

(2019秋季, 网络安全, 编号: CS05154)



# 第2章 基础知识

中国科学技术大学

曾凡平 billzeng@ustc.edu.cn



# 主要内容

- 2.1 常用的Windows命令
- 2.2 常用的Linux命令
- 2.3 批命令及脚本文件
- 2.4 网络端口、服务、进程
- 2.5 网络编程技术基础知识
- 2.6 网络安全实验环境的配置



## 2.1 常用的Windows命令

### 演示环境：Windows2003

- 基本的 **DOS(Disk Operating System)** 命令是在 Windows 系统下运行的一些DOS命令，这些命令又都是从cmd.exe开始。
- 单击“开始”——“运行”命令、在弹出的窗口输入cmd后回车就可以打开cmd了。很多入侵工作都是在这个环境中进行的。
- **cmd.exe**是Windows的控制台程序。
- 可以据个人的偏好配置cmd.exe的界面（**演示**）



# DOS的运行和Path环境的修改

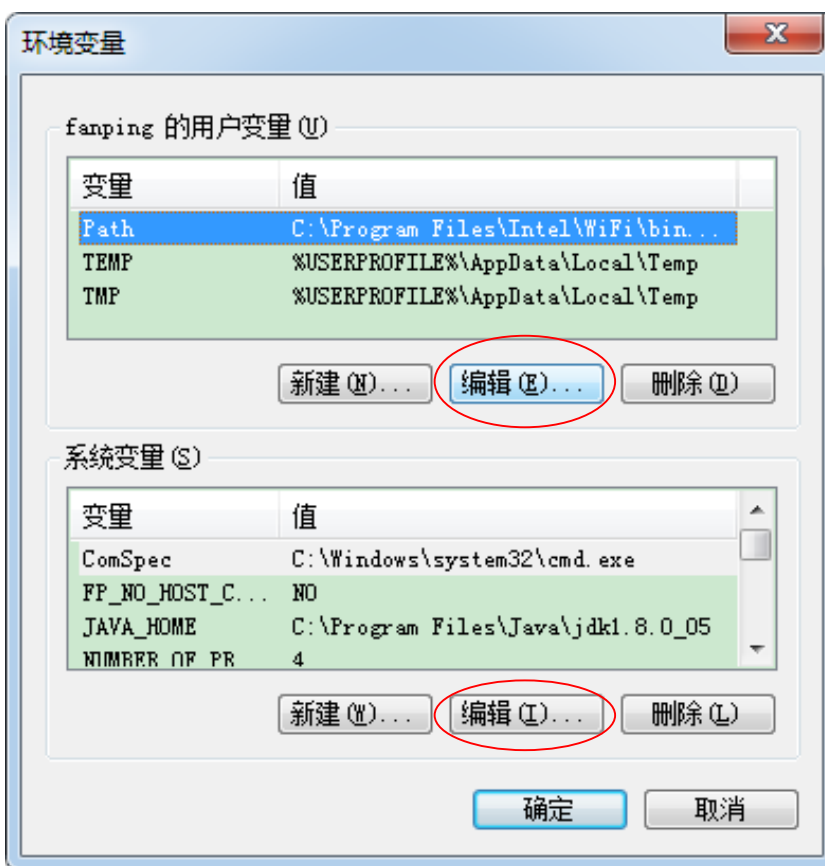
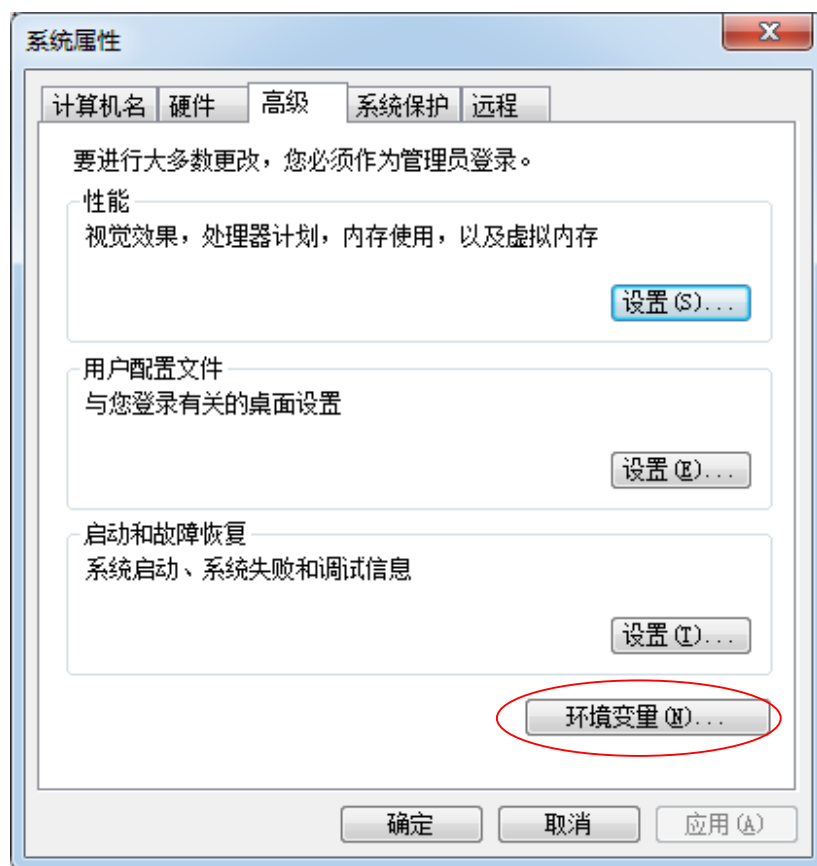
- Windows 下的部分命令程序已经通过系统中的Path环境变量注册过默认的执行路径，可以直接在cmd下执行。例如，telnet、ftp、dir、cd等。但是其他没有通过Path环境变量注册过的命令必须要切换到程序所在目录才能运行。
- 如果某些命令行工具需要经常使用，可以把它直接复制到这些目录下面，那么就不切换路径直接使用了，或者把它的**工具包**所在的目录添加到Path环境变量中。

# 举例-Path环境修改 (Windows7)

- 控制面板\系统和安全\系统 ➔ 高级系统设置



# 修改用户或系统的环境变量





# (1) *net*命令

- *net*命令是很多网络命令的集合，通过*net help*或者*net /?*可以看到这些命令的用法
  - 启动关闭服务：分别用*net start servicename*和*net stop servicename*。
    - (演示) *net start sharedAccess*
    - *net stop sharedAccess*
  - 启动关闭共享：*net share sharename* 和 *net share sharename /del*
    - (演示) *net share c=c:\* 可完全共享C盘，使用*net share*可以查看开放了什么共享。



- 映射磁盘和删除映射磁盘：
  - `net use drivename \\ip\drive /user:username`
  - `net use drivename /del`
- 添加删除用户、将用户加入到组：
  - `net user username password /add` 或 `/del`。
  - `net localgroup administrators username /add` 或 `/del`
- 激活和关闭guest账号
  - `net user guest /active:yes`
  - `net user guest /active:no`



## (2) 远程登录命令telnet

- telnet是一种从客户端登录服务器的方式。比如说在肉鸡（**被入侵的机器**）上留下了一个telnet扩展型后门，都需要使用telnet连接到的指定端口进行连接控制，telnet的使用方式为：

telnet **IP** [Port]

- 比如telnet 192.168.11.1 1234连接到192.168.11.1的1234端口。telnet的默认端口为23，不使用Port参数的时候将默认连接到192.168.11.1的23端口。
- telnet 192.168.11.203 Port （演示）



### (3) 文件传输命令`ftp`

- **`ftp`** 是一种文件传输命令，它可以方便地实现在两台机器间进行文件传输功能。它将文件传输到运行FTP（文件传输协议）服务的计算机或从该计算机上下载文件，可以通过以ASCII文本文件交互地或以批处理模式使用**`ftp`**。其用法如下：
  - FTP [-v] [-d] [-i] [-n] [-g] [-s:filename] [-a] [-w:window size] [-A] [host]
  - -v: 禁止显示FTP服务器响应。
  - -d: 启用调试、显示在FTP客户端和FTP服务器之间传递的所有命令。
  - -i: 传送多个文件时禁用交互提示。



- **-n**: 在建立初始连接后禁止自动登录功能。
- **-g**: 禁用文件名组合。
- **-s: filename**.指定包含FTP命令的文本文件。这些命令在启动FTP后自动运行。该参数不允许带有空格。使用该参数而不是重定向。
- **-a**: 指定绑定FTP数据连接时可以使用任何本地接口。
- **-w: window size**。指定传输缓冲的大小。默认窗口大小为4096字节。
- **-A**: 匿名登录到FTP服务器。
- 该命令最基本用法为“ftp IP”，在输入用户名和密码之后可以使用get或者put来进行下载或上传操作，使用disconnect断开连接，bye或者quit退出FTP。
- 如果在入侵时得到FTP密码，对命令行不太熟悉，可以使用FlashFTP， CuteFTP等图形界面的FTP工具来传输文件。



## (4) 添加计划任务命令at

- 使用at命令可以安排在特定日期和时间运行指定程序，at命令的用法为：
  - at [\\computername] [[id] [/DELETE] | [/DELETE [/YES]]
  - at [\\computername] time [ /INTERACTIVE) [/EVERY : date[, ...] | [/NEXT: date[, ...]] “command”
- 一般在入侵的时候使用该命令指定远程主机在某时间运行的指定程序，比如说将一个木马服务端传到目标主机上，可以使用at命令让它在指定的时间运行。
  - 例如：at \\192.168.11.203 13:42 server.exe。
  - 必须注意的是，主机必须运行Task Scheduler服务，同时当前用户必须是Administrators组的成员。



## (5) 查看修改文件夹权限命令cacs

- `cacs filename [/T][/E][/C] [/G user : perm] [/R user [...]] [/P user : perm [...]] [/D user [...]]`。
- 其中：
  - `filename`: 显示ACL。
  - `/T`: 更改当前目录及其所有子目录中指定文件的ACL。
  - `/E`: 编辑ACL而不替换。
  - `/C`: 在出现拒绝访问错误时继续。
  - `/G user : perm` 赋予指定用户访问权限。
    - `perm`可以是: R读取; W写入; C更改; F完全控制。
  - `/R user`: 撤销指定用户的访问权限。

- /P user : perm 替换指定用户的访问权限。
  - perm可以是：N-无；R-读取；W-写入；C-更改（写入）；F-完全控制。
- /D user 拒绝指定用户的访问。
- 将C:\test.bmp的文件访问权限更改为netkey完全控制，则可以使用如下命令  
cacs C:\test.bmp /G netkey:f
- **入侵成功后**，当被入侵主机对某些文件加上了访问权限，如果此时有足够的权限使用cacs，那么可以利用该命令修改权限，然后查看这些文件。



## (6) 回显命令echo

- 使用echo命令可以在屏幕上显示指定的信息，利用echo和>>符号可以把命令结果导出到某文件中。
  - `echo hacked by netkey > index.html`
    - // 用hacked by netkey覆盖 index.html的内容
  - `echo hacked by netkey >> index.html`
    - //在 index.html的尾部添加hacked by netkey。

### (演示)

- 在上面的命令中，如果文件 `index.html` 不存在，将会自行创建该文件。值得注意的是，在需要写入文件的内容中如果包含>、<、等特殊符号时，需要在前面加上转意字符^，例如：`echo 2 ^>1 >index.html`。

## (7) 命令行下的注册表操作

- Windows系统的所有配置信息都存储在注册表中，通过修改注册表中的相应键值就可以控制程序的启动方式和服务启动类型，因此系统安全与注册表息息相关。入侵成功以后，可以通过修改注册表以实现病毒与木马的自动运行或以服务的方式随系统开机启动。
- 命令行下的注册表工具为reg.exe，该工具的用法为：
- REG Operation [参数列表]

# 注册表操作

- 比如
- `reg export HKEY_LOCAL_MACHINE\Software\Microsoft microsoft.reg`
- 就 是 将 注 册 表 中  
HKEY\_LOCAL\_MACHINE\Software\Microsoft 的  
项值导出到文件microsoft.reg。

**(演示)**



## (8) 查看当前系统用户情况命令query

- query的用法(Windows 2003)如下:
  - QUERY { PROCESS | SESSION | TERMSERVER | USER }
- (演示)
- 使用query user可以来查看当前系统的会话，比如说查看是否有人使用远程终端登录服务器；通过query可以查到某用户的session然后通过logoff命令将他踢出去。
- 注：Windows XP|7|8 不支持该命令



## (9) 终止会话命令 *logoff*

- `logoff [sessionname | sessionid] [server:servername] [/V]`
- 其中的 `sessionname` 或 `sessionid` 选项可以通过 `query` 命令查到，在入侵的时候通常遇到需要把肉鸡的管理员或者其他入侵者踢出去，这时就可以使用 `logoff` 命令。

(演示)



## (10) 物理网络查看命令ping

- 命令ping验证与远程计算机的连接
  - 有时候根据返回的TTL值可以判断出受侵者的操作系统类型，Windows主机的TTL值一般在128左右，\*nix的一般在250左右。
  - 不过一般的主机都屏蔽了，ping无法返回TTL值；其次这个TTL值可以人为修改，根据这个判断操作系统类型并不可靠。

(演示)



## (11) 网络配置查看命令ipconfig

- 使用ipconfig /all命令可以方便地查看网卡的MAC地址、主机的网络设置等，在向内网渗透的过程中，需要了解受侵者机器网络的网络配置，可以使用ipconfig来查看。
- ipconfig /renew 重新获得网络地址。

(演示)



## (12) 查看通信路由命令tracert

- 该诊断实用程序将包含不同生存时间(TTL)值的Internet控制消息协议(ICMP)回显数据包发送到目标，以决定到达目标采用的路由。
- 在转发数据包上的TTL之前递减1，就是必需经过的路由器数，所以TTL是有效的跃点计数。数据包上的TTL到达0时，路由器应该将“ICMP已超时”的消息发送回源系统。



## (13) DNS 查看 nslookup

- 使用 *nslookup* 可以查看主机的 DNS 服务器，*nslookup* 最简单的用法就是查询域名对应的IP地址。
- 其用法是：  
    nslookup 域名 例如：nslookup www.163.com  
        (演示)

## (14) *netstat* 命令

- 显示协议统计和当前 TCP/IP 网络连接。用法为：
- `NETSTAT [-a] [-b] [-e] [-f] [-n] [-o] [-p proto] [-r] [-s] [-t] [interval]`
- 常用选项：
  - `-a` 显示所有连接和侦听端口。
  - `-n` 以数字形式显示地址和端口号。
  - `-r` 显示路由表。
  - `-s` 显示每个协议的统计。

**演示：** `netstat -a`



## (15) *route* 命令

- 操作网络路由表。用法为：
  - `ROUTE [-f] [-p] [-4|-6] command [destination]`  
`[MASK netmask] [gateway] [METRIC metric] [IF interface]`
  - 演示： `route -4 PRINT` 显示当前IPv4的路由表
- 用不带参数的route命令将显示其帮助
  - 演示： `route`

## (16) *tftp*

- 将文件传输到正在运行tftp服务的远程计算机，或从正在运行tftp服务的远程计算机传输文件。其用法如下：
  - `tftp [-4][-6][-v][-l][-m mode] [host [port]] [-c command]`
- 启用tftp后，输入？（或help）可以获得帮助。
- 范例：
  - 先在192.168.11.1用tftpd32建立一个TFTP服务器，然后在192.168.11.2上可以使用`tftp -i 192.168.11.1 get server.exe`就可以把192.168.11.1上和tftpd32同目录下的server.exe下载下来。

## 2.2 常用的Linux命令

- Linux虽然是免费的，但它的确是一个非常优秀的操作系统，与MS-WINDOWS相比具有可靠、稳定、速度快等优点。Linux的维护与管理基本上在命令行界面下进行，最常用的命令行界面是GNOME Terminal。

```
fanping@u14x32: ~/work
welcome to workspace
fanping@u14x32:~/work$ ll
total 24
drwxrwxr-x  4 fanping fanping 4096  9月  8 11:34 ./
drwxr-xr-x 18 fanping fanping 4096  9月 10 10:55 ../
drwxr-xr-x  9 fanping fanping 4096  8月 29 10:40 islab/
-rw-rw-r--  1 fanping fanping  326  9月  8 11:34 mywork.log
-rw-rw-r--  1 fanping fanping  326  9月  8 11:34 mywork.log~
drwxrwxr-x  2 fanping fanping 4096  9月  8 11:29 ns/
fanping@u14x32:~/work$
```

- (1) ls命令：显示指定工作目录下之内容
- (2) mkdir命令：建立子目录
- (3) chown命令：将档案的拥有者加以改变
- (4) chmod命令：改变档案的访问控制模式
- (5) 远程登录命令telnet
  - 其用法同Windows系统下的telnet命令。
- (6) 回显命令echo
  - 其用法同Windows系统下的echo命令。
- (7) 物理网络查看命令ping
  - 其用法同Windows系统下的ping命令。

## (8) 查看通信路由命令tracert

- 其用法同Windows系统下的tracert命令。

## (9) 网络配置查看命令ifconfig

- 其用法同Windows系统下的ipconfig命令。

## (10) netstat命令

- 其用法同Windows系统下的netstat命令。

## (11) grep命令

- 功能说明：查找文件里符合条件的字符串。

(12) ps命令：显示进程(process) 的状态

- ps -A | grep gedit

(13) export命令：设置或显示环境变量。

- 语法：export [变量名称]=[变量设置值]
- 范例：export mydir=/home/fanping/work

(14) lsmod(list modules)命令

- 功能说明：显示已载入系统的模块。

(15) insmod(install module)命令

- 功能说明：载入内核模块。
- rmmod：卸载内核模块

(16) gzip和tar命令

- 功能说明：压缩文件。



## 2.3 批命令及脚本文件

### 2.3.1 批处理文件

- Windows 系统的批处理文件是扩展名为 .bat 或 .cmd 的文本文件，包含一条或多条命令，由 DOS 或 Windows 系统内嵌的命令解释器来解释运行。批处理用于自动地连续执行多条命令，文件的内容就是待执行的命令。在命令提示符下输入批处理文件的名称，或者在资源管理器中双击该批处理文件，系统就会调用 cmd.exe 并按序执行其中的命令。Linux 系统的批命令为 shell 脚本文件。
- 使用批处理文件可以简化日常或重复性的管理任务。



# (1)常用批处理命令

- **echo**: 表示显示此命令后的字符。
- **echo off**: 表示在此语句后所有运行的命令都不显示命令行本身。
- **@**: 与**echo off**类似，但它是加在每个命令行的最前面，表示运行时不显示这一行的命令行（只能影响当前行）。
- **call**: 调用另一个批处理文件（注意：如果不用**call**而直接调用别的批处理文件，那么执行完那个批处理文件后将无法返回当前文件并执行当前文件的后续命令）。
- **pause**: 运行此句会暂停批处理的执行并在屏幕上显示 **Press any key to continue** 的提示，等待用户按任意键后继续。
- **rem**: 表示此命令后的字符为解释行(注释)，不执行，只是给自己今后参考用的（相当于程序中的注释）。

## (2) 批处理文件的参数

- 批处理文件还可以像C语言的函数一样使用参数（相当于DOS命令的命令行参数），这需要用到一个参数表示符“%”。%[1-9]表示参数，参数是指在运行批处理文件时在文件名后加的以空格（或者Tab）分隔的字符串。变量可以从%0~%9，%0表示批处理文件本身，其他参数字符串用%1~%9顺序表示。
- 例：C:根目录下一批处理文件名为t.bat，内容为：  
    @echo off  
    type %1  
    type %2
- 那么运行：C:\>t a.txt b.txt                      将顺序地显示 a.txt 和 b.txt文件的内容



### (3) 特殊命令

- if, goto, choice, for是批处理文件中比较高级的命令

1) if是条件语句，用来判断是否符合条件，从而决定执行不同的命令。它有3种格式。

- if [not] “参数” = “字符串” 待执行的命令

如：if "%1"=="a" format a:

- if [not] exist [路径\]文件名 待执行的命令

如：if exist c:\config.sys echo "exist c:\config.sys"

- if errorlevel <数字> 待执行的命令

### (3) 特殊命令(续)



2) goto将运行批处理文件跳到goto所指定的标号，一般与if配合使用。

```
goto end
```

```
:end
```

```
echo This is the end
```

3) for循环命令，只要条件符合，它将多次执行同一命令。

```
for %variable in (set) do command [command parameters]
```

- 例: for /R %c in (\*.bat \*.txt) do type %c

该命令行会显示当前目录下所有以bat和txt为扩展名的文件的内容。



## 2.3.2 VBScript脚本文件

- VBScript即Microsoft Visual Basic Script Edition（微软公司可视化BASIC脚本版）。VBS（VBScript的进一步简写）是基于Visual Basic的脚本语言。VBS脚本不编译成二进制的可执行文件，直接由宿主(host)解释源代码并执行，即程序不需要编译成EXE，而是直接给用户发送.vbs的源程序，用户就能执行了。
- VBS脚本文件可以用任何文本编辑器编辑，并以扩展名.vbs保存。VBS文件可以通过Cscript和Wscript来解析执行，在命令行下用Cscript来解析，在图形模式下用Wscript解析运行。

- 将Basic程序保存在扩展名为vbs的文本文件中，双击该文件就可以执行该程序。
- 将以下代码保存在hello.vbs中：  
    name=Inputbox("请输入你的姓名:")  
    Msgbox(name)
- 其运行结果如下：



## 2.4 网络端口、服务、进程

### 2.4.1 网络端口

- 物理意义上的端口（比如，ADSL Modem、集线器、交换机、路由器用于连接其他网络设备的接口等）；
- 逻辑意义上的端口，一般是指TCP/IP协议中的端口，即协议(网络)端口。端口号的范围从0~65535（比如用于浏览网页服务的80端口，用于FTP服务的21端口）等。
- 网络端口指的是网络中面向连接服务和无连接服务的通信协议端口。它是一种抽象的软件结构，包括一些数据结构和 I/O (输入输出缓冲区)。它是一个软件结构，被客户程序或服务进程用来发送和接收信息。一个端口对应一个16比特(2字节)的整数。



1. 端口的作用：与进程关联的一种数据结构
2. 端口的分类：知名端口、动态端口；协议端口
3. 端口在入侵中的作用：入侵的门窗
4. 端口的相关工具：netstat和nmap
5. 端口的保护：查看、判断、关闭

# 让内网的主机暴露到外网的方法

- 外部主机只可以访问Internet地址，无法访问局域网内的IP地址，因此无法访问局域网中的服务器。解决这个问题就是采用端口映射，在网关上将内网的地址和端口号映射到Internet地址。Linux系统的Netfilter框架及路由器等(**无线路由器的“转发规则”**、**“DMZ主机”**)均实现了端口映射功能。

(演示)

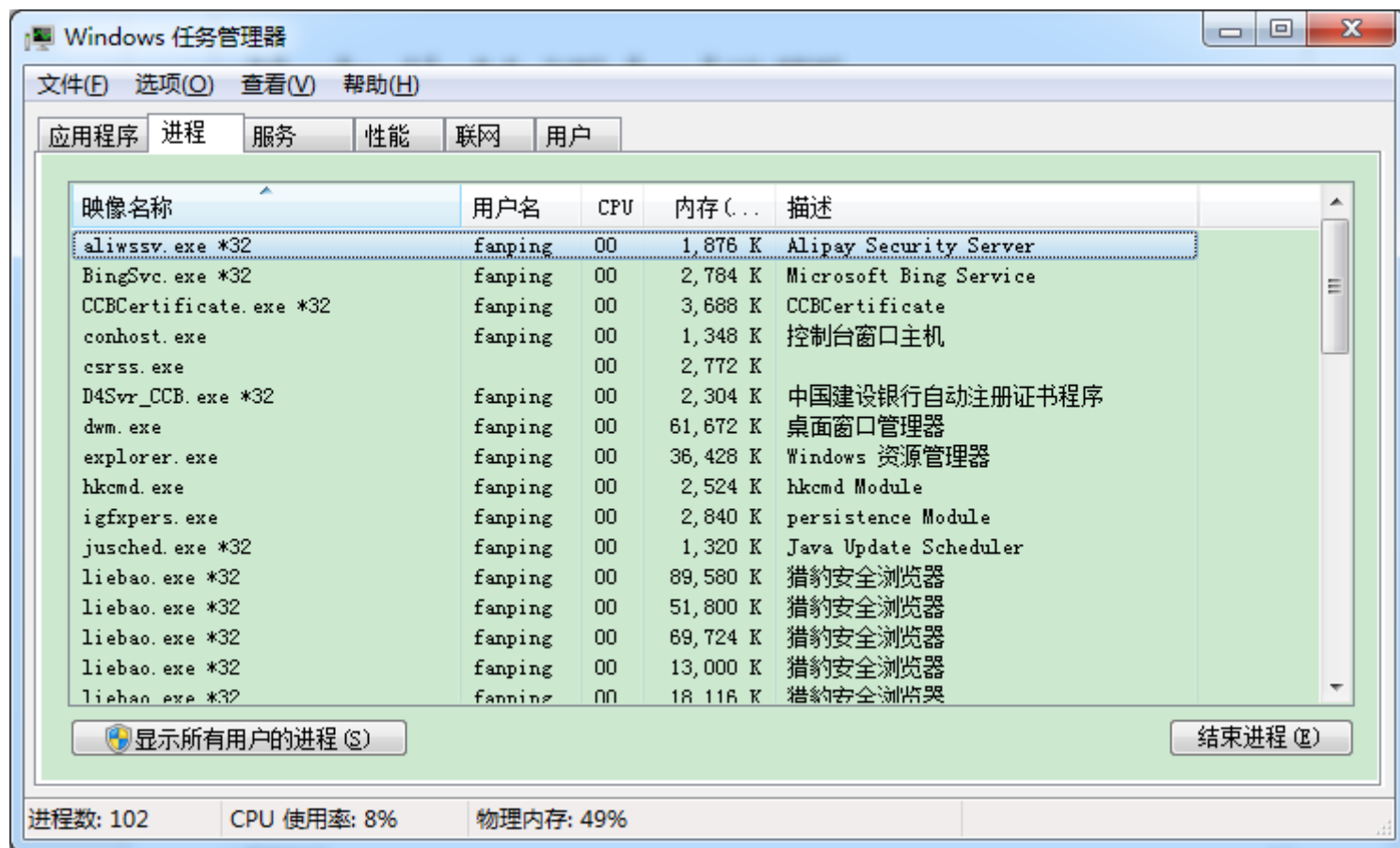


## 2.4.2 服务与进程

- 进程是指在系统中正在运行的一个应用程序。线程是系统分配处理器时间资源的基本单元，或者说进程之内独立执行的一个单元。对于操作系统而言，其调度单元是线程。一个进程至少包括一个线程，通常将该线程称为主线程。一个进程从主线程的执开始进而创建一个或多个附加线程，就是所谓基于多线程的多任务。
- 从操作系统角度来看，进程分为系统进程和用户进程两类。
  - 系统进程执行操作系统程序，完成操作系统的某些功能。用户进程运行用户程序，直接为用户服务。
  - 系统进程的优先级通常高于一般用户进程的优先级。进程与程序之间既有联系又有区别，程序是构成进程的组成部分之一。

- **系统服务(system services)**是执行指定系统功能的程序、例程或进程，以便支持其他程序，尤其是低层(接近硬件)程序。服务一般在后台运行，如Web服务器、数据库服务器以及其他基于服务器的应用程序。
- 与用户运行的其它程序相比，服务不会出现程序窗口或对话框，只有在任务管理器中才能观察到它们的身影。

# Windows的任务管理器



# 进程与程序

- 一个进程的运行目标是执行它所对应的程序，如果没有程序，进程就失去了其存在的实际意义。但是**程序是静态的，而进程是动态的**。
- 进程是有生命周期的，有诞生，也有死亡。
- 一个进程可以执行一个或几个程序，一个程序也可以构成多个进程，进程还具备创建其他进程的功能。

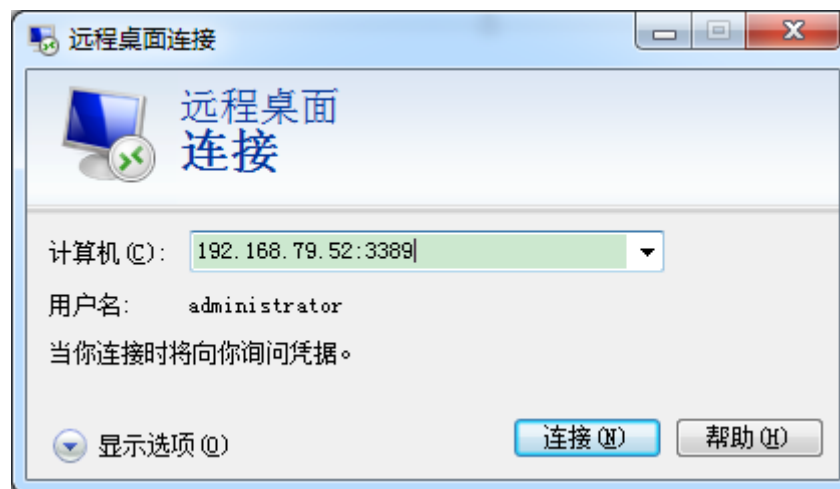


## 2.4.3 Windows终端服务

- 1)配置如何启动服务
- 2)安装终端服务
- 3)如何连接远程主机: mstsc
- 4)修改终端服务端口

- ① HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\TerminalServer\Wds\Repwd\Tds\Tcp”
- ② “HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\TerminalServer\WinStations\RDP-TCP
  - 子键, 修改 “PortNumber”为期望的端口号(如修改成8080端口)。

# win终端服务 (演示)



# 2.5 网络编程技术基础知识



## 2.5.1 套接字socket

- socket接口是TCP/IP网络的API接口函数，最先应用于Unix操作系统，目前已成为网络程序设计的标准接口。

- socket函数原型为：

int socket(int domain, int type, int protocol)

domain

AF\_INET

type

SOCK\_STREAM,  
SOCK\_PACKET

SOCK\_DGRAM,

SOCK\_RAW,

protocol: 一般为“0”

# 面向传输层的Socket编程

- 面向传输层的常用的Socket类型有两种：流式Socket（SOCK\_STREAM）和数据报式Socket（SOCK\_DGRAM）。流式Socket是一种面向连接的Socket，针对于面向连接的TCP服务应用；数据报式Socket是一种无连接的Socket，对应于无连接的UDP服务应用。
- **(1) Socket配置**
  - 通过socket调用返回一个socket描述符后，在使用socket进行网络传输以前，必须配置该socket。面向连接的socket客户端通过调用Connect函数在socket数据结构中保存本地和远端信息。无连接socket的客户端和服务端以及面向连接socket的服务端通过调用bind函数来配置本地信息。



- Bind函数原型为:

```
int bind(int sockfd, struct sockaddr *my_addr, int addrlen);
```

struct sockaddr结构类型是用来保存socket信息的:

```
struct sockaddr {  
    unsigned short sa_family; /* 地址族, AF_XXX */  
    char sa_data[14];        /* 14 字节的协议地址 */  
};
```

sa\_family 一般为 AF\_INET, 代表 Internet (TCP/IP) 地址族;  
sa\_data 则包含该socket的IP地址和端口号。

另外还有一种结构类型:

```
struct sockaddr_in {  
    short int sin_family;    /* 地址族 */  
    unsigned short int sin_port; /* 端口号 */  
    struct in_addr sin_addr; /* IP地址 */  
    unsigned char sin_zero[8]; /* 填充0 以保持与struct sockaddr同样大小 */  
};
```

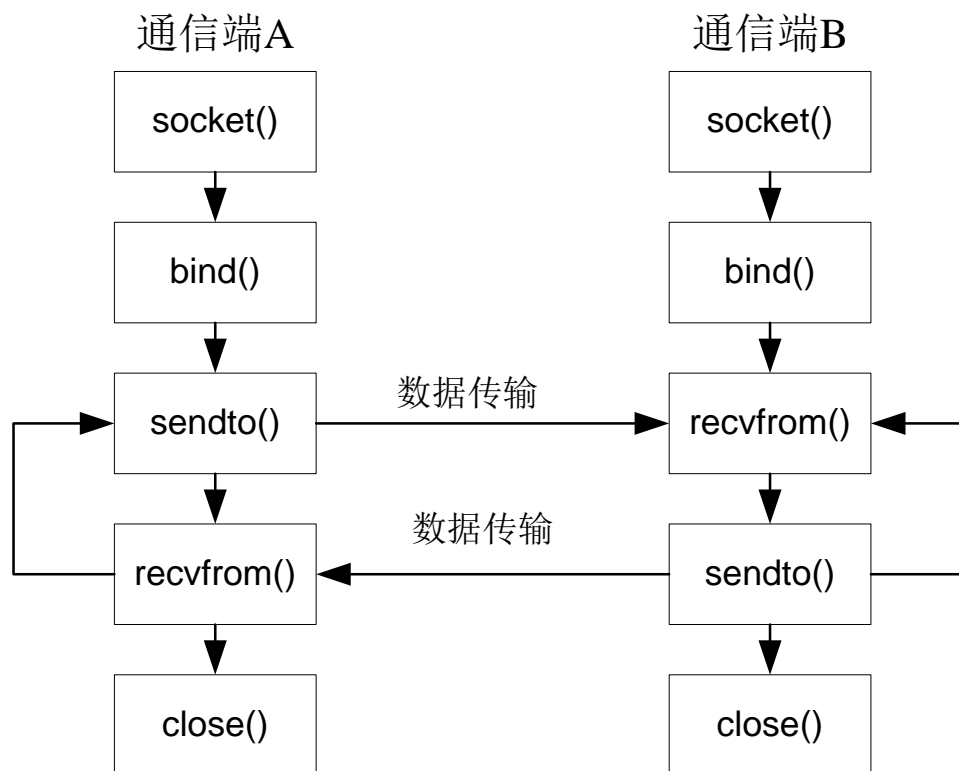
这个结构更方便使用



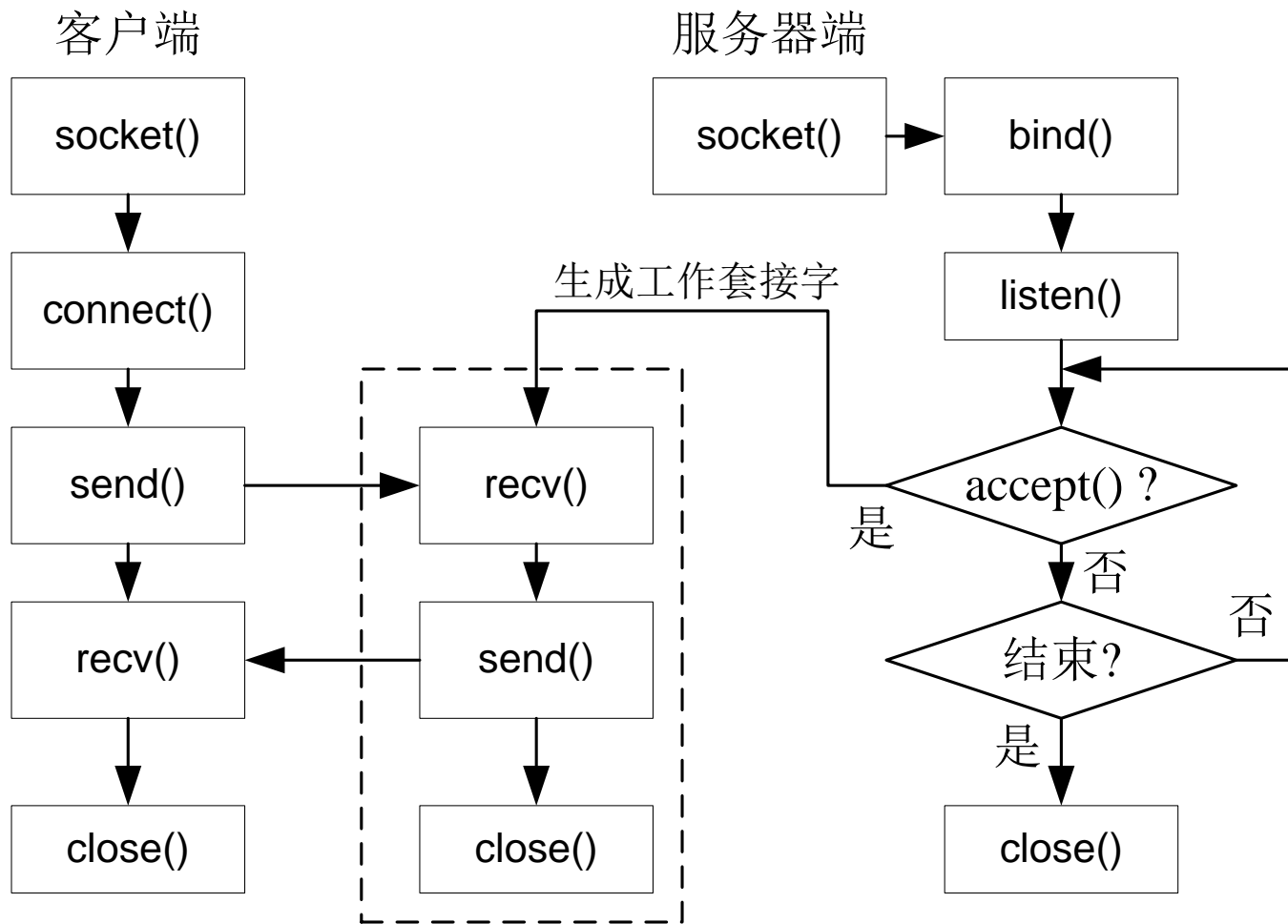
# 主机字节顺序转换成网络字节顺序

- 注意在使用bind函数是需要将sin\_port和sin\_addr转换成网络字节优先顺序。
- 计算机数据存储有两种字节优先顺序：高位字节优先和低位字节优先。Internet上数据以高位字节优先顺序在网络上传输，所以对于在内部是以低位字节优先方式存储数据的机器，在Internet上传输数据时就需要进行转换，否则就会出现数据不一致。
- 下面是几个字节顺序转换函数：
  - htonl(): 把32位值从主机字节序转换成网络字节序
  - htons(): 把16位值从主机字节序转换成网络字节序
  - ntohl(): 把32位值从网络字节序转换成主机字节序
  - ntohs(): 把16位值从网络字节序转换成主机字节序

## (2)无连接的UDP服务应用



### (3) 面向连接的TCP服务应用





# 面向网络层的Socket编程

- 也称为原始套接字(SOCK\_RAW)。应用原始套接字，可以编写出由TCP和UDP套接字不能够实现的功能。原始套接字只能由有root权限的人创建，并且必须自己构造数据包。

- 原始套接字的创建

```
int sockfd=socket(AF_INET, SOCK_RAW, protocol)
```

```
protocol:
```

```
IPPROTO_ICMP、IPPROTO_TCP、IPPROTO_UDP
```

- 几个关键点

```
sockfd=socket(AF_INET,SOCK_RAW,IPPROTO_TCP);
```

```
setsockopt(sockfd,IPPROTO_IP,IP_HDRINCL,&on,sizeof(on));
```

```
setuid(getpid());
```

用sendto和recvfrom函数发送和接收数据



## 2.5.2 网络编程库

- 由于网络安全应用软件通常需要从底层对网络通信链路进行操作，因此需要对网络通信的细节(如连接双方地址/端口、服务类型、传输控制等)进行检查、处理或控制。
- 数据包截获、数据包头分析、数据包重写、中断socket连接等功能几乎在每个网络安全程序中都必须实现，因而采用传统的socket编程技术开发网络安全应用软件就显得非常的烦琐，而且所开发的程序代码维护困难，跨平台移植性较差。

- 为了解决直接用socket技术进行网络安全应用软件开发所存在的弊端，就有必要对常用的socket函数进行封装，在多种平台间提供统一的用户接口界面，使网络应用程序的开发变得简单易行。Linux下的Libnet库、Libpcap库和Windows下的Winpcap(<http://www.winpcap.org/>)库等网络编程库就是为此目的而引入的。
- 利用网络编程库可以很容易编写网络程序，尤其是IP层和数据链路层的网络程序。网络编程库是开放源代码的，也提供了非常详细的开发文档和示例程序，极大地简化了网络底层应用程序的开发。



## 2.5.3 用Windows Sockets编程

- 在Windows环境下进行程序设计，最省事的方法是用MFC的类库，其中的CSocket类封装了TCP协议的大部分功能，并且可以结合Windows的消息映射机制进行异步通讯。
- CSocket类及消息映射  
请参考Windows网络编程技术

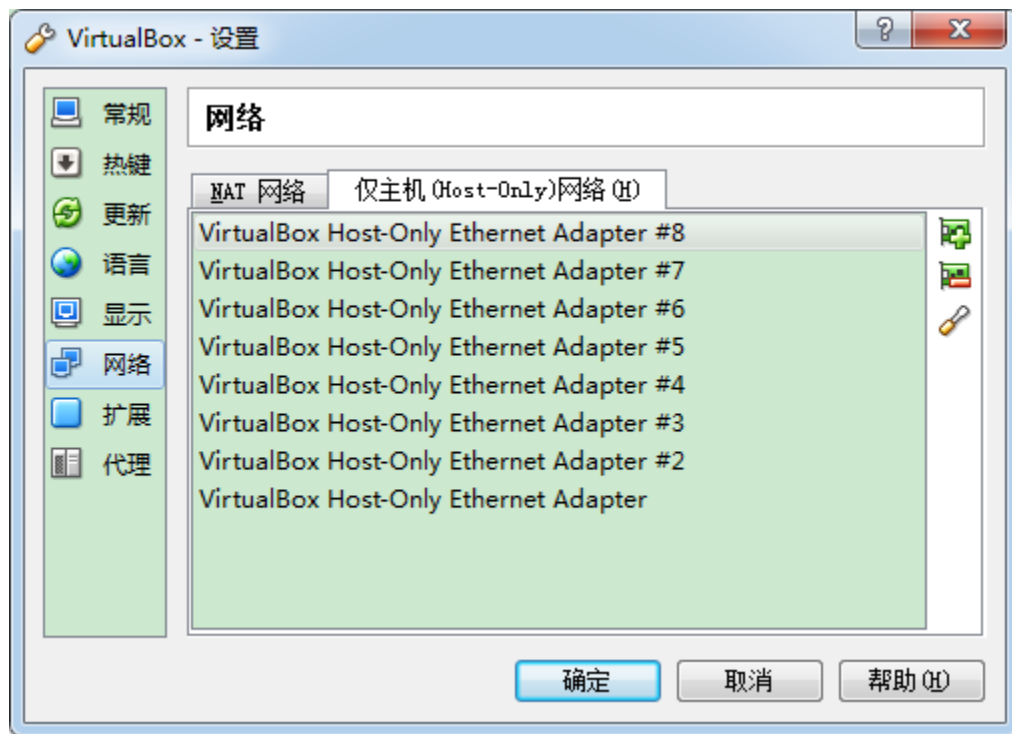


## 2.6 网络安全实验环境的配置

### 2.6.1 安装VirtualBox虚拟机

- 从 <http://www.virtualbox.org/> 下载最新版本的virtualbox(免费软件)，双击安装文件，按照提示进行安装。
- 按默认方式安装，安装完成后打开virtualbox软件（virtualbox管理器）。
- 如果能正确运行virtualbox管理器，则说明virtualbox安装完毕。

## 2.6.2 配置多个虚拟网卡，模拟多个网络交换机





## 2.6.3 安装和配置新的虚拟机系统

- 下载ubuntu安装光盘

安装32bit的ubuntu14.04



## 2.6.4 导入和导出安装好的虚拟机

- 拷贝ova文件到硬盘，导入到期望的目录中。

演示



## 2.6.5 在虚拟机上运行常用的命令行程序

- 在ubuntu Linux和Windows虚拟机下运行常用的命令行程序，比如：
  - chmod, chown, ls, mkdir, cp, rm, ifconfig;
  - dir, mkdir, copy, net, ipconfig, netstat。



# 作业与实践

1. Linux系统中的crontab和Windows系统中的at命令类似，请说明该命令的用法。
  2. 说明Windows系统中的sc命令的3种用法。
  3. 用tracert命令查看并记录下从本地主机到www.sina.com所经过的路由。如果从你的主机无法tracert到[www.sina.com](http://www.sina.com)，分析原因。
- 实践(不考核，自己练习)
    1. 用CSocket实现两台计算机之间的数据通信。
    2. netcat是经典的网络工具，下载该工具，并利用netcat在本机开启一个监听端口。