

未知恶意代码族群归属决策研究

左黎明, 刘二根, 徐保根, 汤鹏志
(华东交通大学 基础科学学院, 江西 南昌 330013)

摘 要: 提出了恶意代码 API 调用函数特征集的概念. 根据不同恶意代码为实现相同功能调用相同 API 函数的特点, 给出了一种基于 API 调用函数集合的恶意代码特征提取方法. 使用集合运算获取族群函数特征集, 通过模糊聚类与熵值法计算出未知恶意代码与已知恶意代码族群的正隶属度, 利用正隶属度极大原则来确定未知恶意代码归属族群, 最后给出算例. 该方法不需要人工干预决策, 易于程序实现.

关键词: 病毒; 数据安全; 恶意代码; 族群; 病毒检测; 决策

中图分类号: TP309; TP393

文献标识码: A

文章编号: 1000-7180(2012)02-0188-04

A Research on Decision-making of Unknown Malicious Code's Family

ZUO Li-ming, LIU Er-gen, XU Bao-gen, TANG Peng-zhi

(School of Natural Science, East China Jiaotong University, Nanchang 330013, China)

Abstract: A new concept of system API calling characteristics set of malicious codes is given. According to the feature of same API calling of the different malicious codes for the same malicious functions, the new extraction and analysis methods of malicious code characteristics which based on the set of API calling is put forward. The method uses set operations to obtain function characteristics set, calculates the membership degree between unknown malicious code and known malicious code families by fuzzy clustering and entropy method, then judges the family of unknown malicious code by the principle of max membership degree. At last, an example is given to verify the method. The proposed method can be realized easily and automatically without any manual intervene.

Key words: viruses; security of data; malicious code; family; virus detection; decision-making

1 引言

虚拟经济和电子商务的发展给人们带来极大便利的同时,也带来了巨大的安全威胁. 恶意代码产生的巨大经济利益驱使恶意代码作者及其使用者针对杀毒软件引擎和病毒库的升级不断改进和开发新变种,在改进过程中参考其他已知的恶意代码源码,模仿它们的一些功能,吸收其中的技术优点,针对操作系统或者应用程序出现的新漏洞,在变种中添加新的攻击代码,同时为了提高变种的生存率,还采用多种新的自我保护技术^[1-2]. 另外恶意代码程序在传播过程中,还可能因为被其他恶意代码感染后被动产

生新的变种^[3]. 恶意代码发展趋势越来越专业化、智能化和商业化,通常集病毒、蠕虫、木马、后门、广告、间谍等于一体,因此恶意代码程序变种往往具有随机性、模糊性和动态性^[4-5]. 文献[6]中分析了网络恶意代码族群发展特点,提出了一个网络恶意代码族群迭代增长模型刻画其发展过程. 文献[7]中进一步提出了恶意代码族群特征的形式化描述,给出了一种基于函数调用图和和内核函数调用集合的恶意代码族群特征提取方法. 由于恶意代码分析受到很多人为或者非人为因素的影响,因此确定一个未知的恶意代码新变种的族群归属是一个复杂的决策问题,本文是文献[7]工作的继续.

收稿日期: 2011-05-15; **修回日期:** 2011-07-20

基金项目: 国家自然科学基金项目(11061014);江西省教育厅青年基金项目(GJJ10129);江西省教育厅科技项目(GJJ10708)

2 族群特征的形式化描述

定义 1^[7] 设 $A_t (1 \leq t \leq n)$ 为 n 个已确定的一族恶意代码, $E_t (1 \leq t \leq n)$ 为 $A_t (1 \leq t \leq n)$ 对应的病毒候选特征集, 则族群特征集定义为: $\hat{E} = \bigcap_{t=1}^n E_t$.

定义 2^[7] 设 B 是一个待分析的恶意代码, 则 B 与已知族群 $A_t (1 \leq t \leq n)$ 族群相关度 λ 定义为

$$\lambda = \frac{|\hat{E} \cap E_B|}{|\hat{E}|},$$

式中, E_B 为 B 的病毒候选特征集, $|\hat{E} \cap E_B|$ 和 $|\hat{E}|$ 分别表示集合 $\hat{E} \cap E_B$ 和 $\hat{E}$ 中元素的个数. 显然 $0 \leq \lambda \leq 1$, 当 λ 越大时, 恶意代码 B 与已知族群 $A_t (1 \leq t \leq n)$ 的族群关系特征越明显.

恶意代码为实现其恶意功能, 会有不同于正常程序的特殊行为特征, 所以可以根据特定的独立行为或复合行为序列来判断一个程序是否是恶意程序, 任何恶意代码程序最终都需要调用系统 API 函数来实现自己的功能和行为^[8], 根据这一原理, 本文提出恶意代码的 API 调用函数特征集概念, 将恶意代码为实现恶意功能而调用的系统 API 函数集合称为恶意代码函数特征集. 而同一族群恶意代码由于功能基本相似, 因此所调用的主要系统 API 函数集合基本一致, 利用定义 1 可以构造已确定的一族恶意代码的族群函数特征集.

设集合 T_1 表示文件操作类函数集, 集合 T_2 表示系统服务类函数集, 集合 T_3 表示注册表类函数集, 集合 T_4 表示网络通讯类函数集, 集合 T_5 表示进程控制类函数集, 集合 T_6 表示系统调试类函数集, 集合 T_7 表示内核对象操作类函数集, $T = T_1 \times T_2 \times T_3 \times T_4 \times T_5 \times T_6 \times T_7$ 表示集合 $T_1, T_2, T_3, T_4, T_5, T_6, T_7$ 的积空间, 对操作系统中七类函数集中每一个主要函数都用定长编码进行编码, T_1 编码范围是“00000001—00000999”, T_2 编码范围是“00001000—00001999”, T_3 编码范围是“00002000—00002999”, T_4 编码范围是“00003000—00003999”, T_5 编码范围是“00004001—00004999”, T_6 编码范围是“00005000—00005999”, T_7 编码范围是“00006000—00006999”. 所有恶意代码为了实现其特定功能必然要调用以上七大类函数集中的函数, 通过静态和动态分析^[7], 可以获取一个恶意代码所有的系统调用函数, 然后将其编码记录到其对应的函数调用集 E 中, 这样每一个未知恶意代码调用系统函数集 E 可以表示为空间 T 中的元素. 设有 n

个已确定的恶意代码族群, 每一个已确定的恶意代码族群都有其特殊的系统 API 函数特征集 $\hat{E}_i (1 \leq i \leq n)$. 现定义函数特征集运算, 设:

$$E_1 = t_1 \times t_2 \times t_3 \times t_4 \times t_5 \times t_6 \times t_7, E_2 = s_1 \times s_2 \times s_3 \times s_4 \times s_5 \times s_6 \times s_7;$$

调用函数集交运算 $E_1 \cap E_2$ 定义为

$$E_1 \cap E_2 = (t_1 \cap s_1) \times (t_2 \cap s_2) \times (t_3 \cap s_3) \times (t_4 \cap s_4) \times (t_5 \cap s_5) \times (t_6 \cap s_6) \times (t_7 \cap s_7).$$

若其中 $t_k \cap s_k = \phi$, 形式上记为 $t_k \cap s_k = \{\}$.

对未知恶意代码, 求其函数调用集 E 与恶意代码族群 i 系统 API 函数特征集 $\hat{E}_i$ 的交集:

$$F_i = \hat{E}_i \cap E, (1 \leq i \leq n), \text{记 } \|F_i\| = (a_1, a_2, a_3, a_4, a_5, a_6, a_7), a_k \text{ 表示 } F_i \text{ 第 } k \text{ 个分量集合元素的个数.}$$

3 恶意代码归属族群判别方法

3.1 相关定义及其说明

定义 2 中族群相关度 λ ^[7] 有一定的局限性, 只适合单个指标, 由于不同种类 API 函数集对恶意代码的影响差异很大, 无法直接比较, 所以需要重新给出一套方法来度量未知恶意代码与族群的相关性. 设备选恶意代码族群有 m 个, 族群特征的评价指标有 n 个, 则对于所有备选恶意代码族群, 其特征评价指标矩阵 C 为

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix} = (c_{ij})_{m \times n},$$

其中 $c_{ij} \geq 0$.

为了便于计算和分析, 可以将特征矩阵 C 转化为对应的归一化隶属度矩阵 R . 设恶意代码族群 i , 族群特征评价指标 j 的隶属度为 r_{ij} , 取 $r_{ij} = c_{ij} / \sum_{i=1}^m c_{ij}$. 则

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} = (r_{ij})_{m \times n}.$$

定义 3 设恶意代码族群特征隶属度矩阵 $R = (r_{ij})_{m \times n}$, 定义 $\varphi = (\varphi_1, \cdots, \varphi_n) = (\bigvee_{i=1}^m r_{i1}, \cdots, \bigvee_{i=1}^m r_{in})$ 为正理想隶属度特征向量, $\psi = (\psi_1, \cdots, \psi_n) = (\bigwedge_{i=1}^m r_{i1}, \cdots, \bigwedge_{i=1}^m r_{in})$ 为负理想隶属度特征向量.

定义 4 设恶意代码族群特征的评价指标权重

向量为 $\omega = (\omega_1, \dots, \omega_n)$, 且第 i 个族群的特征评价指标取值向量为 $R_i = (r_{i1}, \dots, r_{in})$. 定义 $L_1(R_i, \varphi) = \sqrt{\sum_{j=1}^n \omega_j (\varphi_j - r_{ij})^2}$, $L_2(R_i, \psi) = \sqrt{\sum_{j=1}^n \omega_j (\psi_j - r_{ij})^2}$, 称 $L_1(R_i, \varphi)$ 和 $L_2(R_i, \psi)$ 为恶意代码族群 i 的正理想距离和负理想距离.

定义 5 对于恶意代码族群 i , 设 α_i 和 β_i 分别表示未知恶意代码属于恶意代码族群 i 的正隶属度和负隶属度, 且 $\alpha_i + \beta_i = 1$. 则 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)$, $\beta = (\beta_1, \beta_2, \dots, \beta_m)$ 表示恶意代码族群正隶属度向量和负隶属度向量. 定义:

$$H_1(R_i, \varphi) = \alpha_i L_1(R_i, \varphi)$$

$$= \alpha_i \sqrt{\sum_{j=1}^n \omega_j (\varphi_j - r_{ij})^2},$$

$$H_2(R_i, \psi) = \beta_i L_2(R_i, \psi)$$

$$= \beta_i \sqrt{\sum_{j=1}^n \omega_j (\psi_j - r_{ij})^2},$$

称 $H_1(R_i, \varphi)$ 为恶意代码族群 i 的正理想度, $H_2(R_i, \psi)$ 为恶意代码族群 i 的负理想度.

3.2 熵值法确定权重 ω

决策中需要确定恶意代码族群特征的评价指标权重向量为 $\omega = (\omega_1, \dots, \omega_n)$, 确定权重通常用专家经验法或者层次分析法, 为了自动确定权重方便和避免人工评价带来的主观误差, 也可以采用熵值法. 由于恶意代码族群特征隶属度矩阵已经做过归一化, 所以这里直接取 $p_{ij} = r_{ij}$, 显然有 $\sum_{i=1}^m p_{ij} = 1$. 熵值法确定权重 ω 过程如下:

step1 计算第 j 项指标的熵值: $e_j = -\sum_{i=1}^m p_{ij} \ln p_{ij} / \ln m$, 其中 $e_j \geq 0$.

step2 计算第 j 项指标的差异系数: $g_j = 1 - e_j$, $0 \leq g_j \leq 1$.

step3 求权重: $\omega_j = \frac{g_j}{\sum_{j=1}^n e_j}$.

3.3 最小二乘法确定 α 和 β

决策的最终目标要确定恶意代码族群正隶属度向量 α 和负隶属度向量 β , 再利用正隶属度极大原则来确定未知恶意代码属于哪个族群. 这里使用最小二乘法来确定向量 α 和 β . 首先建立目标函数:

$$f_i = [H_1(R_i, \varphi)]^2 + [H_2(R_i, \psi)]^2, 1 \leq i \leq m$$

因为 $\alpha_i + \beta_i = 1$, 所以 $f_i = [\alpha_i L_1(R_i, \varphi)]^2 + [(1 - \alpha_i) L_2(R_i, \psi)]^2$, 再由 $\frac{\partial f_i}{\partial \alpha_i} = 0$, 整理即得

$$\alpha_i = \frac{[L_2(R_i, \psi)]^2}{[L_2(R_i, \psi)]^2 + [L_1(R_i, \varphi)]^2}, 1 \leq i \leq m$$

对所有的恶意代码族群计算其 α_i 的值, 利用正隶属度极大原则来确定恶意代码属于某个族群, 即满足 $\alpha_{\max} = \max_{1 \leq i \leq m} \{\alpha_i\}$ 的族群为未知恶意代码的所属族群.

4 实例分析与已有方法的比较

现有恶意代码族群分类^[7] (粗粒度一级分类) 见表 1.

表 1 恶意代码族群分类

族群编号	1	2	3	4	5	6
族群名称	木马	后门	蠕虫	病毒	Rootkit	广告流氓软件

以下直接给出一个线上游戏窃取者变种预处理后的实验数据:

$$\|F_1\| = (15 \ 12 \ 13 \ 2 \ 4 \ 7 \ 9), \|F_2\| = (23 \ 11 \ 2 \ 11 \ 3 \ 2 \ 5), \|F_3\| = (9 \ 1 \ 0 \ 2 \ 2 \ 3 \ 5),$$

$$\|F_4\| = (4 \ 1 \ 1 \ 4 \ 2 \ 2 \ 0), \|F_5\| = (6 \ 5 \ 7 \ 2 \ 4 \ 4 \ 2), \|F_6\| = (2 \ 4 \ 1 \ 3 \ 3 \ 2 \ 1).$$

构造特征评价指标矩阵 C :

$$C = \begin{bmatrix} 15.000 \ 0 & 12.000 \ 0 & 13.000 \ 0 & 2.000 \ 0 & 4.000 \ 0 & 7.000 \ 0 & 9.000 \ 0 \\ 23.000 \ 0 & 11.000 \ 0 & 2.000 \ 0 & 11.000 \ 0 & 3.000 \ 0 & 2.000 \ 0 & 5.000 \ 0 \\ 9.000 \ 0 & 1.000 \ 0 & 0.000 \ 0 & 2.000 \ 0 & 2.000 \ 0 & 3.000 \ 0 & 5.000 \ 0 \\ 4.000 \ 0 & 1.000 \ 0 & 1.000 \ 0 & 4.000 \ 0 & 2.000 \ 0 & 2.000 \ 0 & 0.000 \ 0 \\ 6.000 \ 0 & 5.000 \ 0 & 7.000 \ 0 & 2.000 \ 0 & 4.000 \ 0 & 4.000 \ 0 & 2.000 \ 0 \\ 2.000 \ 0 & 4.000 \ 0 & 1.000 \ 0 & 3.000 \ 0 & 3.000 \ 0 & 2.000 \ 0 & 1.000 \ 0 \end{bmatrix}$$

将特征矩阵 C 转化为对应的归一化隶属度矩

阵 R :

$$R = \begin{bmatrix} 0.254\ 2 & 0.352\ 9 & 0.541\ 7 & 0.083\ 3 & 0.222\ 2 & 0.350\ 0 & 0.409\ 1 \\ 0.389\ 8 & 0.323\ 5 & 0.083\ 3 & 0.458\ 3 & 0.166\ 7 & 0.100\ 0 & 0.227\ 3 \\ 0.152\ 5 & 0.029\ 4 & 0.000\ 0 & 0.083\ 3 & 0.111\ 1 & 0.150\ 0 & 0.227\ 3 \\ 0.067\ 8 & 0.029\ 4 & 0.041\ 7 & 0.166\ 7 & 0.111\ 1 & 0.100\ 0 & 0.000\ 0 \\ 0.101\ 7 & 0.147\ 1 & 0.291\ 7 & 0.083\ 3 & 0.222\ 2 & 0.200\ 0 & 0.090\ 9 \\ 0.033\ 9 & 0.117\ 6 & 0.041\ 7 & 0.125\ 0 & 0.166\ 7 & 0.100\ 0 & 0.045\ 5 \end{bmatrix}$$

利用熵值法计算权重向量 ω :

$e =$

(0.855 0 0.822 5 0.649 3 0.858 0 0.978 9
0.929 1 0.780 0)

$g =$

(0.145 0 0.177 5 0.350 7 0.142 0 0.021 1
0.070 9 0.220 0)

$\omega =$

(0.128 7 0.157 4 0.311 1 0.126 0 0.018 7
0.062 9 0.195 2)

计算正理想隶属度特征向量 φ 和负理想隶属度特征向量 ψ 和决策指标向量恶意代码族群正隶属度向量 α :

$\varphi =$

(0.389 8 0.352 9 0.541 7 0.458 3 0.222 2
0.350 0 0.409 1)

$\psi =$

(0.033 9 0.029 4 0.000 0 0.083 3 0.111 1
0.100 0 0.000 0)

$L_1 =$

(0.141 7 0.275 6 0.376 7 0.393 9 0.275 1
0.382 9)

$L_2 =$

(0.388 4 0.244 8 0.109 8 0.039 5 0.178 1
0.049 5)

$\alpha =$

(0.882 5 0.441 1 0.078 3 0.010 0 0.295 3
0.016 4)

显然 $\alpha_{\max} = 0.882\ 5$, 按照正隶属度极大原则该未知恶意代码应该属于族群 1—木马。

360 安全卫士和东方微点系统均采用了类似的基于未知恶意代码的函数特征的行为查杀技术,其未知恶意代码归类思想是根据首次命中恶意行为的函数组合来进行的,与本文方案相比,实现速度稍快但是无法比较恰当的全面评估未知恶意代码真正的特征,从而无法进一步准确对未知恶意代码族群共同特征进行数据挖掘。

5 结束语

本文提出的未知恶意代码的族群归类方法,利用模糊聚类熵值法计算未知恶意代码与已知族群的正隶属度,并按照正隶属度极大原则进行决策,由数据内在关系决定,不受人为因素影响,更具有科学性,易于程序实现。恶意代码族群归属的判别决策方法很多,除了文中提出地使用系统 API 函数特征集指标进行决策,还可以采用二进制特征指标或者函数调用图特征指标进行决策。利用数学方法研究恶意代码并从中抽象出数学模型^[9-10]是一个有效的方法。

参考文献:

- [1] Bastrop J, Forest S, Newman MEJ, et al. Technological networks and the spread of computer viruses [J]. Science, 2004 (304): 527-529.
- [2] Christodorescu M, Jha S. Semantics-aware malware detection [C]// Proceedings of IEEE Symposium on Security and Privacy. California: IEEE Press, 2005: 32-46.
- [3] Christodorescu M, Jha S. Static analysis of executables to detect malicious patterns [C]// Proceedings of the 12th USENIX Security Symposium, 2003: 169-186.
- [4] Kinder J, Katzenbeisser S, Schallhart C, et al. Detecting malicious code by model checking [C]// Conference on Detection of Intrusions and Malware & Vulnerability Assessment, Berlin: Springer, 2005.
- [5] Szor P, Ferrie P. Hunting for metamorphic [C]// Proceedings of the 11th International Virus Bulletin Conference, Prague, Czech Republic: VB Press, 2001: 123-144.
- [6] 左黎明, 徐保根, 汤鹏志, 等. 网络恶意代码族群增长模型 [J]. 计算机应用, 2009, 29(1): 152-154.
- [7] 左黎明, 刘二根, 徐保根, 等. 恶意代码族群特征提取与分析技术 [J]. 华中科技大学学报: 自然科学版, 2010, 38(4): 46-49.

(下转第 196 页)

参考文献:

- [1] Arpinar B I, Sheth A, Ramakrishnan C, et al. Geospatial ontology development and semantic analytics [J]. Transactions in GIS, 2006, 10(4): 551-575.
- [2] 梁汝鹏, 李文娟, 李勤超. 基于语义网的地名本体服务框架研究 [J]. 测绘科学, 2011, 36(3): 230-233.
- [3] Zhang C, Li W, Zhao T. Geospatial data sharing based on geospatial semantic web technologies [J]. Journal of Spatial Science, 2007, 52(2): 35-49.
- [4] 岳昆, 王晓玲, 周傲英. Web 服务核心支撑技术: 研究综述 [J]. 软件学报, 2004, 15(3): 428-442.
- [5] 杜云艳, 张丹丹, 苏奋振, 等. 基于本体的海湾空间数据组织方法-以辽东湾为例 [J]. 地球信息科学, 2008, 10(1): 7-13.
- [6] 李德仁, 崔巍. 本体与空间信息多级网格 [J]. 测绘学报, 2006, 35(20): 143-148.
- [7] 安杨, 边馥苓, 关伟红. 基于 Ontology 的网络地理服务描述与发现 [J]. 武汉大学学报: 信息科学版, 2004, 29(12): 1063-1066.
- [8] Smith B, Mark D M. Geographical categories: an ontological investigation [J]. International Journal of Geographical Information Science, 2001, 15(7): 591-612.
- [9] Guarino N. Some organization principle for a unified top-level ontology [C] // Working Notes of AAAI Spring Symposium on Ontological Engineering, Stanford. [s. l]: IEEE: AAAI, 1997.
- [10] Zhao P S, Deng D Di L P. Geospatial web service client [C] // Proceedings of ASPRS 2005 Annual Conference Baltimore, Maryland, 2005.

作者简介:

廖志文 男, (1962—), 博士研究生. 研究方向为数据挖掘、产业经济、投资及衍生工具研究.

(上接第 191 页)

- [8] 王硕, 周激流, 彭博. 基于 API 序列分析和支持向量机的未知病毒检测 [J]. 计算机应用, 2007, 27(8): 1942-41943.
- [9] 左黎明, 刘二根, 汤鹏志. U 盘病毒随机传播模型 [J]. 微电子学与计算机, 2010, 27(3): 106-108.
- [10] 左黎明, 汤鹏志. 一种基于马氏链的可容侵网络状态评估模型 [J]. 微电子学与计算机, 2009, 26(7): 120-122.

作者简介:

左黎明 男, (1981—), 硕士, 讲师. 研究方向为网络信息安全、非线性系统.

刘二根 男, (1968—), 硕士, 教授. 研究方向为图论及其应用.

汤鹏志 男, (1964—), 硕士, 教授. 研究方向为信息系统及其安全.

徐保根 男, (1966—), 教授. 研究方向为图论及其应用.