监督学习主导下恶意代码行为分析与特征码提取的研究

◆黄杰锋　龙华秋　容振邦

（五邑大学计算机学院　广东　529020）

摘要：鉴于近些年来计算机病毒有越来越猖狂的态势，并且很多杀毒软件存在较高的误报或漏报的情况。因此本项目本着降低误报率与漏报率的初衷，设计了一个恶意代码检测系统。本系统包括检测模块和评分模块两个主要功能。根据日常病毒样本的分析逻辑以及平时病毒分析的经验，设计了最优的检测评分逻辑以及自定义了规则库与规则权重分配。系统分析的结果结合了用户的意见以决定是否提取病毒的恶意代码特征码。

关键词：监督学习；病毒；恶意代码；特征码；

0　引言

信息技术日渐发达，给人们带来了许多便利的同时也带来了相应的个人利益问题，最主要的莫过于信息安全威胁。因此电脑用户多数都选择了安装杀毒软件，但是杀毒软件有利也有弊，最突出的地方在于内存资源占用较高，而且存在相当高的误报率，漏报率。特别是人工智能、云查杀一类，如果白名单做的不够全面，就容易误报。根据赛门铁克在全球范围监测的统计数据，单从 2014 年的统计数据来看，全年就新增恶意代码 3.17 亿个，恶意代码总数已达 17 亿[1-2]。由于壳对抗的存在，现阶段的病毒已经完全可以“工厂化”制造，速度大大提高，对黑客的技术门槛大大降低，也使得病毒数量持续增长、感染持续扩散[3]。因此特征码的质量决定了查杀的质量。

1　系统设计

现实场景中，恶意代码可以藏匿在任何地方，常见的是以可执行文件的形式，比如可以单文件完成一系列持久性恶意操作；也可以以白加黑形式，欺骗用户点击，悄然威胁着用户电脑的安全。还有其他形式的存在，比如宏病毒，漏洞利用等，恶意代码现在已经成为混合体的形式，其界限已经不是非常明确[4-5]。

本项目对常见病毒，也就是可执行文件，展开分析。因此分析流程是，先判断是否为自解压包或者安装包类程序，如果是的话，则需要进行解包操作，然后对每个子程序进行检测。如果程序被加壳过，则需要进行脱壳操作。在检测子程序时，如果发现资源节区大于 0x20000 字节则提取资源，进一步匹配特征。

功能模块主要分为两个，第一个是检测模块，第二个是评分模块。由于特征码扫描法被认为是用来检测已知计算机病毒的最简单、最常用的方法，并且基于特征的检测方法，具有效率高、误报率低等优点[6-7]。因此检测模块会首先被执行，用于检测特征，目的是为了检查此前由评分模块判断为恶意并入库的恶意特征库。如果检测模块能命中此前的特征记录，则不会再进行重新评分，可以用于迅速反馈结果。评分模块是核心模块，如图 1 所示，入口方法是 checkSubPE 方法，可以被递归调用，以循环检查待分析程序的资源或者释放出来的子 PE 文件。评分由 scanByClamav 方法、checkAuth 方法、uploadVT 方法、getDynamicScore 方法来决定主要的分数。scanByClamav 方法用于调用 clamav 离线病毒库来扫描程序，病毒库可以进行更新。checkAuth 方法用于检测 PE 文件中的数字签名是否是正常的。uploadVT 方法用于将待分析程序的哈希作为搜索条件，在 VirusTotal 网站上查询结果，而不是上传文件。getDynamicScore 方法用于将待分析程序的哈希作为搜索条件，在 Hybrid-analysis 网站上查询动态行为分析结果，而不是上传文件。extractRes 方

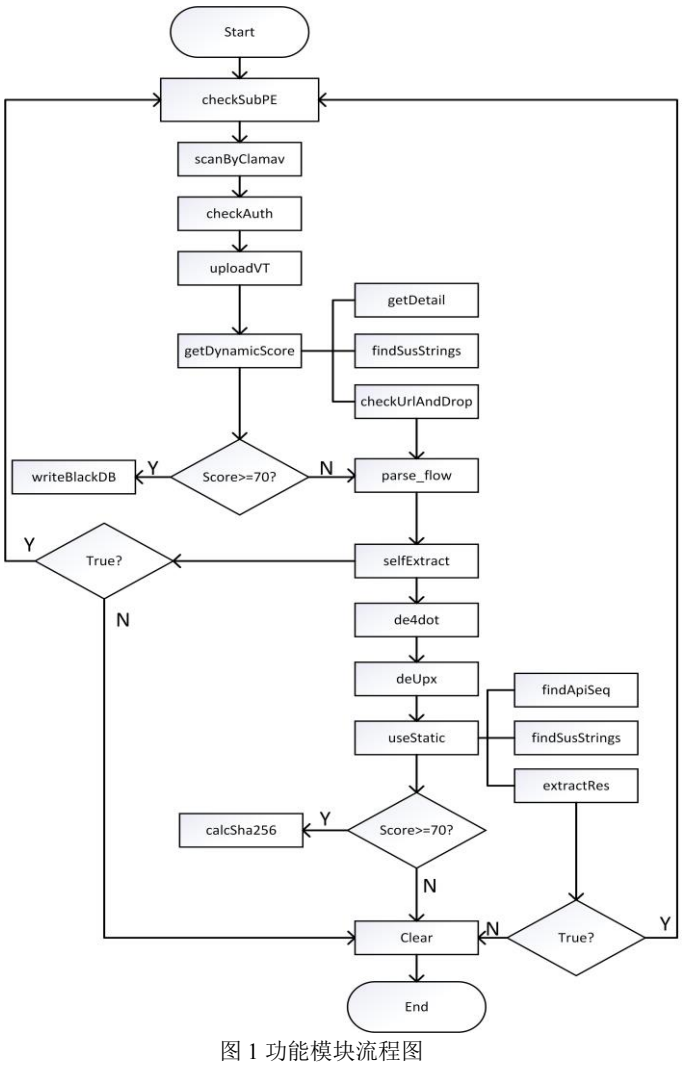

图 1 功能模块流程图

法与 selfExtract 方法用于判断待分析程序是否可以被提取，如果可以被提取则递归分析子 PE 程序。

2 具体实现

如流程图所示，待分析的程序需要先被权重较大的函数进行评分，之后才提取字符串特征。整个项目拟定的一个恶意分数分界线是 70 分，该分数是综合本项目的规则库以及权重分配以及测试之后得出的适合数值。

2.1 评分方法

以下对评分流程所涉及的主要方法进行讲解：

（1）首先经过 scanByClamav 方法进行扫描，clamav 的规则库可以由 python 脚本在线更新，但是由于这个更新流程与本项目评分机制无关，因此不赘述。由于 clamav 的误报率较高，因此这里给定的一个分数是 10 分。

（2）然后经过 checkAuth 方法进行检查，这个方法对待分析程序的数字签名进行校验，如果程序具有正常的数字签名，则恶意评分-20 分，因为 CA 机构在对公司及其程序颁发数字签名时都会对其进行安全性检测，通过了才会对其颁发数字签名。很多杀毒软件厂商很依赖数字签名白名单，假如对数字签名的校验机制不完善，就很有可能被绕过欺骗，因此，很多病毒都会给自己伪造数字签名。

（3）通过 uploadVT 方法，在 virustotal 网站，查询待分析程序的 sha256，然后对返回的 json 数据进行解析，根据平时病毒分析的工作经验，自定义了一些病毒类型名的权重列表，提高相应的权重；在 virustotal 上，众多的杀毒软件厂商中，公认的权威较高的是 Microsoft，ESET-NOD32，Kaspersky，但这些都是外国厂商，对于一些国内的自制特色化病毒并不能检出，因此，还需要借鉴 Tencent，Qihoo-360 的查杀结果。但该两厂商较依赖云查杀，误报率较高，因此程序逻辑定为同时报毒才加分。

（4）通过 getDynamicScore 方法，获取并提取 hybrid-analysis 在线沙盒的分析结果，与自定义的一套字符串与动态行为规则进行匹配，返回特征字符串以及分值数据。

（5）通过 writeBlackDB 方法，定义了写入规则库的格式：[其他壳标志位，模糊哈希，特征串使用标志位，静态字符串列表，动态字符串列表]

（6）通过 parse_flow 方法，检测待分析程序的是否被加壳以及是否是安装包，如果仅是被 upx 加壳的，或者仅是被 dotnet 混淆壳混淆的，则静态脱壳后提取静态字符串；如果是自解压包或者安装包的，则需要提取子程序之后再依次检测子程序；如果是其他壳，则仅是动态分析并尝试提取动态字符串。

（7）通过 selfExtract 方法，检测是否为 NSIS 安装包、或是否为 WinRAR Installer 自解压包、或是否为 Zip 自解压包、或是否为 sfx 自解压包、或是否为 MSI 安装包，如果是，则用 7z 尝试解压缩。

（8）通过 useStatic 方法，提取静态字符串，分析待分析程序的 api 序列。

（9）通过 extractRes 方法，提取并检测资源节区大于 0x20000 字节，资源节区中的 pe 程序。

2.2 规则库以及对应权重分配

Yara 的字符串规则有 19 类，包括了敏感注册表路径、一般木马所特有的字符串、主页锁定、比特币地址正则表达式、系统安全工具类名称、反病毒反虚拟机相关、混淆加密相关等，如图 2 所示。

```
Line 18: rule sensitiveREG
Line 45: rule TrojanSpy
Line 69: rule mainPage_hijack_black_drivers
Line 90: rule mainPage_hijack_Browser
Line 139: rule BitcoinAddress
Line 151: rule System_Tools
Line 190: rule RE_Tools
Line 210: rule Antivirus
Line 675: rule VM_Generic_Detection : AntiVM
Line 694: rule VMWare_Detection : AntiVM
Line 745: rule Sandboxie_Detection : AntiVM
Line 766: rule VirtualPC_Detection : AntiVM
Line 783: rule VirtualBox_Detection : AntiVM
Line 821: rule Parallels_Detection : AntiVM
Line 836: rule Qemu_Detection : AntiVM
Line 846: rule Dropper_Strings
Line 861: rule Obfuscated_Strings
Line 1379: rule Xored_PE
Line 1660: rule BITS_CLSID
```

图 2 规则清单

自定义的字符串规则权重如图 3 所示，展示方式是按权重从大到小顺序，也就是从 30 分到 5 分依次排列。

```
_list_scoreDB = [
    ['general TrojanSpy','30'],
    ['general sensitiveREG','30'],
    ['obfuscated function names','20'],
    ['the BITS service','15'],
    ['XORed PE executable','15'],
    ['base64-encoded executable','10'],
    ['references to system / monitoring tools','15'],
    ['references to security software','15'],
    ['hijack mainPage black_drivers','5'],
    ['hijack all browsers mainPage','5'],
    ['debugging or reversing tools','5'],
    ['Looks for VMWare','5'],
    ['Looks for VirtualPC ','5'],
    ['Looks for Sandboxie','5'],
    ['Looks for VirtualBox','5'],
    ['Looks for Parallels','5'],
    ['Looks for Qemu','5'],
    ['detect virtualized environments','5'],
    ['dropper capabilities','5'],
    ['valid Bitcoin address','5']
]
```

图 3 规则权重

3 功能测试

3.1 运行程序

程序运行后，标签控件显示“please select file by click button”，提示用户点击“FileOpen”按钮以便开始分析文件，如图 4 所示。

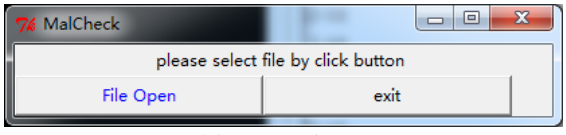

图 4 主界面显示

3.2 展示分析结果

点击“File Open”按钮后，选择要分析的文件后，获得如下分析结果，如图 5 所示。展示的结果项有四项，依次分别为：文件名，分析结果，评分分值，每个子分数的评分缘由。

界面的底部三个按钮：左下角的按钮用于监督学习，由于这里的分析结果是该程序是恶意程序，因此显示是“mark as white”，如果是非恶意程序，则显示“mark as black”，让用户对其进行训练，如果点击该按钮，则会调节相关可疑特征字符串的权重，以后的评分将会根据这个权重进行自动化分析；“confirm”按钮用于让用户确认该结果，然后退出该界面，以便进行后续子程序的分析；“saveLog”按钮用于保存该评分结果。点击“mark as white”按钮后，会对该程序所包含的特征字符串权重修改，然后保存在

“db/learn.db”下，“learn.db”的内容如图 6 所示，与图 3 中的字符串规则权重相对应。

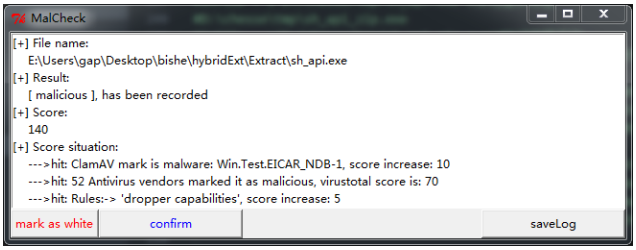

图 5 分析结果

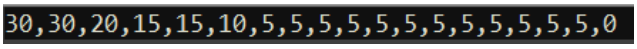

图 6 learn.db 的内容

4 结束语

本项目通过对病毒样本的恶意行为进行分析，根据平时病毒分析经验总结出来的自定义的规则库，对其进行评分，然后结合用户的意见修改特征权重，最后自动化提取其特征码。由于在自动化分析提取的实现方面，字符串特征码的提取较为理想，而且查杀效果好[8]，因此选择字符串特征码。提取的特征码目的是为了实现查杀该类病毒。从最初的手动分析病毒到最后的自动分析系统的实现，提高了工作效率；并且从动静态结合再加上病毒分析经验，降低了误报率、漏报率，以及提高了用户友好度。

参考文献：

[1] 毛蔚轩,蔡忠闽,童力.一种基于主动学习的恶意代码检测方法.软件学报.2017,28(2):384-397

[2] Symantec. Internet security threat report.Vol. 20,2015.https://www4.symantec.com/mktginfo/whitepape-r/ISTR/21347932_GA-internet-security-threat-report-volume-20-2015-social_v2.pdf

[3] 韩奕.基于行为分析的恶意代码检测与评估研究.北京交通大学硕士论文.2014:1-2

[4] 潘剑锋.主机恶意代码检测系统的设计与实现.中国科学技术大学博士论文.2009:4-5

[5] Matt Bishop. Introduction to Computer Security. First printing.Addison-Wesley.2004:6-7

[6] 王蕊,冯登国,杨轶.基于语义的恶意代码行为特征提取及检测方法.软件学报.2012,23(2):378-393

[7] 吴俊军,方明伟,张新访.基于启发式行为监测的手机病毒防治研究.计算机工程与科学.2010(1): 35-38,112

[8] 莫樱.基于病毒行为分析的特征码的提取与检测.电子科技大学工程硕士论文.2011:34-44

基金信息：2017 年校级高等学校大学生创新创业训练计划项目