基于ASP开发的网站的漏洞及对策

胡吉平　（济南军区司令部工程设计研究院　250002）
张淑华　（济山东经济学院　250014）

摘要：本文针对微软的IIS4.0、IIS5.0的漏洞以及用ASP技术开发的网站易泄露ASP源代码等问题，从几个方面进行了探讨，并提出了相应的解决方案。

关键词：ASP　网络安全　IIS　加密　脚本

1　问题的提出

随着互联网的发展及其普及程度的日益提高，越来越多的实体和个人都加入到了网络的环境当中，拥有本部门或自己的网站，已经成为越来越多的人的目标。Microsoft推出的ASP（Active Server Pages）以其简单、易用、多功能、可扩充性等强大的功能而受到越来越多的开发者的青睐，但同时也存在一些问题。

由于ASP的脚本是采用（plain text）方式编写出来的，它的基本工作原理是当客户端用户用浏览器通过internet来访问基于ASP的脚本时，浏览器向Web服务器发出http请求，Web服务器通过ISAPI接口调用ASP脚本解释运行引擎（ASP.DLL），ASP.DLL将从文件系统或内部缓冲区获取指定的脚本文件并解释执行。最终的处理结果形成HTML文件格式返回给浏览器。如果是使用ASP的话，那么你的网络的安全性同时也大大降低了，而其所造成的危害也是多种多样的。如开发人员辛辛苦苦开发出来的“源代码”被流传出去、公司的某些保密技术或产品被非法用户通过种种办法轻易地看到、某些大型的数据库信息网站（如产品的适时价格信息或交通时刻表等）被“黑客”侵入服务器并获取以及删除或格式化数据库，所以，如何保护基于ASP开发的网页或者网站，已经成为当前网络需要解决的迫切问题。

2　通过ASP及其IIS的漏洞入侵Web Server及其解决方案

2.1　在程序名后加特殊符号可看源程序

在ASP程序名后加个特殊符号，这些符号包括小数点(.)、%81、::$data，如http://www.jsrdi.com/default.asp%81等，则能看到default.asp源程序，受影响的版本有win95+pws，IIS3.0。产生这种漏洞的原因是因为Windows NT提供了一种完全不同于Fat的文件系统——NTFS，虽然这种技术使得NT具有了较高的安全机制，但它同时产生了一种隐患，因为NTFS支持包含在一个文件中的多数据流，这个包含了所有内容的数据流被称之为“DATA”，因此使得在浏览器里直接访问NTFS系统的这个特性在文件中的脚本程序中实现成为了可能。然而直接导致::$data的原因是由于IIS在解析文件名的时候出了问题，它没有很好地规范文件名。

解决的方法和建议：如果是Windows NT的用户，安装IIS4.0或IIS5.0，Windows 2000不存在这个问题，如果是Windows 95的用户，安装Windows 98和pws4.0。

2.2　支持ASP的免费主页面临的问题

ASP1.0的例子里有一个文件用来查看ASP源代码，该文件为Aspsamp/samples/code.asp，如果有人把这个程序上传到服务器，而服务器端又没有任何防范措施，那他就很容易地查看他人的程序，例如：code.asp?source=/directory/file.asp，由于这是针对ASP早期的版本，因此是一个比较旧的漏洞了，而诸如http://www.jsrdi.com/iissamples/exair/

howtiworks/code.asp?/article/sousuo.asp=xxx.asp则是比较新的漏洞，最大的危害就是.asa文件可以被上述方法读出，则数据库的密码以明文的形式暴露在非法用户的面前。

解决的方法和建议：对于ASP自带的show asp code的ASP程序文件，删除该文件或者禁止访问该目录即可。

2.3　来自filesystemobject的威胁

IIS3.0、IIS4.0的ASP的文件操作都可以通过filesystemobject实现，包括文本文件的读、写目录操作，文件的拷贝、改名、删除等，但是这个功能是非常危险的，它可以下载Fat分区上的任何文件，即使是NTFS分

区，如果权限没有设定好的话，也同样能遭到破坏。

解决的方法和建议：尽量将Web放在NTFS分区上，目录不要设定“everyone完全控制”，即使是管理组的成员，只要有读出、更改的权限就可以了，也可以把filesystemboject的组件删除或者改名。

2.4 Access mdb数据库有可能被下载的漏洞

在用Access做后台数据库时，如果有人通过各种方法知道或者猜到了服务器的Access数据库的路径或名称，那么他就能够下载这个Access文件，比如，如果你的Access数据库member.mdb放在虚拟目录的database目录下，那么用户在浏览器中键入：http://www.jsrdi.com/database/member.mdb，如果你的member.mdb事先没有加密的话，那么member.mdb中的重要数据就都掌握在别人手中了。

解决的方法和建议如下：

(1) 为你的数据库文件起个复杂的“怪”名，并把它放在几重目录下，如有个数据库保存的是“成员”的信息，不要把它称做“member.mdb”，而起个名如“esf4dlk.mdb”，再把它放在如./kdlkr/l36/stuti/的几层目录下，这样非法用户要想通过“猜”的方法得到你的数据库文件就比较困难了。

(2) 在编写程序时注意两点，一是不要把数据库写在程序中。有些人喜欢把DSN写在程序中，诸如：

```
Dbpath = server.mappath("member.mdb")
Conn.open "driver={Microsofr Access Driver (*.mdb)};dbq=" &dbpath
```

如果万一有人拿到了你的源程序，你的Access数据库的名字则一览无余，因此建议在DSN中设置数据源，因为数据源只是一个别名，用户无法通过这个名字知道你的数据库原名，再在程序中这样写：conn.open "sourcename"。

二是使用Access来为数据文件编码并加密。首先选取“工具→安全→加密/解密数据库”，选取member.mdb，将数据库编码后另存为“member1.mdb",再为数据库加密，以“独占”方式打开member1.mdb,选择功能表的“工具→安全→设置数据库密码”，接着输入密码即可。

为member1.mdb设置了密码之后，接下来再使用Access数据库文件时，则Access会首先要求输入密码，验证正确后才能启动数据库，不过要在ASP程序中的nonnection对象的open方法中增加PWD参数即可，如：

```
udata = "driver={Microsoft Access Driver (*.mdb)};PWD=xderfs"
udata = udata & ";dbq = " &server.mappath("member1.mdb")
conn.open udata
```

这样，即使他人得到了member1.mdb，没有密码也无法看到member1.mdb的内容。

2.5 IIS4.0和IIS5.0中安装有Index Server服务会泄露ASP源程序

在运行IIS4.0或IIS5.0的Index Server，输入特殊字符格式可以看到ASP源程序或者其他页面的程序。即使添加了关于参看源代码的补丁程序的系统，或者没有.htw的系统，一样存在该问题。能够获得ASP程序，甚至global.asa文件的源代码，对系统是一个非常大的安全隐患。往往这些代码中包含了用户密码和ID，以及数据库的源路径和名称等等，这对于攻击者收集系统信息，进行下一步的入侵都是非常重要的。

通过构建下面的特殊程序可以参看该程序源代码：

http://www.jsrdi.com/null.htw?CiWebHitsFile

= /default.asp&CiRestriction = none&CiHikiteType = Full

这样只是返回一些html格式的文件代码，但是当你添加%20到CiWebHitsFile的参数后边，如下：

http://www.jsrdi.com/null.htw?CiWebHitsFile

= /default.asp%20&Cirestriction = none&CiHiliteType = Full

这将获得程序的源代码(注意：/default.asp是以Web的根开始计算，如某站点的http://www.jsrdivcom/welcome/welcome.asp，那么对应就是http://www.jsrdi.com/null.htw?CiWebHitsFile = /welcome/welcome.asp%20&Cirestriction = none&CiHiliteType = Full)，由于null.htw文件并非真正的系统映射文件，所以只是一个储存在系统内存中的虚拟文件，哪怕你已经从你的系统中删除了所有的.htw文件，但是由于对null.htw文件的请求默认是由webhits.dll来处理，所以，IIS仍然受到该漏洞的威胁。

解决的方法和建议如下：

如果该webhits提供的功能是系统必须的，请下载相应的补丁程序，如果没有必要，请用IIS的MMC管理工具简单移除.htw的映象文件。

2.6 ASP程序密码验证漏洞

很多网站把密码放在数据库中，在登录验证中用以下sql语句验证：

```
sql=" select * from user where username = ' " &username&" ' and pass = ' " &pass&" ' "
```

并且判断此sql语句返回的记录集是否为空来确定登录是否失败。其

问题出在没有对收集到的username信息和pass信息进行任何预处理，从而使得用户可以在输入用户名和密码的时候使用“′”这个sql分隔符号，因此用户可以构造出这样的用户名和密码：username=″ ABC ′ or ′ 1 ′ = ′ 1″，然后文中的数据验证部分会产生这样一个select语句并交由ASP执行：″ select * from user where username=″ &ABC ′ or ′ 1 ′ = ′ 1&″ and pass=″ &pass&″，or是一个逻辑运算符，作用是在判断两个条件的时候，只要其中一个条件成立，那么等式将会成立。而在语句中，是以1来代表真的（成立），那么在这行语句中，原语句的“and”验证将不再继续，而因为“1=1”和“or”使语句返回为真值。

解决的方法和建议：

一个可靠的ASP验证页面应该拥有数据预处理和多数据结果的补充验证，并且在验证程序当中要对用户输入的“′”号进行检测。

3 其他安全对策

3.1 用Script Encoder 1.0对脚本进行加密

Script Encoder 1.0是微软推出的一个命令行脚本加密工具，其特点是：它只加密页面中嵌入的脚本代码，其他部分，如HTML的TAG仍然保持原样不变。处理后的文件中被加密过的部分为只读内容，对加密部分的任何修改都将导致整个加密后的文件不能使用。Script Encoder加密过的ASP文件还将使Script Debugger之类的脚本调试工具失效。另外，ScriptEncoder可以对客户端脚本加密，也可对服务器端脚本进行加密。

它的操作非常简单：

SCRENC [/s] [/f] [/xl] [/l defLanguage] [/e defExtension] inputfile outfile

由于篇幅所限，具体参数的用法恕不赘述。

3.2 用ASP2DLL对源代码进行加密

ASP2DLL（可在http://www.xde.net/asp2dll下获取）的基本思路是将ASP程序转换成VB6.0的工程文件，并编译该文件成DLL，然后注册该DLL文件，编程序时再修改你的ASP程序，将ASP改为对DLL对象的调用，经过以上步骤，我们再打开转换后的ASP程序时，只能看到对转换后产生的DLL对象中的子过程调用的代码，而无法了解源程序是如何编写的。

4 结束语

本文只是列举了一些常见的用ASP构建的网站的一些易被攻击的方法，随着互联网的发展和电子商务的普及，网站的安全性也必将成为不得不考虑的一个非常重要的因素，即使“黑客”不攻击你的网站，但其随意窃取你的设计成果也是你所不能容忍的。■

参考文献

1《ASP应用开发指南》，[美]Grey Buczek，MSCD. MCT编著 李博 于赛译