一台图灵机包括 4 个部分：一条无限长的纸带、一个读写头、一个规则集合（程序）和一个状态集合（数据）。当图灵机能够把规则集合当作状态集合来读写时，就会发生很多怪诞的现象，比如图灵机可以自己复制自己！

冯·诺依曼在实现电子计算机时，忽略了图灵机模型中对程序和数据的区分，将程序（规则集）和数据（状态集）放在了同一个物理设备——内存中。因此，现代电子计算机对图灵机模型的实现存在着天然的瑕疵。由于没有明确地区分内存中的程序指令（规则）和普通数据（状态），当年对图灵机自我复制的预言频繁地被黑客攻击所验证，蠕虫的自我复制与传播就是一个生动的例子。

漏洞的万源之本就来自于冯·诺依曼机这种“色即是空，空即是色”的对待代码和数据的态度。高级的变形病毒、软件加壳与脱壳技术等都是基于程序指令可以在运行时当做普通的内存数据进行动态读写的缺陷；堆栈溢出攻击中 shellcode 的执行则是基于计算机错误地把存放在堆栈中的普通内存数据当做程序指令而使用的缺陷；此外，跨站脚本攻击、 SQL 注入攻击等也都是利用计算机把数据和代码混淆这一天然缺陷而造成的。

虽然加强输入验证、分析数据流、分析控制流等方法在增强系统安全性方面起到了一定效果，但总有种“治标不治本”的味道。彻底杜绝黑客攻击需要在计算机体系架构上修复混淆使用数据与代码这一缺陷，而且微软天才的工程师们已经发现了这一点。

在过去的十年中，微软在提高操作系统的安全性方面做着不懈的努力。从 Windows 98 到Windows XP，从 Windows XP 到 W indows Vista，再到最新的 Windows 7，每个新版本的发布都会带来安全性质的飞跃。

从普通用户角度来看，微软在安全方面逐步做了如下几点增强。

（1）增加了 Windows 安全中心，提醒用户使用杀毒软件、防火墙，以及下载最新的安装补丁等。

（2）为 Windows 添加 PC 端的防火墙。

（3）未经用户允许，大多数的 Web 弹出窗口和 Activex 控件安装将被禁止。

（4） Internet Explorer 7 中增加了筛选仿冒网站功能，具有了钓鱼网站过滤器(Phishing Filter)的新功能。

（5）添加 UAC (User Account Control，用户账户控制)机

制，可以防止恶意软件和间谍软件

在未经许可的情况下在计算机上进行安装或对计算机进行更改。

（6）集成了 Windows Defender，可以帮助阻止、控制和删除间谍软件以及其他潜在的恶意软件。

在这些安全功能的保护下，我们操作系统的安全性大大提高了，但是微软所做的工作还远远不止于此。微软还在普通用户看不到的内存保护方面做了很多的工作，下边我们就来看看微软十年间都是如何提高内存保护的安全性。

（1）使用 GS 编译技术，在函数返回地址之前加入了 Security Cooki e，在函数返回前首先检测 Security Cookie 是否被覆盖，从而把针对操作系统的栈溢出变得非常困难。

（2）增加了对 S.E.H 的安全校验机制，能够有效地挫败绝大多数通过改写 S.E.H 而劫持进程的攻击。

（3）堆中加入了 Heap Cookie、 Safe Unlinking 等一系列的安全机制，为原本就困难重重的堆溢出增加了更多的限制。

（4） DEP（Data Ex ecution Protection，数据执行保

护）将数据部分标示为不可执行，阻止了栈、堆和数据节中攻击代码的执行。

（5） ASLR（Address space layout randomization，加载地址随机）技术通过对系统关键地址的随机化，使得经典堆栈溢出手段失效。

（6） SEHOP（Structured Exception Handler Overwrite Protection， S.E.H 覆盖保护）作为对安全 S.E.H 机制的补充， SEHOP 将 S.E.H 的保护提升到系统级别，使得 S.E.H 的保护机制更为有效。

傻眼了吧！事实就是这样，微软在我们看不到的地方已经做了很多保护操作系统的工作，将系统的安全性给予了最大限度的提升，这些安全技术也应用在 Windows 2003、Windows 2008等服务器的操作系统上。如果以安全性为衡量指标对 Windows 家族进行分级的话，Windows XP SP2 以前的操作系统致力于系统的稳定性，忽略了系统的安全性，在这之前的系统可以归为一级；在 Windows XP SP2、Windows 2003 系统中加入了独特安全性设计，在安全性上较前辈有了很大的提高，因此它们属于同一级别； Windows V ista、 Windows 2 008 和最新的 Windows 7等操作系统

中加入了更多的安全机制，从安全性来看它们也是目前 Windows 家族中安全级别最高的。 Windows XP SP2 以后的各版本内存保护机制汇总如表 9-1-1 所示。

表 9-1-1 Windows 安全机制汇总

表 9-1-1　Windows 安全机制汇总

	XP	2003	Vista	2008	Win 7
GS					
安全 C ookies	√	√	√	√	√
变量重排	√	√	√	√	√
安全 S.E.H					
S.E.H 句柄验证	√	√	√	√	√
堆保护					
安全拆卸	√	√	√	√	√
安全快表	×	×	√	√	√
Heap Cookie	√	√	√	√	√
元数据加密	×	×	√	√	√

续表

	XP	2003	Vista	2008	Win 7
DEP					
NX 支持	√	√	√	√	√
永久 DEP	×	×	√[1,2]	√	√
默认 OptOut	×	√	×	√	×
ASLR					
PEB，TEB	√	√	√	√	√
堆	×	×	√	√	√
栈	×	×	√	√	√
映像	×	×	√	√	√
SEHOP*					
S.E.H 链验证	×	×	√[1]	√[0]	√

微软引入的这些安全机制成功地挫败了很多攻击，使得能够应用于 Windows 的漏洞大大减少了。但是，智者千虑必有

一失，在一些特定的攻击场景中，采用一些高级的漏洞利用技术，这些安全机制还是可以被绕过的。 2008 年，Alexander S otirov 和 Mark Dow d 就发表一篇关于 Windows 安全机制的文章“Bypassing Browser Memory Protections”，文中总结了 Windows 各种安全机制及其突破方法。

接下来我们将在前辈们的研究基础上一一介绍这些安全机制和黑客们对付这些安全机制的奇思妙想，带您回顾微软工程师与黑客之间斗智斗勇的故事。