NTFS ADS 带来的 WEB 安全问题

Author:Pysolve
Captain@Xcl0ud.NeT

NTFS ADS 简介

NTFS 流全称为 NTFS 交换数据流（NTFS Alternate Data Streams），ADS 的诞生是为了兼容 Hierarchical File System 。HFS---分层文件系统，是由苹果公司推出的文件系统，其工作模式是将不同数据存在不同的分支文件，文件数据存放在数据分支而文件参数存放在资源分支。类似的，NTFS 流使用资源派生来维持与宿主文件相关的信息。ADS 有点类似文件的属性信息一样，依附于文件的传统边界之外。

来看一个 ADS 的实例，通常这个例子在讲到 ADS 的地方都会提到。新建一个文件，命名为 test.txt，该文件即是宿主文件；打开文件，输入内容”test”。在该目录下执行命令 echo "This is a stream" > test.txt:stream.txt 建立后 cmd 不会有任何提示且对于 Windows 资源管理器来说宿主文件没有发生任何变化（包括其大小、修改时间等）。这是因为 windows 下不是所有程序都能支持 ADS 导致的。同样 dir、type 等也不能看到。Notepad 能够部分支持 ADS，可以打开 test.txt:stream.txt，但 notepad 也不能完全支持，另存为时会出现参数错误。

注：

1、修改宿主文件的内容不会影响流的内容。

2、修改流的内容不会影响宿主文件的内容。

MSDN 上给出了一个完整的流的格式，如下：<filename>:<stream name>:<stream type>

filename	宿主文件的文件名
stream name	流名
stream type	流类型

注：

1、其中 stream type 也叫 attribute type（属性类型）。

2、用户不能创建一个新的流类型，流类型总是以$符号作为开始。

对 NTFS 格式下的一个文件而言，至少包含一个流， 即 data 流（其 stream type 为$DATA），data 流是文件的主流 默认的 data 流其 stream name 为空。ADS 可以省略 stream name ,但不能省略 stream type。默认一个文件如果被指派了流，而该流没有 stream type 的话会在存储时自动添加$DATA。例如上面看到的例子 test.txt:stream.txt 在存储时实际上是为 test.txt:stream.txt:$DATA,但在查询结果中需要去除:$DATA，否则会出现参数错误，这个是 notepad 不能很好的支持流所导致的。

对文件夹而言，没有 data 流，其主流是 directory 流（stream type 为$INDEX_ALLOCATION），directory 流默认的 stream name 是$I30。尽管文件夹默认没有 data 流，但用户可为其指派 data 流。

在 NTFS 中，有如下的这些 attribute type：

Stream Type	**Description**
$ATTRIBUTE_LIST	Lists the location of all attribute records that do not fit in the MFT record
$BITMAP	Attribute for Bitmaps
$DATA	Contains default file data
$EA	Extended attribute index
$EA_INFORMATION	Extended attribute information
$FILE_NAME	File name
$INDEX_ALLOCATION	The type name for a Directory Stream. A string for the attribute code for index allocation

$INDEX_ROOT	Used to support folders and other indexes
$LOGGED_UTILITY_STREAM	Use by the encrypting file system
$OBJECT_ID	Unique GUID for every MFT record
$PROPERTY_SET	Obsolete
$REPARSE_POINT	Used for volume mount points
$SECURITY_DESCRIPTOR	Security descriptor stores ACL and SIDs
$STANDARD_INFORMATION	Standard information such as file times and quota data
$SYMBOLIC_LINK	Obsolete
$TXF_DATA	Transactional NTFS data
$VOLUME_INFORMATION	Version and state of the volume
$VOLUME_NAME	Name of the volume
$VOLUME_VERSION	Obsolete. Volume version

ADS 带来的 WEB 安全问题

早在多年前就有人想出将恶意代码隐藏在 ADS 中并捆绑在自解压文件中，从而躲过杀软的检测的方法。在 WEB 安全领域，ADS 的特殊性和隐蔽性也带来了不少的问题。

::$DATA 请求泄露

Microsoft IIS 3.0/4.0 ::$DATA 请求泄露 ASP 源代码漏洞（MS98-003）CVE-1999-0278

这是一个很古老的漏洞，早期版本的 IIS 在处理文件请求时会先判断文件扩展名是否在可执行文件扩展名列表中，如果在，则执行并返回结果，如果不在，则直接返回文件内容。NTFS 文件系统支持 在文件中包含额外的数据流。$DATA 是在 NTFS 文件系统中存储数据流的属性。当我们对一个在 NTFS 分区中的 ASP 文件发出包含$DATA 请 求，IIS 会检查最后一个“.”后面的扩展名，因为多了“::$DATA”，结果 IIS 不认为这是一个 ASP 文件，而文件系统可以识别该请求，于是 ASP 的源代码被返回。在高版本的 IIS 以及其他的 Web Server 中测试没有发现该问题。

隐藏 webshell 问题

这种利用思路最早由 80sec 提出，很好的利用了 ADS 的隐蔽性。由于目前的 webshell 检测程序都还没能考虑到 ADS 层面的检测，所以，这是种隐藏 web 后门的很好的方式。

来看一个很简单的 demo：在 test.php 中添加<?php include(“1.php:.jpg”); ?>，在 1.php:.jpg 中写入后门代码。访问 test.php 时，后门代码能成功解析。

Bypass HTTP Basic Authentication

对于 IIS6.0+PHP、IIS7.5+asp、IIS7.5+php 环境下，如果某目录通过 HTTP Basic 来认证，假设其目录下有 index.php 文件，我们可以通过构造如下方式来绕过认证直接访问其目录下的文件。

/admin::$INDEX_ALLOCATION/index.php

/admin:$i30:$INDEX_ALLOCATION/index.asp

Bypass 黑名单验证上传

对于 windows 环境的服务器，上传 1.asp:.jpg 类型的文件，当文件传到服务端时，windows 会将该文件识别成 ADS，从而认为其宿主文件名为 1.asp 而将.jpg 识别为流名。在测试过程中（测试环境中 WEB Server 为 IIS、Apache、Nginx）无论如何构造都无法访问可以解析的文件。

前面我们提到，对于 1.asp:.jpg 这类 ADS，其完整的全称为 1.asp:.jpg:$DATA，stream name 可以省略但 stream type 是不能省略且不能自定义的。我们在测试中发现，当上传 1.php::$DATA 时（文件内容为 phpinfo），在存储时会出现逻辑问题，导致生成非空的 1.php 文件，当中的内容为 1.php::$DATA 的内容且可以正常在 webserver 下解析。该逻辑问题与语言无关与 web server 无关，也就是说上传 1.asp::$DATA 也会生成非空的 1.asp 文件，1.jsp::$DATA 同样。

另外，当 1.php::$INDEX_ALLOCATION 或 1.php:$I30:$INDEX_ALLOCATION 时，存储时逻辑同样

会出现问题 ,会把该文件误认为是文件夹 ,从而建立一个 1.php 的空文件夹。(注 :directory 流默认 stream name 为$I30)

Windows 还有个特性，就是当文件夹名或文件名末尾的任意个数的空格或点，在存储时都会自动去除，所以当上传 1.php::$DATA……或 1.php::$INDEX_ALLOCATION……此类文件同样会造成上述的存储时的逻辑错误。

在 MySQL UDF 提权中的利用

MySQL5.1 及其之后的版本，使用 UDF 提权时，指定 UDF 必须导出在 MySQL 目录下的 lib\plugin 下才有用，而非完全版的 Mysql 默认安装后没有 plugin 这个目录，且在 MySQL 中没有可创建文件夹的函数。(通常很多时候在 webshell 中也无权限建立该目录)，可用如下的 SQL 建立文件夹

select ‘xxx’ into outfile ‘c:\\test::$INDEX_ALLOCALTION\\’ ;

同样由于 windows 在处理时会把 c:\test::$INDEX_ALLOCALTION\当做 ADS 处理，从而生成一个 test 文件夹。

参考

[+] http://www.nsfocus.net/index.php?act=sec_bug&do=view&bug_id=3442
[+] http://blog.csdn.net/lake2/article/details/269659
[+] http://www.80sec.com/ntfs-web-security.html
[+] http://www.exploit-db.com/exploits/19033/
[+] http://technet.microsoft.com/zh-cn/aa364404%28v=vs.80%29
[+] http://msdn.microsoft.com/en-us/library/ff469236%28v=prot.10%29
[+] http://www.symantec.com/connect/articles/windows-ntfs-alternate-data-streams