第34卷 第17期 计 算 机 工 程 2008年9月
Vol.34 No.17 Computer Engineering September 2008
·博士论文· 文章编号：1000—3428(2008)17—0007—03 文献标识码：A 中图分类号：TP391

计算机抗恶意代码免疫模型

周 正[1]，刘 毅[2,3]，李 建[3]，沈昌祥[2]
(1. 海军工程大学电气与信息工程学院，武汉 430033；2. 海军计算技术研究所，北京 100841；
3. 解放军信息工程大学电子技术学院，郑州 450002)

摘 要：很多针对计算机恶意代码的免疫模型和算法要求学习训练的代价比较大，另外这些算法本身也不同程度地存在问题，离实际应用有较大距离，该文提出一种新的计算机抗恶意代码免疫模型。该模型不需要计算和识别恶意代码的具体特征，通过直接消除恶意代码传播和实施破坏的前提条件，使得计算机系统对恶意代码具有自身免疫的能力。
关键词：恶意代码；免疫算法；备份函数；恢复函数

Computer Malicious Codes Immune Model

ZHOU Zheng[1], LIU Yi[2,3], LI Jian[3], SHEN Chang-xiang[2]
(1. College of Electrical and Information Engineering, Naval University of Engineering, Wuhan 430033; 2. Naval Institute of Computing Technology, Beijing 100841; 3. Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450002)

【Abstract】 Most recent immune models and algorithms against malicious codes not only take too much cost in learning and training, but also have shortcomings to some extend. This paper provides a new computer malicious codes immune model. Instead of detecting or recognizing the exact characters of malicious codes, the model gives computer system the self-immune ability against malicious codes by getting rid of the prerequisites for malicious codes to spread and do harm.
【Key words】 malicious code; immune algorithm; backup function; recovery function

1 概述

在信息安全领域中，针对恶意代码的特征和异常行为检测的研究已经取得了显著的成果，而且已经发挥了重要的作用，但是随着攻击手段的高明和新型恶意代码的不断出现，恶意代码和正常应用程序之间的差别将越来越不明显，这使得恶意代码的检测比较困难。与此同时，许多学者将人工免疫的思想引入到恶意代码的防御技术中，虽然提出了很多模型和算法[1-2]，但是一方面这些模型和算法目前要求的学习训练代价比较大[2-4]，另一方面这些模型和算法本身都不同程度地存在问题[2,4]。而且这些研究有忽略操作系统的根本作用和计算机终端源头地位的倾向，离开这些基础以及人的现有知识积累而试图依赖机器的学习训练达到智能免疫目的其效果将很难想象。

计算机恶意代码是一种计算机程序，只有在计算机的软硬件环境下才能传染和发作，并产生破坏作用。因此，获得执行机会和能够进行恶意操作是恶意代码能够传播和实施破坏的前提。如果构造出某种机制使计算机操作系统或者软件环境不能够提供这2个前提条件，或者能够阻止这2个前提条件同时存在，那么就能够获得对恶意代码的自身免疫能力。本文通过对信息系统的安全要素属性进行合理的定义和对这些安全要素之间访问关系进行规划，提出一种通过合适的免疫算法、备份和恢复算法以及完整性校验算法构造而成的计算机恶意代码自身免疫模型。

2 模型原理

首先对信息系统中的安全要素进行抽象，这方面笔者曾经在相关文章做出过论述[5]，本文将在其基础上再给出如下定义：

定义1 a, b 和 c 为元素。对于 a，$\forall b, c$ 都有 $c(a,b) \in o$，则称 a 为非攻击元素，$\exists b, c$ 使得 $c(a,b) \notin o$，则称 a 为攻击元素。同样地，定义 b 和 c 是攻击元素或者是非攻击元素。显然 o 为非攻击元素。

2.1 静态免疫原理

根据以上约定和定义可以得到如下定理：

定理 1 任何非攻击元素构成的完整操作都使得攻击不成立。

证明：不妨假设非攻击元素 $a \in A$，$\forall b \in B, \forall c \in C$，根据定义1可知 $c(a,b) \in o$，因此定理得证。

证毕。

定理 2 在未产生攻击的 A，B 和 C 中增加任何非攻击元素都不会产生攻击。

证明：不妨设增加的元素为 $a \in S$

因为 $\forall b, c$ 都有 $c(\alpha,b) \in o$，$C(A,B) \subseteq o$

所以 $C(A+\alpha,B) = C(A,B) + C(\alpha,B) =$

$$C(A,B) \subseteq o$$

证毕。

推论 1 对非攻击元素构成的 A, B 和 C，在 A 中增加攻击元素不会产生攻击。

推论 2 非攻击元素构成的 A, B 和 C 中任意增加一个攻击元素都不会产生攻击。

基金项目：国家“863”计划基金资助项目(2006AA01Z440)
作者简介：周 正(1978－)，男，博士研究生，主研方向：信息安全，安全操作系统；刘 毅，高级工程师；李 建，博士研究生；沈昌祥，博士生导师、中国工程院院士
收稿日期：2007-10-30 **E-mail：**zhouzheng0203@263.net

推论 3 对未产生攻击的 A, B 和 C，如果不添加使得 $c(\alpha,b) \notin o$ 的 a, b 或者 c 将永远不会生成攻击。

以上推论的证明由于篇幅所限，在此省略。

根据定理 1、定理 2 和推论 1~推论 3 可知，如果信息系统中每一个使用者都是经过认证和授权的，其操作都是符合安全要求的，网络上也不会被窃听和插入，那么就不会产生人为的攻击性事故，就能保证整个信息系统的安全[5]。

在开放系统中，攻击元素的存在是不可否认的，针对攻击元素对客体的修改，引入整性检测函数、备份函数和恢复函数[6]。

定义 2 对 $\forall\ e \in E = A \cup B$，在任一时刻 k 有唯一特征码记为 $c_{e,k}$，在 0 时刻的特征码为 $c_{e,0}$，取 $T = \{UNMODIFIED, MODIFIED, NOTFOUND\}$，取 $g: e \to T$，$\forall e \in E$，$g(e)@\begin{cases} UNMODIFIED & \text{iff } e \in E_0 \cap E_k \wedge c_{e,k} = c_{e,0} \\ MODIFIED & \text{iff } e \in E_0 \cap E_k \wedge c_{e,k} \neq c_{e,0} \\ NOTFOUNT & \text{iff } e \in E_k - E_0 \end{cases}$，称 g 为 E 的完整性检测函数。

定义 3 取系统中所有主体 A 和客体 B 构成的集合 $E = A \cup B$，对 $\forall e \in E$，$content(e)$ 表示 e 中包含的信息，如果以 E 为定义域的函数 b 和 h 函数具有如下性质：

(1) b 的输出是另一个客体；

(2) $\forall\ e_1, e_2 \in E, b(e_1) \neq b(e_2)$ iff $e_1 \neq e_2$；

(3) $M = \{content(e) | \forall e \in E\}$，$N = \{content(b(e)) | \forall e \in E\}$，$\exists\ h: N \to M$ s.t. $\forall\ e \in E$，$h(content(b(e))) = content(e)$。

则称 b 为 E 的备份函数，h 为 $b(E)$ 的恢复函数。显然，对 $\forall E = A \cup B$，总是可以构造出 b 和 h。

定义 4 对 $\forall\ e_k \in E_k = A_k \cup B_k$，取 $w: e_k \to E_0$，s.t. $\forall\ e_k \in E_k$，$w(e_k)@\begin{cases} e_k = e_0, \text{iff } g(e_k) = UNMODIFIED \\ h(b(e_0)) = e_0, \text{iff } g(e_k) = MODIFIED \\ 0, \text{iff } g(e_k) = MODIFIED \end{cases}$，称 w 为 E 的静态校验函数。

定理 3 对未产生攻击的 A，B 和 C，A 和 B 中任何元素被修改后，如果该元素存在备份且可以恢复，完整操作之前通过恢复可以使得攻击不成立。

证明：不妨假设某一初始状态元素 a_0 在 k 时刻被修改为 a_k，恢复函数为 h。

因为 $h(a_k) = a_0$ 且 $C(a_0, B) \subseteq o$

所以 $C(h(a_0), B) = C(a_0, B) \subseteq o$

证毕。

定理 3 说明，如果系统是可恢复的，且其初态是安全的，那么一旦发现可执行代码的完整性被破坏，就先启动恢复过程，恢复成功之后再继续代码的执行过程。这样，即使它被恶意代码感染，但经过恢复之后仍能满足定理 1 的要求，因此，仍能保证系统的安全。

2.2 动态免疫原理

在一个开放的动态环境下，系统不可避免地要引入新的主体、客体或者操作。在这种条件下虽然不能保证定理 1、定理 2 的条件，但是只要保证能够阻止恶意代码传播和造成破坏的 2 个前提条件的形成，即只要能够阻止使得恶意代码传播和产生破坏的主体、客体和行为形成一个完整操作就能够避免攻击和破坏的情况发生，经典的 BLP 模型和 Biba 模型已经给出了证明。因此，在如何保证引入的这些元素不危害系统的安全方面引入动态免疫规则集合直接避免攻击的形成。

定理 4 对于添加攻击元素才能产生攻击的 A，B 和 C，攻击元素被植入后采取某种规则抑制产生攻击的完整操作的形成可以避免攻击产生。

证明：不妨设增加的攻击元素为 $\alpha \in S$，$\exists\ b \in B$，$c \in C$，使得 $c(\alpha,b) \notin o$

因为 $C(A,B) \subseteq o$

所以 $C(A+\alpha, B) = C(A,B) + C(\alpha,B)$

取规则集 $G = \{g | g(c(\alpha,b)) = 0, g(c(a,b)) = c(a,b)\}$，则有

$$G(C(A+\alpha,B)) = G(C(A,B) + C(\alpha,B)) = G(C(A,B)) + G(C(\alpha,B)) = C(A,B) \subseteq o$$

证毕。

根据定理 4 可知，系统的安全初态如果得到静态免疫的保证，只要对信息系统做好适当的域划分和隔离，对可能存在的恶意行为做到及时地阻塞，就能保证信息系统不受攻击。这里的关键问题是恶意行为的判断，VISTA 系统原理和基于标识的认证体系以及可信计算芯片有类似技术可供借鉴，笔者在文献[7]中从理论上作出了详细分析，此处不再赘述。

2.3 联合免疫原理

2.1 节“静态免疫原理”介绍了在不考虑引入主体、客体和操作等安全要素的情况下如何通过对系统已有的安全要素进行合理的授权、审查以及备份和恢复来获取并保持系统的安全状态。2.2 节“动态免疫原理”在静态免疫的基础上介绍了需要引入主体、客体和操作等安全要素的情况下如何通过对可能存在的恶意行为做到及时地阻塞来维持系统的安全状态。静态免疫是动态免疫的前提条件，动态免疫是静态免疫之后的必然要求，一个完整的系统需要静态免疫和动态免疫互相联合共同发挥作用。

系统安全是一个动态的概念，与信息系统的具体状态以及安全政策要求有关，随着信息系统具体状态的变化以及安全政策的调整，系统安全指标也会相应地改变。本文把系统初态安全定义为信息系统具体状态和安全政策调整后达到的安全状态定义为系统的初态安全。

定义 5 对于某一系统 $s(A,B,C,D)$，如果 $\exists F$ s.t. $F(s(A,B,C,D)) = s_0(A_0,B_0,C_0,o)$，则称 F 为免疫计算函数。

F 一般情况下是一套规则或者一系列流程的集合，跟信息系统具体条件以及安全政策有关。

根据以上定义、定理和推论得到主体集合、客体集合、行为集合以及安全机构和安全政策确定下的防护模型：

$$\begin{cases} F(s(A,B,C,D)) = s_0(A_0,B_0,C_0,o) & (1) \\ b(A_0) = E,\quad h(E) = A_0 & (2) \\ C_0(w(A_k), w(B_k)) = C_0(A_0,B_0) \subseteq o & (3) \\ G((C_0 + c')((A_0 + a'),(B_0 + b'))) = C_0(A_0,B_0) \subseteq o & (4) \end{cases}$$

其中，F 是免疫计算函数；主体集合 A、客体集合 B 和行为集合 C 通过函数 F 得到安全政策范围内不会产生攻击的子集 A_0，B_0 和 C_0；b 为备份函数；h 为恢复函数；w 为静态校验函数；G 代表抑制形成攻击的完整操作的动态规则集合。式(1)~式(4)形成一个闭环控制流程，使系统即使在开放的环境下避免攻击的发生，实现了对恶意代码的自身免疫功能。

3 模型实现框架

这种安全模型从理论上能够避免恶意代码获得执行的条

件，从而对恶意代码具有自身免疫能力，该模型在实际应用中需要解决如何获取不存在被感染的信息系统的初始状态以及如何有效地动态地实施对引入的新主、客体进行免疫判断，因此引入安全政策机构、可信计算平台(TCP)以及安全审计模块，模型实施逻辑框架如图 1 所示。

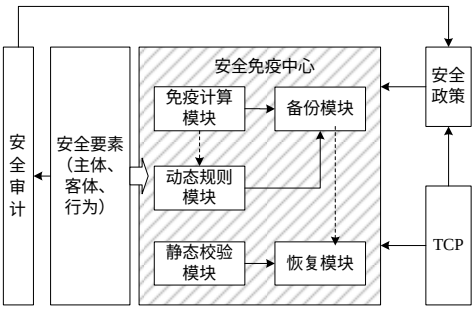

图 1　恶意代码免疫模型的逻辑框架

由安全政策机构、TCP、安全免疫中心、安全审计机构以及系统安全要素中的主体、客体已经行为构成了基于操作系统安全的计算机恶意代码自身免疫模型的实现框架。

其中安全免疫中心由免疫计算模块、备份模块、动态规则模块、静态校验模块和恢复模块组成，它们分别实施模型原理中所介绍的免疫计算函数、备份函数、免疫规则集合、完整性检测函数和恢复函数的功能。

安全要素经免疫计算模块处理后得到安全政策规定下的系统的安全的初始状态，并且通过备份模块将整个安全的环境固化下来。

经过初始的安全固化以后，在接下来的系统运行过程中，首先判断发生作用的安全要素是属于系统原有的还是被新增加的。如果是属于系统原有的，则通过静态校验模块判断是否被修改，如果已经被恶意修改，则调用恢复模块把已经被修改的程序恢复到正常的初始状态。

如果发生作用的安全要素是新增加的则通过动态规则模块的过滤，消除不符合安全政策和系统要求的安全要素或者禁止这些安全要素发生关系。如果经过动态规则模块的过滤后的新增安全要素符合安全政策和系统要求，则通过备份模块加以固化，使得系统得到另一个“安全初态”。为了提高免疫模型的效率，其中的免疫计算模块和动态规则模块的某些功能可以复用。

由于不存在被感染的信息系统的初始状态对本文的模型非常重要，为了有效地获取和保护“系统安全初态”，引入 TCP。TCP 将 BIOS 引导块作为完整性测量的信任的根，可信计算模块(TPM)作为完整性报告的信任的根，对 BIOS、操作系统进行完整性测量，保证计算环境的可信性。TCP 将加密、解密、认证等基本的安全功能写入硬件芯片，并确保芯片中的信息不能在外部通过软件随意获取。在这种情况下除非将硬件芯片从系统中移除，否则理论上是无法突破这层防护的，这也是构建可信的计算机设备以及建立可信的计算机通信的基础。通过系统硬件执行相对基础和底层的安全功能，能保证安全执行机构本身的安全性以及安全政策更新的可信性，为建设安全体系提供完善的底层基础设施。

因此，经过免疫计算模块和动态规则模块处理后的安全要素在通过备份模块固化的时候，这个可信存储过程由 TCP 在底层提供加密保护支持。而静态免疫模块完整性检测的过程则由 TCP 在底层提供加解密和认证保障。

在静态防护的基础上经过动态防护的系统可以根据安全政策在 TCP 的控制下更新静态校验模块的配置，这样既保证了保证系统的完整性，又保证了系统的可用性。

在静态保护和动态保护过程中需要的预警模块和规则执行模块等组件由于概念已经比较直观，而且属于两种保护可以共用的部分，为了保持基本框架的逻辑清晰不再展开描述，因此在实现框架图中不予标注。

系统中涉及到较为敏感的安全要素的操作会经过安全审计的监督，安全审计结果形成报表上报安全执行机构和安全政策，通过安全审计追查可疑操作，确定恶意攻击，以便于清除恶意代码并且为更新安全政策提供依据。

安全政策是制定和更新静态免疫算法和动态免疫算法的依据，伴随着系统的更新以及环境的变化会有相应的安全政策与之相适应，保证选用的免疫算法的实效。

4　结束语

本文从理论上提出了一种新的计算机抗恶意代码免疫模型，这种模型是针对目前流行的各种免疫模型根本缺陷和使用性不强等缺点，通过实际信息系统的各种安全要素进行抽象提出的。通过设计相应的免疫算法和规则，实现系统的安全初始状态，在此基础上对系统实施静态防护和动态防护，使得不合法的程序和代码不能够执行，系统程序和文件不被破坏和篡改，篡改或者被破坏的程序在激活前被修，隐藏的恶意代码能够被清除，必要的程序操作可以被引入但是不会造成危害。这种模型不但能够清除攻击和感染源而且能对潜在的攻击进行预警，对未知恶意代码具有免疫功能。该模型不考虑具体恶意代码的特征，避免了大量的特征码计算比较的工作量以及特征库的存储空间。由于模型是通过对实际信息系统的各种安全要素进行抽象后提出的，因此具有较广泛的适用性。

参考文献

[1] Forrest S. Self-nonself Discrimination in a Computer[C]//Proceedings of 1994 IEEE Symposium on Research in Security and Privacy. Oakland, Calif, USA: IEEE Computer Society Press, 1994: 202-212.

[2] 洪　征, 吴礼发, 胡谷雨, 等. AIS 在计算机安全领域的应用与展望[J]. 解放军理工大学学报: 自然科学版, 2005, 6(6): 531-536.

[3] Kim J, Bentley P. An Evaluation of Negative Selection in an Artificial Immune System for Network Intrusion Detection[C]// Proceedings of the Genetic and Evolutionary Computation Conference. San Francisco, California, USA: [s. n.], 2001: 1330-1337.

[4] 孙夫雄, 黄天戍. 入侵检测系统中非完备性问题研究[J]. 计算机工程, 2007, 33(1): 28-30.

[5] Zhou Zheng, Zhang Jun, Li Jian, et al. Protecting Terminals by Security Domain Mechanism Based on Trusted Computing[J]. Wuhan University Journal of Natural Sciences, 2006, 11(6): 1437-1440.

[6] 陈泽茂, 沈昌祥. 基于操作系统安全的计算机病毒防御策略[J]. 武汉理工大学学报, 2004, 26(9): 75-77.

[7] 周　正. 一种主体行为可信度量模型[M]. 计算机工程, 2008, 34(7): 35-37.