木马雪崩到APT的关联与必然

——对2006年一次技术报告的反思

■安天实验室/江海客

编者按：这是作者在第一届全国网络与信息安全防护峰会（XDEF，武汉大学，2012）上的演讲。根据会议录音整理，有删节。

温故 2006

——这是一个充满预言、谶语和诅咒的年代，这是一个人人皆神、诸佛俱死的年代，我自己也曾因某个只言片语的应验，找到先知先觉的感觉，但推敲起来才发觉自己的狂悖。因此，今天我想认真回顾一篇2006年所做的技术报告，进行一场反思。

2006年对于整个反病毒的发展来说是一个特殊的年份。病毒从最早的狭义的感染式恶意代码的定义，已经开始成为涵盖了蠕虫、木马的统称，而蠕虫被作为研究和产业热点的关注度也已经开始下降，木马也开始呈现爆炸式的增长趋势。2006年全年产生的恶意代码总量比1986年到2005年这20年间所产生的恶意代码总数还要多。

2006年9月25日，笔者在武汉大学做了题为《后冷战时代的病毒捕获体制》的报告，其技术关键词为“木马、蜜罐、旁路捕获、未知检测……”报告的观点为“当前木马与AV之间的对抗已经从辨识对抗、查杀对抗进入到体系对抗阶段”，笔者称之为“‘后冷战’时代”。今天的报告正是从对此的反思开始，因此称其为“温故2006”。

六年前我们认为：木马数量呈几何级数增长的趋势，而溢出技术、驱动技术、流文件技术及信息伪装技术等众多黑客技术都开始被应用到木马程序编写中。木马的发展对反病毒的挑战表现于：数量失控、黑客技术、伪装技术和专有性应用等。

该报告得出“木马动摇了反病毒体系的根基”的结论：传统AV技术的根本链路是编制>>流行>>捕获>>处理，而捕获才是AV的根基。AV的机理是以样本满足一定的流行范围或公开发布为基础，立足于后发式的一对一处理。在此情况下，全面捕获已经趋近不可能，分析处理强度已趋近不收敛，必须有

全新的思路作补充。

此前，我们还提交了一份题为《“中国信息安全将崩盘”于木马的结论》的安天内部分析报告。时隔六年，我们的结论是否得到验证呢？

迷失 2012

——2012，一个让思想者茫然，行动者迷失的年份。

2000~2012 数据回溯

我们分别选取了2000年10月24日、2006年11月10日、2012年11月27日三个时间节点的恶意代码分类数量统计，用以对比分析恶意代码的发展趋势。为了保证数据的可信度，我们并未采用己方数据，而是采用了消重后的某国际知名厂商的病毒库中的病毒名称列表。

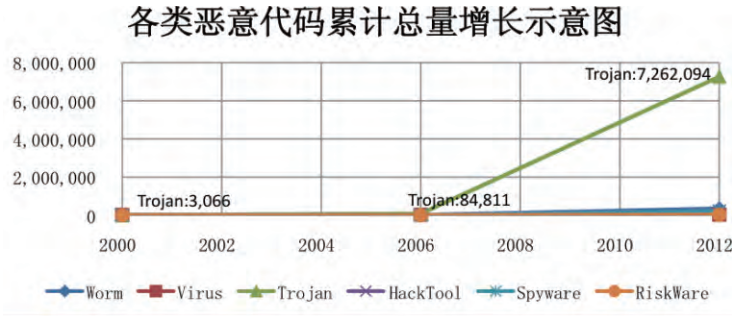

图1 选取2000、2006、2012三个时间点的恶意代码累计总量增长示意图

图2 2000、2006、2012三个时间点的恶意代码分类比例构成

以上数据显示，从2006年到2012年间出现的恶意代码中，比例最大的显然是Trojan。其数量从2006年的8.4万余种增长到2012年的726万余种，而其他类型的恶意代码尽管均有不同程度的增长，但却完全被木马的增长掩盖掉了。那么我们的“木马崩盘”的预言是正确的吗？

失效的预言

确实存在木马数量爆炸式的增长并未得到遏制的事实。但我们不能不面对两个问题：

第一，信息安全整体崩盘了吗？没有。网银、网游等产业依然在快速成长，用户的安全体验实际上在提升。

第二，反病毒的根基动摇了吗？同样没有。我们看到的是整个信息社会和信息安全产业在攻击威胁的增长中所呈现出的一种异常旺盛的发展和适应能力，不但没有崩盘，反而是壮大发展了。

信息安全的状况并非是随着恶意代码数量的激增而急剧恶化的，而是呈现出一种微妙的平衡态，这是值得反思的。因此我们是做出了误判，但我们更大的误判还不在这一点，而是我们过度地思索了数量膨胀带来的挑战，却没有展开更辩证的思考，导致没有把握住今天的威胁——没有对类似APT（高级可持续威胁等趋势）给予足够的预判和警惕。Flame、Duqu、Gauss和Stuxnet，它们既不是木马，不是蠕虫，也不是僵尸网络，而是APT。我们最大的失误不是没有遏制昨天的威胁，而是没有有效应对今天的威胁。

2006 我们预言了APT么？

在2006年，我们是否对APT这种现象有所感知呢？答案是“有”。在“后冷战”

告中，我们提出了一个词汇“专有化”，并提出“经济利益化和政治利益化促进木马向定向性、专有化发展；从传统的散步行为向定向行为转化，不需要大面积传播也能达到一定目的”的观点。纵观这些文字，我们如果自吹当年就预测了APT是“自恋+自欺欺人”的，因为APT不是一种木马，不是一类单一的恶意代码，它是一整套的攻击方法和资源体系，而当时我们并没有跳出一个传统反病毒工程师的角度来看待这一问题。

2006~2012 失落的度量衡

在2006年那场报告中，我们通过2001年、2005年两次反病毒横向检出率对比，展示了木马时代反病毒检出率显著下降的趋势，并以此数据作为恶意代码进入“后冷战”时代的支撑。但面对类似APT等新兴威胁，传统查杀率统计已经失去了意义，安全厂商能力表现的度量衡又是什么呢？

我们曾经对比2000~2006年若干个流行蠕虫的初始流行时间和厂商感知时间，我们发现虽然其中部分蠕虫出现使安全厂商措手不及，但大部分至少是在24小时内被捕获，有的时间稍长但也是以天来计算。而APT时代，我们对相关恶意代码的感知时间则以年为单位来计算，其中Flame是在初始活动近5年后才被发现的，这是由于投放的定向性、条件触发、自毁等方法导致的攻守不平衡。而另一原因也是境内外能力与信息的不对称，无法得到国外工控等厂商的技术支持。如国内机构在Stuxnet的每一个关键分析节点形成成果的时间都比境外晚一个月到45天左右，其中很多分析点我们都不能独立完成，这里既有投入和水平的因素，也有与境外分析团队信息不对称的原因。

弯路 2006~2012

——误判不是无代价的，误判必然导致错误的导向和行动。分析我们没有有效应对木马时代的原因，或许可以为我们有效应对APT时代提供参考。

我们从目前的角度回顾当时对木马的理解：作为传统的AVER，不仅没有预判APT的产生，对于木马的认识也存在诸多错误，误判必将导致错误的导向和行动。接下来列举我们在木马时代做出的错误的技术决策，或许能给我们今后有效应对APT时代提供有价值的参考。

蜜罐还是终端

恶意代码的最佳捕获方式是什么？在2006年报告中，我们介绍了捕风计划，即基于低成本ARM架构做成仿真蜜罐（Honeypot）或蜜池（Honeypool）。但事实证明，最佳的捕获点就是用户终端本身，终端覆盖能力越强，捕获能力越强。但传统反病毒更看重更具有可控性的主动捕获方式，因此我们选择了“向左转”，部署了大量这样的蜜罐节点，但效果并不理想。

更敏感的启发式还是云

在2006年报告中，我们还提出如何应对木马爆炸式增长的思路，即不断提高启发式的敏感性，提交加权值非常低的文件，以采集更多样本。我们自己也清楚，基于可执行对象无条件上报的思路可能是更有效的，但在当时的情况下，我们依然对带宽和计算成本有比较大的焦虑，同时对这种无条件提交的方法，有很强烈的隐私质疑。所以，最终还是采取了提升未知检测能力和敏感性的方法。

还原还是爬虫

在2006年报告中，我们同时提到基于旁路的恶意代码检测架构，即把一个包检测的体系结构变成一个基于流还原的体系结

构，在流还原的体系结构上串接一个异步引擎，来过滤掉多余的文件，实现上报。实际上，从2005年起，已有大量团队陆续投入到基于网页和二进制文件爬虫的研究中，而我们当时尽管在两方面都做了工作，但在还原捕获和检测方面无疑投入更多。

以上所述的每一个时刻，我们都选择了“向左转”，但实际上正确的道路却在右边。事实上，木马这种爆炸式增长趋势得到了有效的对抗，而正是基于海量终端的无条件上报所构筑出的文件鉴定体系使这种对抗成为可能。这种海量终端云鉴定的能力，取代了传统的分析流水线，成为当前反病毒技术新的核心支撑。传统的基于高质量的特征码，包括高质量启发式检测规则的引擎，如果没有这种技术的辅助支撑，无一不会显得落伍。

同时，可以说从传统的磁盘时代，从Debug再到8086的成长过程中，AV业界的资源一直相对匮乏，并存在诸多局限性。我们也看到传统反病毒工程界的一些原则，确实迫使我们会选择一些高成本、低效果的方案，如基于蜜罐的捕获体系和基于旁路的捕获体系，都是我们想要搭建不对用户和网站产生干扰的环境的体现。当我们认识到“贫瘠的资源局限想象力；狭义的道德感局限策略；传统的惯性局限方法”的时候，我们如何建立新的分析方法来有效应对当前的威胁，并避免下一个误判呢？

思索 2012~201x

——信息安全没有崩盘的原因，是因为应用跑得更快；应用跑得令人心惊的原因，是应用忘记了带上安全伴跑。向前走并没有错，我们只是没有绕过路上的石头！但我们不能因为会绊倒石头拒绝向前！

2000~2012的演化原因

表1 2000、2006和2012的三个统计节点的恶意代码分类统计数据

统计时间	Worm	Virus	Trojan	HackTool	Spyware	RiskWare
2000/10/24	512	21,006	3,066	260	37	0
2006/11/10	8,109	27,760	84,811	4,96	4,899	88
2012/11/27	354,049	29,940	7,262,094	217,502	214,570	25,800

如果对前文图表所使用的2000、2006和2012的三个统计节点的数据进一步跟踪的话，我们可以做出更多的分析和解读。

蠕虫的数量增长其实也很快。从2000年的500多种，到2006年的8000多种，再到2012年11月的35万余种。这个数字增长虽然显著，但却被木马数量的剧增所掩盖。蠕虫的膨胀的势头弱于木马，明显与包括Windows系统的安全性提升，漏洞利用的定向性而导致的溢出工具和载荷分离等因素有关。

而对HackTool（黑客工具）和Spyware（广告件和色情件）来说，其在2006年和2012年的规模竟然都十分接近，从某种意义上来说，它们既有恶意代码的基因，但也是正常应用的近亲，它们的发展速度也正反映了应用成长的速度。由此我们不禁在想，由于木马构造的批量化、以及投放端变换（Poly by Server Side）的趋势，从HASH统计上看，黑白名单的规模可能呈现出一种接近趋势，这会很大程度上重新唤起人们对白名单方法的兴趣。

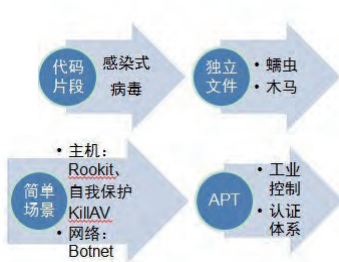

图3 恶意代码的演进

而有趣的是，病毒的数量至今没有突破3万种。特

别是2006年之后，与其他恶意代码类型相比，增量非常少。很多研究者和爱好者将此都简单的概括为：因为病毒感染需要编写者的水平更高，而蠕虫、木马是一个独立文件，因此编写起来更容易。正是因此我们曾经认为病毒到木马是一种技术退化，但如果我们跳出唯技术观来看，就会得出一种相反的结论：感染的方法是把一部分代码注入到其他程序里，其功能空间受到了宿主情况的限制，这种方法局限了攻击者的发挥空间和想象力，因此被淘汰。操作系统的高度复杂已经为独立恶意代码提供了很好的掩盖，不再需要用片段代码注入到宿主进行攻击为掩护。因此我们认为：病毒到蠕虫、木马的过程，不是一个技术退化的过程，而是代码片段能力演进为独立文件功能的能力，这是一种威胁进化。而后又从单一的文件载荷，变成包括多文件、驱动和其他内核技术、Rootkit等隐藏技术在内的复杂的节点环境场景；同时，其网络行为也从单点的窃密回传和一对一的C/S控制等，演化为多层控制、甚至无中心P2P 化的Botnet体制。由此看来，APT所需要的所有技术手段和单

点思想在蠕虫、木马时代均已具备，各种攻击方法也均已成熟。在从最早的几百个字节的感染式病毒，发展到今天的APT攻击的整个过程中，安全事件的复杂度在不断提升。传统信息安全引入复杂巨系统概念，主要是从防御方信息体系的复杂性和规模而言，但当时并未充分估计到单个安全事件本身可以形成高度复杂的体系，当复杂巨系统遭遇高复杂度安全事件，我们的困难也就自然产生。

既然木马时代到APT时代表现出很自然的水到渠成，那么我们为何没有预言APT的发生？我们忽略的关键因素是什么？

在木马时代的地下经济驱动中，各个国家均是地下经济体系和恶意代码的受害者。其基础规则也基于这一共同点搭建。而APT出现的最本质原因是国家和政经集团作为“大玩家”直接介入到网络攻防的游戏体系当中，而攻击的对象也变成了他国政府和其他对立的政经体系。未能预见到威胁与被威胁者主体的突然变化以及这种变化的驱动影响，才是我们没有正确预判APT的出现的原因。

寻找新威胁要素

我曾对传统恶意代码威胁要素进行了概括：第一，入口，恶意代码如何获得权限；第二，介质，恶意代码如何到达节点；第三，格式，恶意代码如何存储。那么新APT威胁要素是

什么呢？APT不是一个简单的恶意代码，而是一个体系，对于这个体系，我们需要思考什么呢？APT之后还会有新的安全威胁形式，那么有没有一个更宏观的视野去分析？

关注新的关键词

第一，生态。

大至整个社会安全保障体系，小到主机环境，都是在遏制恶意代码的旧形态发展的。随着主流操作系统安全性的提升，地址随机化，UAC机制等整套的执行保护体系使这种远程溢出和U盘感染变得越来越困难，主动传播也变得越来越困难，URL欺诈等逐步成为主流，所以我们要关注整个产业链的生态和基础发展。

第二，动机和动力。

在感染式病毒和早期蠕虫活跃的时期，很多人是为了出名而编写病毒和蠕虫，而木马编写多数是利益因素，因此前者往往是个体，后者则形成利益集团。所以在DOS时代，最庞大的感染式恶意代码家族也只有549种，而木马时代，数量最多的木马家族“灰鸽子”变种数超过了27万种。木马之所以如此之多，是因为它们在地下经济体系的驱动下，找到了自身的盈利模式，形成了规模能力。

第三，资源和成本。

我们要充分考虑到编写恶意代码的成本和攻防双方的成本及需要的资源体系。正是国家和政治集团这种强烈的“获利”动机，巨大地影响了生态的能力和无限制的资源可以承担这种巨大的成本，才催生了APT这种集大成者的攻击形式。

有意思的对比

2001年，我们对红色代码发布了仅有一页的BBS预警；2003年，我们对口令猜测类蠕虫做了一个13页的关联分析报告；而今年，我们对Flame的分析报告却长达92页，却感觉只分析了冰山一角。有趣的是，我们却从来没有针对任何一个木马发布长篇的系统的报告，这说明对于不同形态安全事件确实存在不同的应对需求。

AVER工作方法的变迁

在早期感染式病毒为主导的时代，我们面对的是对文件完整性的焦虑，因此，我们要进行人工分析，编写清除病毒的参数、脚本或模块，把文件恢复到感染前的状态；而对于蠕虫，我们面对的是及时性焦虑，类似Slammer在不到十分钟的时间就感染了全球百万台SQL Server服务器等情况，让产业界更关注发现并遏制蠕虫的及时性；而对于木马，我们主要担心其数量级数的膨胀，是一种数量级和处理能力焦虑，我们需要增强海量计算资源的自动分析和终端上的主动防御能力与之抗衡；而对于APT，我们却是难以预见和防御的，我们对APT的恐惧是对其后果的焦虑，那么我们应如何有效应对APT？我们所能做的是如何更快速地感知和深度分析APT，以及对APT的回溯评估。

不同的时代，由于威胁对象和方法的不同，引发了需求和焦虑感的不同，进而导致我们工作方法也随之产生相应地变化。

尾声

比尔·盖茨说：“五年，这就是我们向前能看到最远的时间。”

当想到产业巨子都有如此的时间危机，我更不敢判断我能向前看多远，只知道一定比盖茨近得多。我们难以看远未来，却能更多地反思过去，所以我今天的每一句话，都比六年前谨慎得多，因为我希望六年后，能有勇气回顾今天的话。

（感谢我的同事兰姐、Angel等对此文的贡献）