ＳＱＬ注入是从正常的WWW端口访问，而且表面看起来跟一般的Web页面访问没什么区别， 所以目前市面的防火墙都不会对ＳＱＬ注入发出警报，如果管理员没查看IIS日志的习惯，可能被入侵很长时间都不会发觉。

随着B/S模式应用开发的发展，使用这种模式编写应用程序的程序员也越来越多。但是由于这个行业的入门门槛不高，程序员的水平及经验也参差不齐，相当大一部分程序员在编写代码的时候，没有对用户输入数据的合法性进行判断，使应用程序存在安全隐患。用户可以提交一段数据库查询代码，根据程序返回的结果，获得某些他想得知的数据，这就是所谓的SQL Injection，即SQL注入。

SQL注入是从正常的WWW端口访问，而且表面看起来跟一般的Web页面访问没什么区别，所以目前市面的防火墙都不会对SQL注入发出警报，如果管理员没查看IIS日志的习惯，可能被入侵很长时间都不会发觉。

但是，SQL注入的手法相当灵活，在注入的时候会碰到很多意外的情况。能不能根据具体情况进行分析，构造巧妙的SQL语句，从而成功获取想要的数据，是高手与“菜鸟”的根本区别。

SQL注入漏洞全接触--入门篇

SQL注入是从正常的WWW端口访问，而且表面看起来跟一般的Web页面访问没什么区别，所以目前市面的防火墙都不会对SQL注入发出警报，如果管理员没查看IIS日志的习惯，可能被入侵很长时间都不会发觉。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449834.aspx

SQL 注入漏洞全接触--进阶篇

SQL注入的一般步骤，首先，判断环境，寻找注入点，判断数据库类型，其次，根据注入参数类型，在脑海中重构SQL语句的原貌。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449870.aspx

SQL 注入漏洞全接触--高级篇

看完入门篇和进阶篇后，稍加练习，破解一般的网站是没问题了。但如果碰到表名列名猜不到，或程序作者过滤了一些特殊字符，怎么提高注入的成功率？怎么样提高猜解效率？

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449883.aspx

SQL 注入法攻击一日通

SQL注入的简单原理和攻击一般步骤，文章作者想让人一天学会SQL注入攻击。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449901.aspx

SQL Server应用程序中的高级SQL注入

这份文档是详细讨论SQL注入技术，它适应于比较流行的IIS+ASP+SQLSERVER平台。它讨论了哪些SQL语句能通过各种各样的方法注入到应用程序中，并且记录与攻击相关的数据确认和数据库锁定。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449913.aspx

编写通用的ASP防SQL注入攻击程序

如果编写通用的SQL防注入程序一般的http请求不外乎get 和 post,所以只要我们在文件中过滤所有post或者get请求中的参数信息中非法字符即可，所以我们实现http 请求信息过滤就可以判断是是否受到SQL注入攻击。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449925.aspx

利用instr()函数防止SQL注入攻击

学asp也有一段时间了，这几天一直在写自己的程序，也遇到了好多问题，我就不得不得考虑到一些现在的漏洞，比如，‘ 或 and 1=1等等的一些漏洞！别的先不管，今天我就来说说如何堵这个漏洞！详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449932.aspx

SQL 注入攻击的原理及其防范措施

ASP编程门槛很低，新手很容易上路。在一段不长的时间里，新手往往就已经能够编出看来比较完美的动态网站，在功能上，老手能做到的，新手也能够做到。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449940.aspx

跨站式SQL注入技巧

学习如何从数据库中获取想要获得的内容。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449948.aspx

防范Sql注入式攻击

Sql注入式攻击是指利用设计上的漏洞，在目标服务器上运行Sql 命令以及进行其他方式的攻击 动态生成Sql命令时没有对用户输入的数据进行验证是Sql注入攻击得逞的主要原因。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449951.aspx

Dreamweaver 中sql注入式攻击的防范

在安全性方面，新手最容易忽略的问题就是SQL注入漏洞的问题。用NBSI 2.0对网上的一些ASP网站稍加扫描，就能发现许多ASP网站存在SQL注入漏洞。详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449953.aspx

PHP 与SQL注入攻击

SQL注入攻击是黑客攻击网站最常用的手段。如果你的站点没有使用严格的用户输入检验，那么常容易遭到SQL注入攻击。SQL注入攻击通常通过给站点数据库提交不良的数据或查询语句来实现，很可能使数据库中的纪录遭到暴露，更改或被删除。下面来谈谈SQL注入攻击是如何实现的，又如何防范。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449956.aspx

SQL 注入攻击零距离

一次次的SQL注射入侵，一次次的网站被黑，总是那句话，漏洞在所难免，难道一点办法都没吗？这篇文章就是为大家解析下SQL注 射的原理，以及给出一些预防方法。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449959.aspx

SQL 注入技术和跨站脚本攻击的检测

在最近两年中，安全专家应该对网络应用层的攻击更加重视。因为无论你有多强壮的防火墙规则设置或者非常勤于补漏的修补机制，如果你的网络应用程序开发者没有遵循 安全代码进行开发，攻击者将通过80端口进入你的系统。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449961.aspx

菜鸟入门级：SQL注入攻击

一般国内的小一点的新闻站点程序 都有 ""&request 这种漏洞，下面我讲解攻击方法。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449963.aspx

三步堵死SQL注入漏洞

防御SQL注入有妙法，第一步：很多新手从网上下载SQL通用防注入系统的程序，在需要防范注入的页面头部用 来防止别人进行手动注入测试.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449965.aspx

SQL 注入实战---利用“dbo”获得SQL管理权限和系统权限

如果显示“dbo” 转换数据类型为 int 的列时发生语法错误 那么就可以用我下面介绍的方法来获得系统管理权限，如果是“abc” 转换数据类型为 int 的列时发生语法错误 那么就用不能用我下面的介绍来获得系统权限了.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449967.aspx

两个防SQL注入过滤代码

SQL注入防护的一种简单方法，在网页中嵌入过滤代码，基于认为的安全防护措施.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449969.aspx

蓝雨设计整站SQL注入漏洞

以下文章中就会出现NOWA修改系统漏洞当中从未有的SQL注入漏洞！只怪蓝雨修改程序的时候没有做好注入问题了！这个可不能怪我！谁叫人家程序设计员不会注意安全死角阿？

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449972.aspx

SQL 注入渗透某网络安全公司的网站全过程

写这篇文章不是为了告诉大家如何去做入侵，而是希望提醒大家:“入侵是偶然的，但安全并不是必然的”，不要疏忽运作上的一些小细节。

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449973.aspx

sql 注入防御

SQL注入通过网页对网站数据库进行修改。它能够直接在数据库中添加具有管理员权限的用户，从而最终获得系统管理员权限。黑客可以利用获得的管理员权限任意获得网站上的文件或者在网页上加挂木马和各种恶意程序，对网站和访问该网站的网友都带来巨大危害.详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449976.aspx

终极防范SQL注入漏洞

其实SQL注入漏洞并不可怕，知道原理 ＋ 耐心仔细，就可以彻底防范！下面给出4个函数，足够你抵挡一切SQL注入漏洞！读懂代码，你就能融会贯通.详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449979.aspx

SQL 注入与ASP木马上传

SQL注入后，如何上传木马，一直是比较头疼的事，我这里提供上传木马的一种另一种方法。

1、SQL注入的时候，用xp\_cmdshell 向服务器上写入一个能写文件的asp文件.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449983.aspx

如何在SQL注入时保护数据库

SQL注入是防止数据库攻击的一个有效策略。攻击者将注入一个SQL语句到另外一个语句中，这个通常会损坏你的数据库。有数据库接口的Web站点通常在 SQL注入的时候容易受到攻击，因为它们是基于动态的SQL.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449986.aspx

天晨设计整站SQL注入漏洞

作者从一个安全工作员的角度，对天晨设计整站SQL注入漏洞做出了详细的测试.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449988.aspx

浅谈sql注入式(SQL injection)攻击与防范

没有系统的学习过asp或者php编程，也没有系统的学习过access、sqlserver、mysql等数据库，所以我不是一个程序员，虽然经常干一些类似程序员的事情.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449991.aspx

用 vbs来写sql注入等80端口的攻击脚本

昨天晚上在机器里乱翻时无意打开一个vbs脚本,突然发现一个以前没有见过的对象Test.SendRequest("http://" & g\_sServer & "/testfiles/browser.asp"),虽然对象没有见过,但是意思很明显:发送http请求.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449992.aspx

C# 检查字符串,防SQL注入攻击

这些天,CSDN上讨论SQL注入攻击似乎是如火如荼啊...我也来参合一下..如下,CheckParams函数,接收参数任意,如参数中有字符串,则对字符串进行检查,如参数中有集合(如Array之类,总之是实现了ICollection的),则对集合中的字符串元素进行检查.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449994.aspx

关于对SQL注入80004005 及其它错误消息分析

本文详细讲述了 Microsoft 数据访问组件 (MDAC) 收到 0x800040005 错误的一些常见原因，MDAC 包括 ActiveX 数据对象、OLE DB 和远程数据服务 (RDS)。同时，本文还讨论了其它一些错误消息，包括 80040e21、80040e14 和80040e10.

详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3449996.aspx

SQL 注入入侵动网SQL版论坛

现在动网最新版本是7.0+SP2。应该说安全性已经是很高的了。所以从脚本本身的问题去突破它难度不小。但是我们可以从外部的一些途径间接“搞定”动网.现在IIS+ASP+SQL2000的组合是比较常见的。而一个网站运用大量的ASP脚本程序，难免不出纰漏。如果一台主机上存在某个SQL注入点，而这台主机又安装有动网SQL版的话，基本上可以得出结论：这个动网就是你的了。下面来看一下实例.详细内容>>

http://blog.csdn.net/wufeng4552/archive/2008/12/05/3450001.aspx

利用SQL注入2分钟入侵网站全程实录

说起流光、溯雪、乱刀，可以说是大名鼎鼎无人不知无人不晓，这些都是小榕哥的作品。每次一提起小榕哥来，我的崇拜景仰就如滔滔江水，连绵不绝~~~~（又来了！） 让我们崇拜的小榕哥最新又发布了SQL注入工具，这回喜欢利用SQL注入入侵网站的黑友们有福了。小榕哥的工具就是强！偶用它来搞定我们本地的信息港，从寻找注入漏洞到注入攻击成功，通过准确计时，总共只用了3分还差40秒，呵呵，王者风范，就是强啊！不信吗？看看我的入侵过程吧.

详细内容>>