



计算机安全与维护

第3章

Windows 系统安全问题排查基础



课程内容回顾+探索型实验 (1/2)

- 通过Windows权限设置，是否可以实现：文件内容无法被任何用户读取，但任意用户都可以向该文件写入内容？
- 一个目录只允许一个用户访问，其他用户拒绝访问。
- 一个目录下的文件，禁止被任何用户删除。



课程内容回顾+探索型实验 (1/2)

- 一个目录下，禁止任何用户创建任何类型的文件。
- 通过命令行向系统中添加一个用户名zhangsan，将该用户加入到用户组Administrators，并设置密码为123456。（提示：net help user、net help localgroup）
- 用户名和用户组名是否可以重名？



先修基础知识



常见Windows文件扩展名

- .dll
- .sys
- .bat
- .ini





可执行程序 VS. 可执行文件

- 可执行程序 VS. 进程
 - 可执行程序：静态，文件级别，由一系列计算机指令构成
 - 进程：动态
- 可执行程序 VS. 可执行文件
- 可执行程序 \subseteq 可执行文件
 - 可执行程序：.exe、.bat、.cmd
 - 可执行文件：.dll、.doc、.xls、.pdf



什么是.dll (1/2)

- DLL 是一个包含可由多个程序同时使用的代码和数据的库
 - 例如，在 Windows 操作系统中，Comdlg32 DLL 执行与对话框有关的常见函数。因此，每个程序都可以使用该 DLL 中包含的功能来实现“打开”对话框
- 在 Windows 文件保护中，操作系统禁止未经授权的代理更新或删除系统 DLL。因此，当程序安装操作尝试删除或更新被定义为系统 DLL 的 DLL 时，Windows 文件保护将寻找有效的数字签名



什么是.dll (2/2)

- ActiveX 控件 (.ocx) 文件
 - ActiveX 控件的一个示例是日历控件，它使您可以从日历中选择日期。
- 控制面板 (.cpl) 文件
 - .cpl 文件的一个示例是位于控制面板中的项。每个项都是一个专用 DLL。
- 设备驱动程序 (.drv) 文件
 - 设备驱动程序的一个示例是控制打印到打印机的打印机驱动程序。



如果可执行的代码在作恶?



本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows组件
- 恶意代码的关键技术
- 恶意代码的检测技术



恶意代码的概念

- 广义上讲，恶意代码是一种人为制造的，能够进行自我复制的，对计算机资源具有破坏作用的一组程序或指令的集合。
- 恶意代码的概念最早出现在1977年的一本科幻小说里。
- 1983年，计算机安全专家Frederick Cohen博士首次提出恶意代码的存在。



恶意代码的概念

- 相关术语
 - 恶意代码：Malicious Code
 - 恶意软件：Malware
 - 垃圾（信息）：Spam
- 恶意代码的核心特征
 - 执行结果非用户期望且包含恶意目的
- 恶意软件是由恶意代码编制而成
- 垃圾信息借助恶意代码和恶意软件加速传播，躲避查杀



恶意代码的发展阶段 (1/5)

- 原始病毒阶段

- 产生年限为1986-1989年之间。

- 由于当时的计算机软件少，大多是单机运行，病毒没有大量流行，种类有限，清除相对容易。

- 攻击目标较单一，主要通过截获系统中端向量的方式监视系统的运行状态，并在一定的条件下传染。

- 病毒不具备自我保护的措施。



恶意代码的发展阶段 (2/5)

- 混合型病毒阶段

- 产生年限在1989-1991年之间，是计算机病毒由简单发展到复杂的阶段。
- 计算机局域网的应用和普及给计算机病毒带来了第一次流行高峰。
- 攻击目标趋于混合，采取更隐蔽的方式驻留内存和传染目标，并且没有明显的特征，采取了自我保护的措施，出现很多的病毒的变种。



恶意代码的发展阶段 (3/5)

- 多态性病毒阶段

- 在每次传染目标时，宿主程序中的病毒程序大部分是可变的，防病毒软件查杀时非常困难。
- 这一阶段开始，病毒技术开始向多维化方向发展。

- 网络病毒阶段

- 从20世纪90年代中后期开始，依赖互联网传播的邮件病毒和宏病毒等大量涌现。
- 病毒传播快，隐蔽性强，破坏性大。



恶意代码的发展阶段 (4/5)

- 主动攻击性病毒阶段

- 典型代表是2003年的冲击波病毒和2004年的震荡波病毒。

- 这些病毒利用操作系统的漏洞进行进攻性的扩散，不需要任何的媒介和操作，危害性更大。

- 手机病毒阶段

- 随着移动通信网络的发展及移动终端的不断增强，计算机病毒走进了移动世界，手机用户覆盖面广，数量多，病毒危害和影响也就更大。



恶意代码的发展阶段 (5/5)

- 高级持续威胁 (APT)
 - Advanced
 - Persistent
 - Threat



APT大事件

- 2010年 Google极光攻击，持续数月
- 2011年 夜龙攻击被McAfee曝光，始于2007年
- 2010年 伊朗核电站遭到Stuxnet蠕虫（震网）攻击
- 2011年 震网二代 Duqu肆虐欧洲



恶意代码的发展趋势

- 网络化
- 专业化
- 智能化
- 人性化
- 隐蔽性
- 多样化
- 自动化



本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows组件
- 恶意代码的关键技术
- 恶意代码的检测技术



恶意代码的特征

- 恶意代码的一般特征

- 可执行性

- 传染性

- 非授权性

- 隐蔽性

- 潜伏性

- 可触发性

- 破坏性

- 针对性

- 衍生性

- 寄生性

- 不可预见性

- 持久性



恶意代码分类——按目的和动机

恶意代码形式	破坏	控制	窃密	恶作剧	获利
病毒、蠕虫	✓				
木马、rootkit		✓	✓		
逻辑炸弹、恶作剧程序				✓	
后门		✓	✓		✓
垃圾信息					✓
流氓软件/恶意扣费软件			✓		✓
钓鱼			✓		✓
僵尸网络	✓	✓			✓



恶意代码的特征

- 恶意代码的新特征
 - 抗分析性
 - 诱惑欺骗性
 - 向多元化发展
 - 传播方式多样化，传播速度更快
 - 攻击技术的混合型
 - 造成的破坏日益严重

现代恶意代码已经很难用单一的病毒、木马分类和定义



恶意代码的其他分类方法

- 按照植入方式分类

- 源码型（恶意代码插入到源程序中，编译成为合法程序的一部分）
- 嵌入型（将自身嵌入到现有程序中，使恶意代码与目标程序代码成为一体，清除会破坏合法程序）
- 外壳型（一般链接到宿主程序的首尾，宿主程序执行首先激活恶意代码）
- 操作系统型（加入或者取代部分操作系统进行工作，寄生在计算机磁盘的操作系统区）



恶意代码的其他分类方法

- 按照寄生方式分类

- 引导型（恶意代码程序取代正常的引导记录，占据了引导区的物理位置即可获得控制权）
- 文件型（通过操作系统的文件系统实施感染，以感染可执行文件为主）
- 混合型（综合了引导型恶意代码和文件型恶意代码的特点同时感染文件和引导扇区，同时使用加密和变形算法）



宏病毒 (Macro Virus)

- 宏是微软为office软件设计的功能，提供任务的自动化。如果宏中包含了有破坏能力的命令和自我复制的功能，这个宏就成了宏病毒。
- 宏病毒常见的是针对word，excel，powerpoint等office软件，通过宏录制器和Visual Basic编辑器来创建宏。
- 宏被存储在通用模板中，一执行程序，受感染的模板就会传播到所编辑的文档中去，并以此方式不断地感染。



恶意脚本

- 网页恶意脚本
 - XSS、CSRF
- 操作系统恶意脚本
 - JS、bat、cmd脚本



蠕虫病毒 (Worm)

- 蠕虫强调自身副本的完整性和独立性，主要通过计算机漏洞来进行传染
- 蠕虫的特点和发展趋势
 - 利用系统和程序的漏洞主动攻击
 - 传播方式多样
 - 影响范围大，感染主机数量多
 - 制作技术新
 - 与黑客技术相结合

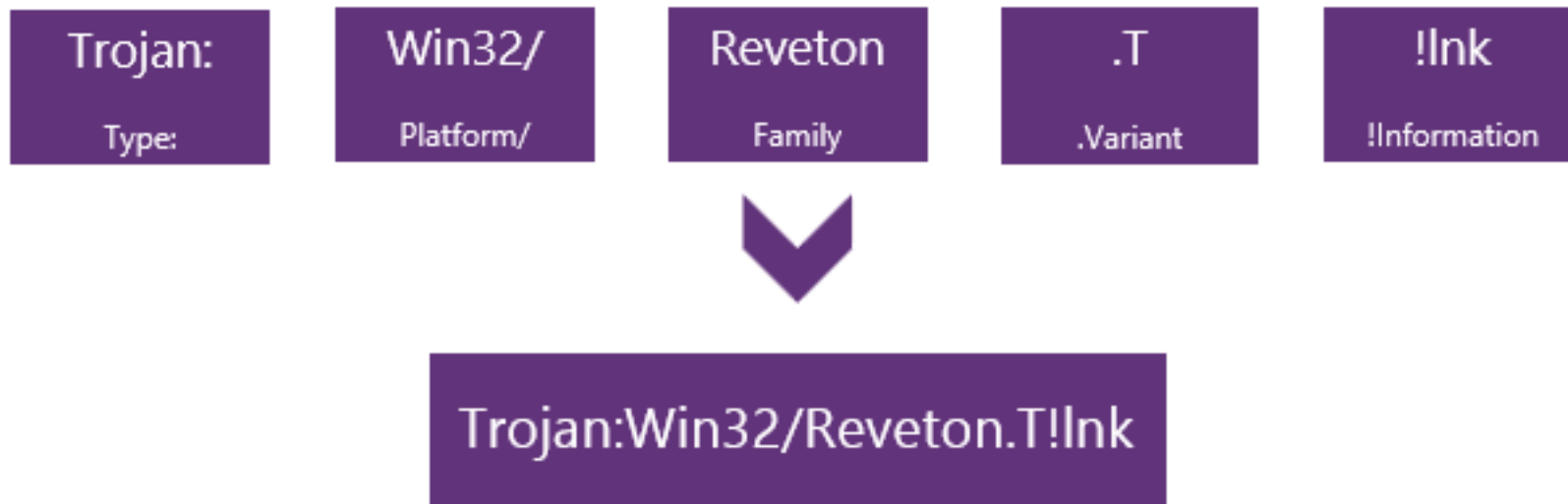


木马 (Trojan Horse)

- 木马一般有客户端和服务端两个程序，植入的木马会发送系统信息给服务器端
 - 客户端：植入端
 - 服务端：控制端
- 木马的基本特征
 - 隐蔽性
 - 自动运行
 - 欺骗性
 - 自动恢复
 - 控制性强（远程控制，键盘记录等）



微软对恶意代码的命名规范



- Type: 恶意代码类型/分类
- Platform: 感染/运行平台环境
- Family: 恶意代码家族名
- Variant: 变种名
- Information: 扩展说明

<http://www.microsoft.com/security/portal/mmpc/shared/malwarenaming.aspx>



本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows组件
- 恶意代码的关键技术
- 恶意代码的检测技术



恶意代码的基本结构

- 引导模块

- 使恶意代码获得执行并使后面的两个模块处于激活状态

- 传染模块

- 传染条件满足时把恶意代码传染到被攻击的对象上

- 破坏模块

- 在恶意代码破坏，发作条件满足时，实施对系统的干扰和破坏活动



恶意代码的引导机理

- 寄生对象

- 磁盘引导扇区
- 特定可执行文件、系统文件
- PDF、Word类文档

- 驻留内存

- 占据磁盘引导区中系统引导程序的位置，系统启动时自动装入内存获得控制权
- 修改原文件使对该文件的操作转入病毒的引导模块

- 窃取系统控制权

- 恢复系统功能



恶意代码的传染机理

- 传染是恶意代码由一个载体传播到另一个载体，由一个系统进入另一个系统
- 被动传染是基于系统的复制和系统I/O传输工作进行的
 - 网络传输
 - 外部存储设备：U盘、NFS等
- 主动传染时系统常驻内存并监视系统运行，伺机采取手段传染



恶意代码的破坏机理

- 恶意代码的破坏模块原理与传染模块相同，基于一个或者若干个设定的破坏条件满足的情况下才触发
- 恶意代码的破坏能力取决于设计者的目的和技术水平，一般会破坏数据区，文件，内存，磁盘，影响系统运行速度



恶意代码的触发机理

- 日期触发
- 键盘触发
- 启动触发
 - 系统启动、应用软件启动
- 调用中断触发
- 邮件触发
- 漏洞触发
- 磁盘访问触发



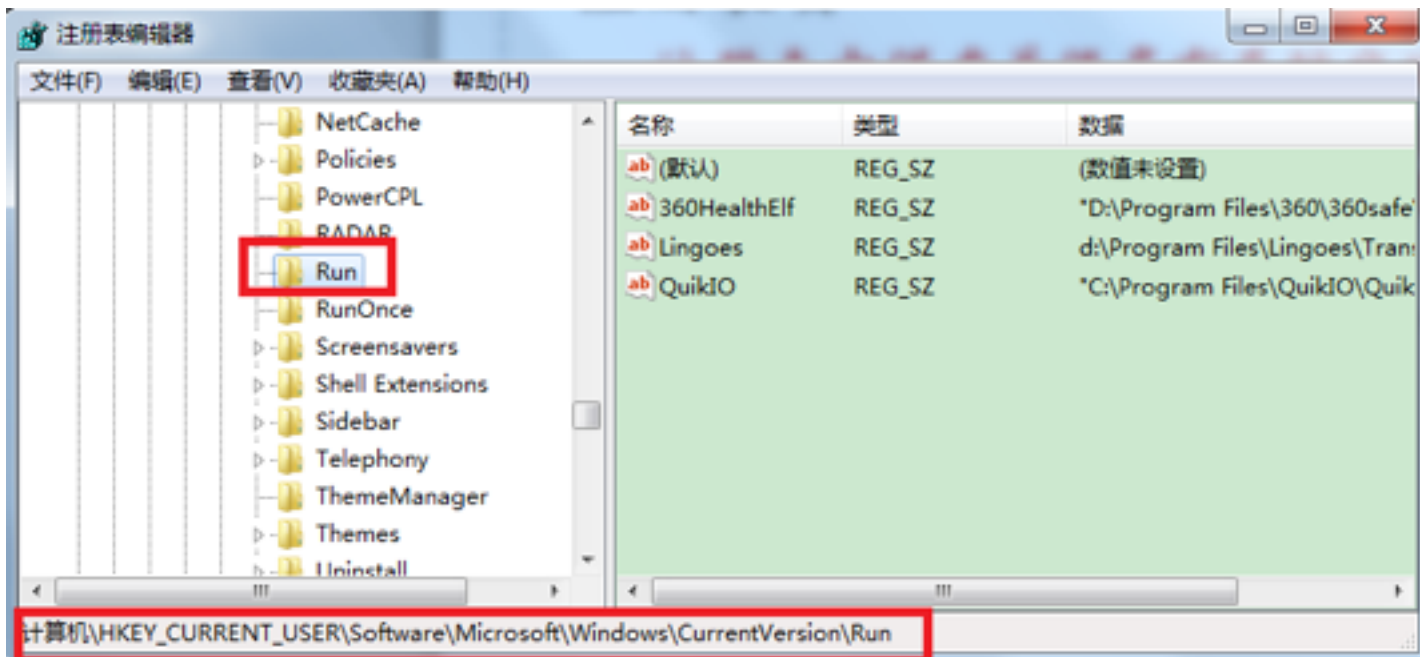
本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows关键组件
- 恶意代码的关键技术
- 恶意代码的检测技术



被恶意代码利用的Windows关键组件——注册表

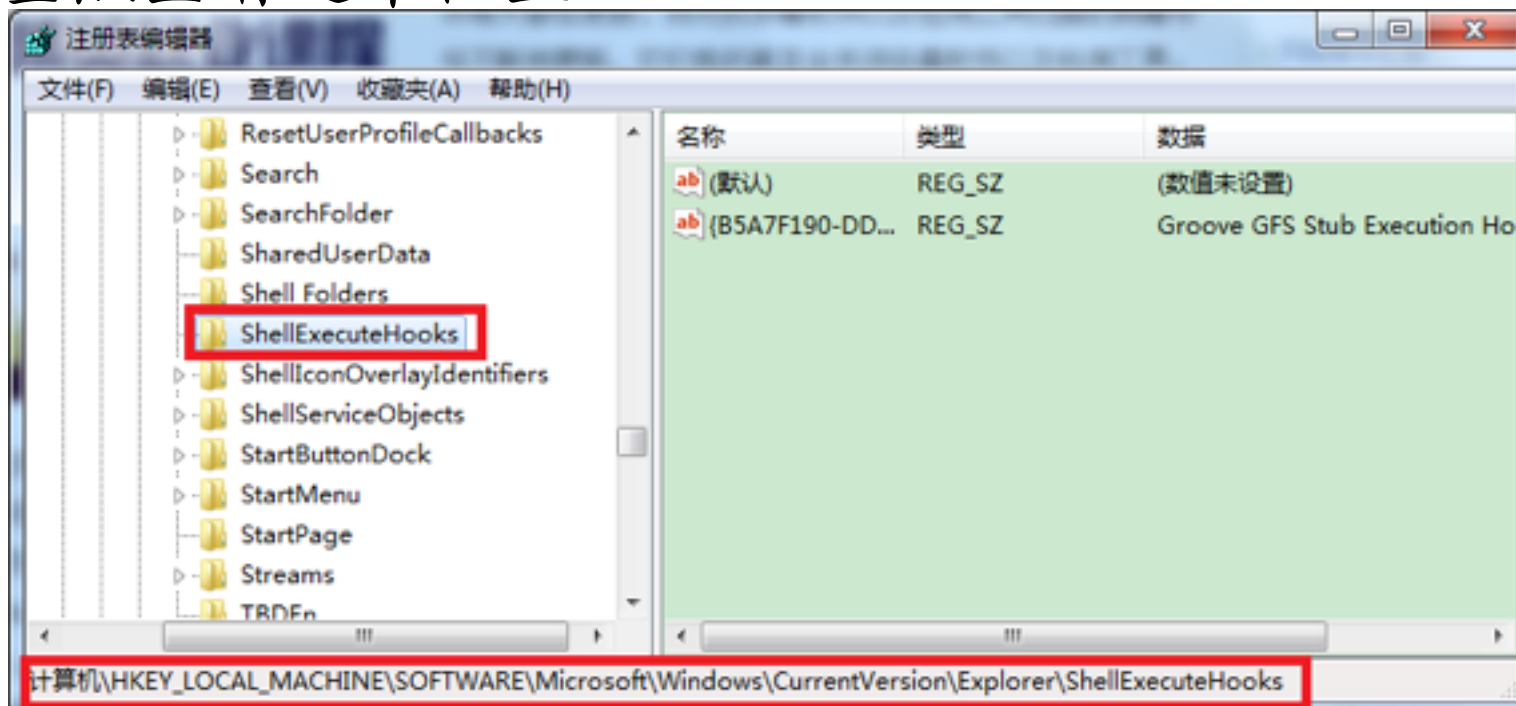
- 注册表中保存着很多有系统启动和运行相关的配置信息，很多的病毒程序会把自身的相关信息写入注册表，用来保护自身或者跟随系统自启动
- 最常见的自启动项





被恶意代码利用的Windows关键组件——注册表

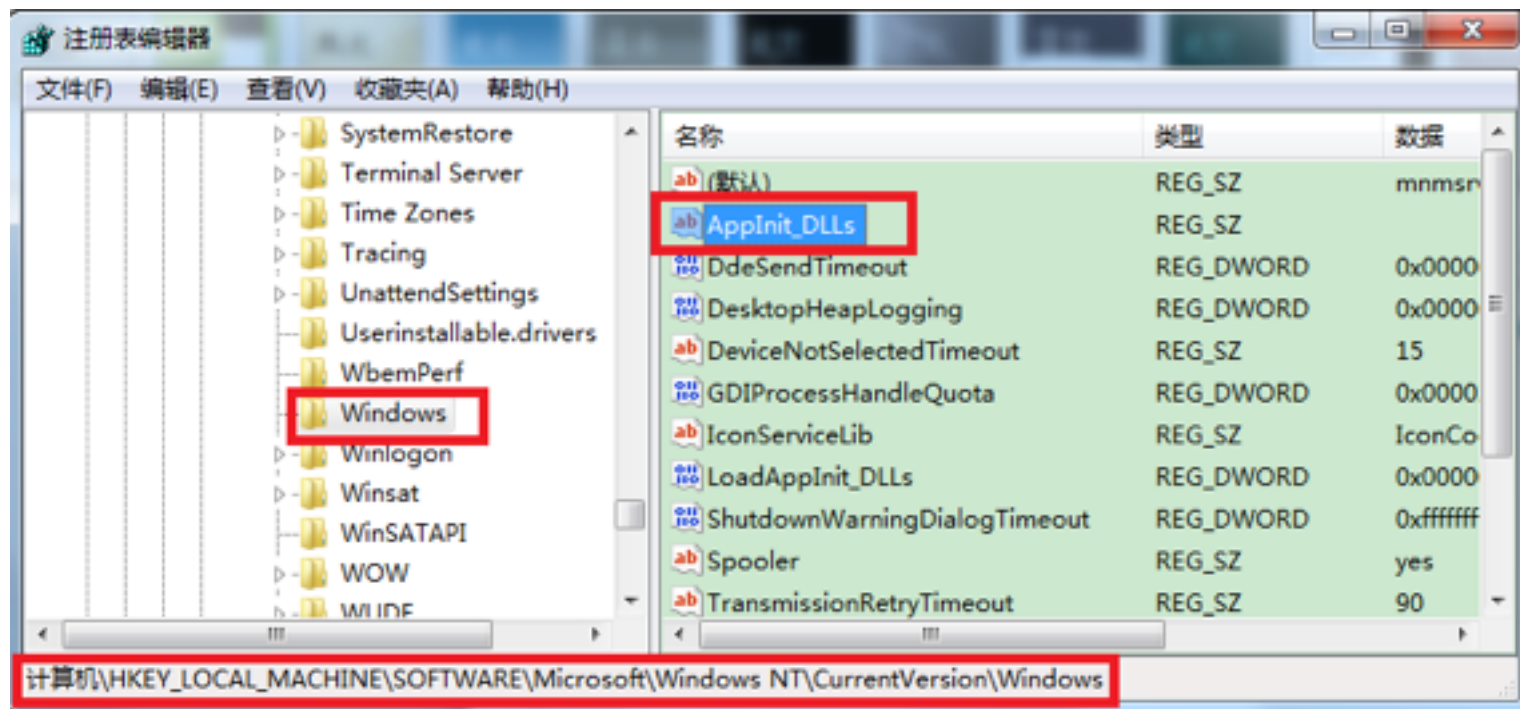
- 这个注册表项的每个值都是监控Explorer消息的钩子程序GUID
- 若病毒不易查杀，或者杀软被关闭，可能被挂钩，可以重点查看这个位置





被恶意代码利用的Windows关键组件——注册表

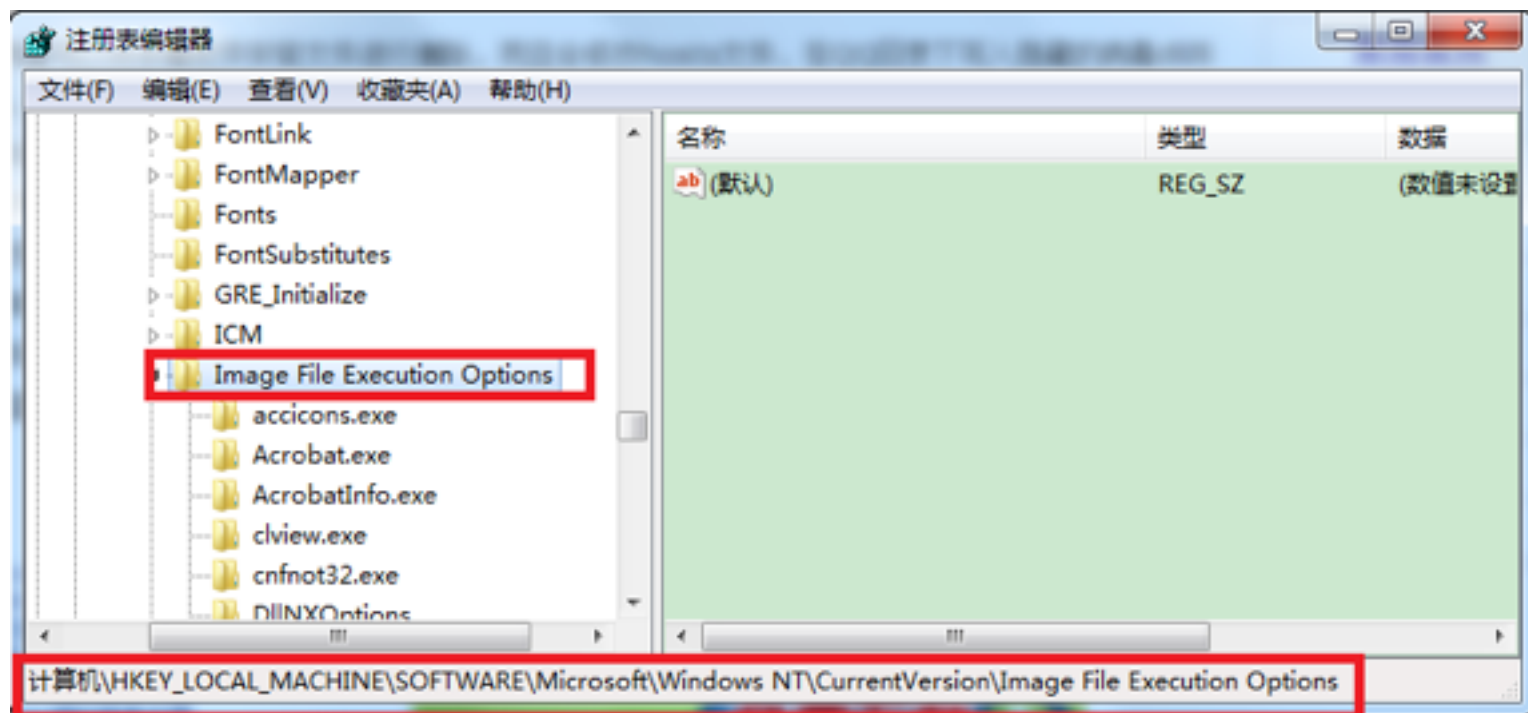
- 该值指定了系统级钩子程序的文件名，程序运行时加载，监控程序的消息，默认为空
- 很少有合法的程序会使用该键值





被恶意代码利用的Windows关键组件——注册表

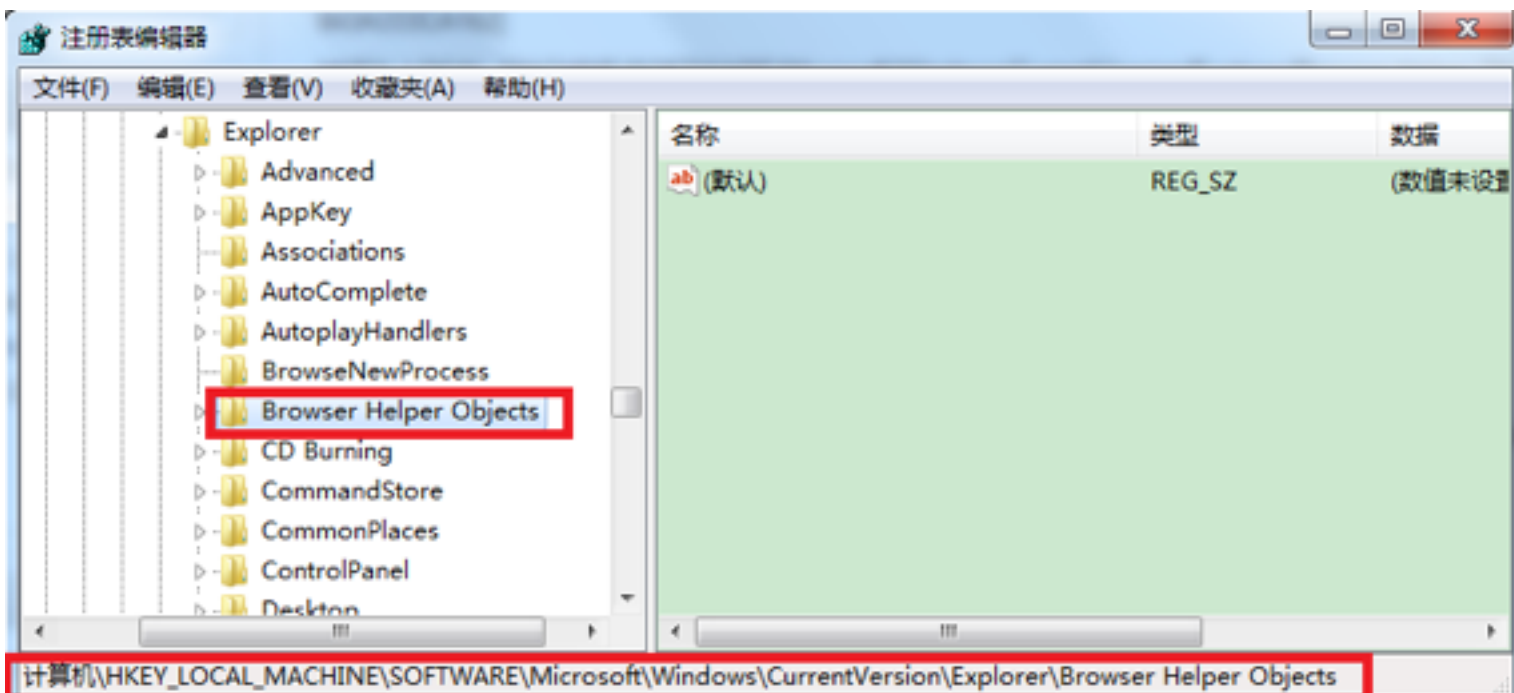
- 如果发现某个特定文件名的文件无法执行了，可能是被映像劫持了
- 这个位置是排查的重点





被恶意代码利用的Windows关键组件——注册表

- 这个注册表项的每个子项都保存了一个浏览器辅助插件的信息
- 很多的浏览器恶意软件通过该键值在浏览器启动时加载





被恶意代码利用的Windows关键组件——注册表

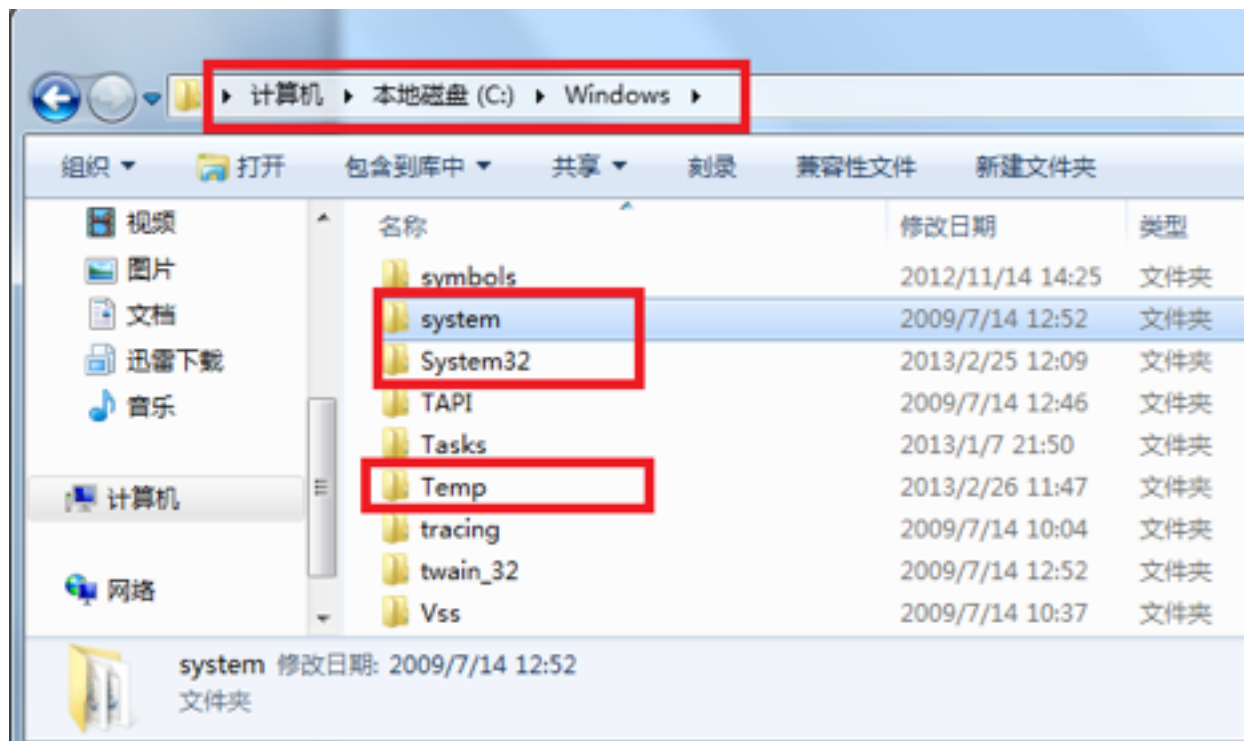
- 其他常被恶意代码利用和篡改的注册表项详见第二章《常见注册表项》一节



被恶意代码利用的Windows关键组件——系统文件

- 关键系统位置

—主要位于系统安装路径下的windows文件夹，其中存放了系统运行必须的程序和驱动，配置信息





被恶意代码利用的Windows关键组件——系统文件夹

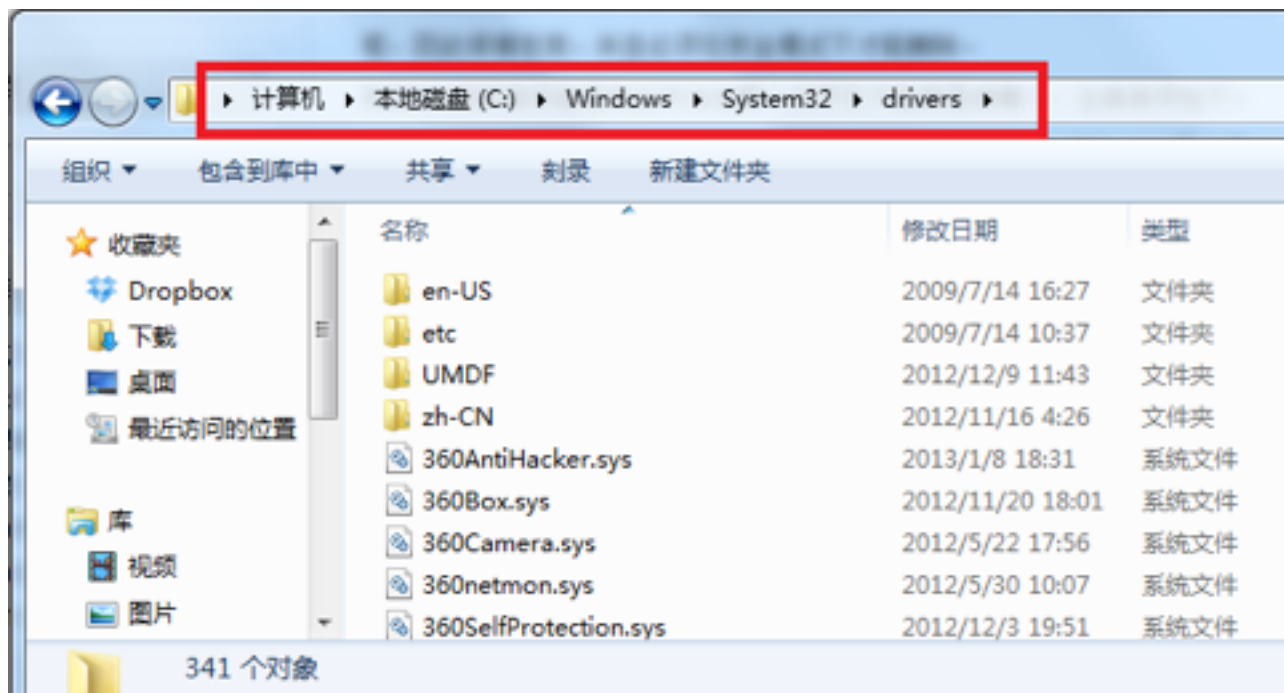
- system和system32文件夹

- 该文件夹下存放着很多关键的系统库文件，这些文件是系统和程序运行时的必须加载的模块
- 病毒可以通过对这些关键文件的替换和注入来实现对正常程序的感染和破坏
- 有时会把病毒的相关文件放置在这些文件夹中，伪装成正常的系统文件



被恶意代码利用的Windows关键组件——系统文件夹

- System32文件夹下的drivers文件夹中存放着系统的驱动文件
- 病毒程序有可能会把自己的驱动文件放置进来替换原有的驱动文件





被恶意代码利用的Windows关键组件——系统文件夹

- temp临时文件夹

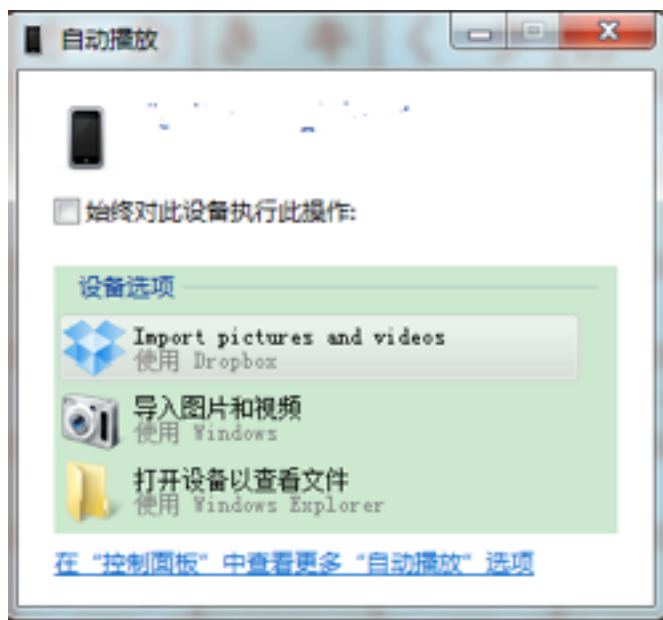
—该文件夹通常会被病毒程序用来放置一些临时的配置文件或者记录日志文件

—由于在该文件夹下创建文件默认所有进程都有写权限，所以这一文件夹经常被病毒程序利用



被恶意代码利用的Windows关键组件——其他

- Windows的自动播放和文件隐藏属性
—Autorun等病毒文件就借助windows的自动播放和文件隐藏属性来启动执行指定恶意程序






被恶意代码利用的Windows关键组件——其他

- Windows 系统漏洞

—Windows 系统漏洞可以被用来植入木马，病毒等
恶意程序

<input checked="" type="checkbox"/>		高危漏洞	这些漏洞可能会被木马、病毒利用，破坏您的电脑，请立即修复。		
<input checked="" type="checkbox"/>		KB982316 - Windows 电话应用程序编程...	2010-08-10	0.29MB	
<input checked="" type="checkbox"/>		KB2641690 - 系统证书吊销列表安全更新	2011-11-09	0.22MB	
<input checked="" type="checkbox"/>		KB2798897 - 虚假数字证书可导致欺骗的...	2013-01-02	0.16MB	
<input checked="" type="checkbox"/>		KB2792100 - IE 累积性安全更新	2013-02-08	13.39MB	
<input checked="" type="checkbox"/>		KB2797052 - Windows 矢量标记语言远程...	2013-02-08	0.48MB	
<input checked="" type="checkbox"/>		KB2778344 - Windows 内核模式驱动权限...	2013-02-11	1.31MB	
<input checked="" type="checkbox"/>		KB2789642 - .NET Framework 4 远程代...	2013-02-11	4.06MB	
<input checked="" type="checkbox"/>		KB2789644 - .NET Framework 3.5.1 远...	2013-02-11	4.63MB	
<input checked="" type="checkbox"/>		KB2790113 - Windows 客户端/服务器运...	2013-02-11	0.94MB	
<input checked="" type="checkbox"/>		KB2790655 - Windows TCP/IP 组件拒绝...	2013-02-11	1.00MB	



被恶意代码利用的Windows关键组件——其他

- Windows的网络端口

—一些后门程序，木马会打开或监听一些比较特殊的端口，用来给服务器传输数据或者发送指令

```
C:\Users\santiago>netstat -ano
```

活动连接

协议	本地地址	外部地址	状态	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	968
TCP	0.0.0.0:443	0.0.0.0:0	LISTENING	856
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:902	0.0.0.0:0	LISTENING	6052
TCP	0.0.0.0:912	0.0.0.0:0	LISTENING	6052
TCP	0.0.0.0:1025	0.0.0.0:0	LISTENING	604
TCP	0.0.0.0:1026	0.0.0.0:0	LISTENING	1040
TCP	0.0.0.0:1027	0.0.0.0:0	LISTENING	1144
TCP	0.0.0.0:1029	0.0.0.0:0	LISTENING	708
TCP	0.0.0.0:1031	0.0.0.0:0	LISTENING	656
TCP	0.0.0.0:7712	0.0.0.0:0	LISTENING	5028
TCP	0.0.0.0:17500	0.0.0.0:0	LISTENING	4280
TCP	0.0.0.0:21049	0.0.0.0:0	LISTENING	1840
TCP	0.0.0.0:34383	0.0.0.0:0	LISTENING	7000
TCP	10.109.34.194:139	0.0.0.0:0	LISTENING	4
TCP	10.109.34.194:2279	199.47.217.149:80	ESTABLISHED	4280



被恶意代码利用的Windows关键组件——其他

- winsock.dll
- wsock32.dll
- imm32.dll



本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows组件
- 恶意代码的关键技术
- 恶意代码的检测技术



恶意代码关键技术分类

恶意代码形式	复制	隐藏	传播	控制	自我保护
病毒、蠕虫	✓		✓		✓
木马、rootkit		✓		✓	✓
逻辑炸弹、恶作剧程序		✓			
后门		✓		✓	✓
垃圾信息			✓		
流氓软件/恶意扣费软件		✓		✓	✓
钓鱼			✓		
僵尸网络		✓	✓	✓	✓



- ## ——群转发手段

[百度](#) [新闻](#) [网页](#) [贴吧](#) [知道](#) [音乐](#) [图片](#) [视频](#) [地图](#) [文库](#) [更多>](#)

[请转发给4个群,10秒后再看看你的头像](#)

[请转发给4个群,10秒后再看看你的头像爱 - xc](#)
上一篇下一篇返回列表转发 [请转发给4个群,10秒后再看看你的头像 爱''爱'''...''爱'](#) 转发四个群,
让更多的朋友关注四川地震,希望爱传递,谢谢!! 我们继续 ...
home.51.com/wwwtwwwet1314520/diary/?id... 2012-10-11 - [百度快照](#)

[请转发给4个群,10秒后再看看你的头像爱''''爱''''''''爱''爱'.._..](#)
【评论】 [请转发给4个群,10秒后再看看你的头像爱''爱''...](#) 发表于 2008-05-21 13:42:52[请转发给4个群,10秒后再看看你的头像 爱''爱''....](#)
www.2100book.com/modules/article/rev... 2012-11-1 - [百度快照](#)

[请转发给4个群,10秒后再看看你的头像爱-51.com个人空间](#)
来源:xiao li. [请转发给4个群,10秒后再看看你的头像爱爱爱爱爱爱爱爱爱爱汶川-挺住爱爱爱爱中国加油爱爱爱爱让更多的朋友关注四川地震,希望爱传递,谢谢!](#)
diary.51.com/item/lixiaoli1232002/dl... 2012-10-26 - [百度快照](#)

[请转发给4个群,10秒后再看看你的头像。这是哪个中蛋的发起的](#)
[请转发给4个群,10秒后再看看你的头像。](#) 这是哪个中蛋的发起的只看楼主 收藏 回复 十月里的雨 回复 意粉 喜欢飞的小鱼 我捡个2楼容易吗!该粉的粉,...
tieba.baidu.com/p/860695... 2012-11-4 - [百度快照](#)

[请转发给4个群，10秒后再看看你的头像 - 好友](#)
[请转发给4个群，10秒后再看看你的头像](#) 勤奋是成功者的必要“物” ... 四川南充,南充市,24岁 最好的自己，我可以！ 伪我疯癫(yfsnyy) 红...
home.51.com/chenyuyan415/fr... 2012-9-7 - [百度快照](#)



恶意代码关键技术（隐藏）

- 变形，混淆，加密
- Rootkit技术
 - 文件隐藏
 - 注册表隐藏
 - 网络连接隐藏
 - 进程隐藏
- 不驻留文件系统，注册表
 - 驻留BIOS，MBR，内存
- 寄生于代码，进程，内存中



- 病毒的混淆变形技术

- 目前很多的病毒程序通过混淆变形的方式实现形态和特征的变化，呈现多态性，使得借助特征代码来查杀病毒的安全软件显得束手无策
- 加壳技术目前常用的是压缩壳UPX,加密壳PECompact
- 代码混淆技术也是通过对程序代码的混淆形成不同的版本文件，使得病毒的代码特征呈现无规律性，有效的隐藏自身



- Rootkit的文件隐藏技术

- 在Windows系统中枚举一个目录下的文件通常是使用两个API函数，FindFirstFile和FindNextFile。而这两个API函数实际上对应的内核Native API函数是NtQueryDirectoryFile
- Rootkit在系统的内核挂钩该API函数，当该函数遍历到需隐藏的文件时，从内核中把该文件信息删除，再把Rootkit处理过的信息返回



- Rootkit的进程隐藏技术

- Taskmgr.exe（任务管理器）进程获取系统上的进程列表时，调用ZwQuerySystemInformation的函数，在内核中是NtQuerySystemInformation
- Rootkit通过在内核中挂钩SSDT表来实现对返回的进程结果进行过滤，隐藏指定的进程



恶意代码关键技术（传播）

- 社会工程学手段
 - 电子邮件，即时通信，网站，SNS
- 基于漏洞的传播
 - 漏洞利用代码中包含恶意代码
- 文件捆绑
 - 在下载的破解软件中捆绑恶意代码
 - 手机第三方定制ROM中预置恶意代码
 - 电子邮件附件，自解压缩文件
 - 漏洞利用代码，宏代码



恶意代码关键技术（控制）

- 木马

- 主动监听，等待连接
- 反向连接，绕过防火墙
- 主动接受指令
- 远程控制(网页，邮件，DNS解析，IRC)
- 屏幕监控
- 键盘记录



恶意代码关键技术（自我保护）

- 隐藏
 - 自身隐藏来躲避查杀
- 对抗检测
 - 变形，混淆，加密
- 对抗清除
 - 双守护进程保护
 - 变种更新，对抗查杀



本章内容提要

- 恶意代码的基本概念和发展阶段
- 恶意代码的特征与分类
- 恶意代码的基本结构和机理
- 被恶意代码利用的Windows组件
- 恶意代码的关键技术
- 恶意代码的检测技术



计算机病毒的检测技术

- 特征代码法

—计算机病毒中一般都带有明显的特征代码，该方法是最为普遍的病毒检测方法

—优点：

- 检测准确快速，误报率低

—缺点：

- 从未见过的病毒无法检测，必须不断更新病毒库
- 随着病毒种类的增加，检索时间变长
- 不能检测多态和隐蔽性病毒



• 校验和法

—根据正常文件的信息（名称，大小，内容），计算其校验和，通过检查校验和来判断是否发现病毒

—优点：

— 方法简单，可以发现已知和未知的病毒

—缺点：

— 对文件内容的变化过于敏感，误报率高

— 影响文件运行速度，不能对付隐蔽性病毒



- 行为监测法

—利用病毒的特有行为特征来监测病毒的方法称为行为监测法

—优点：

- 可发现未知病毒，准确的预报未知的多种病毒

—缺点：

- 可能误报警

- 不能识别病毒名称

- 实现有一定的难度



- 软件模拟法

- 又称虚拟执行法，在虚拟机或者虚拟内存中用软件的方式模拟和分析不明程序的运行。
- 这种方式不会对系统造成危害
- 从虚拟机环境中截获文件数据，如果含有可疑的病毒代码，则进行杀毒后还原文件
- 这种方式可以对各类可执行文件内病毒进行查杀



计算机病毒的检测技术

- 比较法

- 注册表比较法

- 病毒通过注册表自动加载，破坏用户配置

- 长度和内容比较法

- 感染文件的长度和内容会发生变化，需要与其他方法配合使用

- 内存比较法

- 病毒驻留内存需要申请空间，通过正常内存占用的比较检测

- 中断比较法

- 通过修改中断向量的方法驻留，调用时激活



- 感染实验法

- 系统中有异常行为时，运行一些正常的程序，若正常程序被感染，则断定系统有病毒
- 这种方式简单实用，利用了病毒最基本的特性感染性，检测出检测工具不认识的新病毒



- 分析法

- 静态分析

- 利用反汇编工具将恶意代码转换为源代码或汇编代码进行分析
 - 发现恶意代码的模块组成，编程技巧，感染方法，标示特征代码

- 动态分析

- 在恶意代码执行的情况下，利用程序调试工具进行跟踪和观察，确定工作过程