基于可信根的计算机终端免疫模型

徐 甫[1,2]

(1. 解放军信息工程大学,河南郑州 450002;2. 北京市信息技术研究所,北京 100094)

摘 要: 人工免疫系统方法中的否定选择(NS)算法已广泛应用于病毒防护、入侵检测、垃圾邮件检测等.然而,由于当前的计算机中不存在类似"免疫器官"的硬件部件,无法对NS算法的运行提供保护,可能造成其运行过程遭受恶意干扰,成熟检测器和中间变量遭受篡改,进而导致其检测结果不可信.借鉴自然免疫系统的组成和原理,提出一种基于可信根的计算机终端免疫模型(TRBCTIM),引入可信计算技术中的可信根作为"免疫器官",对NS算法实施保护.采用无干扰可信模型理论对新模型进行分析,并通过构建新模型的原型系统来进行性能实验.理论分析及实验结果表明,新模型能够确保NS算法的运行过程和检测结果可信.

关键词: 人工免疫系统;可信计算;非自体检测;可信平台控制模块;否定选择算法

中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2016)03-0653-05

电子学报URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2016.03.024

Trusted Root Based Computer Terminal Immune Model

XU Fu[1,2]

(1. *PLA Information Engineering University, Zhengzhou, Henan* 450002, *China*;
2. *Information Technology Institute of Beijing City, Beijing* 100094, *China*)

Abstract: Negative selection (NS) algorithm, a kind of artificial immune system method, is widely applied in virus detection, intrusion detection, spam detection, etc. However, for there is no hardware component like "immune organ" in current computer systems, NS algorithm runs in an unprotected environment, with its running process interfered, mature detectors and intermediate variables interpolated, and then detection results trustless. The composition and principle of natural immune system are simulated and then a trusted root based computer terminal immune model (TRBCTIM) is proposed, which introduces trusted root as "immune organ" from trusted computing technology to protect NS algorithm. Theory of non-interference trusted model is used to analyze the proposed model and the prototype system is constructed to test its performance. Theoretical analysis and experimental results show that the running processes and results of NS algorithm are trusted in the model.

Key words: artificial immune system; trusted computing; non-self detection; trusted platform control module; negative selection algorithm

1 引言

1994年,Forrest等借鉴自然免疫系统中T淋巴细胞生成时的"否定选择"过程,设计了用于病毒检测的NS(Negative Selection,否定选择)算法,在计算机上建立了一个人工免疫系统[1],并于1997年提出了计算机免疫系统的概念[2].之后,人工免疫系统方法广泛应用于计算机安全领域,如病毒防护[1,3,4],恶意软件检测[5]、网络防御及入侵检测[6,7]、垃圾邮件检测[8]、域间路由系统监控[9]等.其中,病毒防护、恶意软件检测和基于主机的入侵检测对计算机终端实施保护,其他手段对计算机网络实施保护.本文研究如何利用人工免疫系统方法对计算机终端实施保护,将构建的安全系统称为计算机终端免疫系统.

2000年,Hofmeyr等利用NS算法构建了ARTIS系统[10],形成了人工免疫系统方法在计算机安全领域应用的一般框架.为避免降低鲁棒性,导致单点失效,ARTIS系统未考虑类似自然免疫系统的"中心耐受"机制,

收稿日期:2014-12-17;修回日期:2015-03-17;责任编辑:覃怀银
基金项目:国家科技重大专项(No.2012ZX03002003)

也未设置类似“胸腺”等免疫器官的部件,而是采用了类似“外围耐受”的分布式否定选择,这一方法被众多学者所接受[4~7]. 在网络中,分布式否定选择可以由多个节点实现,每个节点扮演淋巴节的角色. 然而,在计算机终端免疫系统中,所谓分布式否定选择仅仅是指 NS 算法使用不同的检测器在存储器的不同区域实施否定选择,而算法的运行必须依赖 CPU,受操作系统调度[1,10]. 这一特点与自然免疫系统不符,在自然免疫系统中,经否定选择成熟的 T 淋巴细胞在胸腺、淋巴结、脾等专门的免疫器官中执行“非自体”识别任务,受到免疫器官的有效保护[11]. 而目前的计算机终端免疫系统中却不存在“免疫器官”[10],NS 算法运行于 CPU,算法代码和成熟检测器直接存储于外存,算法运行产生的中间变量直接存储于内存,均可能遭受恶意进程干扰或篡改,导致运行不可信,进而可能出现误判情况. 因此,要想实现真正意义上的计算机终端免疫系统,必须改变现有计算机系统的体系结构,为其添加类似“免疫器官”的硬件部件,对 NS 算法实施有效保护.

可信计算技术正是为了改善计算机系统的体系结构,从根本上解决其面临的安全问题而产生. 其基本思想是[12]:通过建立信任根和信任链,一级测量认证一级,一级信任一级,从而把信任扩展到整个计算机系统. 其中,信任根又称为可信根,由主板上的可信芯片实现,不依赖 CPU 而独立运行,其可信性由物理安全、技术安全和管理安全共同保证[12]. 可信根的这些特点使其能够胜任计算机终端免疫系统中“免疫器官”的角色,为 NS 算法提供有效的保护.

2 NS 算法、可信计算技术与理论研究现状

2.1 NS 算法

NS 算法借鉴自然免疫系统中 T 淋巴细胞生成时的“否定选择”,首先执行检测器生成过程,即随机生成检测器,用其与自体集合中的每一元素进行匹配操作,如果不能匹配,则将其纳入成熟检测器集合,否则,丢弃该检测器;然后执行非自体检测过程,即从受保护的数据中选择个体,与成熟检测器集合中的所有元素进行匹配操作,如果能够匹配,则判定为非自体,否则,判定为自体.

近年来 NS 算法在国内外得到了迅速发展,衍生出许多变种,逐渐形成了 NS 算法簇. 关于 NS 算法的详细分类及描述,可参阅文献[13].

2.2 可信计算技术与理论

2.2.1 可信根

目前,受到广大研究者关注的可信根主要有三种[14]:TCG(Trusted Computing Group,可信计算组织)的 TPM(Trusted Platform Module,可信平台模块)、我国的 TCM(Trusted Cryptography Module,可信密码模块)和 TPCM(Trusted Platform Control Module,可信平台控制模块). 其中,TPM 和 TCM 的设计在总体上是很成功的,但也存在明显的不足[14],如被设计为被动部件,缺少对平台的主动控制作用;存储于 TPM/TCM 之外的可信度量根易受到恶意攻击等. 针对这些问题,我国正在研究制定自己的 TPCM 规范[12,14,18],其主要技术特点在于具备主动度量功能,既包括平台启动时首先掌握对平台的控制权,又包括平台启动后对平台关键部件的完整性度量,详细描述可参阅文献[14].

2.2.2 无干扰可信模型理论

由于可信计算技术来源于工程实践,导致目前尚没有公认的可信计算理论模型[12],但许多学者对此进行了有益的探索. 我国学者张兴等[15]将无干扰理论[19]引入了基于进程的可信模型研究,给出了进程运行可信的条件,并证明了系统运行可信定理. 本文将采用文献[15]的方法对计算机系统进行形式化,相关符号及函数的定义可参阅该文献.

秦晰等[16]分析指出,进程的运行过程仅满足无干扰理论的条件时,尚不能够确定其是否可信,需同时保证其代码的完整性,即:

定义 1[16] 称进程 p 运行可信,当满足条件

$\mathrm{output}(\mathrm{run}(s_0,\alpha),a)=\mathrm{output}(\mathrm{run}(s_0,\mathrm{purge}(\alpha,\mathrm{proc}(a))),a)$ 和 $\mathrm{Hash}(p)=\mathrm{expect}(p)$,其中 $p=\mathrm{proc}(a)$,$\mathrm{Hash}(p)$ 表示对进程 p 的代码进行 Hash 运算,$\mathrm{expect}(p)$ 为进程 p 的代码的完整性校验预期值.

3 TRBCTIM 模型

3.1 NS 进程可信运行的条件

NS 进程(用符号表示为 p_{NS})表示 NS 算法在运行时形成的进程. 它对计算机中存储的所有数据进行扫描,能够观察到所有的存储区域. 因此,系统中运行的任何进程都能够改变其系统视图[17],即对其造成干扰. 系统策略必须规定:$\forall p\in P, p\sim>p_{NS}$,才能够保证 p_{NS} 能够正常发挥作用. 但是,这并不意味着某一进程可以篡改 p_{NS} 的成熟检测器和中间变量,否则,p_{NS} 的检测结果将不准确. 也就是说,任何进程对 p_{NS} 的干扰只能通过改变 p_{NS} 系统视图中的输入区域进行,而不能改变其成熟检测器和中间变量. 因此,要使得 p_{NS} 运行可信,不仅需要满足定义 1 中的一般进程运行可信的条件,还需要满足成熟检测器和中间变量不被除 p_{NS} 自身以外的任何进程修改这一条件. p_{NS} 运行可信的条件可形式化表示为:

定义 2 称 p_{NS} 运行可信,当满足条件:

(1) $\mathrm{output}(\mathrm{run}(s_0,\alpha),a)=\mathrm{output}(\mathrm{run}(s_0,\mathrm{purge}(\alpha,\mathrm{proc}(a))),a)$,其中 $p_{NS}=\mathrm{proc}(a)$;

(2) $\mathrm{Hash}(p_{NS})=\mathrm{expect}(p_{NS})$;

(3) $\forall o\in \mathrm{DetectorString}(p_{NS})\cup \mathrm{IntermediateVariable}$

(p_{NS}), $\forall p \in P/\{p_{NS}\}$, $o \notin \text{alter}(p)$,其中,DetectorString(p_{NS})表示 p_{NS}运行时需访问的所有成熟检测器集合,IntermediateVariable(p_{NS})表示 p_{NS}运行时需访问的所有中间变量集合,alter(p)表示进程 p 能够修改的对象集合.

3.2　TRBCTIM 模型原理及运行流程

由于 TPM 和 TCM 不能保证输入数据和度量结果的可信,而采用主动度量的 TPCM 则能够有效解决这一问题[20]. 因此,为满足定义 2 给出的 p_{NS}运行可信的三个条件,我们选用 TPCM 来保护 NS 算法,并设计了 TRBCTIM 模型的基本框架,如图 1 所示. 含 NS 算法引擎的 TPCM 与普通 TPCM 相比,有如下几点不同:

(1)增加了 NS 算法引擎,由 TPCM 芯片对 NS 算法代码实施保护;

(2)在 TPCM 芯片外配置非易失存储器,用于保存成熟检测器,只能由 NS 算法引擎访问;

(3)NS 算法引擎能够通过接口控制器直接访问计算机内存和外存;

(4)NS 算法的中间变量存储于片内易失存储器,由 TPCM 芯片实施保护.

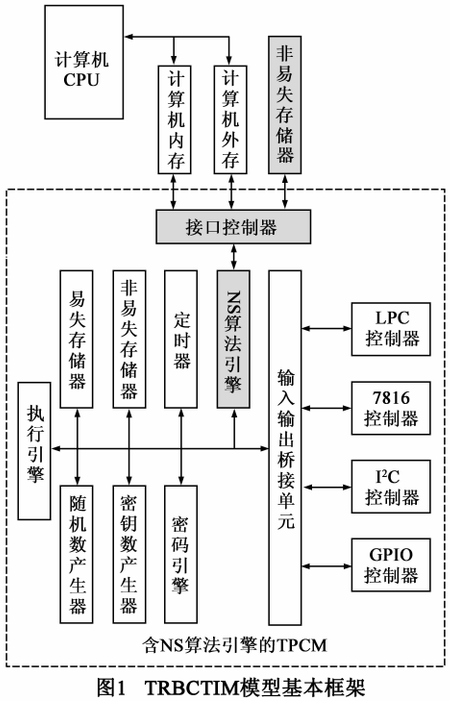


图1　TRBCTIM模型基本框架

TRBCTIM 模型的运行步骤如下:

Step1 在无外来入侵的封闭环境下,计算机系统正常运行,NS 算法引擎对内存和外存中的数据实施检测器生成过程,将成熟检测器存储于片外非易失存储器.

Step2 每次开机时,TPCM 首先启动,对外存中的数据实施非自体检测过程,如果发现非自体,则将其清除.

Step3 TPCM 将系统控制权交给 CPU,在系统启动及启动后的运行过程中,NS 算法引擎对成功写入至内存和外存的数据实施非自体检测,发现非自体即清除.

4　TRBCTIM 模型分析

4.1　理论分析

定理 1　TRBCTIM 能够确保 p_{NS}运行可信.

证明　要证明 p_{NS}运行可信,只要证明其满足定义 2 的三个条件即可.

(1)令 $\alpha = a_1 a_2 \cdots a_n$,由于 p_{NS}能够访问计算机内存和外存的所有数据,因此有 $\forall a_i$, $\text{proc}(a_i) \sim > p_{NS}$,那么,

$$\text{purge}(\alpha, p_{NS}) = \text{purge}(a_1 a_2 \cdots a_n, p_{NS}) = a_1 \circ \text{purge}(a_2 a_3 \cdots a_n, p_{NS}) = a_1 a_2 \circ \text{purge}(a_3\ a_4 \cdots a_n, p_{NS}) = \cdots = a_1 a_2 \cdots a_n = \alpha \quad (1)$$

由式(1)可得,$\text{output}(\text{run}(s_0, \text{purge}(\alpha, p_{NS})), a) = \text{output}(\text{run}(s_0, \alpha), a)$,即定义 2 的条件(1)成立.

(2) 由于 p_{NS}运行于 TPCM 芯片内部,受到 TPCM 芯片的有效保护,因此,其代码不会受到篡改,即 Hash(p_{NS}) = expect(p_{NS})成立;

(3) 对于任意成熟检测器 $d \in \text{DetectorString}(p_{NS})$,由于其存储于片外非易失存储器,且该存储器只能由 NS 算法引擎访问,因此,

$$\forall p \in P/\{p_{NS}\}, d \notin \text{alter}(p) \quad (2)$$

对于任意中间变量 $v \in \text{IntermediateVariable}(p_{NS})$,由于其存储于片内易失存储器,受到 TPCM 芯片的有效保护,因此, $\forall p \in P/\{p_{NS}\}$, $v \notin \text{alter}(p)$ (3)

综合式(2)、式(3),得 $\forall o \in \text{DetectorString}(p_{NS}) \cup \text{IntermediateVariable}(p_{NS})$, $\forall p \in P/\{p_{NS}\}$, $o \notin \text{alter}(p)$,即定义 2 的条件(3)成立. 证毕.

4.2　仿真分析

4.2.1　原型系统构建

在 Xen 虚拟机[21]框架下构建了 TRBCTIM 模型的原型系统,如图 2 所示. Dom0 为 Xen 的特权虚拟域,Dom1 和 Dom2 均为运行 Windows XP 操作系统的硬件虚拟域. 在 Xen Hypervisor 中为 Dom1 构建了含 NS 算法引擎的 TPCM. 带箭头实线表示 Dom1 对物理硬盘、物理内存、物理 CPU 等的访问,带箭头虚线表示 NS 算法引擎对 Dom1 所占用的物理硬盘、物理内存的访问,以及对 Dom1 向硬盘、内存写入数据行为的监视. Dom2 通过虚拟网络连接同 Dom1 相连,模拟攻击者对 Dom1 实施攻击.

在设计 NS 算法引擎时选择了一种基于字符串的 NS 算法——r 可变 NS 算法[22]. 字符串长度 l 选择为 80,初始匹配阈值 r_1选择为 13,最大匹配阈值 r_c选择为 15,随机生成的未成熟检测器数量 N_{R0}选择为 1100,经过耐受生成的成熟检测器数量为 $N_R = 946$.

4.2.2　抗攻击性能仿真

为了同一般计算机终端免疫系统进行比较,我们

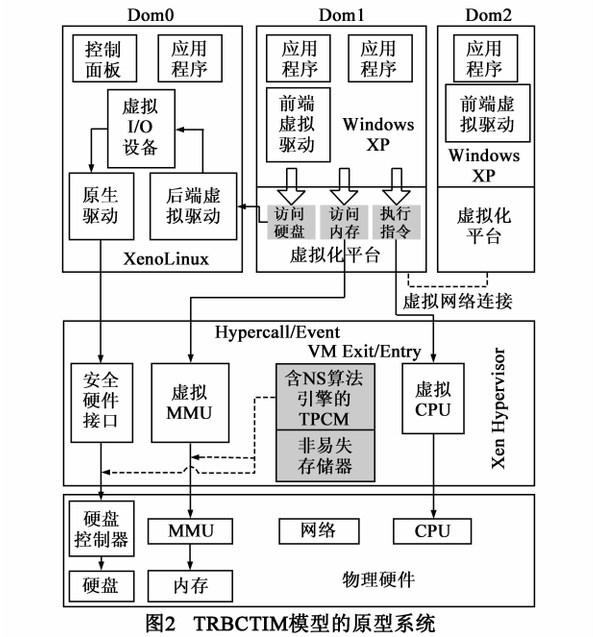


图2 TRBCTIM模型的原型系统

构建了另一种保护系统,即去掉图 2 中的 TPCM,而在 Dom1 的 Windows XP 操作系统中实现 r 可变 NS 算法,对其虚拟硬盘中的某一固定存储区域实施保护. Dom2 利用 Windows XP 系统提供的远程桌面连接功能,模拟黑客远程控制 Dom1,向该存储区域拷贝 100 个良性病毒文件作为非自体,并采用以下三种攻击方法分别对该保护系统进行攻击实验:

攻击实验 1:用随机字符串替换 Dom1 中 NS 算法的成熟检测器. 随着被替换的检测器数量的增加,NS 算法的检测率变化如图 3 所示.

攻击实验 2:将 Dom1 中 NS 算法执行时输入缓冲区中的数据篡改为随机数. 随着被篡改的输入缓冲区的比重的增加,NS 算法的检测率变化如图 4 所示.

攻击实验 3:将 Dom1 中 NS 算法的可执行程序的前 100 字节篡改为随机数. 篡改后,NS 算法程序无法启动.

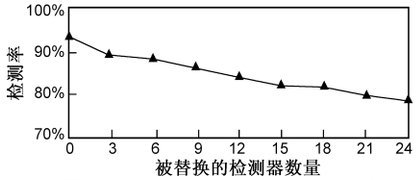


图3 攻击实验1中NS算法的检测率

在图 2 所示的原型系统中,Dom2 采用相同的方法,向 Dom1 的虚拟硬盘中拷贝 100 个良性病毒文件作为非自体,计算 NS 算法对非自体的检测率,并重复 8 次该过程. 由于成熟检测器、中间变量和算法代码都受到 TPCM 芯片或片外非易失存储器的保护,三种攻击方法都无法实施. r 可变 NS 算法的检测率随实验次数的变

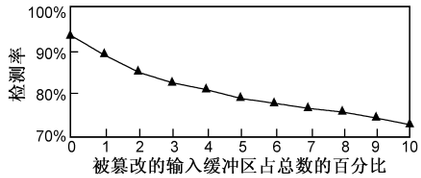


图4 攻击实验 2 中NS算法的检测率

化曲线如图 5 所示.

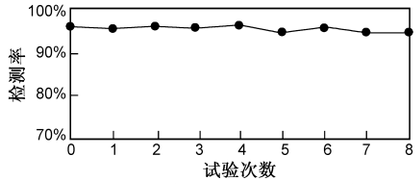


图5 TPCM中实现的NS算法的检测率

由图 5 可知,TPCM 中实现的 r 可变 NS 算法的检测率一直保持在 95% 左右,这与文献[22]给出的实验结果一致. 而在对 Windows XP 中实现的 r 可变 NS 算法的前两种攻击实验中,其检测率最初也接近 95%,但随着被篡改的检测器和输入缓冲区的增多,本来能够匹配的检测器和非自体不再能够匹配,导致其检测率逐渐下降,算法运行变得不可信. 攻击实验 3 表明当算法代码受到篡改后,算法无法运行,当然更加不可信. 而在图 2 所示的 TRBCTIM 模型原型系统中,三种攻击方法都无法实施,因此,r 可变 NS 算法能够可信运行,保证了算法的检测率.

5 结束语

本文构建了基于可信根的计算机终端免疫模型 TRBCTIM,为计算机终端免疫系统引入了采用主动度量机制的可信根 TPCM,使用其充当类似于自然免疫系统中的“免疫器官”的角色,同时在片外引入了非易失存储器,用来保存成熟检测器. TPCM 及片外非易失存储器为 NS 进程提供独立的运行环境和运行时的保护,避免了恶意进程对 NS 算法代码、NS 算法运行时的中间变量以及成熟检测器进行恶意篡改. 理论分析及实验结果表明,TRBCTIM 模型中,NS 进程能够可信运行.

参考文献

[1] Forrest S, Perelson A S, Allen L, et al. Self-nonself discrimination in a computer[A]. Proceedings of 1994 IEEE Symposium on Research in Security and Privacy[C]. Oakland, CA, USA: IEEE, 1994. 202 – 212.

[2] Somayaji A, Hofmeyr S A, Forrest S. Principles of acomputer immune system[A]. Proceedings of New Security Paradigms Workshop[C]. Langdale, UK: ACM, 1997. 75 – 82.

[3] 程春玲,柴倩,徐小龙,等. 一种用于病毒检测的协作免疫

网络算法[J]. 电子学报,2013,41(12):2518 – 2522.
Cheng Chun-ling, Chai Qian, Xu Xiao-long, et al. A cooperative immune network algorithm for virus detection[J]. Acta Electronica Sinica, 2013, 41(12): 2518 – 2522. (in Chinese)

[4] 芦天亮. 基于人工免疫系统的恶意代码检测技术研究[D]. 北京:北京邮电大学,2013.
Lu Tian-liang. Research on Malcode Detection Technology Based on Artificial Immune System[D]. Beijing: Beijing University of Post and Telecommunications, 2013. (in Chinese)

[5] Ali M A M, Maarof M A. Dynamic innate immune system model for malware detection[A]. Proceedings of 2013 International Conference on IT Convergence and Security[C]. Macau, China: IEEE, 2013. 1 – 4.

[6] Elhaj M M K, Hamrawi H, Suliman M M A. Amulti-layer network defense system using artificial immune system[A]. Proceedings of 2013 International Conference on Computing, Electrical and Electronic Engineering[C]. Khartoum, Sudan: IEEE, 2013. 232 – 236.

[7] Kumar G V P, Reddy D K. An agent based intrusion detection system for wireless network with artificial immune system (AIS) and negative clone selection[A]. Proceedings of 2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies[C]. Nagpur, India: IEEE, 2014. 429 – 433.

[8] Zitar R A, Hamdan A. Genetic optimized artificial immune system in spam detection: a review and a model[J]. Artificial Intelligence Review, 2013, (40): 305 – 377.

[9] Guo Yi, Wang Zhen-xing. An immune-theory-based model for monitoring inter-domain routing system[J]. Science China: Information Sciences, 2012, 55(10): 2358 – 2368.

[10] Hofmeyr S A, Forrest S. Architecture for an artificial immune system[J]. Evolutionary Computation, 2000, 7(1): 45 – 68.

[11] 金伯泉. 医学免疫学[M]. 北京:人民卫生出版社,2008. 12 – 23.

[12] Shen Chang-xiang, Zhang Huan-guo, Feng Deng-guo, et al. Survey of information security[J]. Science in China Series F: Information Science, 2007, 50(3): 273 – 298.

[13] 金章赞,廖明宏,肖刚. 否定选择算法综述[J]. 通信学报,2013,34(1):159 – 170.
Jin Zhang-zan, Liao Ming-hong, Xiao Gang. Survey of negative selection algorithms[J]. Journal on Communications, 2013, 34(1): 159 – 170. (in Chinese)

[14] 张焕国,赵波. 可信计算[M]. 武汉:武汉大学出版社,2011. 61 – 107.

[15] 张兴,陈幼雷,沈昌祥. 基于进程的无干扰可信模型[J]. 通信学报,2009,30(6):6 – 11.
Zhang Xing, Chen You-lei, Shen Chang-xiang. Non-interference trusted model based on processes[J]. Journal on Communications, 2009, 30(6): 6 – 11. (in Chinese)

[16] 秦昕,常朝稳,沈昌祥,等. 容忍非信任组件的可信终端模型研究[J]. 电子学报,2011,39(4):934 – 939.
Qin Xi, Chang Chao-wen, Shen Chang-xiang, et al. Research on trusted terminal computer model tolerating untrusted components[J]. Acta Electronica Sinica, 2011, 39(4): 934-939. (in Chinese)

[17] 张兴,黄强,沈昌祥. 一种基于无干扰模型的信任链传递分析方法[J]. 计算机学报,2010,33(1):74 – 81.
Zhang Xing, Huang Qiang, Shen Chang-xiang. Aformal method based on noninterference for analyzing trust chain of trusted computing platform[J]. Chinese Journal of Computers, 2010, 33(1): 74 – 81. (in Chinese)

[18] 张兴,沈昌祥. 一种新的可信平台控制模块设计方案[J]. 武汉大学学报 · 信息科学版,2008,33(10):1011 – 1014.
Zhang Xing, Shen Chang-xiang. Anovel design of trusted platform control module[J]. Geomatics and Information Science of Wuhan University, 2008, 33(10): 1011 – 1014. (in Chinese)

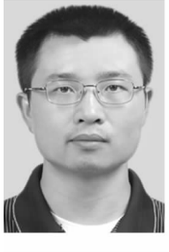
[19] Rushby J. Noninterference, Transitivity, and Channel-Control Security Policies[R]. Menlo Park: Stanford Research Institute, 1992.

[20] 郭颖,毛军捷,张翀斌,等. 基于可信平台控制模块的主动度量方法[J]. 清华大学学报(自然科学版),2012,52(10):1465 – 1473.
Guo Ying, Mao Jun-jie, Zhang Chong-bin, et al. Active measures based on a trust platform control module[J]. Journal of Tsinghua Unversity (Science & Technology), 2012, 52(10): 1465 – 1473. (in Chinese)

[21] 石磊,皱德清,金海. Xen 虚拟化技术[M]. 武汉:华中科技大学出版社,2009. 33 – 33.

[22] 张衡,吴礼发,张毓森,等. 一种 r 可变阴性选择算法及其仿真分析[J]. 计算机学报,2005,28(10):1614 – 1619.
Zhang Heng, Wu Li-fa, Zhang Yu-sen, et al. An algorithm ofr-adjustable negative selection algorithm and its simulation analysis[J]. Chinese Journal of Computers, 2005, 28(10): 1614 – 1619. (in Chinese)

作者简介



徐　甫　男,1983 年 3 月出生于江苏省淮安市. 现为解放军信息工程大学博士研究生. 主要研究方向为信息安全、可信计算.
E-mail:ilyxufu@163.com