《Linux 操作系统》第七讲：

网络配置

学时计划：4 学时 理论，0 学时 实验

（无实验教学内容）

教学大纲：

1、网络配置文件

2、配置 Linux 加入网络

3、网络测试工具

4、讨论与思考

Linux 是互联网的产物，这是无可争议的。正是得益于互联网通信，世界上各地的程序员才能够共同合作开发了 Linux 操作系统。也是因为此，Linux 对网络的支持以及对各种网络的适应性都很好，毕竟 Linux 是和互联网共同成长的。

网络功能的实现是 Linux 内核最复杂、牵涉最广的一部分。除了经典的以太网协议(例如 TCP、UDP)和相关的 IP 传输机制之外，Linux 还支持许多其他的互连方案（例如 FDDI、ATM)。Linux 也支持大量的网络硬件设备，例如以太网卡和令牌环网络适配器以及 ISDN 卡以及 Modem。

Linux 关于网络的内容非常多，例如网络子系统的 c 语言实现的代码在 Linux 内核中占到了 15MB。本讲不对具体的内容和实现网络的每个细节进行介绍，仅简单的介绍如何管理和配置 Linux 的网络。

一、网络配置文件

在对 Linux 进行网络配置之前，首先介绍 Linux 中网络配置相关的一些主要配置文件。了解这些配置文件如何控制和影响网络，是对自定义 Linux 的网络配置的基础。

Linux 中关于网络配置的文件主要有 7 个，分别是/etc/hosts、/etc/services 、 /etc/hostname 、 /etc/host.conf 、 /etc/nsswitch.conf、/etc/resolv.conf、/etc/network/interfaces。

1.1/etc/hosts

该文件包含（本地网络中）已知主机的一个列表。如果系统的 IP 不是动态获取，就可以使用此文件。对于简单的主机名解析（点分表示法),在请求 DNS 或 NIS 网络名称服务器之前,/etc/hosts.conf 通常会告诉解析程序先查看此文件。

/etc/hosts 的文件格式是 ip 地址 主机名 别名。

```
zhaodongfeng@TeachServer:~$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       ubuntu
211.69.44.22    HactcmServer2-2

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

1.2 /etc/services

Internet 网络服务文件，将网络服务名转换为端口号 / 协议。由 inetd、telnet、tcpdump 和一些其它程序读取。文件中的每一行对应一种服务，它由 4 个字段组成，中间用 TAB 或空格分隔，分别表示“服务名称”、“使用端口”、“协议名称”以及“别名”。

/etc/services 的文件格式为：

服务　　端口/端口　　类型　　别名

```
zhaodongfeng@TeachServer:~$ cat /etc/services |more
tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
…
qotd            17/tcp          quote
ftp             21/tcp
fsp             21/udp          fspd
```

```
ssh             22/tcp                  # SSH Remote Login Protocol
ssh             22/udp
telnet          23/tcp
smtp            25/tcp          mail
…
```

1.3 /etc/hostname

主机名配置文件，该文件只有一行，记录着本机的主机名。

/etc/hostname 的文件格式:主机名。

```
zhaodongfeng@TeachServer:~$ cat /etc/hostname
TeachServer
```

1.4 /etc/host.conf

当系统中同时存在 DNS 域名解析和/etc/hosts 主机表机制时，由该/etc/host.conf 确定主机名解释顺序。

```
zhaodongfeng@TeachServer:~$ cat /etc/host.conf
# The "order" line is only used by old versions of the C library.
order hosts,bind                #名称解释顺序
multi on                        #允许主机拥有多个 IP 地址
nospoof on                      #禁止 IP 地址欺骗
```

order 是关键字，定义先用本机 hosts 主机表进行名称解释，如果不能解释，再搜索 bind 名称服务器(DNS)。

1.5 /etc/nsswitch.conf

名称服务交换设定档，此文件控制了数据库搜寻的工作，包括承认的主机、使用者、群组等。此外，此文件还定义了所要搜寻的数据库。

```
zhaodongfeng@TeachServer:~$ cat /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the `glibc-doc-reference' and `info' packages installed, try:
# `info libc "Name Service Switch"' for information about this file.

passwd:         compat
group:          compat
shadow:         compat
```

```
hosts:          files dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       nis
```

例如：hosts: files dns

指明主机数据库来自两个地方，files（/etc/hosts file）和DNS，并且hosts file优先级高于DNS。

1.6 /etc/resolv.conf

该文件是DNS域名解析的配置文件，它的格式很简单，每行以一个关键字开头，后接配置参数。resolv.conf 的关键字主要有四个，分别是：

```
nameserver    #定义DNS服务器的IP地址
domain        #定义本地域名
search        #定义域名的搜索列表
sortlist      #对返回的域名进行排序
```

例如：

```
zhaodongfeng@TeachServer:~$ cat /etc/resolv.conf
nameserver 211.69.32.58
nameserver 211.69.32.48
```

1.7 /etc/network/interfaces

网络接口参数配置文件。

```
zhaodongfeng@TeachServer:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 211.69.44.22
```

```
netmask 255.255.255.0
network 211.69.44.0
boardcask 211.69.32.255
gateway 211.69.44.1

dns-nameservers 211.69.32.58 211.69.32.48 211.69.33.2
dns-search hactcm.edu.cn
```

eth0 表示第一块网卡，lo 是表示主机的回坏地址。

1.8 不同发型版本的网络配置文件

不同的发型版本，网络配置文件会有不同。关键的网络设备文件如下表所示。

表 7-1 网络配置文件

系统	配置文件	设置内容
RedHat、Fedora	/etc/sysconfig/network Network-scripts/ifcfg-ifname	主机名、默认路由 IP 地址、网络掩码、广播地址
SuSE	/etc/rc.config /etc/route.conf	主机名、IP 地址、网络掩码等 默认路由
Debian、Ubuntu	/etc/hostname /etc/network/interfaces	主机名 IP 地址、网络掩码、默认路由

本讲的网络配置依据 Ubuntu 11.10 Server 的配置文件进行讲解，其他发行版本的配置方法基本相同。

本讲所讲授的网络是指以太网，如果让 Linux 在 FDDI 或者其他类型的网络中通信，请参阅相关内容。

二、配置 Linux 加入网络

如果需要把一台 Linux 主机或者 Linux 服务器加入到以太网中的步骤有四步。

第一步：分配唯一的 IP 地址和主机名。

第二步：设置 Linux 在启动时配置并启用网络接口。

第三步：设置一条默认路由，可以根据需要设置多条静态路由。

第四步：配置 DNS 服务器，使能够使用域名进行通信。

如果使用 DHCP，则只需要为网络接口卡配置为使用 DHCP 即可。

本部分以 Ubuntu 11.10 Server 64bit 接入以太网（Ethernet）为例，介绍配置 Linux 加入网络的过程。

2.1 配置主机名

2.1.1hostname

命令：

```
hostname [-v] [-a, --alias] [-d, --domain] [-f, --fqdn, --long] [-i, --ip-address] [-s, --short] [-y, --yp, --nis] [-F, --filename] [-h, --help] [-V, --version]
```

命令参数：

```
-s, --short    短主机名
-a, --alias    别名
-i, --ip-address    IP 地址
-f, --fqdn, --long 长主机名
-d, --domainDNS 域名
-y, --yp, --nis      NIS/YP 域名
-v 运行时，显示详细的处理过程
-F, --filename      读取指定文件
-h, --help    帮助信息
-V, --version       版本信息
```

命令说明：

（1）hostname 用以显示或设置系统的主机名称。

（2）hostname 命令进行主机名修改后，可以即时生效。但是当重新启动设备时,配置的信息将丢失。如果需要永久性的修改主机名，需要修改/etc/hostname 文件。

2.1.2 编辑/etc/hostname 文件

直接编辑/etc/hostname 文件可以永久性的修改主机名，修改完成后，需要重新启动网络服务才能够生效。

2.2 配置主机名和 IP 地址

把名字映射到 IP 地址的最古老、最简单的方式是/etc/hosts 文件，在 Windows 操作系统中也有此文件。该文件中最少包含的内容为指定 127.0.0.1 和 localhost 的对应关系。

/etc/hosts 文件对于保护一些不希望 DNS 知道的映射关系非常

重要。/etc/hosts 文件在以前的系统中非常重要，因为没有 DNS 的情况下，该文件是进行 IP 地址和主机名对应的唯一手段。现在的 Linux 发行版中此文件除了实现主机本身和环回（loopback）地址的映射关系外，基本上没有其他的作用。

可以通过编辑/etc/hosts 文件实现主机名和 IP 地址的映射。

2.3 配置网络接口：ifconfig

ifconfig 是一个用来查看、配置、启用或禁用网络接口的工具。可以用 ifconfig 来临时性的配置网卡的 IP 地址、掩码、广播地址、网关等，也可以把网络接口信息写入 Linux 的网络配置文件中，系统引导后，会读取网络配置文件，为网卡设置 IP 地址。

命令：

```
ifconfig [网络设备]
        [down up -all multi -arp -promisc]
        [add<地址>][del<地址>]
        [<hw<网络设备类型><硬件地址>]
        [io_addr<I/O 地址>]
        [irq<IRQ 地址>]
        [media<网络媒介类型>]
        [mem_start<内存地址>]
        [metric<数目>]
        [mtu<字节>]
        [netmask<子网掩码>]
        [tunnel<地址>]
        [-broadcast<地址>]
        [-pointopoint<地址>]
        [IP 地址]
```

命令参数：

```
add<地址>       设置网络设备 IPv6 的 IP 地址。
del<地址>       删除网络设备 IPv6 的 IP 地址。
down            关闭指定的网络设备。
<hw<网络设备类型><硬件地址>       设置网络设备的类型与硬件地址。
io_addr<I/O 地址>           设置网络设备的 I/O 地址。
irq<IRQ 地址>               设置网络设备的 IRQ。
media<网络媒介类型>         设置网络设备的媒介类型。
mem_start<内存地址>         设置网络设备在主内存所占用的起始地址。
metric<数目>        指定在计算数据包的转送次数时，所要加上的数目。
mtu<字节>           设置网络设备的 MTU。
netmask<子网掩码>       设置网络设备的子网掩码。
tunnel<地址>            建立 IPv4 与 IPv6 之间的隧道通信地址。
```

up	启动指定的网络设备。
-broadcast<地址>	将要送往指定地址的数据包当成广播数据包来处理。
-pointopoint<地址>	与指定地址网络设备建立直接连线，具有保密功能。
-promisc	关闭或启动指定网络设备的 promiscuous 模式。
IP 地址	指定网络设备的 IP 地址。
网络设备	指定网络设备的名称。

2.3.1 查看网络接口状态

ifconfig 如果不接任何参数，就会输出当前设备的所有网络接口的信息。

```
zhaodongfeng@TeachServer:~$ ifconfig
eth0          Link encap:Ethernet  HWaddr 00:19:e0:2f:38:c1
              inet addr:211.69.44.22  Bcast:0.0.0.0  Mask:255.255.255.0
              inet6 addr: fe80::219:e0ff:fe2f:38c1/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
              RX packets:4645 errors:2 dropped:0 overruns:0 frame:0
              TX packets:1713 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:406080 (406.0 KB)  TX bytes:336092 (336.0 KB)
              Interrupt:18 Base address:0xb400

lo            Link encap:Local Loopback
              inet addr:127.0.0.1  Mask:255.0.0.0
              inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING  MTU:16436  Metric:1
              RX packets:42 errors:0 dropped:0 overruns:0 frame:0
              TX packets:42 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:2332 (2.3 KB)  TX bytes:2332 (2.3 KB)
```

信息含义说明：

eth0 表示第一块网卡，其信息的含义如下。
HWaddr：网卡的物理地址
inet addr：网卡的 IPv4 地址
Bcast：广播地址
Mask:子网掩码
inet6 addr：网卡的 IPv6 地址
Scope：业务状态为主机
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1：
网卡的运行状态
TX 和 RX：网卡收发包的情况，统计的信息包括包的数量，
错误包数量及丢包的数量等
collisions:0：以太网发生冲突的次数

```
txqueuelen:1000：传输缓区长度大小
Interrupt:18：中断次数
Base address:0xb400：地址
```

如果想知道主机所有网络接口的情况，命令如下。

```
zhaodongfeng@TeachServer:~$ ifconfig -a
```

如果想查看某个端口，例如 eth0 的状态，命令如下。

```
zhaodongfeng@TeachServer:~$ ifconfig eth0
```

2.3.2 配置网络接口

ifconfig 可以用来配置网络接口的 IP 地址、掩码、网关、物理地址等，但是使用 ifconfig 为网卡指定 IP 地址只是用来调试网络用的，并不会更改系统关于网卡的配置文件，也就是说使用 ifconfig 对网络接口的配置在系统重新启动后将消失。

如果需要永久性的修改网络接口的 IP 地址，通常有三个方法：一是通过各个发行版本专用的工具来修改 IP 地址；二是直接修改网络接口的配置文件；三是修改特定的文件，加入 ifconfig 指令来指定网卡的 IP 地址，比如在 RedHat 或 Fedora 中，把 ifconfig 的语名写入/etc/rc.d/rc.local 文件中。

命令：

```
ifconfig 网络端口 IP 地址
        hw <HW> MAC 地址
        netmask 掩码地址
        broadcast 广播地址
        [up/down]
```

命令举例：

```
#启用指定的接口
zhaodongfeng@TeachServer:~$ ifconfig eth0 up
#禁用指定的接口
zhaodongfeng@TeachServer:~$ ifconfig eth0 down

#配置指定接口的 IP 地址，注意不能够配置网关。
zhaodongfeng@TeachServer:~$ ifconfig eth0 192.168.25.100
                        broadcast 192.168.25.200 netmask 255.255.255.0
#查看指定接口的状态
zhaodongfeng@TeachServer:~$ ifconfig eth0
```

```
#配置指定接口的 IP 地址、MAC 地址等信息，并启用该接口。
zhaodongfeng@TeachServer:~$ ifconfig eth1 192.168.25.200
                              hw ether 00:11:00:00:11:11
                              netmask 255.255.255.0
                              broadcast 192.168.25.255 up
```

2.3.3 配置虚拟网络接口

有些情况下，需要为同一个网络接口配置多个 IP 地址，这就需要配置虚拟网络接口。例如，需要在一台服务器上配置多个 IP 地址，发布多个网站。

虚拟网络接口指的是为一个网络接口指定多个 IP 地址，虚拟接口的名称是 eth0:0、eth0:1、eth0:2 ... .. eth0:N。

命令举例：

```
#配置虚拟网络接口 IP 地址、MAC 地址等信息。
zhaodongfeng@TeachServer:~$ ifconfig eth1:0 192.168.25.201
                              hw ether 00:11:00:00:11:12
                              netmask 255.255.255.0
                              broadcast 192.168.25.255 up
```

2.3.4 激活和终止网络接口

使用 ifconfig 可以激活和终止网络接口，也可以用专用工具 ifup 和 ifdown 工具。

命令举例：

```
#激活 eth0 接口，使其可用。
zhaodongfeng@TeachServer:~$ ifconfig eth0 up
#终止 eth0 接口。
zhaodongfeng@TeachServer:~$ ifconfig eth0 down
#激活 eth0 接口，使其可用。
zhaodongfeng@TeachServer:~$ ifup eth0
#终止 eth0 接口。
zhaodongfeng@TeachServer:~$ ifdown eth0
```

需要注意的是，对 DHCP 自动分配的 IP，ifconfig 和 ifup、ifdown 命令是不能够激活和终止的。

2.4 永久性配置网络接口

永久性的配置网络接口的信息，需要通过 Linux 发行版本自带的工具进行，或者通过直接编辑网络接口配置文件进行。

使用静态 IP 地址的配置文件如下所示。

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 211.69.44.22
        netmask 255.255.255.0
        network 211.69.44.0
        boardcask 211.69.32.255
        gateway 211.69.44.1

        dns-nameservers 211.69.32.58 211.69.32.48 211.69.33.2
        dns-search hactcm.edu.cn
```

使用 DHCP 方式进行 IP 管理的配置文件如下所示。

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

2.5 配置默认路由：route

在网络接口的配置中，配置了 IP 地址、子网掩码、广播地址，但是却没有配置网关，这是因为，在 Linux 系统中，需要定义一条默认的路由把所有在内核路由表中找不到目的网络的数据包发送到指定的网关。

定义默认路由和定义静态路由使用的命令是 route。关于定义静态路由的内容，请参考相关资料。定义默认路由的命令如下。

```
route add default gw  网关 IP 地址
```

当然，通过这种方式定义的默认路由是临时的，如果需要永久的

定义默认路由，需要修改配置文件来定义。

不同发行版本的 Linux 系统，默认路由存放的文件也不尽相同。具体如表 7-2 所示。

表 7-2 默认路由的配置文件

系统	配置文件	需要修改的变量
RedHat、Fedora	/etc/sysconfig/network	GATEWAY
SuSE	/etc/route.conf	加入新的一行： Default IP 地址 子网掩码 网络接口
Debian、Ubuntu	/etc/network/interfaces	gateway

2.6 配置 DNS

如果需要访问互联网，最后需要进行的就是配置 DNS 服务，从而实现域名和 IP 地址的转换。

在 Linux 系统中配置 DNS，只需要编辑 1 个或者 2 个文件。所有的 Linux 发行版都需要修改/etc/resolv.conf 文件，部分 Linux 发行版本需要修改“服务开关（service switch)”文件。

/etc/resolv.conf 文件列出了要解析不完全的名字(也就是说，不是全名，如 www，而不是 www.51xueweb.cn）应该搜索的 DNS 域和名字服务器的 IP 地址以便查找名字。

/etc/resolv.conf 的文件内容如下所示。

```
zhaodongfeng@TeachServer:~$ cat /etc/resolv.conf
search 51xueweb.cn
nameserver 211.69.32.58
nameserver 211.69.32.48
```

在访问域名的时候，使用不完全的名字，会直接在指定的域内进行搜索。

```
zhaodongfeng@TeachServer:~$ ping i -c 3
PING i.51xueweb.cn (218.28.226.16) 56(84) bytes of data.
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=1 ttl=46 time=40.4 ms
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=2 ttl=46 time=40.3 ms
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=3 ttl=46 time=41.1 ms
```

```
--- i.51xueweb.cn ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 40.387/40.655/41.167/0.397 ms
```

有些Linux发行版本在默认的情况下不使用DNS，甚至在有配置正确的resolv.conf文件存在的情况下也不使用DNS。这类发行版本都有一个“服务开关（service switch)”文件，这个文件能确定用那种机制来解析从主机到IP地址的映射。服务开关文件能够允许指定在进行主机到IP地址的映射时参考DNS、NIS以及/etc/hosts的顺序，也可以指定其中的一种方式。

表 7-3 服务开关文件

系统	开关文件	主机名查找的默认设置
Ubuntu	/etc/nsswitch.conf /etc/host.conf	Files dns mdns Hosts, bind
其他 Linux 发行版本	/etc/nsswitch.conf /etc/host.conf	Files dns Hosts, bind

三、网络测试工具

3.1ping

Ping工具是向网络主机发送ICMP回显请求（ECHO_REQUEST）分组，程序使用 ICMP 协议的强制回显请求数据报以使主机或网关发送一份 ICMP 的回显应答。回显请求数据报(pings IP 及 ICMP 的报头，后跟一个时间值关键字，然后是一段任意长度的填充字节用于把保持分组长度为16的整数倍。

命令：

```
ping [-t] [-a] [-n count] [-l length] [-f] [-i ttl] [-v tos] [-r count] [-s count]
            [[-j computer-list] | [-k computer-list] [-w timeout] destination-list
```

命令参数：

```
-t ： 一直 Ping 指定的计算机，直到从键盘按下 Ctrl+C 中断。
-a ： 将地址解析为计算机 NetBios 名。
-n ： 发送 count 指定的 ECHO 数据包数，通过这个命令可以自己定义发送的个数，对衡量网络速度很有帮助。能够测试发送数据包的返回平均时间，及时
```

间的快慢程度。默认值为 4。
-l ： 发送指定数据量的 ECHO 数据包。默认为 32 字节；最大值是 65500byt。
-f ： 在数据包中发送“不要分段”标志，数据包就不会被路由上的网关分段。通常你所发送的数据包都会通过路由分段再发送给对方，加上此参数以后路由就不会再分段处理。
-i ： 将“生存时间”字段设置为 TTL 指定的值。指定 TTL 值在对方的系统里停留的时间。同时检查网络运转情况的。
-v ： tos 将“服务类型”字段设置为 tos 指定的值。
-r ： 在“记录路由”字段中记录传出和返回数据包的路由。通常情况下，发送的数据包是通过一系列路由才到达目标地址的，通过此参数可以设定，想探测经过路由的个数。限定能跟踪到 9 个路由。
-s ： 指定 count 指定的跃点数的时间戳。与参数-r 差不多，但此参数不记录数据包返回所经过的路由，最多只记录 4 个。
-j ： 利用 computer-list 指定的计算机列表路由数据包。连续计算机可以被中间网关分隔(路由稀疏源) IP 允许的最大数量为 9。
-k ： computer-list 利用 computer-list 指定的计算机列表路由数据包。连续计算机不能被中间网关分隔(路由严格源)IP 允许的最大数量为 9。
-w：timeout 指定超时间隔，单位为毫秒。
destination-list：指定要 ping 的远程计算机。

命令举例：

```
#Ping 本地主机
zhaodongfeng@TeachServer:~$ ping 127.0.0.1 -c 4
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_req=1 ttl=64 time=0.041 ms
64 bytes from 127.0.0.1: icmp_req=2 ttl=64 time=0.033 ms
64 bytes from 127.0.0.1: icmp_req=3 ttl=64 time=0.031 ms
64 bytes from 127.0.0.1: icmp_req=4 ttl=64 time=0.033 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.031/0.034/0.041/0.007 ms
#Ping 网关
zhaodongfeng@TeachServer:~$ ping 211.69.44.1 -c 4 -s 1024 -t 10
PING 211.69.44.1 (211.69.44.1) 1024(1052) bytes of data.
1032 bytes from 211.69.44.1: icmp_req=1 ttl=64 time=2.12 ms
1032 bytes from 211.69.44.1: icmp_req=2 ttl=64 time=2.10 ms
1032 bytes from 211.69.44.1: icmp_req=3 ttl=64 time=2.10 ms
1032 bytes from 211.69.44.1: icmp_req=4 ttl=64 time=2.09 ms

--- 211.69.44.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 2.096/2.106/2.120/0.008 ms
#Ping 域名
zhaodongfeng@TeachServer:~$ ping i.51xueweb.cn -c 4
```

```
PING i.51xueweb.cn (218.28.226.16) 56(84) bytes of data.
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=1 ttl=46 time=43.4 ms
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=2 ttl=46 time=42.3 ms
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=3 ttl=46 time=42.7 ms
64 bytes from pc0.zz.ha.cn (218.28.226.16): icmp_req=4 ttl=46 time=42.6 ms

--- i.51xueweb.cn ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 42.323/42.800/43.499/0.431 ms
```

3.2traceroute

traceroute 是用来发出数据包的主机到目标主机之间所经过的网关的工具。

Traceroute 的原理是试图以最小的 TTL 发出探测包来跟踪数据包到达目标主机所经过的网关，然后监听一个来自网关 ICMP 的应答。发送数据包的大小默认为 38 个字节。

命令：

```
traceroute [-dFlnrvx]
           [-f <存活数值>][-g <网关>...]
           [-i <网络接口>][-m <存活数值>]
           [-p <通信端口>][-s <来源地址>]
           [-t <服务类型>][-w <超时秒数>]
           主机名称或 IP 地址
           [数据包大小]
```

或者：

```
traceroute [ -dFIlnrvx ]
           [ -f first_ttl][-g gateway]
           [-i iface][ -m max_ttl]
           [-p port][-q nqueries]
           [-s src_addr][-t tos]
           [-w waittime][-z pausemsecs]
           Host
           [packetlen]
```

命令参数：

```
-d 使用 Socket 层级的排错功能。
-f<存活数值> 设置第一个检测数据包的存活数值 TTL 的大小。
-F 设置勿离断位。
-g<网关> 设置来源路由网关，最多可设置 8 个。
```

-i<网络界面> 使用指定的网络界面送出数据包。
-I 使用 ICMP 回应取代 UDP 资料信息。
-m<存活数值> 设置检测数据包的最大存活数值 TTL 的大小。
-n 直接使用 IP 地址而非主机名称。
-p<通信端口> 设置 UDP 传输协议的通信端口。
-r 忽略普通的 Routing Table，直接将数据包送到远端主机上。
-s<来源地址> 设置本地主机送出数据包的 IP 地址。
-t<服务类型> 设置检测数据包的 TOS 数值。
-v 详细显示指令的执行过程。
-w<超时秒数> 设置等待远端主机回报的时间。
-x 开启或关闭数据包的正确性检验。

命令举例：

```
#traceroute 的目标为 IP 地址。
zhaodongfeng@TeachServer:~$ traceroute 211.69.31.1
traceroute to 211.69.31.1 (211.69.31.1), 30 hops max, 60 byte packets
 1  211.69.44.1 (211.69.44.1)  8.964 ms * *
 2  222.21.219.89 (222.21.219.89)  11.531 ms  12.777 ms *
 3  210.43.146.1 (210.43.146.1)  13.327 ms  13.665 ms *
 4  210.43.146.37 (210.43.146.37)  15.500 ms *  16.181 ms
 5  210.43.146.26 (210.43.146.26)  18.118 ms  18.812 ms  19.341 ms
 6  210.43.146.150 (210.43.146.150)  26.469 ms  8.903 ms  8.827 ms
 7  * * *
 8  211.69.21.85 (211.69.21.85)  4.395 ms  4.639 ms  4.837 ms
 9  211.69.21.78 (211.69.21.78)  5.034 ms *  5.216 ms
10  211.69.21.54 (211.69.21.54)  6.919 ms  7.325 ms  7.700 ms
11  211.69.31.1 (211.69.31.1)  7.513 ms  7.906 ms  8.180 ms
#traceroute 的目标是域名地址。
zhaodongfeng@TeachServer:~$ traceroute i.51xueweb.cn
traceroute to i.51xueweb.cn (218.28.226.16), 30 hops max, 60 byte packets
 1  211.69.44.1 (211.69.44.1)  1.621 ms  1.677 ms  2.048 ms
 2  222.21.219.89 (222.21.219.89)  2.485 ms  3.337 ms *
 3  * * *
 4  202.112.38.33 (202.112.38.33)  2.045 ms  2.176 ms *
 5  202.112.61.49 (202.112.61.49)  2.080 ms  2.220 ms  2.285 ms
 6  bjcd3.cernet.net (202.112.46.161)  12.412 ms  11.603 ms *
 7  202.112.62.50 (202.112.62.50)  10.996 ms *  11.185 ms
 8  202.38.123.18 (202.38.123.18)  11.473 ms *  11.661 ms
 9  202.38.123.10 (202.38.123.10)  11.857 ms  12.044 ms *
10  202.97.10.233 (202.97.10.233)  33.000 ms * *
11  219.158.5.181 (219.158.5.181)  33.196 ms *  33.315 ms
12  * 219.158.8.222 (219.158.8.222)  40.514 ms *
13  * * *
14  * * *
15  pc22.zz.ha.cn (61.168.251.22)  47.767 ms  47.496 ms  48.260 ms
```

```
16  pc0.zz.ha.cn (218.28.226.34)  44.630 ms  43.315 ms  43.708 ms
17  pc0.zz.ha.cn (218.28.226.16)  42.192 ms  42.176 ms  42.770 ms
#设置最大为 5 跳。
zhaodongfeng@TeachServer:~$ traceroute -m 5 i.51xueweb.cn
traceroute to i.51xueweb.cn (218.28.226.16), 5 hops max, 60 byte packets
 1  211.69.44.1 (211.69.44.1)  1.651 ms  1.706 ms  2.026 ms
 2  222.21.219.89 (222.21.219.89)  2.540 ms  3.359 ms *
 3  * * 210.43.146.1 (210.43.146.1)  2.411 ms
 4  202.112.38.33 (202.112.38.33)  2.025 ms  2.154 ms *
 5  202.112.61.49 (202.112.61.49)  2.058 ms  2.197 ms  2.262 ms
```

3.3netstat

netstat 是一个监控 TCP/IP 网络的非常有用的工具，它可以显示路由表、实际的网络连接以及每一个网络接口设备的状态信息。

命令：

```
netstat [-acCeFghilMnNoprstuvVwx][-A<网络类型>][--ip]
```

命令参数：

```
-a 或--all 显示所有连线中的 Socket。
-A<网络类型>或--<网络类型> 列出该网络类型连线中的相关地址。
-c 或--continuous 持续列出网络状态。
-C 或--cache 显示路由器配置的快取信息。
-e 或--extend 显示网络其他相关信息。
-F 或--fib 显示 FIB。
-g 或--groups 显示多重广播功能群组组员名单。
-h 或--help 在线帮助。
-i 或--interfaces 显示网络界面信息表单。
-l 或--listening 显示监控中的服务器的 Socket。
-M 或--masquerade 显示伪装的网络连线。
-n 或--numeric 直接使用 IP 地址，而不通过域名服务器。
-N 或--netlink 或--symbolic 显示网络硬件外围设备的符号连接名称。
-o 或--timers 显示计时器。
-p 或--programs 显示正在使用 Socket 的程序识别码和程序名称。
-r 或--route 显示 Routing Table。
-s 或--statistice 显示网络工作信息统计表。
-t 或--tcp 显示 TCP 传输协议的连线状况。
-u 或--udp 显示 UDP 传输协议的连线状况。
-v 或--verbose 显示指令执行过程。
-V 或--version 显示版本信息。
-w 或--raw 显示 RAW 传输协议的连线状况。
-x 或--unix 此参数的效果和指定"-A unix"参数相同。
--ip 或--inet 此参数的效果和指定"-A inet"参数相同。
```

命令举例：

```
#按照各种协议分组统计数据。
zhaodongfeng@TeachServer:~$ netstat -s
#显示以太网的统计数据。
zhaodongfeng@TeachServer:~$ netstat -e
#显示路由表。
zhaodongfeng@TeachServer:~$ netstat -r
#显示所有的有效链接信息列表。
zhaodongfeng@TeachServer:~$ netstat -a
#显示所有已经建立的有效链接。
zhaodongfeng@TeachServer:~$ netstat -n
```

3.4tcpdump

tcpdump 将网络中传送的数据包的“头”完全截获下来提供分析。它支持针对网络层、协议、主机、网络或端口的过滤，并提供 and、or、not 等逻辑语句来帮助你去掉无用的信息。

tcmdump 是 Linux 中强大的网络数据采集分析工具，也是一个嗅探器（sniffer）工具。用简单的话来定义 tcpdump，就是：dump the traffic on a network，根据使用者的定义对网络上的数据包进行截获的包分析工具。

命令：

```
tcpdump [-aAbdDefIKlLnNOpqRStuUvxX] [ -B size ] [ -c count ]
                [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
                [ -i interface ] [ -M secret ] [ -r file ]
                [ -s snaplen ] [ -T type ] [ -w file ] [ -W filecount ]
                [ -y datalinktype ] [ -z command ] [ -Z user ]
                [ expression ]
```

命令参数：

```
-a 将网络地址和广播地址转变成名字;
-d 将匹配信息包的代码以人们能够理解的汇编格式给出;
-dd 将匹配信息包的代码以 c 语言程序段的格式给出;
-ddd 将匹配信息包的代码以十进制的形式给出;
-e 在输出行打印出数据链路层的头部信息;
-f 将外部的 Internet 地址以数字的形式打印出来;
-l 使标准输出变为缓冲行形式;
-n 不把网络地址转换成名字;
-t 在输出的每一行不打印时间戳;
```

-v 输出一个稍微详细的信息，例如在 ip 包中可以包括 ttl 和服务类型的信息；
-vv 输出详细的报文信息；
-c 在收到指定的包的数目后，tcpdump 就会停止；
-F 从指定的文件中读取表达式,忽略其它的表达式；
-i 指定监听的网络接口；
-r 从指定的文件中读取包(这些包一般通过-w 选项产生)；
-w 直接将包写入文件中，并不分析和打印出来；
-T 将监听到的包直接解释为指定的类型的报文，常见的类型有 rpc（远程过程调用）和 snmp（简单网络管理协议；）

命令举例：

```
#显示 eth0 接口的通信。
zhaodongfeng@TeachServer:~$ sudo tcpdump -i eth0
#显示 eth0 接口上和指定 IP 地址进行的通信。
zhaodongfeng@TeachServer:~$ sudo tcpdump -i eth0 host 211.69.32.33
#显示 eth0 接口上在指定端口上进行的通信。
zhaodongfeng@TeachServer:~$ sudo tcpdump -i eth0 port 3306
```

四、讨论与思考

4.1 静态路由

（1）如何修改 Linux 系统的静态路由表？

（2）如果一台服务器上安装了两个网卡，这两个网卡分别配置中国联通的 IP 地址和中国电信的 IP 地址，且两条链路同时联通，应该如何配置静态路由以实现两条链路的网络访问？

4.2Linux 系统对其他网络的支持

（1）Linux 系统如何使用 ADSL 链接网络？

（2）Linux 系统是否可以使用 VPN?

（3）Linux 系统能够支持以太网以外的其他结构的网络么？

（4）IPv6 的支持如何？