计算机程序基因技术初探*

苏　宏,丁建伟,陈周国
(中国电子科技集团公司第三十研究所保密通信重点实验室,四川 成都 610041)

摘　要:计算机程序基因技术是在恶意程序不断演化与检测识别技术的对抗博弈中发展起来的,传统的恶意程序检测技术已难以应对恶意程序的快速动态增长趋势,计算机程序基因技术开启了恶意程序检测分析的一个新的思路,从汇编指令层中萃取出计算机程序的真实行为意图,从而对程序进行分析和识别。根据给出的计算机程序基因的定义以及提取方法,可以找出计算机程序最本原的特征,有效提高新型的或变种的恶意程序的分析和检测效率。

关键词:静态分析;动态分析;程序基因;基本块
中图分类号:TP309　　**文献标识码**:A　　DOI: 10.19358/j.issn.2096-5133.2020.01.004
引用格式:苏宏,丁建伟,陈周国.计算机程序基因技术初探[J].信息技术与网络安全,2020,39(1):19-23.

A study of computer program genetic technology

Su Hong, Ding Jianwei, Chen Zhouguo
(Science and Technology on Communication Security Laboratory 30th Research Institute of China Electronics Technology Group Corporation, Chengdu 610041, China)

Abstract: Computer program genetic technology is developed in the confrontation game between the evolution of malicious programs and the detection and recognition technology. Traditional malicious program detection technology has been difficult to cope with the rapid and dynamic growth trend of malicious programs. Computer program genetic technology finds a new way for malicious program detection and analysis, by extracting the real behavior intention of the computer program from the assembly instruction layer so as to analyze and identify it. This paper gives the definition of the computer program gene and the method of extracting the computer program gene. In this way, the most original characteristics of the computer program can be found out. The analysis and detection can be improved effectively for the new malicious programs or the variant of malicious programs.

Key words: static analysis; dynamic analysis; program gene; basic block

0　研究背景

随着网络和计算机技术日益渗透到人类社会生活的方方面面,人们在享受网络和计算机所带来的便捷与高效的同时,也不断受到网络安全问题所带来的困扰。当前,网络安全面临的最主要威胁之一便是恶意程序的侵扰,从目前公布的网络安全事件分析报告可以看出,几乎所有的安全事件最终都可以归结于恶意程序的生产、传播和爆发。因此,恶意程序的检测和识别成为了重要的网络关键技术,也是维护计算机和网络等信息基础设施正常运行的必备手段。

恶意程序是指一种用于破坏计算机正常运行、内嵌各种用户未知的异常操作如收集敏感信息、非法访问各种资源等的计算机可执行代码。根据恶意程序攻击形式,大致分为:病毒(Viruses)、木马(Trojan horses)、蠕虫(Worms)、僵尸(Worpse)、间谍软件(Spyware)、后门程序(Backdoor)、逻辑炸弹(Logic bombs)等。近年来,恶意代码的肆意传播已经严重威胁到计算机等网络基础设施的安全,恶意代码的攻击手段复杂多变,其攻击方式可以通过组合变换,使得恶意代码的攻击强度和隐蔽性得到显著增强,让杀毒软件防不胜防。

随着多年的技术发展和积累,针对恶意程序的检测和识别的技术也是应运而生,主要有三种分析技术,分别是静态分析、动态分析和同源性分析技术。

* 基金项目:国家重点研发计划资助(2016YFE0206700)

恶意程序静态分析技术是通过二进制反汇编技术实现的,主要提取程序的离散特征,包括程序静态反汇编指令信息、文件格式、程序 PE 文件结构信息、程序字符串常量、导入表资源信息等特征信息,在此基础上对程序进行分析识别,这种技术的适用范围和应用场景比较有限[1-3]。恶意程序动态分析技术也指沙箱技术,主要采用基于 QEMU、VMware、Pin 全虚拟机的环境,通过针对全系统级别的信息搜集,获取恶意程序在沙箱中的行为信息和指令信息等特征,包括用户函数调用序列、系统调用序列、指令流序列的程序行为、网络数据流信息、程序运行时数据流信息和汇编指令流信息,然后对程序进行分析识别,动态分析技术具有对常见抗静态分析技术的对抗能力,但是由于二进制程序的复杂性,提取信息繁多,难以从海量指令数据流中提取出具有识别作用的特征数据[4-6]。

软件同源性分析技术是在如何保护软件的知识产权背景下发展起来的,随着计算机产业的不断发展,计算机软件所带来的巨大经济效益和社会效益,使得软件的价值越来越被人们所重视,这势必带来软件的剽窃和抄袭行为。而软件同源性分析鉴别是指比较两个或多个软件系统的源代码,找出它们的相同或相似之处,为软件知识产权的归属提供有力的证据[7]。虽然软件同源性分析是比较两个计算机程序之间的相似度,但也仍然可以应用于未知程序是否与某一已知恶意程序相似性的判定上,从而界定程序是否存在恶意性。

应该说程序静态分析技术、动态分析技术和软件同源性分析技术是专为恶意程序的检测分析而生,虽然上述几种方法在恶意程序的检测和识别中或多或少存在一些弊端,但仍然能够发挥一定的作用,一定程度上检测识别出历史上已知的恶意程序。然而任何事物都是不断发展变化的,针对恶意程序的检测识别技术,恶意程序本身也在寻求着变化,以躲避被检测识别的威胁。恶意程序最常用的方法是采用新型混淆和特征隐藏技术,如代码多态、指令变形、代码混淆、程序加壳等,来对抗恶意程序的分析和程序特征的提取。

恶意程序的不断变化必然导致恶意程序检测识别技术的相应变化。计算机程序基因技术正是这一变化的最直接体现。目前针对恶意程序的分析和检测最大的难题在于基于特征的分析检测技术难以应对恶意程序的快速动态增长趋势,计算机程序基因技术旨在设计一种新型的恶意程序特征模型,能够有效表征恶意程序,尤其是针对新型恶意程序或者恶意程序的变种,也能够有效提高恶意程序的分析和检测效率。

借鉴生物基因学的概念内涵,计算机程序基因是计算机程序静态元素被动态表达的最小单元,其内涵体现在计算机程序运行模式的体系结构基础上,结合大数据技术手段和机器学习方法,从计算机程序运行的底层汇编指令流中萃取出计算机的程序行为特征,用于语义描述和唯一表征计算机程序的行为模式。这种行为模式是从程序本身实际执行过程中动态提取的,反映了程序真实的行为意图,能够有效地抵御新型混淆和特征隐藏技术,是分析检测恶意程序的大胆尝试。

国内郑州信息工程大学最早从 2006 年开始,持续跟踪恶意代码领域相关研究,在系统收集海量恶意代码样本的同时,创新性地提出软件基因理论和分析方法。与传统的特征定义不同,基因是指软件体上具有功能或承载信息的二进制片段,是物质性与信息性的统一,利用基因的原子性、表意性、稳定性、进化性等特性,能够解释代码的同源传播、遗传变异、衍生进化的复杂问题,是在当前海量恶意代码威胁态势下,解决各类复杂问题的有利抓手,为漏洞挖掘、武器生产、态势感知、风险发现等工作提供了直接技术支撑,这些应用和分析都依赖于海量恶意代码基因库,因此构建海量的恶意代码基因库,是计算机程序各种应用分析的基础。

虽然计算机程序基因借鉴了生物基因的概念,但两者之间还是有很大的区别,程序基因则是人为定义的,目的是换一种视角来看待各种计算机程序,希望探索出一条全新的思路来分析识别计算机程序。

1 计算机程序基因定义

计算机程序基因的研究是在底层汇编指令层级对计算机程序进行研究,每一个计算机程序在底层运行时都是由一系列汇编指令的序列集合构成的,一个计算机程序的某个计算功能(行为),运行时由一个指令序列片段实现。因此在程序的功能特性与汇编指令序列片段之间必然存在一个对应关系,通过该序列片段就可以找出该计算机程序内在的基因。基于此,可以将程序基因与生物基因进

行类比:一条汇编指令是一个静态的计算机元素,通过一定的排列组合得到一段汇编指令流,动态表达计算机程序的一个行为(功能)。一条计算机程序行为的汇编指令,相当于于生物基因学的脱氧核糖核酸,即程序 DNA,一段汇编指令的序列就可以组成为程序行为的 DNA 序列。

1.1 生物基因的定义

(1)脱氧核糖核苷酸

脱氧核糖核苷酸是组成 DNA 的基本单位,根据碱基的不同可以分为 4 种,脱氧核苷酸按不同的碱基(A、G、C、T)排列顺序就组成了 DNA,不同的排列组合构成了不同的 DNA 片段,具有不同的遗传效应。

(2)DNA

DNA 是主要遗传物质,基本单位是脱氧核苷酸,DNA 分子分为有遗传功能和无遗传功能的片段,其中,有遗传功能的片段可以控制生物体的某一项生命活动。

(3)基因

基因是 DNA 分子上具有遗传效应的片段,是实现遗传效应的基本单位,一个 DNA 分子上存在着很多个基因,一个特定的基因片段对应一种生命活动的遗传控制。

1.2 程序基因的定义

(1)汇编指令

汇编指令是程序的最小单位,一条指令表示一种计算机操作,每条指令由操作码和操作数组成,其中,操作码决定指令功能。汇编指令不同的排列组合构成不同的基本块,实现不同的功能,类比于生物中脱氧核糖核苷酸,只是脱氧核糖核苷酸只有四种,而汇编指令根据运行平台的不同可以有成百上千种。

(2)基本块

基本块是单入单出结构的最小顺序执行汇编语句集,程序功能实现的基本单位。程序的基本块分为用户自实现代码的基本块以及库函数基本块。基本块功能的组合构成程序的完整功能,程序的功能就由一个个基本块顺序执行完成,这若干个基本块的序列就类似于生物中的 DAN 序列,可以称之为程序 DNA 序列。

(3)程序基因

程序基因是程序基因是在汇编指令层级具有一定功能、能反映程序行为的特定汇编指令及相关数据的有机组合。直观地,程序基因可能是部分基本块或基本块内部分汇编指令的组合,汇编指令的先后顺序也包含在基因信息之列。

在生物界,脱氧核苷酸的排列组合构成了 DNA 序列,DNA 序列中的某一段形成了生物基因,其中 4 种碱基的不同排列顺序蕴含了不同的遗传信息,代表了某一类生物特有的生物性能和特征,表现为生物物种的多样性。而计算机程序中,若干条汇编指令的排列组合构成了基本块序列,若干个基本块或某个基本块中的几条汇编指令构成了计算机程序的基因,其中汇编指令不同的排列顺序代表了不同的程序行为,表现为不同的程序功能。生物基因与计算机程序基因比较如图 1 所示。

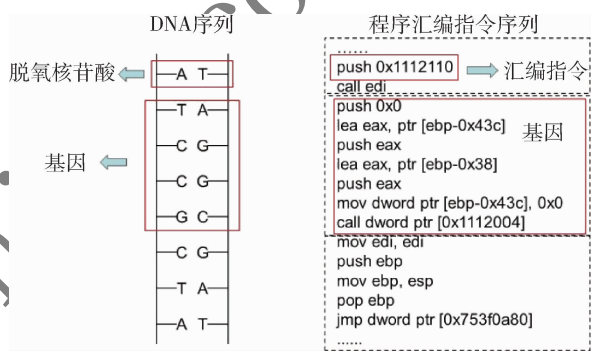

图 1 生物基因与计算机程序基因比较图

2 计算机程序基因提取

为了提取出计算机的程序基因,需要从以下几个环节开展相关工作:编指令流序汇提取、程序样本库构建和基因测绘。

(1)汇编指令流提取

汇编指令流都能以基本块 BBL(Basic Block)的形式呈现,基本块是最小的程序功能单元,为单入单出的程序结构,每个程序由若干个基本块组成,利用动态二进制插桩技术和基于基本块索引的程序指令流快照技术,可以详细地记录程序运行过程中的汇编指令流信息,充分还原程序的真实执行流程。

(2)程序样本库构建

借助互联网平台等各类途径,下载收集各类样本程序,以此构建程序样本库,对程序样本库中的样本进行程序行为分析,根据分析结果标注出程序的行为特征,提取其相关的语义特征,并建立程序样本与语义特征之间的映射关系。

(3)程序基因测绘

在充分还原了程序完整的汇编指令流的基础上,分析样本库中同一类程序行为的所共有的基本块信息,提取出该类型程序的基因,即具有一定功能、能反映某些行为的特定汇编指令集及相关数据的有机组合,相关数据包括通过反编译工具获取的静态属性等信息,如文件名、网站地址等,对于未知程序的基因测绘则是将未知程序的基本块信息与样本程序的基因进行比对,以此判定未知程序是否具有样本程序的基因。

3 文件操作类程序基因提取

基因是代表了相同一类群体所共有的特征,与生物基因一样,计算机程序基因也是代表了相同一类程序行为的共有特征。因此就单一的程序个体而言,基因没有什么实质的意义,正确的做法应该是先广泛收集某一类样本,分析、提取其共有的部分作为该类程序的共性基因,共有的部分可能是几个 BBL 的组合,也可能是某一个 BBL 中的几条汇编指令。当有一个新的样本需要检测时,首先通过基于模拟执行的基本块相似性比较算法,比较样本程序基本块与共性基因基本块是否相似,如果相似则已知样本类型的共性基因即为待测样本的基因,同时还可以得到程序基因在样本程序中的位置,以及汇编指令流的具体表现形式,即待测样本的个性基因;如果不相似则提取其语义特征,通过人工方式打上标签,标明其程序类型,并收集更多该类型的样本组成新的程序样本库,待一定规模后再形成该类型程序的程序基因。

通过对已收集到的文件操作类样本程序进行分析,提取其公共的 BBL 作为文件操作类的共性基因,其结果如下:

```
BBL ZwOpenFile
mov eax,0x30
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL ZwQueryDirectoryFile
mov eax,0x32
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL NtCreateFile
mov eax,0x52
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL ZwQueryInformationFile
mov eax,0xe
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL ZwReadFile
mov eax,0x3
mov ecx,0x1a
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL NtQueryFullAttributesFile
mov eax,0x113
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL ZwWriteFile
mov eax,0x5
mov ecx,0x1a
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
BBL ZwLockFile
mov eax,0xe0
xor ecx,ecx
lea edx,ptr [esp+0x4]
call dword ptr fs:[0xc0]
0
```

以上八个 BBL 模块包括了文件操作类程序的所有共性基因,具体到某一个文件操作类文件,可能只含有其中的部分共性基因,但只要包含了其中的任意一个 BBL,则该待识别的程序即可以判定为文件操作类程序。

对于待测定程序进行基因的提取,首先需要通过二进制插桩技术获取程序的基本块 BBL 信息,然后用已知的文件操作类程序行为的共性基因 BBL 与待测样本程序的所有 BBL 进行模拟执行相似性

比较,比较结果相似,则待测样本程序具有相应类型的程序基因,进一步可提取出具体的个性基因。

由于动态插桩提取得到的是程序运行的基本块汇编指令,而同一基本块内各汇编指令排列顺序的多样性,以及相同功能指令的多样性,普通的基于语义或词频的相似度比较算法,较难反映出程序基本块的功能相似性。因此需要使用基于模拟执行的基本块相似性比较算法,该比较算法利用了基本块是一个功能连续的单入口、单出口的汇编指令序列,通过定义内存、寄存器等数据结构,并解析基本块中各汇编指令的操作码、操作数语义,作用于相对应的数据结构中,模拟该基本块的执行逻辑,通过内存等数据结构中保存的汇编指令流执行结果,判断待测程序基本块与共性基因基本块之间的相似性。

程序静态属性信息一般包含字符串等信息,程序静态字符串中一般含有与程序功能相关的字符,可以将此字符串作为程序基因的补充。文件操作属性的内容是文件路径,通过反编译工具可以得到文件操作类程序所要操作的文件名,例如,c:\users\test\desktop\test.txt。

4 结论

借鉴生物基因的概念,本文提出了计算机程序基因的概念,从计算机程序运行的底层汇编指令流中萃取出计算机程序基因,用于语义描述和唯一表征计算机程序的行为模式。计算机程序基因技术开启了恶意程序检测分析的一个新的思路,后续围绕计算机程序基因的应用,可以建立计算机程序统一的、完整的、详尽的基因图谱世系图,旨在识别、分析、梳理计算机程序的行为模式以及发展进化趋势。

参考文献

[1] MILLER B,KANTCHELIAN A,TSCHANTZ M C,et al. Reviewer integration and performance measurement for malware detection[M]. Detection of Intrusions and Malware, and Vulnerability Assessment,2016.

[2] KOLOSNJAJI B,ZARRAS A,LENGYEL T,et al. Adaptive semantics-aware malware classification[C]. International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Springer-Verlag New York,Inc.,2016:419-439.

[3] RAFF E,SYLVESTER J,NICHOLOAS C. Learning the PE header,malware detection with minimal domain knowledge[C]. ACM Workshop on Artificial Intelligence and Security. ACM,2017:121-132.

[4] KI Y,KIM E,KIM H K. A novel approach to detect malware based on API call sequence analysis[J]. International Journal of Distributed Sensor Networks,2015,58(7):3201-3206.

[5] KIRAT D,VIGNA G. MalGene:automatic extraction of malware analysis evasion signature[C]. ACM SIGSAC Conference on Computer and Communications Security. ACM,2015:769-780.

[6] XU Z,ZHANG J,GU G,et al. GoldenEye:efficiently and effectively unveiling malware's targeted environment[M]. Research in Attacks,Intrusions and Defenses. Springer International Publishing,2014:22-45.

[7] 任颜珠. 结合文本和抽象语法树比对的源代码同源性鉴别系统的研究与设计[D]. 北京:北京邮电大学,2011.

(收稿日期:2019-11-21)

作者简介:

苏宏(1966-),男,硕士,研究员,主要研究方向:网络安全、信息系统、深暗网监测。

丁建伟(1986-),男,博士,高级工程师,主要研究方向:威胁情报分析、暗网安全分析等。

陈周国(1980-),男,硕士,高级工程师,主要研究方向:威胁情报分析、暗网安全分析等。

版权声明