快速通道

文档分类
- 红盟公告
- 新手上路 <=
- 漏洞分析
- 工具介绍
- 安全配置
- 编程技术
- 安全技术
- 业界动态

本类热门
- ftp使用经典
- 一个木马的查杀过程
- 基于arp欺骗的网络攻击程序(BtNet.exe)
- 终端服务全攻略
- 握着你的“手”清除DLL后门
- 巧用asp木马和KV2004得到管理员权限

握着你的“手”清除DLL后门

提交文章： dahubaobao
提交日期： 2004-03-21
文章属性： 原创
文档类别： 新手上路
相关下载： **=>下载<=**
浏览次数： 今4次/总15954次

文：握着你的“手”清除DLL后门
作者：dahubaobao [Nuke Group]
主页：http://www.ringz.org
团体站点：http://www.digitalnuke.com/main/
邮件：dahushibaobao@vip.sina.com
QQ：47809945
发表于黑客X档案2004年2期
工具及图片都在压缩包中，解压密码：www.ringz.org
http://dahubaobao.go.nease.net/DLL.rar
欢迎进入环形区，一群技术狂热者的社区，www.ringz.org欢迎你的加入！
==========
握着你的“手”来清除DLL后门
------DLL后门清除完全篇
前言
后门！相信这个词语对您来说一定不会陌生，它的危害不然而欲，但随着人们的安全意识逐步增强，又加上杀毒软件的“大力支持”，使传统的后门无法在隐藏自己，任何稍微有点计算机知识的人，都知道“查端口”“看进程”，以便发现一些“蛛丝马迹”。所以，后门的编写者及时调整了思路，把目光放到了动态链接程序库上，也就是说，把后门做成DLL文件，然后由某一个EXE做为载体，或者使用Rundll32.exe来启动，这样就不会有进程，不开端口等特点，也就实现了进程、端口的隐藏。本文以“DLL的原理”“DLL的清除”“DLL的防范”为主题，并展开论述，旨在能让大家对DLL后门“快速上手”，不在恐惧DLL后门。好了，进入我们的主题。

一，DLL的原理
1，动态链接程序库
动态链接程序库，全称：Dynamic Link Library，简称：DLL，作用在于为应用程序提供扩展功能。应用程序想要调用DLL文件，需要跟其进行“动态链接”；从编程的角度，应用程序需要知道DLL文件导出的API函数方可调用。由此可见，DLL文件本身并不可以运行，需要应用程序调用。正因为DLL文件运行时必须插入到应用程序的内存模块当中，这就说明了：DLL文件无法删除。这是由于Windows内部机制造成的：正在运行的程序不能关闭。所以，DLL后门由此而生！

2，DLL后门原理及特点
把一个实现了后门功能的代码写成一个DLL文件，然后插入到一个EXE文件当中，使其可以执行，这样就不需要占用进程，也就没有相对应的PID号，也就可以在任务管理器中隐藏。DLL文件本身和EXE文件相差不大，但必须使用程序（EXE）调用才能执行DLL文件。DLL文件的执行，需要EXE文件加载，但EXE想要加载DLL文件，需要知道一个DLL文件的入口函数（既DLL文件的导出函数），所以，根据DLL文件的编写标准：EXE必须执行DLL文件中的DLLMain（）作为加载的条件（如同EXE的mian（））。做DLL后门基本分为两种：

1）把所有功能都在DLL文件中实现；2）把DLL做成一个启动文件，在需要的时候启动一个普通的EXE后门。

常见的编写方法：
(1)，只有一个DLL文件

这类后门很简单，只把自己做成一个DLL文件，在注册表Run键值或其他可以被系统自动加载的地方，使用Rundll32.exe来自动启动。Rundll32.exe是什么？顾名思意，“ 执行32位的DLL文件 ”。它的作用是执行DLL文件中的内部函数，这样在进程当中，只会有Rundll32.exe，而不会有DLL后门的进程，这样，就实现了进程上的隐藏。如果看到系统中有多个Rundll32.exe，不必惊慌，这证明用Rundll32.exe启动了多少个的DLL文件。当然，这些Rundll32.exe执行的DLL文件是什么，我们都可以从系统自动加载的地方找到。

现在，我来介绍一下Rundll32.exe这个文件，意思上边已经说过，功能就是以命令行的方式调用动态链接程序库。系统中还有一个Rundll.exe文件，他的意思是“ 执行16位的DLL文件 ”，这里要注意一下。在来看看Rundll32.exe使用的函数原型：

```
Void CALLBACK FunctionName (
HWND hwnd,
HINSTANCE hinst,
LPTSTR lpCmdLine,
Int nCmdShow
);
```

其命令行下的使用方法为：Rundll32.exe DLLname,Functionname [Arguments]
DLLname为需要执行的DLL文件名；Functionname为前边需要执行的DLL文件的具体引出函数；[Arguments]为引出函数的具体参数。

(2)，替换系统中的DLL文件
这类后门就比上边的先进了一些，它把实现了后门功能的代码做成一个和系统匹配的DLL文件，并把原来的DLL文件改名。遇到应用程序请求原来的DLL文件时， DLL后门就启一个转发的作用，把“ 参数 ”传递给原来的DLL文件；如果遇到特殊的请求时（比如客户端），DLL后门就开始，启动并运行了。对于这类后门，把所有操作都在DLL文件中实现最为安全，但需要的编程知识也非常多，也非常不容易编写。所以，这类后门一般都是把DLL文件做成一个“ 启动 ”文件，在遇到特殊的情况下（比如客户端的请求），就启动一个普通的EXE后门；在客户端结束连接之后，把EXE后门停止，然后DLL文件进入“ 休息 ”状态，在下次客户端连接之前，都不会启动。但随着微软的“ 数字签名 ”和“ 文件恢复 ”的功能出台，这种后门已经逐步衰落。

提示：
在WINNT\system32目录下，有一个dllcache文件夹，里边存放着众多DLL文件（也包括一些重要的EXE文件），在DLL文件被非法修改之后，系统就从这里来恢复被修改的DLL文件。如果要修改某个DLL文件，首先应该把dllcache目录下的同名DLL文件删除或更名，否则系统会自动恢复。

(3)，动态嵌入式
这才是DLL后门最常用的方法。其意义是将DLL文件嵌入到正在运行的系统进程当中。在Windows系统中，每个进程都有自己的私有内存空间，但还是有种种方法来进入其进程的私有内存空间，来实现动态嵌入式。由于系统的关键进程是不能终止的，所以这类后门非常隐蔽，查杀也非常困难。常见的动态嵌入式有：“ 挂接API ”“ 全局钩子（HOOK） ”“ 远程线程 ”等。

远程线程技术指的是通过在一个进程中创建远程线程的方法来进入那个进程的内存地址空间。当EXE载体（或Rundll32.exe）在那个被插入的进程里创建了远程线程，并命令它执行某个DLL文件时，我们的DLL后门就挂上去执行了，这里不会产生新的进程，要想让DLL后门停止，只有让这个链接DLL后门的进程终止。但如果和某些系统的关键进程链接，那就不能终止了，如果你终止了系统进程，那Windows也随即被终止！！！

3，DLL后门的启动特性

启动DLL后门的载体EXE是不可缺少的，也是非常重要的，它被称为：Loader。如果没有Loader，那我们的DLL后门如何启动呢？因此，一个好的DLL后门会尽力保护自己的Loader不被查杀。Loader的方式有很多，可以是为我们的DLL后门而专门编写的一个EXE文件；也可以是系统自带的Rundll32.exe，即使停止了Rundll32.exe，DLL后门的主体还是存在的。3721网络实名就是一个例子，虽然它并不是“ 真正 ”的后门。

二，DLL的清除
本节以三款比较有名的DLL后门例，分别为“ SvchostDLL.dll ”“ BITS.dll ”“ QoServer.dll ”。详细讲解其手工清除方法。希望大家在看过这三款DLL后门的清除方法之后，能够举一反三，灵活运用，在不惧怕DLL后门。其实，手工清除DLL后门还是比较简单

的，无非就是在注册表中做文章。具体怎么做，请看下文。

1，PortLess BackDoor
这是一款功能非常强大的DLL后门程序，除了可以获得Local System权限的Shell之外，还支持如“ 检测克隆帐户 ”“ 安装终端服务 ”等一系列功能（具体可以参见程序帮助），适用Windows2000/xp/2003等系统。程序使用svchost.exe来启动，平常不开端口，可以进行反向连接（最大的特点哦），对于有防火墙的主机来说，这个功能在好不过了。

在介绍清除方法之前，我们先来简单的介绍一下svchost.exe这个系统的关键服务：
Svchost只是做为服务的宿主，本身并不实现什么功能，如果需要使用Svchost来启动服务，则某个服务是以DLL形式实现的，该DLL的载体Loader指向svchost，所以，在启动服务的时候由svchost调用该服务的DLL来实现启动的目的。使用svchost启动某个服务的DLL文件是由注册表中的参数来决定的，在需要启动服务的下边都有一个Parameters子键，其中的ServiceDll表明该服务由哪个DLL文件负责，并且这个DLL文件必须导出一个ServiceMain()函数，为处理服务任务提供支持。

呵呵！看了上边的理论，是不是有点蒙（我都快睡着了），别着急，我们来看看具体的内容（如图1）。从图1中，我们可以看到HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RpcSs下的Parameters子键，其键值为%SystemRoot%\system32\rpcss.dll。这就说明：启动RpcSs服务时。Svchost调用WINNT\system32目录下的rpcss.dll。

在来看看图2，这是注册表的HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost，里边存放着Svchost启动的组和组内的各个服务，其中netsvcs组的服务最多。要使用Svchost启动某个服务，则该服务名就会出现在HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost下。这里有四种方法来实现：
1， 添加一个新的组，在组里添加服务名
2， 在现有组里添加服务名
3， 直接使用现有组里的一个服务名，但是本机没有安装的服务
4， 修改现有组里的现有服务，把它的ServiceDll指向自己的DLL后门

我测试的PortLess BackDoor使用的第三种方法。
好了，我想大家看完了上边的原理，一定可以想到我们清除PortLess BackDoor的方法了，对，就是在注册表的Svchost键下做文章。好，我们现在开始。

注：由于本文只是介绍清除方法，使用方法在此略过。

后门的Loader把SvchostDLL.dll插入Svchost进程当中，所以，我们先打开Windows优化大师中的Windows进程管理2.5，查看Svchost进程中的模块信息（如图3），从图3中我们可以看到，SvchostDLL.dll已经插入到Svchost进程中了，在根据“ 直接使用现有组里的一个服务名，但是本机没有安装的服务 ”的提示，我们可以断定，在“ 管理工具 ”—“ 服务 ”中会有一项新的服务。图4证明了我的说法，此服务名称为：IPRIP，由Svchost启动，-k netsvcs表示此服务包含在netsvcs服务组中。

我们把该服务停掉，然后打开注册表编辑器（开始—运行--regedit），来到HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IPRIP下，查看其Parameters子键（如图5）。Program键的键值SvcHostDLL.exe为后门的Loader；ServiceDll的键值C:\WINNT\system32\svchostdll.dll为调用的DLL文件，这正是后门的DLL文件。现在我们删除IPRIP子键（或者用SC来删除），然后在来到HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost下，编辑netsvcs服务组，把49 00 70 00 72 00 69 00 70 00 00 00删除，这里对应的就是IPRIP的服务名，具体如图6所示。然后退出，重启。重启之后删除WINNT\system32目录下的后门文件即可。

2，BITS.dll
这是榕哥的作品，也是DLL后门，和SvchostDLL.dll原理基本一样，不过这里使用的是上边介绍的第四种方法，即“ 修改现有组里的现有服务，把它的ServiceDll指向自己的DLL后门 ”。换句话说，该后门修改现有的某一个服务，把其原有服务的DLL指向自己（也就是BITS.dll），这样就达到了自动加载的目的；其次，该后门没有自己的Loader，而是使用系统自带的Rundll32.exe来加载。我们还是用Windows 进程管理2.5来查看，从图7中，我们可以看到bits.dll已经插入到Svchost进程当中。

好，现在我们来看看具体的清除方法，由于该后门是修改现有服务，而我们并不知道具体是修改了哪个服务，所以，在注册表中搜索bits.dll，最后在HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RasAuto下搜索到了bits.dll，查看Parameters子键下的ServiceDll，其键值为C:\WINNT\system32\bits.dll（如图8）。原来，该后门把RasAuto服务原来的DLL文件替换为bits.dll了，这样来实现自动加载。知道了原因就好办了，现在我们把ServiceDll的键值修改为RasAuto服务原有的DLL文件，

即%SystemRoot%\System32\rasauto.dll，退出，重启。之后删除WINNT\system32目录下的bits.dll即可。

3，NOIR--QUEEN
NOIR--QUEEN(守护者)是一个DLL后门&木马程序，服务端以DLL文件的形式插入到系统的Lsass.exe进程里，由于Lsass.exe是系统的关键进程，所以不能终止。在来介绍清除方法之前，我先介绍一下Lsass.exe进程：
这是一个本地的安全授权服务，并且它会为使用Winlogon服务的授权用户生成一个进程，如果授权是成功的，Lsass就会产生用户的进入令牌，令牌使用启动初始 的Shell。其他的由用户初始化的进程会继承这个令牌。
从上边的介绍我们就可以看出Lsass对系统的重要性，那具体怎么清除呢？请看下文。

后门在安装成功后，会在服务中添加一个名为QoSserver的服务，并把QoSserver.dll后门文件插入到Lsass进程当中，使其可以隐藏进程并自动启动（如图9）。现在我们打开注册表，来到HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\QoSserver，直接删除QoSserver键，然后重启。重启之后，我们在来到服务列表中，会看到QoSserver服务还在，但没有启动，类别是自动，我们把他修改为“已禁用”；然后往上看，会发现一个服务名为AppCPI的服务，其可执行程序指向QoSserver.exe（原因后边我会说到），具体如图11所示。我们再次打开注册表，来到
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AppCPI，删除AppCPI键，重启，再删除QoSserver，最后删除WINNT\system32目录下的后门文件。

本人和这个后门“搏斗”了3个多小时，重启N次。原因在于即使删除了QoSserver服务，后门还是在运行，而且服务列表中的QoSserver服务又“死灰复燃”。后来才知道原因：在我删除了QoSserver服务并重启之后，插入到Lsass进程当中的QoSserver.dll文件又恢复了QoSserver服务，并且生成了另外一个服务，即AppCPI，所以我们必须在到注册表中删除AppCPI服务才算是把该后门清除。由此可以看出，现在的后门的保护措施，真是一环扣环。

注意：在删除QoSserver服务并重启之后，恢复的QoSserver的启动类别要修改为“已禁用”，否则即便删除了AppCPI服务，QoSserver服务又运行了。

三，DLL的防范
看了上边的例子，我想大家对清除DLL后门的方法有了一定的了解，但在现实中，DLL后门并不会使用默认的文件名，所以你也就不能肯定是否中了DLL后门。对于DLL后门，system32目录下是个好地方，大多数后门也是如此，所以这里要非常注意。下面我来具体介绍一下怎么发现DLL后门，希望对大家有所帮助。

1，安装好系统和所有的应用程序之后，备份system32目录下的EXE和DLL文件：打开CMD，来到WINNT\system32目录下，执行：dir *.exe>exe.txt & dir *.dll>dll.txt，这样，就会把所有的EXE和DLL文件备份到exe.txt和dll.txt文件中；日后，如发现异常，可以使用相同的命令再次备份EXE和DLL文件(这里我们假设是exe0.txt和dll0.txt)，并使用：fc exe.txt exe0.txt>exedll.txt & fc dll.txt dll0.txt>exedll.txt，其意思为使用FC命令比较两次的EXE文件和DLL文件，并将比较结果保存到exedll.txt文件中。通过这种方法，我们就可以发现多出来的EXE和DLL文件，并通过文件大小，创建时间来判断是否是DLL后门。

2，使用内存/模块工具来查看进程调用的DLL文件，比如Windows优化大师中的Windows 进程管理 2.5。这样，可以发现进程到底调用了什么DLL文件，在结合上边用FC命令比较出来的结果，又能进一步来确定是否中了DLL后门。如果没有优化大师，可以使用TaskList，这个小工具也可以显示进程调用的DLL文件，而且还有源代码，方便修改。

3，普通后门连接需要打开特定的端口，DLL后门也不例外，不管它怎么隐藏，连接的时候都需要打开端口。我们可以用netstat －an来查看所有TCP/UDP端口的连接，以发现非法连接。大家平时要对自己打开的端口心中有数，并对netstat －an中的state属性有所了解。当然，也可以使用Fport来显示端口对应的进程，这样，系统有什么不明的连接和端口，都可以尽收眼底。

4，定期检查系统自动加载的地方，比如：注册表，Winstart.bat，Autoexec.bat，win.ini，system.ini，wininit.ini，Autorun.inf，Config.sys等。其次是对服务进行管理，对系统默认的服务要有所了解，在发现有问题的服务时，可以使用Windows 2000 Server Resource Kit中的SC来删除。以上这些地方都可以用来加载DLL后门的Loader，如果我们把DLL后门Loader删除了，试问？DLL后门还怎么运行？！

通过使用上边的方法，我想大多数DLL后门都可以“现形”，如果我们平时多做一些备份，那对查找DLL后门会启到事半功倍的效果。

后记
本文详细的介绍了DLL后门的一些知识。其实，从上文中不难看出，DLL后门并没有想象的这么可怕，清除起来也比较简单。在文章中的开头我以说到：旨在能让大家对DLL后门“快速上手”，所以，希望这篇拙文能对大家有所帮助，文中如有错误，还请大家多多包涵，谢

谢！最后，给大家拜个年，祝大家幸福快乐。

本贴由环形区原创，欢迎转帖！欢迎进入环形区，一群技术狂热者的社区，www.ringz.org欢迎你的加入！

返回上页

Copyright © 2000-2004 **HUC** All Rights Reserved.
中国红客网络技术联盟 www.cnhonker.com
webmaster@cnhonker.net