
    user           = root
    server         = /usr/sbin/in.telnetd
    log_on_failure += USERID
    disable        = yes
}
```

文件中参数的含义如表 5-2 所示。

表 5-2 Telnet 文件中参数的含义

参 数	含 义	参 数	含 义
flags=REUSE	额外使用的参数	socket_type=stream	TCP 封包的联机形态
wait=no	联机时不需要等待	user=root	启动程序的使用者身份
server=/usr/sbin/in.telnetd	服务启动的程序	log_on_failure+=USERID	记录下登录错误的信息
disable=yes	服务预设是关闭		

### 4. 安装 Telnet 的方法

Telnet 的安装包分为客户端和服务器端,客户端的软件包为 telnet-0.17-46.el6.i686.rpm,服务器端的软件包为 telnet-server-0.17-46.el6.i686.rpm。在安装之前建议先用命令:

```
rpm - qa | grep telnet
```

检查系统中是否已经安装了 Telnet 软件,若没有安装,可以使用 rpm 或 yum 命令进行安装。

**注意:** 客户端既可使用 rpm 也可使用 yum 命令安装,而服务器端软件存在依赖关系,建议使用 yum 命令安装,因为 yum 可以解决软件包之间的依赖关系。

### 5. 启动 Telnet 的方法

启动 Telnet 服务可以通过 chkconfig 命令完成,也可以通过修改 Telnet 服务的配置文件完成。

#### 1) 直接修改配置文件

编辑/etc/xinetd.d/telnet 文件,将其中的 disable 的值改为 no, 使用 service xinetd start 命令,重启 xinetd 服务,此时 Telnet 服务将生效。

## 2) 使用 chkconfig 命令

chkconfig 命令主要用来更新(启动或停止)和查询系统服务的运行级别信息。注意：chkconfig 不是立即自动禁止或激活一个服务,它只是简单地改变了符号连接。

chkconfig 的语法如下：

```
chkconfig [ --add ][ --del ][ --list ][ 系统服务 ]
```

或

```
chkconfig [ --level <等级代号> ][ 系统服务 ][ on/off/reset ]
```

参数用法如下。

(1) chkconfig --list [name]：显示所有运行级的系统服务的运行状态信息(on 或 off)。如果指定了 name,那么只显示指定的服务在不同运行级的状态。

(2) chkconfig --add name：增加一项新的服务。chkconfig 确保每个运行级有一项程序入口。若有缺少则会从默认的 init 脚本自动建立。

(3) chkconfig --del name：删除服务，并把相关符号连接从/etc/rc[0-6].d 删除。

(4) chkconfig [--level levels] name：设置某一服务在指定的运行级是被启动、停止还是重置。

(5) --level <等级代号>：指定系统服务要在哪一个执行等级中开启或关闭。

等级 0 表示：表示关机。

等级 1：表示单用户模式。

等级 2：表示无网络连接的多用户命令行模式。

等级 3：表示有网络连接的多用户命令行模式。

等级 4：表示不可用。

等级 5：表示带图形界面的多用户模式。

等级 6：表示重新启动。

需要说明的是,level 选项可以指定要查看的运行级而不一定是当前运行级。对于每个运行级,只能有一个启动脚本或者停止脚本。当切换运行级时,init 不会重新启动已经启动的服务,也不会再次去停止已经停止的服务。

若需要在运行级 3、5 运行 Telnet 服务,可使用下面命令：

```
[root@localhost ~]# chkconfig -- level 35 telnet on
```

此时/etc/xinetd.d/telnet 文件中的 disable 的值会由 yes 变为 no。

## 6. Telnet 客户端的登录方法

Telnet 是一种远程连接协议,若不加参数将进入 Telnet 的客户端命令状态。此状态也可以在终端中通过按 Ctrl+] 键实现,如图 5-5 所示。在客户端命令状态下输入“help”可以查看帮助,输入“open IP”命令将登录到具有指定 IP 的主机。也可以直接使用 Telnet IP 的方式进入目标主机。

成功进入目标主机后,会出现如图 5-6 所示的内容(设服务器的 IP 地址为 172.17.2.250/24,



图 5-5 telnet 命令参数

客户端的 IP 为 172.17.2.202/24)。

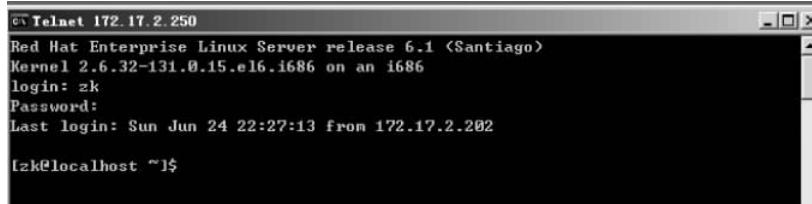


图 5-6 成功登录界面

**注意:** 在默认情况下, Telnet 客户端不允许使用管理员账户 root 远程登录, 因为 root 账户的权限过高, 而 Telnet 用明文传输用户名及密码对, 一旦密码丢失, 会对系统带来致命的损害。若希望使用 root 账户直接在 Telnet 客户端登录, 可以将/etc/securetty 文件改名, 设改名为 securetty.back, 则命令如下:

```
[root@localhost ~]# mv /etc/securetty /etc/securetty.back
```

执行完此操作后就可以使用 root 账户在 Telnet 客户端登录了。不过, 一般不建议这么做, 正确的方法是使用普通用户登录, 再用 su 命令切换为 root 账户, 这样可以提高系统的安全性。

### 5.3.2 SSH 配置

SSH(Secure Shell)由 IETF 的网络工作小组(Network Working Group)所制定, 是建立在应用层和传输层基础上的安全协议。SSH 是目前较可靠、专为远程登录会话和其他网络服务提供安全性的协议。利用 SSH 协议可以有效地防止远程管理过程中的信息泄露问题。

#### 1. SSH 协议的组成

SSH 协议主要由以下 3 部分组成:

1) 传输层协议(SSH-TRANS)

提供了服务器认证、保密性及完整性, 有时还提供压缩功能。SSH-TRANS 通常运行

在 TCP/IP 连接上,也可能用于其他可靠数据流上。SSH-TRANS 提供了强大的加密技术、密码主机认证及完整性保护。

### 2) 用户认证协议(SSH-USERAUTH)

用于向服务器提供客户端用户鉴别功能,运行在传输层。当 SSH-USERAUTH 开始后,它从低层协议那里接收会话标识符。会话标识符唯一标识此会话并且适用于标记以证明私钥的所有权。

### 3) 连接协议(SSH-CONNECT)

将多个加密隧道分成逻辑通道,运行在用户认证协议上,提供了交互式登录方式,允许远程执行命令,可以转发 TCP/IP 连接和 X11 连接。

## 2. SSH 的结构

SSH 是由客户端和服务器端的软件组成的。服务器端是一个守护进程(daemon),在后台运行并响应来自客户端的连接请求,一般是 sshd 进程,提供了对远程连接的处理,一般包括公共密钥认证、密钥交换、对称密钥加密和非安全连接;客户端包含 SSH 程序以及像 scp(远程复制)、slogin(远程登录)、sftp(安全文件传输)等其他应用程序。

SSH 的工作过程是:本地客户端发送一个连接请求到远程的服务器端,服务器端检查申请的包和 IP 地址,发送密钥给 SSH 的客户端,本地再将密钥发回给服务器端,自此连接建立。

## 3. SSH 服务器的配置

### 1) 安装

在 RHEL 6.1 中,SSH 的服务器端软件包为 openssh-server-5.3pl52.el6.i686,可以使用 yum 或 rpm 命令进行安装。

### 2) 配置

在一般情况下无须对 SSH 服务器做任何配置,只需要启动 SSH 服务即可。SSH 的配置文件位于/etc/ssh 目录下,名为 ssh\_config,文件部分内容如下:

```
Host *          # 只对匹配后面字串的计算机有效,“*”代表所有计算机
ForwardAgent no      # 设置连接不经过认证代理,若存在则转发给远程计算机
ForwardX11 no       # 设置 X11 连接不被自动重定向到安全的通道和显示集
RhostsRSAAuthentication no # 设置不使用基于 rhosts 的安全认证
RSAAuthentication yes    # 设置使用 RSA 算法的基于 rhosts 的安全认证
PasswordAuthentication yes # 设置使用密码认证
HostbasedAuthentication no # 不使用主机认证
BatchMode no        # 如果设置为 yes,交互式输入密码的提示将被禁止
CheckHostIP yes     # 设置 SSH 查看连接到服务器主机的 IP,以防止 DNS 欺骗,建议为 yes
AddressFamily any
Port 22           # 设置 sshd 监听的端口号
Protocol 2,1
Cipher 3des        # 设置加密算法
Ciphers aes128 - ctr,aes192 - ctr,aes256 - ctr,arcfour256,arcfour128,aes128 - cbc,3des - cbc
MACs hmac - md5,hmac - sha1,umac - 64@openssh.com,hmac - ripemd160
EscapeChar ~        # 设置 esc
```

## 3) 启动

```
[root@localhost ~]# service sshd start
```

## 4) 在 Windows 中测试

在 Windows 中的默认情况下没有安装 SSH 工具,需要单独安装。这里使用的工具为 putty。设服务器的 IP 为 172.17.2.220/24,客户端的 IP 为 172.17.2.202/24,登录服务器的界面如图 5-7 所示。

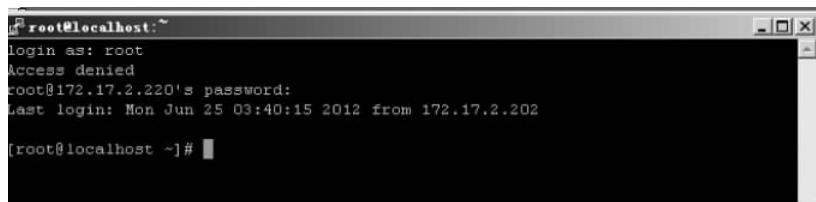


图 5-7 使用 SSH 访问服务器

### 5.3.3 远程桌面

远程桌面连接是一种远程操作计算机的模式,可用于可视化访问远程计算机的桌面环境,用于在客户端上对远程计算机服务器进行管理。远程桌面的前身是 Telnet。Telnet 是一种字符界面的登录方式,微软首先将其扩展到图形界面上,并提供了非常强大的功能。现在几乎所有的图形化操作系统都支持远程桌面功能,远程桌面在实际应用中是非常有用的。

在 Linux 下的 VNC 可以同时启动多个 vncserver,各个 vncserver 之间用编号区分,每个 vncserver 服务监听 3 个端口,分别如下。

- (1) 5800+ 编号: VNC 的 httpd 监听端口,如果 VNC 客户端为 IE、Firefox 等非 vncviewer 时,此端口必须开放。
- (2) 5900+ 编号: VNC 服务器端与客户端通信的真正端口,无条件开放。
- (3) 6000+ 编号: 监听端口,可选。

#### 1. 安装 VNC 服务端软件

在 RHEL 6.1 中的默认情况下,VNC 服务器端的软件包是没有安装的。VNC 服务器端的软件包为 tigervnc-server-1.0.90-0.15.20110314svn4359.el6.i686.rpm,可以通过 yum 安装也可以使用 rpm 安装。本例中使用 rpm 进行安装,方法如下:

```
[root@localhost 桌面] # rpm -ivh /media/RHEL_6.1\ i386\ Disc\ 1/Packages/tigervnc - server - 1.0.90 - 0.15.20110314svn4359.el6.i686.rpm
warning:      /media/RHEL_6.1          i386          Disc 1/Packages/tigervnc - server - 1.0.90 - 0.15.20110314svn4359.el6.i686.rpm: Header V3 RSA/SHA256 Signature, key ID fd431d51: NOKEY
Preparing...          ##### [100 %]
1:tigervnc - server      ##### [100 %]
```

#### 2. 修改 VNC 服务器端的配置文件

vncservers 文件内容如下:

```
[root@localhost 桌面]# vim /etc/sysconfig/vncservers
# The VNCSERVERS variable is a list of display:user pairs.
# Uncomment the lines below to start a VNC server on display :2
# as my 'myusername' (adjust this to your own). You will also
# need to set a VNC password; run 'man vncpasswd' to see how
# to do that.
# DO NOT RUN THIS SERVICE if your local area network is
# untrusted! For a secure way of using VNC, see this URL:
# http://kbase.redhat.com/faq/docs/DOC-7028
# Use "-nolisten tcp" to prevent X connections to your VNC server via TCP.
# Use "-localhost" to prevent remote VNC clients connecting except when
# doing so through a secure tunnel. See the "-via" option in the
# 'man vncviewer' manual page.
# VNCSERVERS = "2:username"
# VNCERVERARGS[2] = "-geometry 800×600 -nolisten tcp -localhost"
```

将文件最后两行内容修改如下：

```
VNCSERVERS = "2:root"
VNCERVERARGS[2] = "-geometry 800×600"
```

VNC 服务器端的编号、开放的端口分别由/etc/sysconfig/vncservers 文件中的 VNC SERVERS 和 VNC SERVERARGS 控制。VNC SERVERS 的设置方式为：

```
VNCSERVERS = "编号 1: 用户名 1 … … "
```

例如：

```
VNCSERVERS = "1:root 2:user1"
```

VNC SERVERARGS 的设置方式为：

```
VNCERVERARGS[ 编号 1 ] = "参数一参数值一参数二参数值二 … … "
```

例如：

```
VNCERVERARGS[1] = "geometry 800 * 600 - nohttpd"
```

VNC SERVERARGS 的详细参数如下。

- (1) **geometry**: 桌面分辨率，默认为 1024×768。
- (2) **-nohttpd**: 不监听 HTTP 端口(58××端口)。
- (3) **-nolisten tcp**: 不监听×端口(60××端口)。
- (4) **-localhost**: 只允许从本机访问。
- (5) **-AlwaysShared**: 默认只同时允许一个 vncviewer 连接，此参数允许同时连接多个 vncviewer。
- (6) **-SecurityTypes None**: 登录不需要密码认证 VncAuth 默认值，要密码认证。

### 3. 为 VNC 用户创建密码

```
[root@localhost 桌面]# vncpasswd root
Password:
Verify:
```

### 4. 修改/root/.vnc/xstartup 文件

```
[root@localhost 桌面]# vim /root/.vnc/xstartup
```

注释掉文件中的“twm &.”, 加入“gnome-session &.”, 目的是在 VNC 客户端中使用 GNOME 桌面系统。修改后的文件内容如下：

```
#!/bin/sh
[ -r /etc/sysconfig/i18n ] && . /etc/sysconfig/i18n
export LANG
export SYSFONT
vncconfig -iconic &
unset SESSION_MANAGER
unset DBUS_SESSION_BUS_ADDRESS
OS = `uname -s`
if [ $OS = 'Linux' ]; then
    case "$WINDOWMANAGER" in
        *gnome*
            if [ -e /etc/SuSE-release ]; then
                PATH = $PATH:/opt/gnome/bin
                export PATH
            fi;;
        esac
    fi
    if [ -x /etc/X11/xinit/xinitrc ]; then
        exec /etc/X11/xinit/xinitrc
    fi
    if [ -f /etc/X11/xinit/xinitrc ]; then
        exec sh /etc/X11/xinit/xinitrc
    fi
[ -r $HOME/.Xresources ] && xrdb $HOME/.Xresources
xsetroot -solid grey
# xterm -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
# twm &
gnome-session &
```

**注意：**如果在用户主文件夹下找不到.vnc/xstartup 文件, 在启动 VNC 服务后, 正常情况下会在/etc/sysconfig/vncservers 文件 VNCSERVERS 参数指定的用户主文件夹中产生一个.vnc/xstartup 文件, 本例中的用户为 root。

### 5. 启动 VNC 服务

```
[root@localhost 桌面]# service vncserver start
```

提示信息如下：

```
正在启动 VNC 服务器: 2:root xauth: creating new authority file /root/.Xauthority
```

```
New 'localhost.localdomain:2 (root)' desktop is localhost.localdomain:2
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/localhost.localdomain:2.log
```

[确定]

## 6. 设置防火墙

若熟悉 Iptables 的使用方法,可以直接修改/etc/sysconfig/iptables 文件后执行 service iptables restart 命令重启防火墙服务,对于初学者建议直接将防火墙关闭,命令如下:

```
[root@localhost 桌面]# iptables -F
```

## 7. 远程桌面登录

在客户端打开 VNC 客户端工具,本例中使用的是 VNC Viewer,结果如图 5-8 所示。设服务器的 IP 地址为 172.17.2.220/24,客户端的 IP 地址为 172.17.2.202/24。

在图 5-8 中输入 VNC 服务器的 IP,注意要添加好端口号,若成功连接,则显示结果如图 5-9 所示。

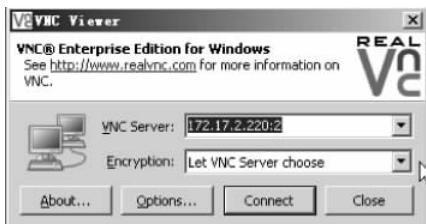


图 5-8 VNC Viewer 界面



图 5-9 客户端连接后的界面

在图 5-9 的 Password 后面的文本框中输入 VNC 密码(在服务器用 vncpasswd 命令设置的密码)。若认证通过,则会显示服务器的桌面,如图 5-10 所示。

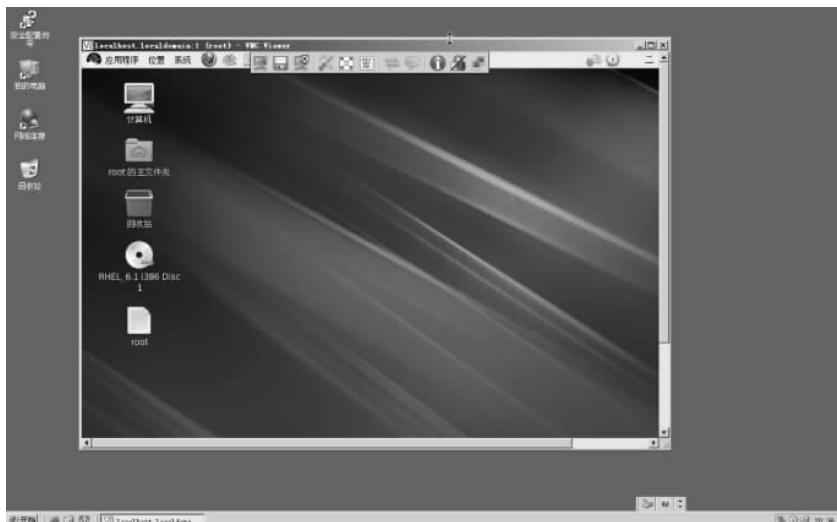


图 5-10 远程桌面登录成功

当远程配置结束后,可单击 VNC Viewer 工具中的 Close connection 按钮断开与服务器的连接。若希望在服务器端停止 VNC 服务,可以执行下面的命令来停止 VNC 服务。

```
[root@localhost 桌面]# vncserver -kill :1
Killing Xvnc process ID 2360
```

## 5.4 FTP 配置

### 5.4.1 FTP 介绍

FTP(File Transfer Protocol,文件传输协议)在 Internet 中有着广泛的应用,早在 Internet 发展初期就与 Web 服务、E-mail 服务一起被列为 Internet 的三大应用。利用 FTP 可以方便地实现软件、文件等资源的共享。使用 FTP 协议可以在 Internet 上传输文件数据,下载或者上传各种软件、文档等资料。FTP 服务需要使用的两个端口分别为 20 和 21,其中 20 号端口用于控制连接,发布 FTP 命令信息;21 号端口用于控制数据的上传和下载。

#### 1. FTP 连接模式

FIP 连接模式分为主动模式(Active FTP)和被动模式(Passive FTP)。

##### 1) 主动模式(Active FTP)

在主动模式下,FTP 客户端随机开启一个大于 1024 的端口(端口 AA)向服务器的 21 号端口发起连接,然后开放 BB 号端口进行监听,并向服务器发出 PORT N+1 命令。服务器接收到命令后,会用其本地的 FTP 数据端口(通常是 20 端口)连接客户端指定的端口 BB,进行数据传输,如图 5-11 所示。

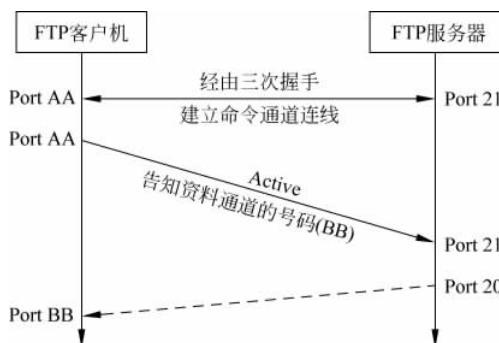


图 5-11 主动模式的 FTP 连接

主动模式连接步骤如下:

- (1) 建立命令通道连接。
- (2) 通知 FTP 服务器端使用 Active 且告知连接的端口号。
- (3) FTP 服务器主动向客户端连接。

##### 2) 被动模式(Passive FTP)

在被动模式下,FTP 客户端随机开启一个大于 1024 的端口 N 向服务器的 21 号端口发起连接,同时会开启 N+1 号端口;然后向服务器发送 PASV 命令,通知服务器自己处于被

动模式；服务器收到命令后，会开放一个大于 1024 的端口 P 进行监听，然后用 PORT P 命令通知客户端服务器的数据端口是 P；客户端收到命令后，会通过 N+1 号端口连接服务器的端口 P，然后在两个端口之间进行数据传输，如图 5-12 所示。

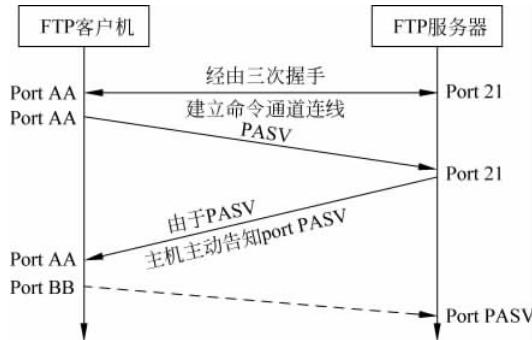


图 5-12 被动模式的 FTP 连接

被动模式连接步骤如下：

- (1) 建立命令通道。
- (2) 发出 PASV 的连接要求。
- (3) FTP 服务器启动数据端口，并通知客户端连接。
- (4) 客户端随机取用大于 1024 的端口进行连接。

## 2. VSFTP 介绍

目前在 RedHat 中支持的 FTP 服务器软件为 VSFTP，其含义为 Very Secure FTP，是一个基于 GPL 发布的类 UNIX 系统上使用的 FTP 服务器软件。VSFTP 除了与生俱来的安全性之外，还具有高速、稳定的特性。

### 1) VSFTP 的特点

- (1) 它是一个安全、高速、稳定的 FTP 服务器。
- (2) 它可以做基于多个 IP 的虚拟 FTP 主机服务器。
- (3) 匿名服务设置十分方便。
- (4) 匿名 FTP 的根目录不需要任何特殊的目录结构、系统程序或其他系统文件。
- (5) 不执行任何外部程序，从而减少了安全隐患。
- (6) 支持虚拟用户，并且每个虚拟用户可以具有独立的属性配置。
- (7) 可以设置从 inetd 中启动或者独立的 FTP 服务器两种运行方式。
- (8) 支持两种认证方式(PAP 或 xinetd/tcp\_wrappers)。
- (9) 支持带宽限制。

### 2) VSFTP 的配置文件

VSFTP 的配置文件主要有 vsftpd.conf、vsftpd.ftpusers、vsftpd.user\_list 等，如表 5-3 所示。

### 3) VSFTP 支持的账户类型

- (1) 匿名账户：在登录 FTP 服务器时不需要输入密码就可以访问 FTP 服务器，匿名账户名称为 anonymous 或 ftp，匿名账户的登录目录为 /var/ftp/pub。

表 5-3 VSFTP 配置文件

配置文件或路径	说 明
/etc/vsftpd/vsftpd.conf	VSFTP 服务器的主要配置文件
/etc/pam.d/vsftpd	PAM 认证文件,用来标识虚拟用户
/etc/vsftpd.ftpusers	禁止使用 VSFTP 服务的用户列表
/etc/vsftpd.user_list	禁止或允许使用 VSFTP 服务的用户列表,但这个文件能否生效取决于 vsftpd.conf 配置文件中的 userlist_enable 和 userlist_deny 两个参数
/usr/sbin/vsftpd	VSFTP 的主程序
/var/ftp	默认匿名用户登录的主目录
/var/ftp/pub	匿名用户的下载目录

(2) 本地实体账户：具有本地权限的账户，登录 FTP 服务器时需要输入用户名、密码，登录目录为自己的主目录。

(3) 虚拟账户：虚拟账户只具有从远程登录 FTP 服务器的权限，只能访问为其提供的 FTP 服务，密码和用户名都是由用户密码库指定，采用 PAM 认证。虚拟账户不能在本地登录。

### 3. VSFTP 的主配置文件 vsftpd.conf 的主要内容

```

anonymous_enable = YES/NO          # 是否允许匿名者登录
local_enable = YES/NO             # 是否允许/etc/passwd 内的账户以实体用户方式登录
write_enable = YES/NO              # 是否允许本地实体账户具有上传权限
anon_upload_enable = YES/NO        # 是否允许匿名者具有上传的权限
anon_mkdir_write_enable = YES/NO   # 是否允许匿名者具有建立目录的权限
anon_other_write_enable = YES/NO   # 是否允许匿名者改名或删除文件
dirmessage_enable = YES/NO         # 当用户进入某个目录时,会显示该目录的提示信息
xferlog_enable = YES/NO            # 是否记录用户上传的文件记录
connect_from_port_20 = YES/NO       # 连接时打开 20 号端口
chown_uploads = YES/NO             # 上传身份是否改变
chown_username = whoever           # 改变上传文件的属主身份为 whoever
xferlog_file = /var/log/vsftpd.log # 日志文件所在目录和文件名
xferlog_std_format = YES           # 日志格式
idle_session_timeout = 600          # 如果用户在 600 秒内都没有命令操作,则强制退出
data_connection_timeout = 120        # 如果服务器与客户端的数据连接已经建立(不论主动还是被动连接),若 120 秒内还是无法顺利完成数据传输,那么客户端的连接就会被强制剔除
nopriv_user = ftpsecure            # 运行 vsftpd 需要的非特权系统用户,默认是 nobody
ascii_upload_enable = YES/NO        # client 是否可以使用 ASCII 格式上传文件
ascii_download_enable = YES/NO       # 是否可以使用 ASCII 格式下载文件
ftpd_banner = Welcome to blah FTP service. # 登录服务器的欢迎信息
deny_email_enable = YES/NO          # 拒绝邮箱登录
banned_email_file = /etc/vsftpd.banned_emails # 如果上一项设置为 YES,可以在这个文件定义不允许登录的 E-mail 地址
chroot_local_user = YES/NO          # 是否将本地用户限制在他们的默认目录中
chroot_list_enable = YES/NO          # 是否将用户限制在他们的默认目录里
chroot_list_file = /etc/vsftpd.chroot_list # 如果上一项为 YES,则实体用户无法离开他们的默认目录,必须结合 chroot_list_enable = YES 使用

```

```

userlist_enable = YES/NO          # 是否借助 vsFTP 的阻止机制(/etc/vsftpd/user_list 文件中的内容)来处理那些不受欢迎的账户
userlist_deny = YES              # 是否禁用/etc/vsftpd/user_list 中的用户
pam_service_name = vsftpd        # 认证方式
listen = YES/NO                  # 若设置为 YES 表示 vsFTP 是以 stand alone 的方式启动的
tcp_wrappers = YES/NO            # 是否设置为 TCP 封装
local_root = /                   # 使用本地用户登录到 FTP 时的默认目录
anon_root = /                   # 匿名用户登录到 FTP 时的默认目录
local_max_rate = 50000           # 本地用户的传输速度为 50bps
anon_max_rate = 30000            # 匿名用户的传输速度为 30bps
max_clients = 200                # vsFTP 服务器最大的并发连接数为 200,若此值为 0,则说明不限制并发连接数,与服务器性能有关
max_per_ip = 4                  # 单个 IP 地址最多的并发连接数为 4

```

## 5.4.2 FTP 的登录方式及常用命令

### 1. FTP 的命令格式

FTP 命令的一般格式为：

```
ftp 主机名/IP 或 ftp 用户名@主机名/IP
```

其中,主机名/IP 是所要连接的远程机的主机名或 IP 地址。在命令行中,主机名属于选项,如果指定主机名,FTP 将试图与远程机的 FTP 服务程序进行连接;如果没有指定主机名,FTP 将给出提示符“`ftp >`”,等待用户输入命令,此时在提示符后面可以输入 FTP 的内部命令,可以用 `help` 命令取得可供使用的命令清单,也可以在 `help` 命令后面指定具体的命令名称,获得这条命令的说明。系统中默认的匿名账户为 `anonymous`(也称为匿名 FTP),系统各为匿名账户专门提供了两个目录: `pub` 目录和 `incoming` 目录。`pub` 目录用于存放供下载的文件;`incoming` 目录需要自己创建,目录中存放上传到该站点的文件。

### 2. FTP 命令提示符中常用的命令

在 FTP 的客户端提供了丰富的命令,用于对 FTP 服务器进行操作,例如上传、下载等,此处只列举出部分常用的命令。

- (1) `ls`: 列出 FTP 服务器的当前目录。
- (2) `cd`: 在 FTP 服务器上改变工作目录。
- (3) `lcd`: 在本地机上改变工作目录。
- (4) `ascii`: 设置文件传输方式为 ASCII 模式。
- (5) `binary`: 设置文件传输方式为二进制模式。
- (6) `close`: 终止当前的 FTP 会话。
- (7) `hash`: 每次传输完数据缓冲区中的数据后就显示一个#号。
- (8) `get(mget)`: 下载到客户端。
- (9) `put(mput)`: 上传到服务器。
- (10) `open`: 连接远程 FTP 站点。

**例 5-38 启动 FTP 会话。**

```
open 主机名/IP
```

如果在 FTP 会话期间要与一个以上的站点连接,通常只用不带参数的 FTP 命令。如果在会话期间只想与一台计算机连接,那么在命令行上指定 FTP 服务器的域名或 IP 地址作为 FTP 命令的参数。

**例 5-39 终止 FTP 会话。**

可以使用 close、disconnect 和 bye 命令终止与远程主机的会话。区别在于 close 和 disconnect 命令关闭与远程机的连接后用户仍留在本地计算机的 FTP 程序中;而 bye 命令关闭用户与远程机的连接后退出用户机上的 FTP 程序。

**例 5-40 改变目录。**

cd [目录]命令用于在 FTP 会话期间改变在 FTP 服务器上的目录。lcd 命令用于改变本地目录,使用户能指定查找或放置本地文件的位置。

**例 5-41 列出远程目录。**

使用 ls 命令,列出远目录的内容。ls 命令的一般格式是:

```
ls [目录] [本地文件]
```

如果指定了目录作为参数,那么 ls 就列出该目录的内容;如果给出一个本地文件的名字,那么这个目录列表就被放入本地主机上指定的这个文件中。

**例 5-42 从远程系统获取文件。**

使用 get 命令和 mget 命令。get 命令的一般格式为:

```
get 文件名
```

mget 命令一次下载多个远程文件,其一般格式为:

```
mget 文件名列表
```

mget 命令使用空格分隔的或带通配符的文件名列表来指定要下载的文件,对其中的每个文件都要求用户确认是否下载。

**例 5-43 上传文件。**

put 命令和 mput 命令用于上传文件。put 命令的一般格式为:

```
put 文件名
```

mput 命令一次上传多个本地文件,其一般格式为:

```
mput 文件名列表
```

mput 命令使用空格分隔的或带通配符的文件名列表来指定要上传的文件,对其中的每个文件都要求用户确认是否上传。

**例 5-44 改变文件传输模式。**

默认情况下,FTP 按 ASCII 模式传输文件,用户也可以指定其他模式如 binary 模式。ASCII 模式用于传输纯文本文件,brinary 模式用于传输二进制文件。

**例 5-45 检查传输状态。**

传输大型文件时,可能会发现让 FTP 提供关于传输情况的反馈信息是非常有用的。使用 hash 命令在每次传输完数据缓冲区中的数据后,就在屏幕上打印一个#字符。本命令在发送和接收文件时都可以使用。

**例 5-46 运行 Shell 命令。**

字符“!”用于在 FTP 会话中向本地主机上的 Shell 发送命令。若要建立目录,可输入“!mkdir dir\_name,”Linux 会在用户当前的本地目录中创建一个名为 dir\_name 的目录。

### 5.4.3 任务 5-1: 匿名账户和实体账户登录 FTP 实验

#### 1. 任务描述

设某公司内部有一台 FTP 服务器,本地实体账户可以上传下载资源,匿名账户只能下载;FTP 客户端登录的用户不能改变登录的目录位置。设实体账户为 user1,FTP 服务器的 IP 地址为 172.17.2.1/24;客户端的 IP 地址为 172.17.2.202/24。其中 IP 地址已经配置好,此处不再详述。

#### 2. 操作步骤

##### 1) 安装 FTP 服务软件包

在 RHEL 6.1 中默认情况下是没有安装 VSFTPD 服务软件包的,查询结果如下:

```
[root@localhost 桌面]# rpm -qa | grep vsftpd
```

安装结果为:

```
[root@localhost 桌面]# rpm -ivh /media/RHEL_6.1\ i386\ Disc\ 1/Packages/*vsftpd*  
warning: /media/RHEL_6.1 i386 Disc 1/Packages/vsftpd-2.2.2-6.el6_0.1.i686.rpm: Header V3  
RSA/SHA256 Signature, key ID fd431d51: NOKEY  
Preparing... ##### [100 %]  
1:vsftpd ##### [100 %]
```

##### 2) 创建实体用户 user1

```
[root@localhost 桌面]# useradd user1  
[root@localhost 桌面]# passwd user1  
更改用户 user1 的密码。  
新的 密码:  
无效的密码: 它没有包含足够的不同字符  
无效的密码: 是回文  
重新输入新的 密码:  
passwd: 所有的身份认证令牌已经成功更新。
```

3) 在 FTP 服务器的默认下载目录中创建实验用文件 1.txt

```
[root@localhost 桌面]# echo how are you > /var/ftp/pub/1.txt
```

4) 修改配置文件

```
[root@localhost 桌面]# vim /etc/vsftpd/vsftpd.conf
```

修改内容如下：

```
anonymous_enable = YES
local_enable = YES
write_enable = YES
local_umask = 022
ftpd_banner = You are fine!
chroot_list_enable = YES
chroot_list_file = /etc/vsftpd/chroot_list
```

5) 创建/etc/vsftpd/chroot\_list 文件

由于该文件默认情况是不存在的,所示需要自己创建。文件中出现的用户不允许切换登录的目录,一个用户占一行,本处只添加一个用户 user1。

```
[root@localhost 桌面]# touch /etc/vsftpd/chroot_list
```

内容如下：

```
user1
```

6) 重启 FTP 服务

```
[root@localhost 桌面]# service vsftpd start
为 vsftpd 启动 vsftpd: [确定]
```

7) 关闭防火墙及 SELinux

```
[root@localhost 桌面]# iptables -F
[root@localhost 桌面]# setenforce 0
```

8) 在客户端中进行测试(以 Windows 为例)

(1) 测试匿名账户的权限。

在 IE 地址栏中输入“`ftp://172.17.2.1`”,结果如图 5-13 所示。双击 pub 图标可看到 FTP 上的资源 `1.txt`,如图 5-14 所示。

说明：在 IE 地址栏中输入 FTP 命令后,若没有用户名则默认使用匿名访问。下载文件时可直接拖动目标文件到本地目录中。

**提示：**若在下载文件时出现错误提示,如图 5-15 所示,则说明在本地的 IE 中不信任 FTP 服务器站点。修改方法为：单击 IE 的“工具”→“Internet 选项”→“安全”→“受信任的站点”,将 FTP 服务器的 IP 地址添加到受信任的站点中,此时刷新 IE 浏览器,就可以下载了。

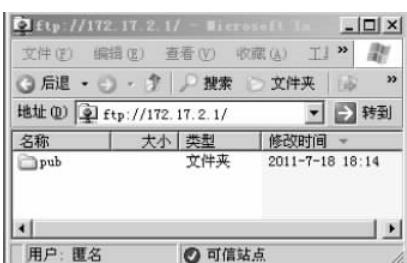


图 5-13 匿名成功访问 FTP 服务器

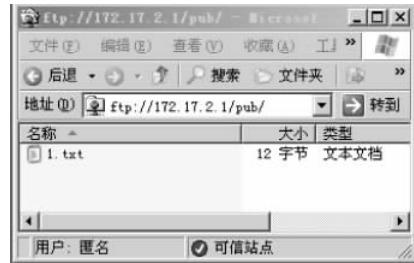


图 5-14 可供下载的资源



图 5-15 在 IE 中不允许下载的对话框

## (2) 测试实体账户 user1 的权限。

在 IE 地址栏中输入“`ftp:user1@172.17.2.1`”，结果如图 5-16 所示，输入 user1 的密码后，出现窗口如图 5-17 所示。user1 用户可上传/下载 FTP 服务器上的资源，下载方法同匿名访问的下载方法，上传时只需要将文件拖动到 FTP 服务器即可，结果如图 5-18 所示。



图 5-16 user1 账户登录 FTP 服务器



图 5-17 user1 成功登录 FTP 服务器



图 5-18 user1 正在上传文件

操作完成。

#### 5.4.4 任务 5-2：虚拟账户登录 FTP 实验

##### 1. 任务描述

设某公司内部有一台安装好 Linux 系统的主机，要求创建 FTP 服务器，管理员 manager 可上传/下载/删除文件，传输速率为 1Mbps；员工 user1 可上传/下载文件，但不能删除文件，传输速率为 500Kbps；用户 user2 只能下载文件，不能上传文件，传输速率为 300Kbps。设 FTP 服务器的 IP 地址为 172.17.2.1/24，客户端的 IP 地址为 172.17.2.202/24。由于实体账户不但可以在 FTP 客户端登录也可以在系统中直接登录，权限过大，会给系统带来不安全的隐患；而匿名账户只能从远程的 FTP 客户端登录，受限制较多，权限又太少，操作不灵活。所以在此任务中，使用虚拟账户。虚拟账户不能在本地登录，但可以在远程 FTP 客户端登录，可以对虚拟账户进行灵活的权限设置。

##### 2. 操作步骤

###### 1) 建立虚拟用户密码库文件

```
[root@localhost 桌面]# vim /etc/vsftpd/vftppuser.txt
```

奇数行是用户名，偶数行是密码，不能有空格，内容如下：

```
manager
123456
user1
123456
user2
123456
```

###### 2) 创建的密码库文件生成 vsftpd 认证文件

需要使用 db4-utils 工具，此包在 RHEL 6.1 中默认已经安装好，查看结果如下：

```
[root@localhost 桌面]# rpm -qa | grep db4-utils
db4-utils-4.7.25-16.el6.i686
```

### 3) 创建 PAM 配置文件

```
[root@localhost 桌面] # db_load -T -t hash -f /etc/vsftpd/vftpuser.txt /etc/vsftpd/vftpuser.db
```

其中,-T 和-t 为 db\_load 命令的固有参数; hash 表示用 hash 算法对认证文件进行密码加密;-f 为 hash 的固有参数; /etc/vsftpd/vftpuser.txt 为记录用户名密码的文本文件;/etc/vsftpd/vftpuser.db 为生成的认证数据库文件。

### 4) 修改认证模块文件

注释掉原有内容,在文件的最后面加上两行,内容如下:

```
[root@localhost 桌面] # vim /etc/pam.d/vsftpd
# %PAM-1.0
# session optional pam_keyinit.so force revoke
# auth required pam_listfile.so item=user sense=deny file=/etc/vsftpd/ftusers onerr=succeed
# auth required pam_shells.so
# auth include password-auth
# account include password-auth
# session required pam_loginuid.so
# session include password-auth
auth required /lib/security/pam_userdb.so db=/etc/vsftpd/vftpuser
account required /lib/security/pam_userdb.so db=/etc/vsftpd/vftpuser
```

### 5) 建立一个本地用户供虚拟用户使用并设置权限

```
[root@localhost 桌面] # useradd -d /home/vftpsite -s /sbin/nologin vftpuser
```

### 6) 设置虚拟用户主目录的访问权限

```
[root@localhost 桌面] # chmod 700 /home/vftpsite
```

### 7) 为虚拟用户创建主目录

```
[root@localhost 桌面] # mkdir /home/vftpsite/manager
[root@localhost 桌面] # mkdir /home/vftpsite/user1
[root@localhost 桌面] # mkdir /home/vftpsite/user2
```

### 8) 修改虚拟用户主目录权限

```
[root@localhost 桌面] # chmod 700 /home/vftpsite/manager/
[root@localhost 桌面] # chown vftpuser.vftpuser /home/vftpsite/manager/
[root@localhost 桌面] # chmod 700 /home/vftpsite/user1/
[root@localhost 桌面] # chown vftpuser.vftpuser /home/vftpsite/user1/
[root@localhost 桌面] # chmod 700 /home/vftpsite/user2
[root@localhost 桌面] # chown vftpuser.vftpuser /home/vftpsite/user2
```

## 9) 修改 FTP 的配置文件

```
[root@localhost 桌面]# vim /etc/vsftpd/vsftpd.conf
```

修改内容如下：

```
pam_service_name = vsftpd
userlist_enable = YES
tcp_wrappers = YES
guest_enable = YES
guest_username = vftpuser
user_config_dir = /etc/vsftpd_user_conf
```

其中, guest\_enable= YES 作用为允许以 guest 方式访问 FTP 服务器; guest\_username=vftpuser 作用是访问 FTP 服务器时将数据库中的虚拟用户转换为 vftpuser 账户; user\_config\_dir=/etc/vsftpd\_user\_conf 作用是指出虚拟用户权限的配置目录。若希望所有虚拟用户都使用同一个目录, 可以使用 local\_root=path 参数进行指定, 此处每个用户都使用自己的文件夹。

## 10) 建立虚拟用户权限目录

```
[root@localhost 桌面]# mkdir /etc/vsftpd_user_conf
```

## 11) 建立虚拟用户权限配置文件

在/etc/vsftpd\_user\_conf 目录下为每个用户建立权限配置文件, 注意文件名必须与虚拟用户名名称一致, “=”两侧不能有空格。

(1) 虚拟用户 manager 的配置文件如下：

```
[root@localhost 桌面]# vim /etc/vsftpd_user_conf/manager
local_root = /home/vftpsite/manager
anon_world_readable_only = YES
anon_upload_enable = YES
anon_mkdir_write_enable = YES
anon_other_write_enable = YES
local_max_rate = 1M
```

(2) 虚拟用户 user1 的配置文件如下：

```
[root@localhost 桌面]# vim /etc/vsftpd_user_conf/user1
local_root = /home/vftpsite/user1
anon_world_readable_only = NO
anon_upload_enable = YES
anon_mkdir_write_enable = YES
local_max_rate = 500K
```

(3) 虚拟用户 user2 的配置文件如下：

```
[root@localhost 桌面]# vim /etc/vsftpd_user_conf/user2
```

```
local_root = /home/vftpsite/user2  
anon_world_readable_only = NO  
local_max_rate = 300K
```

其中,local\_root=/home/vftpsite/username 作用是指定虚拟用户的目录;参数anon\_mkdir\_write\_enable=YES 表示用户具有建立目录的权限,不能删除目录;参数anon\_other\_write\_enable=YES 表示用户具有文件改名和删除文件的权限;参数anon\_upload\_enable=YES 表示用户可以上传文件;参数anon\_world\_readable\_only=NO 表示用户可以浏览FTP 目录和下载文件,若此参数为 YES,则在客户端用 ls 命令访问 FTP 服务器时会出现“226 Transfer done (but failed to open directory)”提示,这是因为/home/vftpsite 目录不允许任意访问,但不妨碍用户上传/下载文件,若希望去掉此提示,可将参数值设置为 NO。

#### 12) 关闭防火墙及 SELinux

```
[root@localhost 桌面]# iptables -F  
[root@localhost 桌面]# setenforce 0
```

#### 13) 重启服务

```
[root@localhost 桌面]# service vsftpd restart  
关闭 vsftpd: [失败]  
为 vsftpd 启动 vsftpd: [确定]
```

#### 14) 测试(使用 FTP 命令行方式完成)

在 Windows 中以命令行的方式对 FTP 服务器进行访问,打开 cmd 命令提示符环境,输入“ftp 172.17.2.1”后回车,在提示符下输入虚拟用户名及密码即可。

(1) 测试 manager 的权限,结果如图 5-19 所示。

从图 5-19 中可以看出,虚拟用户 manager 可上传、下载文件,并可删除文件。

(2) 测试 user1 的权限,结果如图 5-20 所示。

从图 5-20 中可以看出,虚拟用户 user1 可以上传或下载文件,但不能删除文件。

(3) 测试 user2 的权限,结果如图 5-21 所示。

从图 5-21 中可以看出,虚拟用户 user2 可以下载文件,但不能上传文件,也不能删除文件。

**提示:** 启用虚拟用户后,本地用户将默认不能作为 FTP 登入账户,匿名用户不会受到影响。若需要控制虚拟用户对 FTP 服务器的访问,可以在主配置文件/etc/vsftpd/vsftpd.conf 中添加记录:

```
userlist_enable = YES
```

或:

```
userlist_deny = YES
```

并将被允许/禁用的用户名写入/etc/vsftpd/user\_list 文件。

```

C:\Documents and Settings\Administrator>ftp 172.17.2.1
Connected to 172.17.2.1.
220 vsFTPD 2.2.2
User <172.17.2.1:<none>>: manager
331 Please specify the password.
Password:
230 Login successful.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 5 Jul 18 16:11 manager.txt
226 Directory send OK.
ftp: 69 bytes received in 0.00Seconds 69000.00Kbytes/sec.
ftp> get manager.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for manager.txt (5 bytes).
226 Transfer complete.
ftp: 5 bytes received in 0.00Seconds 5000.00Kbytes/sec.
ftp> put download.txt
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
ftp: 6 bytes sent in 0.00Seconds 6000.00Kbytes/sec.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw----- 1 518 511 6 Jul 18 17:21 download.txt
-rw-r--r-- 1 0 0 5 Jul 18 16:11 manager.txt
226 Directory send OK.
ftp: 139 bytes received in 0.00Seconds 139000.00Kbytes/sec.
ftp> delete download.txt
250 Delete operation successful.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 5 Jul 18 16:11 manager.txt
226 Directory send OK.
ftp: 69 bytes received in 0.00Seconds 69000.00Kbytes/sec.
ftp> quit

```

图 5-19 虚拟用户 manager 的权限测试结果

```

C:\Documents and Settings\Administrator>ftp 172.17.2.1
Connected to 172.17.2.1.
220 vsFTPD 2.2.2
User <172.17.2.1:<none>>: user1
331 Please specify the password.
Password:
230 Login successful.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 6 Jul 18 14:41 user1.txt
226 Directory send OK.
ftp: 67 bytes received in 0.00Seconds 67000.00Kbytes/sec.
ftp> get user1.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for user1.txt (6 bytes).
226 Transfer complete.
ftp: 6 bytes received in 0.00Seconds 6000.00Kbytes/sec.
ftp> put download.txt
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
ftp: 6 bytes sent in 0.00Seconds 6000.00Kbytes/sec.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw----- 1 518 511 6 Jul 18 17:26 download.txt
-rw-r--r-- 1 0 0 6 Jul 18 14:41 user1.txt
226 Directory send OK.
ftp: 137 bytes received in 0.00Seconds 137000.00Kbytes/sec.
ftp> delete user1.txt
550 Permission denied.
ftp> quit

```

图 5-20 虚拟用户 user1 的权限测试结果

```

命令提示符 - ftp 172.17.2.1

C:\Documents and Settings\Administrator>ftp 172.17.2.1
Connected to 172.17.2.1.
220 <vsFTPd 2.2.2>
User <172.17.2.1:<none>>: user2
331 Please specify the password.
Password:
230 Login successful.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 6 Jul 18 16:08 user2.txt
226 Directory send OK.
ftp: 67 bytes received in 0.00Seconds 67000.00Kbytes/sec.
ftp> get user2.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for user2.txt <6 bytes>.
226 Transfer complete.
ftp: 6 bytes received in 0.00Seconds 6000.00Kbytes/sec.
ftp> put download.txt
200 PORT command successful. Consider using PASV.
550 Permission denied.
ftp> delete user2.txt
550 Permission denied.
ftp> _

```

图 5-21 虚拟用户 user2 的权限测试结果

## 5.5 小结

本章主要介绍了在 Linux 中与网络配置有关的命令及文件,着重介绍了 Telnet、SSH、VNC、FTP 服务的使用方法。作为服务器,IP 地址的配置是非常重要的,在 Linux 中可以通过命令或图形方式设置 IP 地址,若使用静态 IP,当系统重启后需要激活网卡。Telnet、SSH、VNC 都可以用来完成远程接入服务,均分为服务器端和客户端,其中 Telnet 的用户名及密码对是明文传输的,安全性较差,但 Telnet 工具在 Windows 中是默认提供的,使用方便,适用于对系统安全要求不高的情况; SSH 方式使用密文传输用户名及密码对,安全性较高,但 SSH 不是 Windows 系统自带的工具,需要自己下载安装工具软件; VNC 不同于前两种方法,提供了一种图形化的远程操作方式,更适合现代人的操作习惯,通过 VNC 可以像使用本地系统一样控制远程主机。FTP 服务是 Internet 上的一项非常重要的服务,可通过匿名访问、实体账户访问或虚拟用户账户访问,其中实体账户可以在系统中直接登录,权限不易控制;匿名账户权限过小,操作不灵活;建议使用虚拟账户,既控制了安全风险,又满足了灵活性的要求。

## 5.6 习题

### 1. 选择题

- (1) 提供了主机名与 IP 地址间映射的文件是\_\_\_\_\_。
- |                           |                 |
|---------------------------|-----------------|
| A. /etc/hosts             | B. /etc/network |
| C. /etc/sysconfig/network | D. /etc/host    |

- (2) 设网卡的名称为 eth0，则用于保存网卡信息的文件是\_\_\_\_\_。  
A. /etc/sysconfig/network      B. /etc/network  
C. /etc/sysconfig/network-scripts      D. /etc/resolve.conf
- (3) 为网卡 eth0 设置临时 IP 地址的命令是\_\_\_\_\_。  
A. ipconfig eth0      B. ifconfig eth0  
C. ipconfig      D. ifconfig
- (4) 激活网络卡 eth0 的方法是\_\_\_\_\_。  
A. ifup eth0      B. ifconfig up eth0  
C. ifup ethernet0      D. ifconfigup eth0
- (5) 显示本地主机名的命令是\_\_\_\_\_。  
A. hostname      B. host  
C. name      D. hsname
- (6) 显示网络接口状态信息的命令是\_\_\_\_\_。  
A. netstat - i      B. netstat - nr  
C. netstat - t      D. netstat -n
- (7) Telnet 服务使用的端口号为\_\_\_\_\_。  
A. 21      B. 22  
C. 23      D. 24
- (8) Telnet 的配置文件是\_\_\_\_\_。  
A. /etc/xinetd.d/telnet      B. /etc/telnet  
C. /etc/telnet/telnet      D. /etc/sysconfig/telnet
- (9) 需要在运行级 5 运行 Telnet 服务，可使用\_\_\_\_\_命令。  
A. chkconfig --level =5 telnet on      B. chkconfig -level =5 telnet=on  
C. chkconfig --level 5 telnet on      D. chkconfig
- (10) SSH 服务主要由 3 部分组成，即\_\_\_\_\_。  
A. 传输层协议、用户认证协议、连接协议  
B. 传输层协议、连接协议、UDP 协议  
C. 传输层协议、隧道协议、SSH 协议  
D. 用户认证协议、TCP/IP 协议、Telnet 协议
- (11) 远程桌面服务中，服务器与客户端通信的端口是\_\_\_\_\_。  
A. 5900+编号      B. 6100+编号  
C. 6300+编号      D. 5700+编号
- (12) 创建 VNC 用户密码的命令是\_\_\_\_\_。  
A. passwd username      B. vncpasswd username  
C. vnc passwd username      D. vncpassword username
- (13) FTP 服务会使用\_\_\_\_\_端口进行通信。  
A. 21、20      B. 23、24  
C. 21、23      D. 20、23
- (14) FTP 服务器的主配置文件名是\_\_\_\_\_。

- A. /etc/vsftpd/vsftpd.conf
  - B. /etc/vsftpd.conf
  - C. /etc/ftp/ftpd.conf
  - D. /etc/ftp/ftp.conf
- (15) FTP 服务中的匿名账户是\_\_\_\_\_。
- A. ftp
  - B. root
  - C. administrator
  - D. admin

## 2. 简答题

- (1) 简述 Telnet 的工作过程。
- (2) 简述 SSH 的组成。
- (3) 简述 FTP 服务中虚拟账户的创建方法。