Hidost a static machine-learning-based detector of malicious files

* Nedim Šrndić

Malicious software, i.e., malware, has been a persistent threat in the information security landscape since the early days of personal computing. The recent targeted attacks extensively use non-executable malware as a stealthy attack vector. There exists a substantial body of previous work on the detection of non-executable malware, including static, dynamic, and combined methods. While static methods perform orders of magnitude faster, their applicability has been hitherto limited to specific file formats.

This paper introduces Hidost, the first static machine-learning-based malware detection system designed to operate on *multiple file formats*. Extending a previously published, highly effective method, it combines the logical structure of files with their content for even better detection accuracy. Our system has been implemented and evaluated on two formats, PDF and SWF (Flash). Thanks to its modular design and general feature set, it is extensible to other formats whose logical structure is organized as a hierarchy. Evaluated in realistic experiments on timestamped datasets comprising 440,000 PDF and 40,000 SWF files collected during several months, Hidost outperformed all antivirus engines deployed by the website VirusTotal to detect the highest number of malicious PDF files and ranked among the best on SWF malware.

从个人计算的早期开始，恶意软件即恶意软件一直是信息安全领域的持续威胁。最近的有针对性的攻击广泛使用不可执行的恶意软件作为隐身攻击媒介。在检测不可执行的恶意软件方面存在大量以前的工作，包括静态，动态和组合方法。虽然静态方法执行速度更快，但它们的适用性迄今仅限于特定的文件格式。

本文介绍Hidost，这是第一个基于静态机器学习的恶意软件检测系统，旨在以*多种文件格式*进行操作。扩展了以前发布的高效方法，它将文件的逻辑结构与其内容相结合，以实现更好的检测准确性。我们的系统已通过PDF和SWF（Flash）两种格式实施和评估。由于其模块化设计和通用功能集，可扩展到逻辑结构按层次结构组织的其他格式。在包含440,000 PDF和40,000 SWF文件的时间戳数据集的实际实验中评估，Hidost胜过了网站VirusTotal部署的所有防病毒引擎，以检测最高数量的恶意PDF文件并在SWF恶意软件中名列前茅。

用于在企业和个人之间交换文件的众所周知格式的文件感染。这种感染为攻击者提供了以下好处：

1. 诱使用户打开文档比启动可执行程序更容易。

1. 2。

近年来，文档查看器中出现了一系列新的漏洞，这是因为文档格式的复杂性导致了其高度复杂性。

1. 3。

文档格式的灵活性和多功能性为混淆嵌入式恶意内容提供了充足的机会。

The favorite formats used by attackers are PDF ，Non-executable files are especially popular as a means for *targeted attacks*非可执行文件作为有*针对性的攻击*手段尤其受欢迎

提出了一种*新的静态分析方法 ， 检测恶意非可执行文件的主要困难是理解复杂格式的必要性*它有可能在各种格式中更具可移植性。我们的实验证明，通过合并适当的格式解析器，它可以应用于PDF和Flash文件。

1.静态方法

PDF是一种复杂的文件格式，只有完整的PDF解析器才能正确解除混淆，。在这方面，n-gram方法过于简单。第一种使用PDF解析器的方法是PJScan [ [10](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR10)]。它基于嵌入在PDF文件中的JavaScript代码的词法属性使用

异常检测，随后提出了两种简单的基于学习的方法，Malware Slayer [ [11](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR11) ]和PDFrate [ [12](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR12) ]，均使用基于PDF文件原始字节的启发式特征。PJScan是迄今为止唯一提出的深层方法。浅层方法的一个常见弱点是PDF格式物理结构伪造的相对容易性，以PDFrate [ [13](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR13) ] 为例。*所有*纯静态方法的一个共同缺点是它们无法检测动态加载的威胁，例如，当分析的文件不包含攻击代码，而是通过网络或其他文件加载它时。

2.动态方法

虽然动态方法比静态方法更为准确，但它们的执行时间使其不适合实时检测繁忙网络上的恶意文档。此外，构建和维护动态检测器能够模拟每个版本的易受攻击的软件产品，并结合每个支持的操作系统和库的每个版本，这是一项成本高昂且技术上具有挑战性的任务。另一方面，从检测器中省略目标软件的一个组合就足够了，并且为该特定版本设计的威胁将不会被检测到。

3.动静结合

为了达到静态方法的速度*和动态*方法的精度，随后开发了*静态和动态组合*方法。MDScan执行静态JavaScript提取和动态代码执行[ [21](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR21) ]，由于其设计，它仅适用于单个版本的Adobe Reader上的恶意软件检测，其动态组件需要几秒钟才能运行。

4. 全自动方法

Nissim等人 建议使用*主动学习*方法，其中人类专家手动标记机器学习算法的有趣样本，目标是使检测器保持最新的威胁[ [23](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR23) ]。他们概述了一种结合了签名检测和目前描述的多种方法的设计，但将其实施和评估留给未来的工作。有关许多提及的PDF恶意软件检测方法的更详细的调查，我们请读者参考[ [23](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR23) ]

贡献：

所提出的检测方法基于对分级文档结构的分析，并且此后缩写为Hidost，是使用*逻辑结构*来表征恶意和良性PDF文件，PDF逻辑结构是由PDF标准定义的高级构造，它将基本PDF构建块组织为功能性文档。

Hidost继承了SL2013的所有优点。它在PDF文件中保持了近乎完美的检测性能和高吞吐量，该文件专门针对繁忙网络上的集中部署SL2013进行了量身定制。作为深度静态方法的进一步优势，Hidost不受PDF混淆和物理结构伪造的影响。

开发了*结构路径整合*（SPC），这是一种用于合并相似特征的技术，更好地保留逻辑结构的语义并减少特征集对特定数据集的依赖性。

SPC的好处有三个：

1. 逃避的攻击面减少;
2. 随时间变化的特征集是有限的;
3. 功能的数量急剧减少。

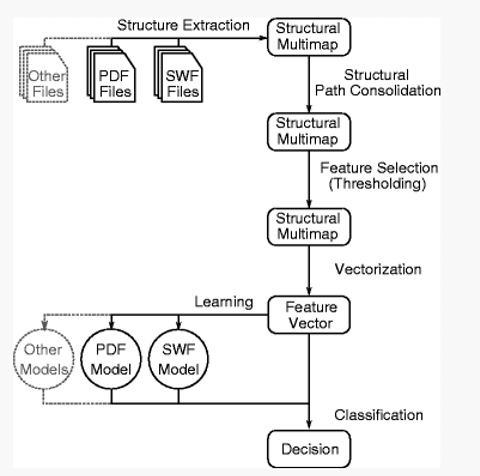
最重要的是，本文介绍了一种新颖的Hidost系统设计，使其能够推广到多种不相关的文件格式。Hidost不仅考虑了文件的逻辑结构，而且还考虑了其​​内容

总之，本文的主要贡献如下：

* 基于静态机器学习的恶意软件检测器Hidost，这是第一个基于逻辑结构和内容适用于不同文件格式的系统。
* Hidost对PDF和SWF两种格式进行了实验性评估，旨在反映恶意软件检测器的操作环境，该文件在440,000个PDF文件和前所未有的40,000个SWF文件的数据集上执行。在我们的评估中，Hidost在PDF上的性能优于VirusTotal上的所有防病毒引擎，并在SWF文件中名列前茅。
* Hidost针对两种文件格式（PDF和SWF）的原型实现，作为开源软件发布。
* 复制这项工作所需的源代码，包括作为开源软件发布的实验和绘图。
* 数据集需要重现这项工作。

我们的设计明确地将格式特定的处理步骤与检测方法分开。所提出的方法是作为一个研究原型来实现的，其特征提取子系统被公开为开源软件[ [29]](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR29)]，Hidost的系统设计如图[8所示](https://link.springer.com/article/10.1186/s13635-016-0045-0" \l "Fig8)：

Hidost有六个主要阶段：结构提取，结构路径合并，特征选择，矢量化，学习和分类。



结构抽取将特定格式的结构特征转换为表示结构层次结构中的路径的公共数据结构 - 结构化多图。结构路径合并旨在将结构路径转换为更一般的形式，消除工件。特征选择涉及找到成功的机器学习应用程序所需的最小特征集。矢量化将结构化多图表转换为机器学习方法处理的数字矢量。学习根据编码在特征向量中的属性生成恶意和良性文件的区分模型

特别是对PDF而言，我们有充足的机会进一步匿名化和层级化，例如我们的规则中未涉及的名称树和数字树。通常，为了扩展列表，建议阅读PDF标准，查找PDF文档结构中允许使用用户定义名称或有明确定义的列表或层次结构的位置

关键优势：

1.减少攻击面。

2.有限的功能集在时间上漂移

3.特征空间降维

6结论

在本文中，我们介绍了Hidost，一种基于机器学习的恶意软件检测系统。它代表了先前发表的方法SL2013 [ [26](https://link.springer.com/article/10.1186/s13635-016-0045-0#CR26) ] 的扩展。Hidost是第一个基于静态机器学习的恶意软件检测器，旨在操作多种文件类型。它通过根据结构和内容制作恶意和良性样本模型来实现这一点。

在一个实际的实验中对实际数据集进行了评估，这些实验包括多个月的定期再训练，Hidost胜过了网站VirusTotal部署的所有防病毒引擎，并检测到最高数量的恶意PDF文件。它也在SWF恶意软件中名列前茅。与其前身SL2013相比，它不太容易受到隐藏在PDF文件模糊部分的恶意软件的影响。通过定期再训练，Hidost也变得更加强大，能够不断适应恶意软件，以更新防御。最后，其大大减少的特征集维度使其能够在非常大的数据集上高效地应用。

对于Hidost来说，合理的下一步是对其他分层结构文件格式的实施和评估。特别重要的是Microsoft Office使用的格式的应用程序，因为它们广泛用于最近的针对性攻击。本文提出了该应用的概念设计。展望未来，更高级的字符串处理方法的发展可能证明对于逻辑结构和数字内容不能提供足够区分能力的格式来检测恶意软件是不可或缺的。