恶意代码行为分析的交互式视觉辅助工具

非官方中文译本·安天实验室译注

文档信息			
论文题目	Interactive, Visual-Aided Tools to Analyze Malware Behavior		
论文作者	Andre Ricardo Abed Gregio, Alexandre Or Cansian Baruque, Vitor Monte Afonso, Dario Simoes Fernandes Filho, Paulo Licio de Geus, Mario Jino，Rafael Duarte Coelho dos Santos		
发布单位	阿切尔雷纳托研究中心，坎皮纳斯大学，巴西空间研究所		
原文链接/出处	http://www.las.ic.unicamp.br/paulo/papers/2012-ICCSA-gregio-alexandre-vitor-dario-jino-rafael-visualization.malware.behavior.pdf		
论文发布日期	单击此处输入日期。	**译文发布日期**	2014/9/30
论文摘要&关键词	**摘要：**本文展示了一种攻击行为的可视化框架，可以快速识别有趣的入侵行为。对不同家族的 400 多个恶意样本进行分析后的结果显示可以根据这些样本的可视化特征将它们进行分类。 **关键词：**安全数据可视化；恶意代码分析		
译者	安天技术公益翻译组	**校对者**	安天技术公益翻译组
免责声明	• 本译文译者为安天实验室工程师，本文系出自个人兴趣在业余时间所译，本文原文来自互联网的公共方式，译者力图忠于所获得之电子版本进行翻译，但受翻译水平和技术水平所限，不能完全保证译文完全与原文含义一致，同时对所获得原文是否存在臆造、或者是否与其原始版本一致未进行可靠性验证和评价。 • 本译文对应原文所有观点亦不受本译文中任何打字、排版、印刷或翻译错误的影响。译者与安天实验室不对译文及原文中包含或引用的信息的真实性、准确性、可靠性、或完整性提供任何明示或暗示的保证。译者与安天实验室亦对原文和译文的任何内容不承担任何责任。翻译本文的行为		

恶意代码行为分析的交互式视觉辅助工具

Andre Ricardo Abed Gregio[1,2], Alexandre Or Cansian Baruque[2],
Vitor Monte Afonso[2], Dario Simoes Fernandes Filho[2],
Paulo Licio de Geus[2], Mario Jino[2]，Rafael Duarte Coelho dos Santos[3]
[1]巴西，圣保罗州，坎皮纳斯，阿切尔雷纳托研究中心（CTI/MCT），
argregio@cti.gov.br
[2]巴西，圣保罗州，坎皮纳斯，坎皮纳斯大学（Unicamp），fvitor, dario,
paulog@las.ic.unicamp.br, jino@dca.fee.unicamp.br,
orcansian@gmail.com
[3]巴西，圣保罗州，圣约瑟杜斯坎普斯，巴西空间研究所（INPE/MCT），
rafael.santos@lac.inpe.br

摘要：恶意攻击能够破坏信息系统，打破可用性、保密性和完整性的安全原则。攻击者使用恶意程序获取控制权、窃取数据、入侵系统并掩盖痕迹。对恶意代码的动态分析有助于获取其执行路径，判断攻击的破坏程度。对截获的恶意代码进行分析可以为分析人员提供有关其行为的信息，帮助他们重现恶意代码在目标系统中的恶意行为。分析时获取的行为数据包括文件系统和网络活动痕迹。分析人员从庞杂的数据中挑选出与攻击相关的数据是相当费时费力的。我们在此展示了一种攻击行为的可视化框架，可以快速识别有趣的入侵行为。而且，我们分析了不同家族的 400 多个恶意样本，结果显示可以根据样本的可视化特征将它们进行分类。最后，我们将其中一种工具免费开放使用。

关键字：安全数据可视化；恶意代码分析

1. 简介

恶意代码是信息系统最主要的威胁。恶意代码一般通过互联网进行传播，严重危害系统和数据的保密性、完整性和可用性。大多数恶意代码没有特定的攻击目标，它们的胃口很大，目标越多越好，这样攻击者才能更多的获取被入侵系统的控制权，窃取私密数据。然而，很多情况下，恶意代码有特定的攻击目标，其设计的目标就是欺骗受害者，让无辜的用户成为数据泄露的牺牲者，正如其对政府基础设施的攻击一样。无论哪种情况，恶意攻击都会破坏整个网络系统，无论总部还是分支，无一幸免。

当计算机系统被成功入侵后，犯罪取证过程也便开始了。取证过程需要调查清楚攻击发生的方式，数据汇集的端点位置（工具下载和数据发送），系统被攻击期间发生了什么。

就恶意代码攻击而言，收集影响系统的二进制数据或者目标系统被攻击后下载的二进制数据都是十分重要的。这些二进制数据可能会在后续工作中提供一些线索，这些线索能让分析人员对攻击者所使用的攻击技术进行更深入地了解。这种取证工作可以通过在被控制的环境中运行该恶意程序，并监控所有的文件系统和网络活动以追踪恶意代码的行为轨迹。

恶意行为轨迹实际上就是恶意程序在被入侵系统中的事件日志，但是这种数据块可能会很大而且很难进行分析，因为我们既要找到恶意行为的蛛丝马迹，还要对攻击的整体情况进行把握。然而，通过这种分析方式获得的信息可以提供恰当的事件响应以及风险缓和措施。在对大规模的文本数据进行分析时，我们可以引入可视化技术以加快对日志的分析速度，快速锁定特定恶意程序的重要攻击行为，从而更好地了解导致目标系统被入侵的恶意事件链。

本文的主要贡献：

我们开发了两种可视化工具：行为螺旋（behavioral spiral）和恶意时间轴(malicious timeline)。它们能够帮助分析人员观测到恶意软件在进行攻击时的行为。这些工具具有交互的特性，允许用户对不同的恶意行为特征进行观测，并收集每种恶意行为的详细信息。

我们讨论了对恶意代码家族的可视化分类。我们的工具可以基于对已知恶意代码的比对，而对未知恶意样本进行可视化识别。

我们在网络上发布了原型的测试版，以供有识之士免费使用。

2. 相关研究

利用可视化工具来处理文本日志中的庞杂数据，相关研究已经见诸报端。有些研究成果尚未对外公开，其他成果或者不够直观，或者缺乏交互性，也有一些研究的可视化解析难度比较大。

Quist 和 Liebrock 利用可视化技术来了解被编译的可执行程序的行为。他们的 VERA（Visualization of Executables for Reversing and Analysis）框架可以帮助分析人员对可执行文件的执行流有更好的理解，加快逆向工程的进程。

Conti et al.开发的系统可以对二进制和数据文件进行不依赖于环境的分析，通过可视化实现对文件的整体图景和内部结构的快速浏览。在取证环境中，这一系统对于分析未存档格式的文件以及搜索二进制文件的隐藏文本信息，都是十分有优势的。

Trinius et al. 利用可视化技术来研究恶意代码的行为。他们使用树形图和线程图来展示可执行文件的行为，帮助分析人员对恶意行为进行识别和分类。他们的线程图存在过多的重叠信息并且缺乏交互性，因此很容易让分析人员产生困惑，而我们的时间轴允许分析人员对某一恶意代码样本创建的不同进程所衍生的事件链进行预览，还可以放大某些有趣的区域，甚至对被选择的行为进行信息收集和注释。我们的行为螺旋展示的是一种短暂的行为，而他们的树形图包含行为的分布频率，缺乏交互性，数据也过于庞大，而我们仅关注能够导致目标系统发生改变的行为。然而，与我们研究相似的是，他们也无法将每个恶意代码家族进行可视化分类，因为属于某一家族的变种样本可能表现出完全不同的行为特征。

最后，检查入侵检测系统的日志对于识别并掌握网络攻击行为是十分重要的。一些可视化工具可以实现这一目的，每种都各有千秋。利用这些工具，DEViSE（将安全事件可视化的数据交换）框架可以帮助分析人员实现数据在不同工具间的交互，以获得对数据的更好理解。

3. 数据收集

为了实现恶意行为数据的可视化，我们首先需要收集当前流行恶意代码的样本，然后分析并提取它们的行为特征。本节讨论了样本收集和行为提取的方式。

3.1 样本收集

为了收集到流行的恶意代码样本，我们使用了混合的蜜罐技术（低等和中等交互程度）来捕获 MS Windows 系统中恶意二进制文件。蜜罐是被投放的、用于入侵的、高度受控的系统。可以通过蜜罐来引诱攻击者暴露他们的攻击手段和工具。该样本收集架构包含一个 Honeyd 节点，用于将攻击引诱到 Dionaea 系统的某些漏洞端口，该系统实则是在下载恶意样本。通过这种方式，我们在 2010 年捕获到了 400 多个独特的样本，这些样本在本文的研究中被用作测试用的数据集。

3.2 行为提取

为了提取到恶意代码的行为，我们将其在受控的环境中运行并监控其在目标系统中的执行情况。这些行为是基于系统调用的，比如修改系统状态，获取敏感信息，具体包括写入文件、创建进程、修改注册表值、网络连接、创建互斥体等。用于行为提取的动态分析环境是 BehEMOT，这是用于创建日志的系统，日志中的每行信息都代表着受控的恶意代码所表现出的行为，日志的格式为 "*时间戳*、*来源*、*操纵*、*类型*、*目标*"。例如，恶意代码样本 *mw.exe* 创建了名为 *downloader.exe* 的进程，连接到 IP 地址为 *X.Y.W.Z* 的网站下载一个名为 *a.jpg* 的文件，并保存到 *TEMP* 中。BehEMOT 所创建的日志文件就会有下面三行内容：

ts1, mw.exe, CREATE, PROCESS, downloader.exe

ts2, downloader.exe, CONNECT, NET, X.Y.Z.W:80

ts3, downloader.exe, WRITE, FILE, TEMP/a.jpg

因此，我们使用 BehEMOT 行为格式将文本数据输入到我们的可视化工具中，这一点我们将在下一节进行介绍。值得注意的是，恶意代码样本所创建的日志可能会有数千行，可视化工具的使用可以加快人工分析的速度。

4. 用于行为分析的交互式可视化工具

恶意程序的行为可以被解析为一系列相继发生在系统中的行为，包含操作系统交互和网络连接活动。对这些操作进行分析可以还原攻击发生的过程，这样有助于对事件进行整体把握，并掌握系统变化的细节信息，比如被重新编辑的目录、受感染的文件、下载的数据甚至是被隐藏的信息。

通常，反病毒开发者对未知的恶意样本进行分析的目的是为了创建用于检测的签名或启

发算法。这是一个复杂的过程，涉及大量的人工工作，因为分析人员需要搜索到能够将一个样本归类为某一已知类别的信息条目，或创建一个新的类别来标识这一样本。实际上，这一过程越来越复杂，因为各种花样繁多的病毒制造工具层出不穷。有时候，一个样本的分类可能会基于它所创建的互斥体的值，或它所创建的带有特定名称的进程，或它发送到网络中的信息类型。

我们的目标在于开发一种可视化工具，使特定信息的处理和甄别更简便，从而帮助分析人员更好地关注与恶意样本的行为。通过将恶意样本的整体行为进行可视化，并且将分析人员认为可疑或重要的行为进行放大，便可以实现对恶意样本的快速分析。另外，如果分析人员掌握了可视化技术，那么各种公开的动态分析系统所提供的文本报告中的大量信息，就很容易解析了。为了弥补这一空白，我们开发了两种工具，能够将动态分析系统提供的文本报告转换成交互式的可视化行为。

4.1 时间轴和螺旋

恶意代码的行为可以通过简单的 x-y 标绘的方式来呈现，x 轴代表事件，y 轴代表事件信息。X 轴上的时间信息可以是：1）时间发生的绝对时间；2）事件发生的相对时间（从某一特定起始时间算起）；3）事件发生的次序。

事件信息可以使用不同的方法来绘制，通常都是针对某一特定目的而设计。如果已知给定事件发生的频率，严重程序和强度，那么可以用 y 轴的高度来表示这些信息。还可以对事件的额外特征进行不同的标记，并将这些标记信息用于呈现更多的数据维度。当然，还可以应用其他绘图成分来丰富内容，但需要注意的是，过多的信息可能会混淆分析人员的视听，使其无法快速获得有效的信息。

图 1 和图 2 中的例子展示了两种工具的绘制成果。它将带有恶意行为的文件按照事件的时间戳进行了排序，并采用了一种简单的交互式界面。所有的时间序列被绘制在两个平面上，在上面的平面中所有的点都被绘制出来了，x 轴代表事件发生的次序，y 轴代表与时间相关的行为。而且，事件根据进程 ID 被绘制成不同的颜色。例如，如果与恶意代码相关的进程创建了两个子进程，并且需要已知进程提供相关服务，则恶意代码、第一个子进程、第二个子进程以及正在运行的进程会在图像上表现出不同的颜色。并不是所有被监控的行动都是单个恶意样本的行为，因此 y 轴根据当前被捕获的行为轨迹而有所不同。

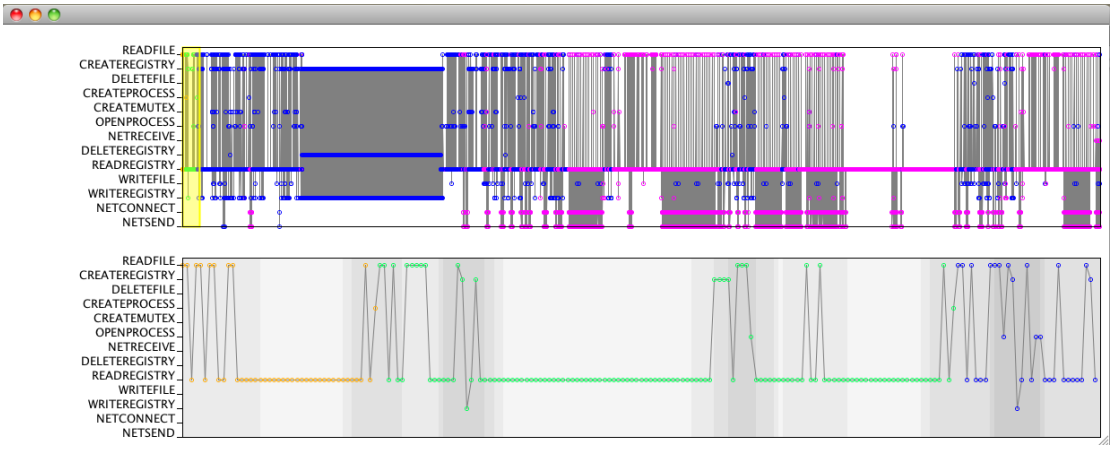

图 1：呈现恶意事件的时间轴和螺旋工具

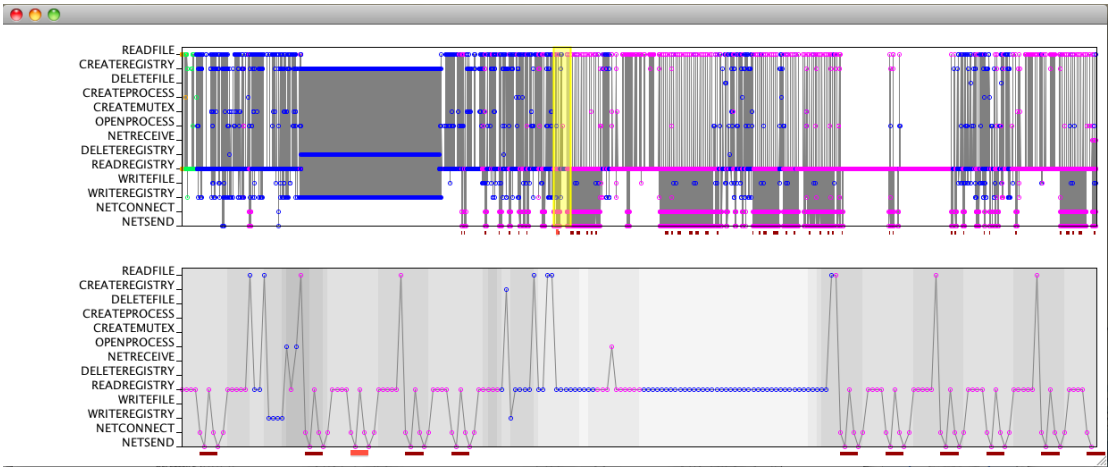

图 2：事件顺序（亮红色）和自动匹配的事件（暗红色）

工具所绘制的图像也是具有交互性的。因为在绘制图的顶部有很多事件，很难看清楚事件的排列顺序，所以一个半透明的黄色区域还可供用户选择。用鼠标拖拽这一区域，这一部分就会在绘制图的底部平面被放大显示。绘制图的底部还包含着灰色背景的，显示操作类型丰富多样性的信息（暗色背景表示多样程度高，浅色背景表示相似程度高）。

4.2 恶意螺旋

该工具的设计目的是以一种螺旋的形式来呈现恶意攻击行为。这种方式可以对恶意样本行为的整体图景进行描述，也可以实现不同恶意样本的快速比对。

这种可视能力的实现可以帮助分析人员对恶意行为进行各种形式的观察。我们提供的不仅仅是攻击图景，还有对某些行为模式的识别能力，这将有助于对恶意样本进行分类。

表 1 显示了被监控且用于生成行为日志的恶意操作和恶意代码类型，以及代表这些操作和类型的图标。

5. 对结果的测试与分析

尽管我们开发的两种工具具有相同的日志格式，它们的目的却不尽相同。时间轴和放大工具可以用来对行为进行多种形式的观察，从而掌握进程的数量、行为以及类型。螺旋工具则被用来对同一恶意家族的恶意代码进行识别，同时采用一种图像的形式来呈现行为的具体信息。

在本节，我们展示了螺旋工具作为一种可视化词典，如何进行同一家族恶意代码的识别工作。当前研究的原型为文献 9，其中附有更大的截图。我们当前的研究提取了 425 个恶意样本（见 3.1 节）的执行行为，并对其进行了动态分析（见 3.2 节）。

数据集中的所有恶意样本都是当前流行的，包含 31 个家族的变种。用 ClamAV 反病毒引擎对其进行扫描后发现 94 种未知样本。在文献 9 中，我们可以了解到每个已知样本的行为螺旋的图片以及各自的日志信息。

通过对生成的螺旋进行观察，我们意识到可以通过它们的可视化行为对其进行分类。而且， 这些来自不同家族的恶意样本中，同一类别的样本呈现出相似的行为模式，而非同一类别的样本则呈现出不同的行为模式。可视模式中呈现出的这种区别表明，可以将聚类算法、人工智能和数据挖掘技术应用到我们的日志中，从而对恶意代码进行基于行为相似性的分类。

在图 3 中，我们选择了两个木马家族 Pincav 和 Zbot，每个家族选取了三个样本。即便同一家族的恶意样本表现出不同的行为，但仍然存在能够将它们划分到同一类恶意代码中的行为模式。

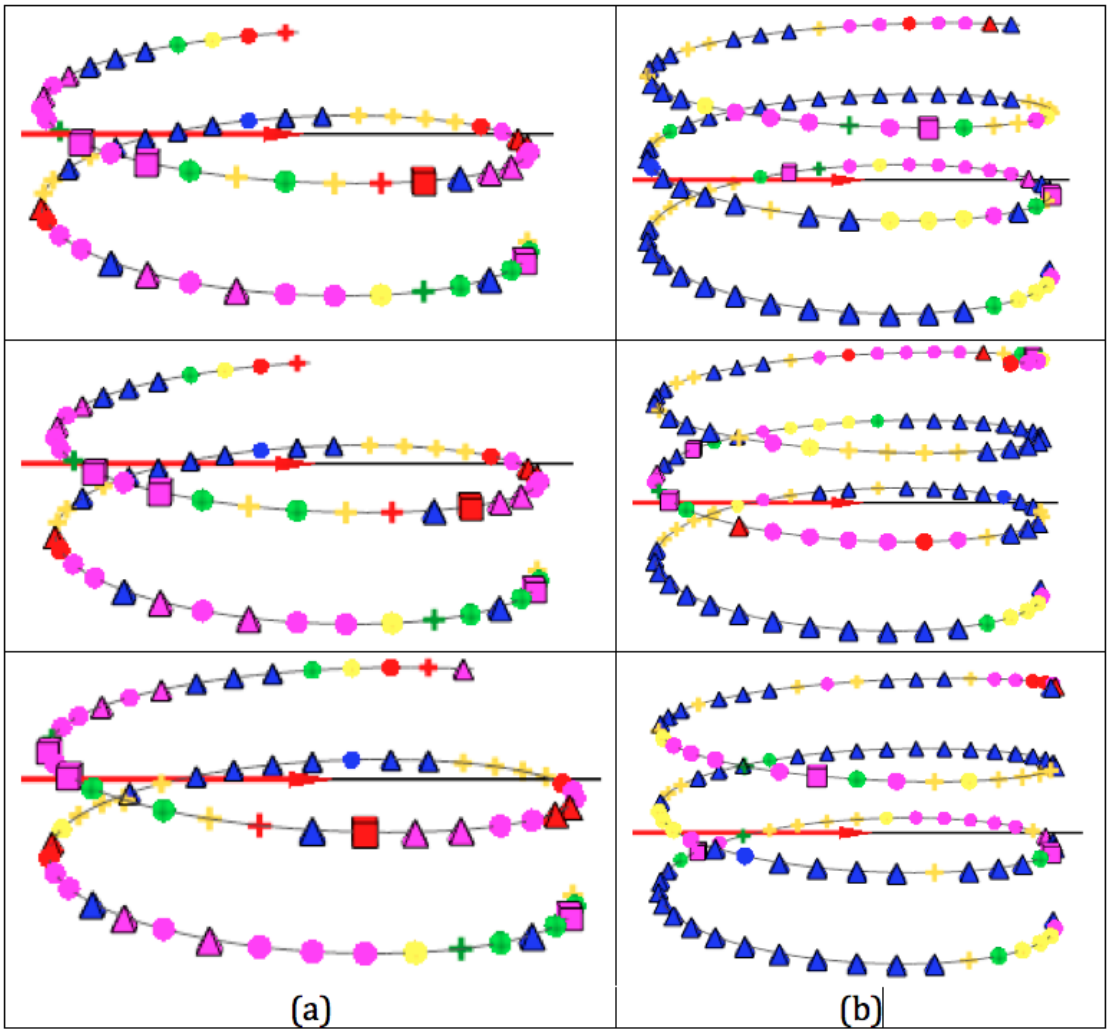

图 3：Pincav（a）和 Zbot（b）木马家族三个样本的行为螺旋

另一个有趣的事实是 ,如果某一恶意样本试图进行更多的网络连接 ,而同一家族(比如，恶意扫描) 的其他样本则未表现出此类行为，或者试图连接的次数很少甚至停止了行动，那么就很容易将同一家族不同变种行为的相似性表现出来，如图 4 所示，其中包含 *Allape* 家族的三个样本。

图 5 展示了不同恶意家族——蠕虫 *Palevo* 和 *Autorun* ,木马 *Buzus* 和 *FakeSSH* 的行为螺旋。我们可以将同一家族（ *Palevo* ，*Autorun* 和 *FakeSSH* ）不同样本间微小的行为差异可视化。*Buzus* 家族中，前两个样本的行为明显不同于最后一个样本的行为，将这些样本彼此区分开来的是一种自动分类框架。然而，在图 6 中，我们也意识到，一个被反病毒引擎标识为“未知”的样本，其可视化行为与 *Inject* 家族的一个样本极其相似。

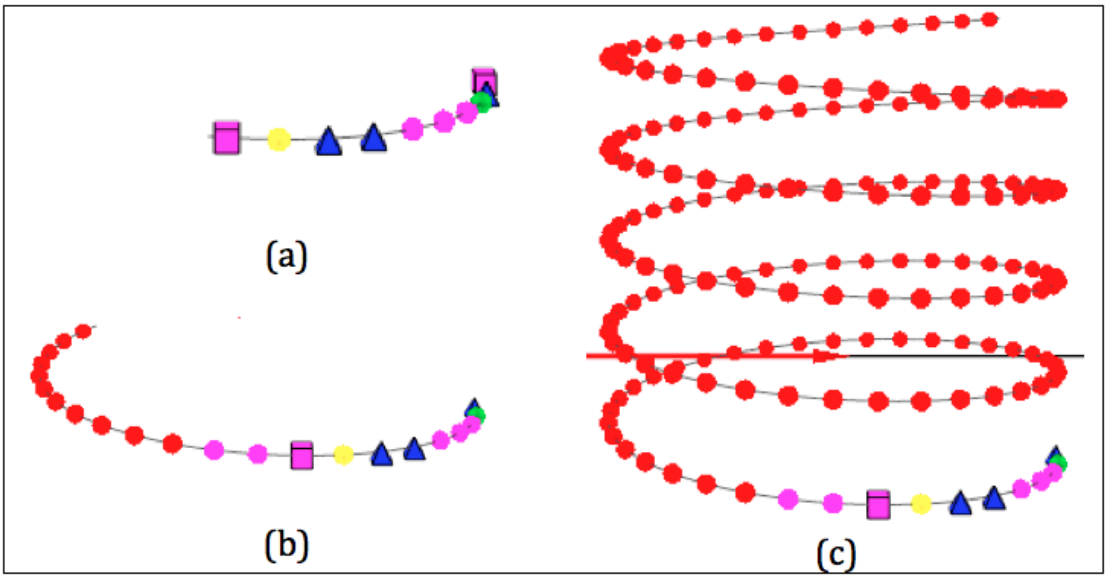

图 4：Allape 蠕虫的变种:(a)未能连接到网络并停止行动的样本;(b)进行了短暂网络扫描的样本;(c)进行了大规模扫描的样本，每个红球代表连接到一个不同的ＩＰ地址。

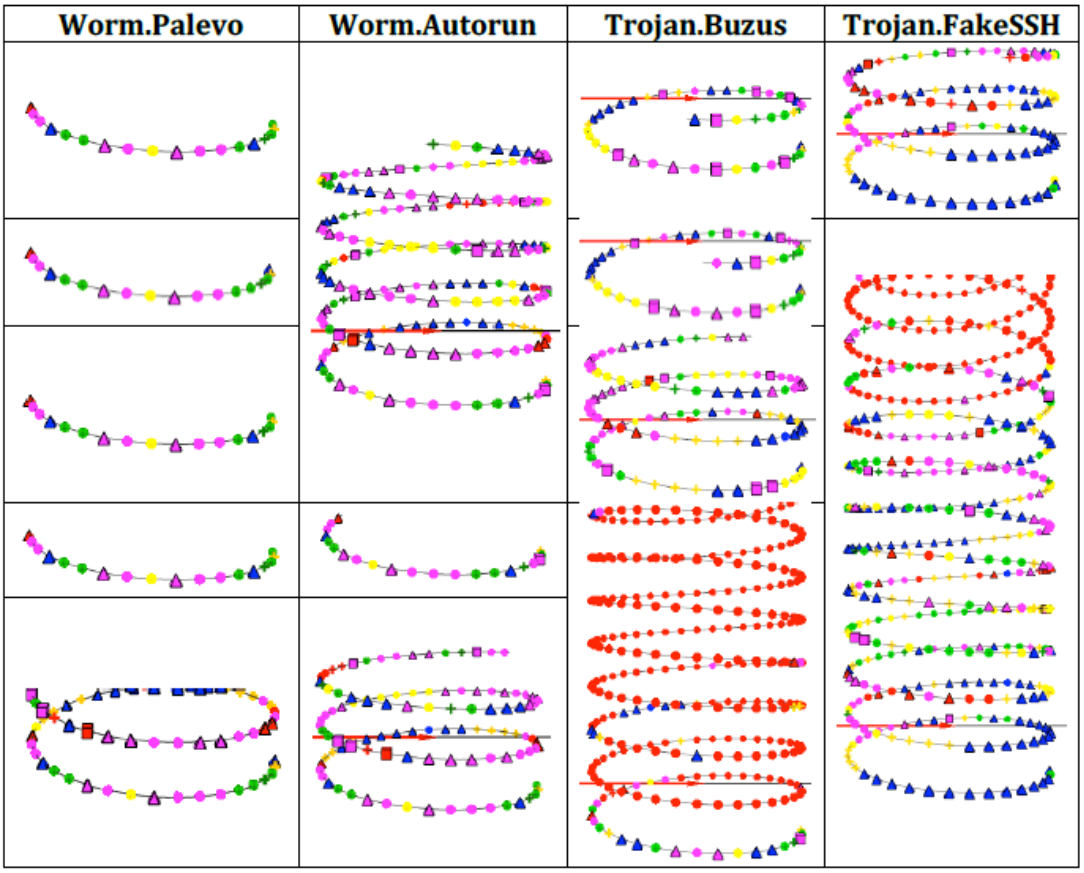

图 5：从 4 个不同恶意代码家族提取出的可视化行为

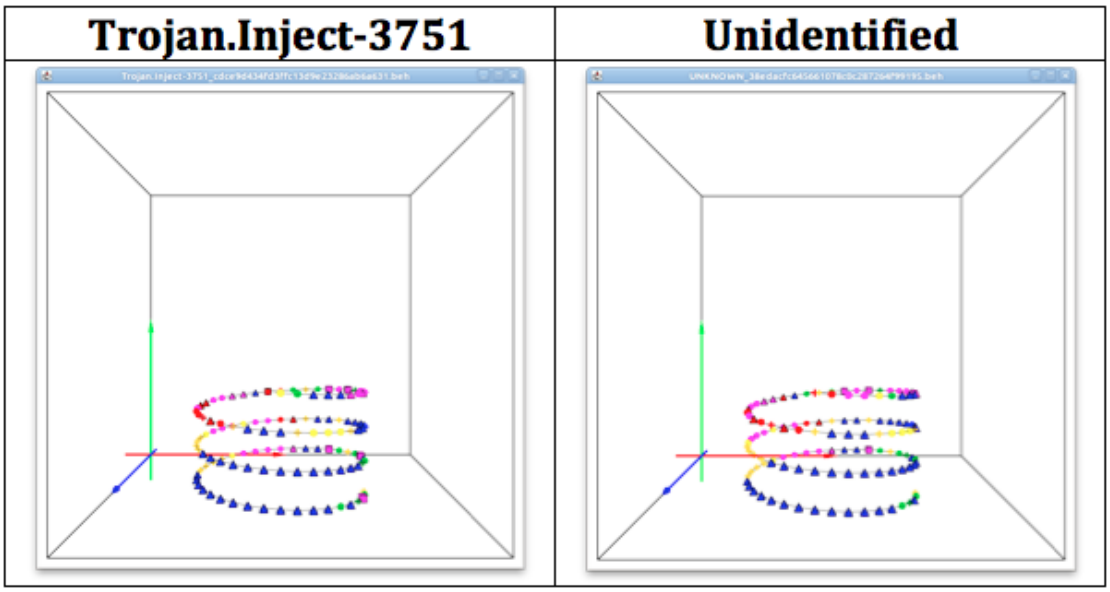

图 6：将未知的恶意代码样本（右）进行可视化操作后，其被归类为已知的 *Injcet* 木马家族（左）

6. 结论

本文提出了两种交互式的视觉辅助工具，其作用是增强恶意代码分析的有效性，为安全分析人员提供有关恶意代码行为的整体视角。而且， 这些工具可以实现对大量日志信息的分析，对有趣恶意行为的注释和强调，对攻击模式的搜索，对目标系统损坏程度的深入了解以及对恶意样本的可视化比对。因此，我们是有可能对病毒家族进行可视化区分的，这表明，在未来我们可以采用一种自动化技术来对行为数据进行分类和挖掘。而且，我们还可以将行为与其他恶意代码相似的样本进行部分可视化。我们还将在未来的工作中开发一个行为数据库，这一数据库能够为时间轴/放大器工具的注释过程带来某些启迪，并提供同时加载多种日志，将数个螺旋工具平行可视化的性能，最后，还会将一种分类算法整合到螺旋工具中，以便自动识别高度相似的样本以及样本行为。

参考文献

1. S. Buehlmann and C. Liebchen. Joebox: a secure sandbox application for windows to analyse the behaviour of malware. http://www.joebox.org.

2. Clam antivirus. http://www.clamav.net.

3. G. Conti, E. Dean, M. Sinda and B. Sangster. Visual Reverse Engineering of Binary and Data Files. *Proceedings of the 5th international workshop on Visualization for Computer Security* (VizSec), 2008, pp. 1-17.

4. S. G. Eick, J. L. Ste_en and E. E. Sumner, Jr. Seesoft— A Tool for Visualizing Line Oriented Software Statistics. In *IEEE Transactions on Software Engineering*, vol. 18, no. 11, pp. 957-968, 1992.

5. A. R. A. Gr_egio, I. L. Oliveira, R. D. C. dos Santos, A. M. Cansian and P. L. de Geus. Malware distributed collection and pre-classification system using honeypot technology. P*roceedings of SPIE*, vol. 7344, pp. 73440B-73440B-10, 2009.

6. A. R. A. Gr_egio, D. S. Fernandes Filho, V. M. Afonso,R. D. C. dos Santos, M. Jino and P. L. de Geus. Behavioral analysis of malicious code through network traffic and system call monitoring. *Proceedings of SPIE*, vol. 8059, pp. 80590O-80590O-10, 2011.

7. The Honeynet Project. Dionaea. http://dionaea.carnivore.it.

8. C. Kruegel, E. Kirda and U. Bayer. Ttanalyze: A tool for analyzing malware. In *Proceedings of the 15th European Institute for Computer Antivirus Research (EICAR 2006) Annual Conference*, 2006.

9. MBS Tool. Malicious Behavior's Spiral - Beta version. http://www.las.ic. unicamp.br/~gregio/mbs

10. N. Provos and T. Holz. Virtual Honeypots: from botnet tracking to intrusion detection. *Addison-Wesley Professional*, 2007.

11. N. Provos. Honeyd - A Virtual Honeypot Daemon. In *10th DFNCERT Workshop*,

2003.

12. D. Quist and L. Liebrock. Visualizing Compiled Executables for Malware Analysis. *Proceedings of the Workshop on Visualization for Cyber Security*, 2009, pp. 27-32.

13. H. Read, K. Xynos and A. Blyth. Presenting DEViSE: Data Exchange for Visualizing Security Events. *IEEE Computer Graphics and Applications*, vol. 29,pp.6-11, 2009.

14. ThreatExpert. http://www.threatexpert.com.

15. P. Trinius, T. Holz, J. Gobel and F. C. Freiling. Visual analysis of malware behavior using treemaps and thread graphs. *International Workshop on Visualization for Cyber Security* (VizSec), 2009, pp. 33-38.