基于 Clark-Wilson 完整性策略的安全监视模型*

卿斯汉[1,2], 温红子[1,2+], 雷 浩[2,3], 王 建[2,4]

[1](中国科学院 软件研究所 信息安全技术工程研究中心,北京 100080)

[2](中国科学院 研究生院,北京 100039)

[3](中国科学院 软件研究所 信息安全国家重点实验室,北京 100080)

[4](中国科学院 计算机网络信息中心 超级计算中心,北京 100080)

A Secure Monitoring Model Based on the Clark-Wilson Integrity Policies

QING Si-Han[1,2], WEN Hong-Zi[1,2+], LEI Hao[2,3], WANG Jian[2,4]

[1](Engineering and Research Center for Information Security Technology, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

[2](Graduate School, The Chinese Academy of Sciences, Beijing 100039, China)

[3](State Key Laboratory of Information Security, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

[4](Supercomputing Center, Computer Network Information Center, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: +86-10-62561197 ext 8006, E-mail: wenhongzi@msn.com, http://www.iscas.ac.cn

Received 2003-11-03; Accepted 2004-03-31

Qing SH, Wen HZ, Lei H, Wang J. A secure monitoring model based on the Clark-Wilson integrity policies. *Journal of Software*, 2004,15(8):1124~1132.

http://www.jos.org.cn/1000-9825/15/1124.htm

Abstract: The redundant data in log files and the delay for detecting abnormal trails are the inherent problems existing in the traditional secure monitoring subsystem of a computer system. In this paper, it is identified that the system security policies determine the logging data items in a secure monitoring function. By formally describing and analyzing the famous Clark-Wilson integrity policies with the corresponding relation patterns, the minimal logging data items set involved in these security policies is precisely determined. A formal secure monitoring model based on Clark-Wilson integrity policies (CW-SMM) is proposed. The CW-SMM has the characteristics of both minimal logging data and auto-detecting of the system abnormal trails in time, and can thoroughly solve the problems mentioned above.

Key words: Clark-Wilson integrity policy; relation pattern; secure monitoring; logging; auditing

摘 要: 传统的计算机设计系统的安全监视功能存在日志数据冗余和异常线索检测时延过长等固有问题.由于

* Supported by the National Natural Science Foundation of China under Grant No.60083007 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.G1999035810 (国家重点基础研究发展规划(973))

作者简介: 卿斯汉(1939－),男,湖南隆回人,研究员,教授,博士生导师,主要研究领域为信息安全理论与技术;温红子(1969－),男,工程师,主要研究领域为信息安全理论与技术;雷浩(1975－),男,博士生,主要研究领域为系统安全与信息安全技术;王建(1976－),男,博士生,主要研究领域为并行非线性最优化技术,并行计算安全技术.

安全监视功能的日志数据项主要是由系统实施的安全策略所决定,所以采用关系模式,通过形式地描述、分析著名的 Clark-Wilson 完整性策略,从而精确确定了与各条策略相关的最小日志项集,然后将其应用于基于 Clark-Wilson 完整性策略的形式化安全监视模型(CW-SMM).该模型不但可以有效解决 Clark-Wilson 安全策略适用系统的日志数据冗余问题,而且也可以彻底解决异常线索检测中的时延问题.

关键词: Clark-Wilson 完整性策略;关系模式;安全监视;日志;审计

中图法分类号: TP316 **文献标识码**: A

审计的安全监视功能是安全信息系统的一个基本安全功能.它通过记录和检查系统安全相关事件来检测由于外部渗透和内部误用所引起的各种系统异常[1].2001 年,Simone 在文献[2]中有关审计的综合分析认为:在多数现有的安全系统中,日志是自动的,而基于日志的审计活动则是以人工分析为主*.为了利用审计活动达到相当程度的安全性,就要求尽可能多的日志系统活动数据.这样,一方面给系统的性能和存储空间带来了较重的负担,另一方面也使得审计分析日志数据时延增加,从而系统异常可能不被侦测到或者只能在它们发生后的很长时间才能被侦测到.

显然,要有效解决上述问题,就应从日志数据项精确选择和系统异常机器自动检测两方面着手.由于系统审计的目的在于"为系统安全策略的有效实施提供一个附加级别的用户保障"[4],这种保障主要通过基于审计的安全监视功能来提供,亦即安全监视功能通过对当前系统状态一致性的判定,给用户提供系统安全策略有效实施的情况.所以,依据系统安全策略来精确确定日志数据项和自动检测系统异常是一条有效的途径.

在商务安全领域,Clark 和 Wilson 于 1987 年提出的 Clark-Wilson 完整性策略[5]被认为是完整意义上的完整性目标、策略和机制的起源,可以有效满足企业信息系统所追求的完整性安全需求[2,6].因此,研究 Clark-Wilson 完整性策略下安全系统的安全监视功能具有重要的理论和实践意义.本文提出了基于形式化 Clark-Wilson 完整性策略的安全监视模型,用以克服 Clark-Wilson 安全策略适应系统中存在的有关安全监视的前述问题.基本做法是,首先参照 Clark-Wilson 完整性策略生成各种审计目标和日志项目集,然后系统主要参照日志项目集来生成日志数据,最后依据审计目标和策略来对日志数据进行合法性分析,从而达到对系统的状态进行及时的一致性检测的目的.

本文第 1 节给出 Clark-Wilson 完整性策略关系模式.第 2 节提出基于该安全策略的形式化安全监视模型.第 3 节是对模型的综合分析.第 4 节为结论.

1 Clark-Wilson 完整性策略的形式化关系模式

当系统中出现违反安全策略的事件时,系统安全状态就处于非法状态(非一致性状态),相关的状态信息具体是由日志数据来体现的.为了验证系统状态的一致性,必须把安全策略应用于这些状态信息,评估当前状态信息对安全策略的满足情况,从而达到安全监视的目的.但是,Clark-Wilson 完整性策略的陈述形式是很难直接应用于这种以机器为主的自动判定操作之中的,因此必须先把 Clark-Wilson 完整性策略条目改写为适当的易于应用的形式.本文采用关系模式这种形式表示方式.

1.1 关系和模式

Özsu 指出,各种完整性规则的实质是用于定义维持系统一致性状态的各种约束,这些约束常用一致性断言的形式表达[7].但是,关系是状态相关的,而策略是状态无关的,显然关系不能直接用来描述策略,因此本文采用关系模式来描述策略.这种策略的关系模式只有和与其相关的系统状态信息(称为系统状态相关部分)结合后才可以形成可用以判定当前状态合法性(完整性)的具体关系.

---

* 尽管在许多文献中,日志和审计这两个概念可以互换使用,但本文为了精确起见,依据 Bishop 在文献[3]中的提法,认为日志(logging)和审计(auditing)分别描述了两个不同的行为:日志是简单的形成一条记录,而审计则被用以分析这条记录.但是,仍把日志操作和审计操作一起统称为审计.例如,当提到审计系统时就意味着它包括日志和审计两部分.

Clark-Wilson 完整性策略的一个特点在于全面规范了影响系统完整性的各个方面,但这些影响是由系统中的各种操作来施加的,只有先显式定义隐含在各策略条目中的操作,然后才可以籍此更加深入地研究这些操作对系统的具体影响方式和结果.所以,下面我们首先定义用以表示这些操作的基础函数.

1.2 基础函数

1. 身份分类函数: $IdentityClass: USERIDs \to IDCLASSs$. 用以甄别系统中的用户身份类型.这里,$IDCLASSs=\{AuthUser,EcexUser\}$,为用户身份类别集.上述函数是说一个用户身份不是属于 AuthUser 类(安全官员、系统所有者和系统管理员等特权用户身份),就是属于 ExecUser 类(执行用户身份,也即执行具体变换过程的用户身份).特权用户(AuthUser)和执行用户(ExecUser)分别用以刻画被授权去验证实体的代理(AuthUser)和普通执行具体变换过程的用户(ExecUser).特权用户可以更改一个实体和其他实体之间的关联列表,但是同时要求他不可以具有任何有关那个实体的执行权限,后者则为可以执行变换过程的普通用户.

2. 验证函数模式: $Cert_{ENT}: ENTs \times USERIDs \times ATTRs \to CERTENTs$. $Cert_{ENT}$是一个针对所有验证策略条目的通用验证函数模式.这里,$ENTs$ 为系统实体集(这里主要有 3 种类型的实体集,即完整性验证过程集 $IVPs$、变换过程集 TPs 和变换过程片集 $PTPs$),$CERTENTs$ 为已被验证的实体集(可以分别表示为 $CERTIVPs$,$CERTTPs$ 和 $CERTPTPs$),相应地,$Cert_{ENT}$ 可以分别表示为 $Cert_{IVPs}$,$Cert_{TPs}$ 和 $Cert_{PTPs}$.$USERIDs$ 为认证者用户身份,$ATTR_s$ 为被认证实体通过验证时所必须保持的关系集.

对一个实体进行验证的目的在于考察该实体对一个给定约束集的可满足性,该约束集一般包括对验证实体的结构、授权控制、组件关联等方面的约束.在本文中是用关系来描述这种可满足性的,这样就把约束集转化为关系集.此时只有当关系集中的所有关系都保持时,实体才通过了验证.

3. 变换过程执行函数: $Exec: USERIDs \times TPs \to CTPs$. 这里,$CTPs$ 为当前激活变换过程集,通过某一用户身份 $userID$ 使用 $Exec$ 函数把一个变换过程 TP 变成活动变换过程 CTP.此执行过程必须满足变换过程激活关系:

$$R_{Exec-TP}: \forall CTP \in CTPs \quad \exists TP \in TPs, userID \in USERIDs$$
$$[CTP = Exec(userID, TP) \wedge IdentityClass(userID) = ExecUser].$$

任何一个激活变换过程都是通过 $ExecUser$ 类用户执行变换过程执行函数来获得的.

4. 聚组函数: $Group: (ENTITY, s) \to 2^{ENTITY}$. 这里,$ENTITY$ 为目标集,s 为聚组标准.该函数的作用是把实体集 $ENTITY$ 中与 s 相关的所有元素取出,形成一个新的 s 相干集合.

5. 通用元组项提取函数: $GetItem(Tuple, index) \to Tuple[index]$. 此处,$Tuple$ 为元组名,$index$ 为元组项的索引值,其值从 1 开始,$Tuple[index]$代表元组中第 $index$ 项的值.该函数主要用于提取一个元组中的特定项的值.

6. 变换过程分割函数: $Part_{TP-PTPs}: TPs \to 2^{PTPs}$. 其中,$TPs$ 为变换过程集,$PTPs$ 为变换过程片集,任务分割函数的作用是把一个变换过程划成几个互不相同的变换过程片,也即各变换过程片分别具有两两不同的操作集.

7. 用户身份认证函数: $Assign_{user-IDToken}: USERs \to IDTOKENs$. 用户身份认证函数用于处理系统用户的身份和权能之间的关联问题.当一个用户通过系统的身份认证后,就获得一个用户身份权能元组 $IDToken$,其结构为 $(userID, Token)$. $userID \in USERIDs$ 为用户身份. $Token \in TOKENs$ 为用户所获得的权能令牌,也即可以存取的变换过程标志符集合.系统中所有的 $IDToken$ 构成了用户身份权能元组集 $IDTOKENs$.这里,$USERs$ 为用户集.

1.3 Clark-Wilson完整性策略的关系模式

为了便于和 Clark-Wilson 完整性策略进行比较、分析,Clark-Wilson 完整性策略的关系模式分 6 条来描述,分别标识为 C1,C2&C5,E1,E2,C3 和 E3,它们之间的关系如图 1 所示.图中 CDI 为受控数据项,UDI 为非受控数据项,LOG CDI 为日志受控数据项,TP 为变换过程,IVP 为完整性验证过程.具体定义参见文献[5].

1. (C1)本策略要求只有当所有的 IVP 都通过验证后,才可以确认系统中的 $CDIs$ 处于一致状态.这里,用 Denning[8]的关系定义来描述 $CDIs$ 上的一致性状态条件.$r(A_1:D_1,\ldots,A_n:D_n)$是一个数据关系模式,这里,对于每一个 $A_i, 1 \le i \le n, Dom_i$ 是 D_i 域的值集,一个数据关系 r(满足该模式中的域约束,也称为实例)是一个有 n 个字段的元组集: $r: \{\langle a_1:d_1,\ldots,a_n:d_n\rangle | d_1 \in Dom_1,\ldots,d_n \in Dom_n\}$,$\langle\ldots\rangle$表明一个元组的字段.关系模式是状态独立的,而关系(实例)则是状态相关的.当 $CDIs=\{D_1,\ldots,D_n\}$时,则称 $r:\langle a_1:d_1,\ldots,a_n:d_n\rangle$为受控数据关系,记为 r_{CDIs},记 R_{CDIs} 为作用在 $CDIs$

上的所有受控数据关系的集合,称为受控数据关系集.

在 C1 中,实体集 *ENTs* 为完整性验证过程集 *IVPs*,相应的验证实体集 *CERTENTs* 为验证完整性验证过程集 *CERTIVPs*,验证函数为 $Cert_{IVP}$.任一个 *IVPs* 要通过验证,必须满足关系集 R_{CDIs} 中与该 IVP 相关的部分 R_{IVP}(由蕴含在 IVP 编程逻辑中的用以检查部分 *CDIs* 一致性的程序功能来体现).当然,只有所有 IVP 都通过验证后才可以确保整个系统处于一致状态,则有以下关系:

$$R_{C1}:\left[\forall CERTIVP\in CERTIVPs \quad \exists IVP\in IVPs, userID\in USERIDs, R_{IVP}=Group(R_{CDIs}, IVP)\right.$$

$$\left.[CERTIVP=Cert_{IVPs}(IVP, userID, R_{IVP})\wedge IdentityClass(userID)=AuthUser]\right]\wedge\left[\bigcup_{IVP\in CERTIVPs} R_{IVP}=R_{CDIs}\right].$$

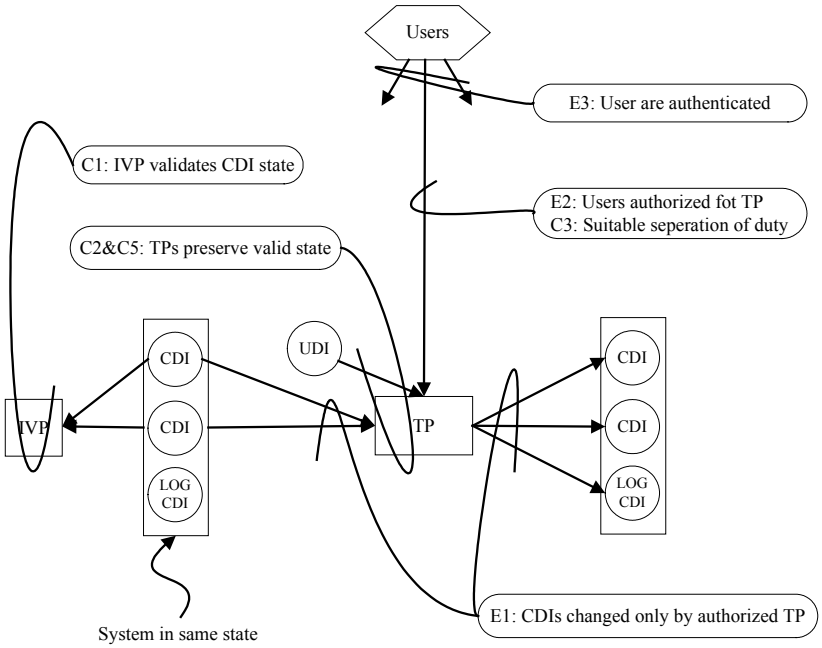

Fig.1 Illustration of Clark-Wilson relation patterns

图 1 Clark-Wilson 关系模式示意图

对 R_{C1} 可从两方面来认识.一方面,对每个 IVP 而言,只有当与它相关的作用在 *CDIs* 上的关系集 R_{IVP} 中的所有关系都为真时,该验证过程才可以通过.这也就表明,对当前 IVP 而言,此时相关的受控数据处于一致状态.而且也要求该验证过程只能由 AuthUser 用户身份来完成.另一方面,由于整个受控数据状态的一致性由 R_{CDIs} 来体现,所以要求当所有的 IVP 通过验证后,总效果必须等同于 R_{CDIs},才可以确信此时整个系统处于一致状态.

由于关系模式 R_{C1} 通过读取 *CDIs*(由关系 R_{CDIs} 来间接引用),*IVPs* 和 *USERIDs* 等系统状态信息来生成具体关系,所以 C1 的策略系统状态相关部分为 $\sigma_{C1}=\{CDIs, IVPs, USERIDs\}$.

2. (C2&C5)主要用于处理对变换过程的合法性验证问题.在 Clark-Wilson 完整性安全策略 C2 和 C5 中是把 *CDIs* 和 *UDIs* 分别对待的.但如果把 *UDIs* 看作一种特殊的 *CDIs*,相关变换过程的语义和结构就可以认为基本上相同.其差异之处只是在于对作为变换过程输入的 *UDIs* 进行适当的限制,亦即当把 *UDIs* 作为普通 *CDIs* 看待之前必须进行必要的质量保证工作,这种要求可以转化为验证变换过程所应具有的一个特性 R_{UDIs},当某个变换过程存在对 *UDIs* 的引用时,要求必须满足 R_{UDIs}.

变换过程的完整性要求可以归结为两个方面:对变换过程的结构要求和对变换过程的语义要求.对于前者,使用变换过程的结构定义来满足,而对于后者则用状态保持关系来描述.

一个 TP 由作用在 *CDIs* 中各个元素上的操作序列构成,这些操作都是预先确定的操作集 *OPs* 中的元素,所以,一个 TP 可以形式化地定义为 $TP=\{(op, CDI)\,|\,op\in OPs, CDI\in CDIs\}$.

需要注意的是,由于在 TP 的定义中已经蕴涵了关系$(TP_i, CDI_a, CDI_b, CDI_c, \ldots)$,所以关系$(TP_i, CDI_a, CDI_b,$

CDI_c,…)此时已转化为对 TP 结构的要求,则有变换过程结构蕴涵关系:

$$R_{TPStruct}:\forall TP\in TPs \quad [\forall (op,CDI)\in TP \quad [op\in OPs, CDI\in CDIs]].$$

这就是说,对于一个变换过程中的所有(*op*,*CDI*)元组,其中 *op* 和 *CDI* 分属于预先定义的操作集和受控数据集.

变换过程的语义要求是,一个变换过程必须能够把 *CDIs* 从一个一致状态带到下一个一致状态.由于系统的受控数据状态是由作用在其上的受控数据关系集 R_{CDIs} 决定的,所以,对变换过程必须把 *CDIs* 从一个一致状态(使用 R_{CDIs} 来表示)带到下一个一致状态(使用 R'_{CDIs} 来表示)的语义要求,可以进一步抽象为状态保持关系 R_{sp}:

$$R_{sp}:\forall TP\in TPs \quad [R_{CDIs}\rightarrow R'_{CDIs}].$$

也就是说,任何一个变换过程,都有能力确保系统在一致状态之间移动.

另外,变换过程作用在非受控数据 UDI 上的效果有两种:转换 UDI 为 CDI,或者拒绝 UDI(这种情况约定把 UDI 转换为 ∅,这种互斥关系用操作符"||"来表示).使用非受控数据转换关系 R_{UDIs} 来描述接收 UDI 为输入的变换过程的作用效果是 $R_{UDIs}:UDIs\times TPs\rightarrow CDIs\,||\,\varnothing$.

在针对变换过程的验证中,实体集 *ENTs* 为变换过程集 *TPs*,相应的验证实体集 *CERTENTs* 为验证变换过程集 *CERTTPs*,验证函数为 $Cert_{TPs}$,则有关系*:

$$R_{C2\&C5}:\forall CERTTP\in CERTTPs \quad \exists TP\in TPs, userID\in USERIDs$$
$$[CERTTP=Cert_{TPs}(TP,userID,R_{sp}\cup R_{TPStruct}\cup R_{UDIs})\wedge IdentityClass(userID)=AuthUser].$$

亦即对于任何变换过程的验证过程,都是由 *AuthUser* 类别的用户身份来执行认证函数 $Cert_{TPs}$ 来完成的,关系集 $R_{sp}\cup R_{TPStruct}\cup R_{UDIs}$ 的保持是验证得以完成的前提.此时,C2&C5 策略的系统状态相关部分为

$$\sigma_{C2\&C5}=\{UDIs,CDIs,OPs,TPs,USERIDs\}.$$

3. (E1)E1 要求当执行一个变换过程时,对一个与当前激活变换过程相关的 CDI 进行的操作都属于该变换过程,因此就有以下激活变换过程操作蕴涵关系:

$$R_{CTP-OPs}:\forall CTP\in CTPs \quad \exists TP\in TPs, userID\in USERIDs$$
$$\big[CTP=Exec(TP,userID)\wedge[\forall CDI\in Group(CDIs,TP) \quad (op,CDI)\in CTP]\big].$$

执行一个变换过程的结果是使这个变换过程转化成激活变换过程,并且作用在与该变换过程相关的 *CDIs* 元素上的操作(*op*,*CDI*)都属于当前变换过程.这里假设任一时刻在一个 CDI 上只有 1 个变换过程作用.

另外,对于任何已经激活的变换过程,都应该使 $R_{Exec\text{-}TP}$ 成立,于是有 $R_{E1}=R_{Exec\text{-}TP}\cup R_{CTP\text{-}OPs}$. 该式意味着使变换过程正确执行的条件是,一方面要求满足能使该变换过程执行的条件 $R_{Exec\text{-}TP}$,另一方面要求变换过程满足一定的结构约束 $R_{CTP\text{-}Ops}$.E1 策略的系统状态相关部分为 $\sigma_{E1}=\{CDIs,OPs,TPs,CTPs,USERIDs\}$.

4. (E2)一个变换过程的执行动作必须与一个用户身份相关联,于是必须维持以下关系:

$$R_{TP-userID}:\forall CTP\in CTPs \quad \exists TP\in TPs, userID\in USERIDs$$
$$\big[CTP=Exec(userID,TP)\wedge\big[\exists IDToken\in IDTOKENs$$
$$[GetItem(IDToken,1)=userID\wedge TP\in GetItem(IDToken,2)]\big]\big].$$

可从以下几个方面来理解 $R_{TP\text{-}userID}$.首先,对任何已经激活的变换过程而言,都是由用户身份 *userID* 通过 *Exec* 函数执行一个变换过程所得.其次,要求存在一个包括这个用户身份的用户身份权能元组,它的权能令牌中包括当前变换过程的标识符.当然,为使一个变换过程得以顺利执行,还需要它的执行条件,于是就有

$$R_{E2}=R_{Exec\text{-}TP}\cup R_{TP\text{-}userID}.$$

E2 策略的系统状态相关部分为 $\sigma_{E2}=\{TPs,CTPs,USERIDs,IDTOKENs\}$.

5. (C3)为了处理职责隔离方面的完整性要求,首先对一个变换过程应用变换过程分割函数以形成相关的变换过程片集,并且确保变换过程片集中的变换过程片都是互不相同的,这种差异是由在不同变换过程片中包含不同的(*op*,*CDI*)元组来定义的.也就是说,存在关系:

---

* 在代数运算中,运算符"∪"是指两个集合间的运算,但在本文的以下部分为了简洁起见,对∪的功能进行了适当的扩展,使其同时具有进行集合与集合、集合与元素和元素与元素之间运算的功能.这里,假设 *A*,*B* 为集合,*a*,*b* 为单个元素,则 $A\cup B$,$A\cup a$ 和 $a\cup b$ 都是合法的,后两者分别等同于 $A\cup\{a\}$和$\{a\}\cup\{b\}$.

$$R_{PTPs}:\forall TP\in TPs \quad \exists Pset, Pset=Part_{TP-PTPs}(TP)$$
$$[\forall p,q\in Pset, p\neq q \quad \exists (op,CDI), op\in OPs, CDI\in CDIs \quad [(op,CDI)\in p\wedge (op,CDI)\notin q]].$$

其次,要求每个 PTP 必须被授予不同的用户身份,也即有职责隔离维持关系:

$$R_{ds}:\forall TP\in TPs \quad \exists Pset=Part_{TP-PTPs}(TP) \quad \left[\forall p,q\in Pset, p\neq q \quad \exists IDToken\in IDTOKENs,\right.$$
$$IDTOKENs_p=\bigcup_{p\in GetItem(IDToken,2)} IDToken, IDTOKENs_q=\bigcup_{q\in GetItem(IDToken,2)} IDToken$$
$$\left.\left[\left[\bigcup_{IDToken\in IDTOKENs_p} GetItem(IDToken,1)\right]\cap\left[\bigcup_{IDToken\in IDTOKENs_p} GetItem(IDToken,1)\right]=\varnothing\right]\right].$$

其中,$IDTOKENs_p$ 和 $IDTOKENs_q$ 分别是指和变换过程片 p,q 相关的身份权能元组集.在上述关系中,首先应用变换过程分割函数来对变换过程进行划片是实现职责隔离的基础;然后针对不同的变换过程片,分别收集含有当前变换过程片的所有身份权能元组,形成相应的身份权能元组集;最后还必须确保与不同变换过程片集相关的用户身份集是各不相同的,这样就达到了严格的职责隔离目的.同时,注意到这里允许分割函数为特征函数,也就是说划片的结果为 1,这时关系仍然保持.

这里,实体集 *ENTs* 为变换过程片集 *PTPs*,相应的验证实体集 *CERTENTs* 为验证变换过程片集 *CERTPTPs*,验证函数为 $Cert_{PTPs}$.于是有关系:

$$R_{C3}:\forall CERTTP\in CERTTPs \quad \exists PTP\in PTPs, userID\in USERIDs$$
$$[CERTTP=Cert_{PTPs}(PTP,userID,R_{TPStruct}\cup R_{PTPs}\cup R_{ds})\wedge IdentityClass(userID)=AuthUser].$$

在上述关系中,任何变换过程片的验证过程都是由 *AuthUser* 类别的用户身份执行认证函数 $Cert_{PTPs}$ 来完成的,关系集 $R_{TPStruct}\cup R_{PTPs}\cup R_{ds}$ 的保持是验证得以完成的前提.C3 策略的系统状态相关部分为

$$\sigma_{C3}=\{CDIs, OPs, TPs, PTPs, USERIDs, IDTOKENs\}.$$

6. (E3)这条安全策略主要用于解决变换过程执行所需的身份问题.由前述可知,系统中一个用户首先通过身份认证获得一个用户身份权能元组,然后再根据元组中的权能令牌存取所授权的变换过程.这里,用户和用户身份权能元组之间是一一对应关系,所以可进一步写为(用户,用户权能元组)的形式.于是,如果一个用户要执行一个变换过程,则必须满足下列激活变换过程用户身份保持关系:

$$R_{CTP-user}:\forall CTP\in CTPs \quad \exists (user\in USERs, IDToken\in IDTOKENs)$$
$$[CTP=Exec(GetItem(IDToken,1),TP)\wedge TP\in GetItem(IDToken,2)\wedge Assign_{user-IDToken}(user)=IDToken].$$

对于任何一个已经被执行的变换过程,必然存在着一个用户身份令牌元组 *IDToken*,使得在该令牌中存在包括该变换过程的描述符.这时,要求从用户身份令牌元组中获得的用户身份就是用以执行该变换过程的用户身份.与 E3 相关的关系为 $R_{E3}=R_{Exec\text{-}TP}\cup R_{CTP\text{-}user}$.要使变换过程正确执行,一方面要求满足能使该变换过程执行的条件 $R_{Exec\text{-}TP}$,另一方面还要求满足有关这个变换过程的用户和用户身份之间的认证约束关系 $R_{CTP\text{-}users}$.

E3 策略的系统状态相关部分为 $\sigma_{E3}=\{TPs, CTPs, USERIDs, IDTOKENs, USERs\}$.

2 基于 Clark-Wilson 完整性策略的安全监视模型(CW-SMM)

基于 Clark-Wilson 完整性策略的安全监视模型(CW-SMM)可用图 2 来表示.具体来讲,它可以分为 4 个功能模块:确定审计目标、确定日志项目、系统日志和审计监视等功能模块.这里,PolicyItem 为安全策略关系模式条目,AuditTarget 为审计目标,LogItem 为日志项目,SysData 为系统数据,LogData 为日志数据,SysSRP 为系统状态相关部分,*Result* 为裁判结果.相应地,*POLICYITEMs* 为安全策略关系条目集,*AUDITTARGETs* 为审计目标集,*LOGITEMs* 为日志项目集,*SYSDATAs* 为系统数据集,*LOGDATAs* 为日志数据集,*SYSSRPs* 为系统状态相关部分集,*RESULTs* 为裁判结果集.各主要功能的定义和规范如下:

1. 确定审计目标: $CreatAuditTargets: 2^{POLICYITEMs}\rightarrow AUDITTARGETs$. 其功能是参照安全策略(1-1)来生成审

计目标(1-2).相应的审计目标到安全策略集的投射为 $Map_{AutiTarget-POLICYITEMs}: AUDITTARGET \to 2^{POLICYITEMs}$. 这里, *POLICYITEMs*={C1,C2&C5,C3,E1,E2,E3},是由 Clark-Wilson 完整性策略关系模式条目构成,所有的审计目标都由其中的元素决定.

策略关系条目到系统状态相关部分的映射函数为 $Map_{PolicyItem-SysSRP}: POLICYITEMs \to SYSSRPs$. 这里,$SYSSRPs=\{\sigma_{C1},\sigma_{C2\&C5},\sigma_{C3},\sigma_{E1},\sigma_{E2},\sigma_{E3}\}$,是由各 Clark-Wilson 完整性策略关系模式条目的系统状态相关部分构成,对于策略关系模式条目 $PolicyItem \in POLICYITEMs$,其相应的系统状态相关部分在 *SYSSRPs* 中的索引形式为 $\sigma_{PolicyItem}$,例如,策略关系模式条目 C1 的系统状态相关部分就是 σ_{C1}.

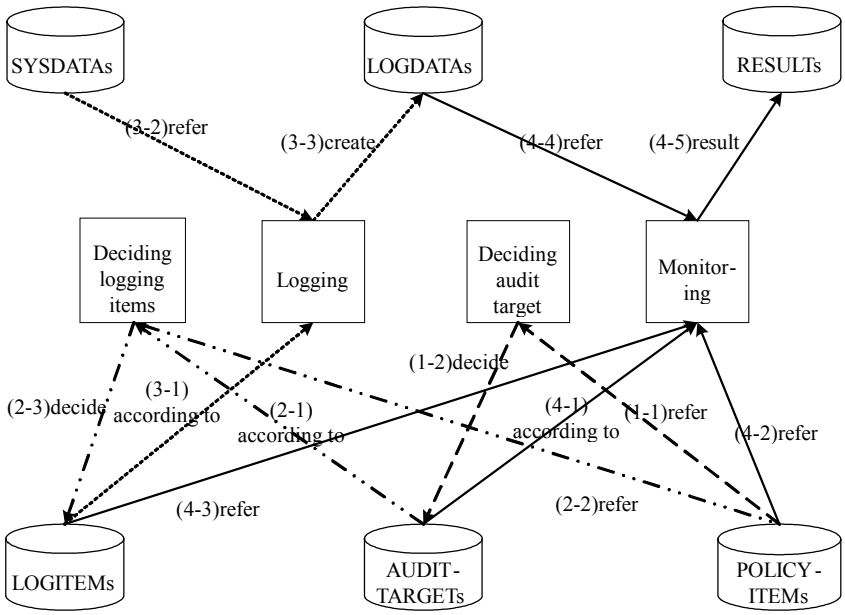

Fig.2 Illustration of secure monitoring model
图 2 安全监视模型示意图

2. 确定日志项目集: $CreatLogItems: AUDITTARGETs \times 2^{POLICYITEMs} \to 2^{LOGITEMs}$. 该功能依据审计目标(2-1)参照相关安全策略(2-2)确定日志项目集合(2-3).下面给出该功能的函数规范*:

$$CreatLogItems(lis: 2^{LOGITEMs}) \lhd$$

$$at: AuditTarget; p: PolicyItem;$$

$$lis = \bigcup_{at \in AUDITTARGETs} \left(\bigcup_{p \in Map_{AuditTarget-POLICYITEMs}(at)} Map_{PolicyItem-SysSRP}(p) \right) \rhd$$

在上式中,先计算出与每个安全审计目标相关的策略集,然后再计算出与该策略集中所有策略相应的状态相关部分的集合,然后据此计算出与系统所有审计目标相关的系统状态相关部分集,亦即日志项目集 *LOGITEMs*.

3. 系统日志: $SysLog: 2^{SYSDATAs} \times LOGITEMs \to LOGDATAs$. 系统日志功能依据日志项目集(3-1)来日志系统数据(3-2),生成日志数据(3-3).

为了处理日志项目到系统数据项上的映射问题,定义日志项目映射函数:

$$Map_{LogItem-SysData}: LOGITEMs \to SYSDATAs.$$

为了处理单个系统数据项的日志问题,定义单项日志函数: $SingleLog: SYSDATAs \to LOGDATAs$, 则系统日志函数规范为

* 这里采用Z语言[9]来形式地描述CW-SMM中的几个关键功能函数的规范,唯一的变化是关于模式的表示:

模式名(声明) ◁ 命题;…;命题 ▷ .

$$SysLog(lds:2^{LOGDATAs})\lhd$$
$$sd:SysData;li:LogItem;$$
$$lds=\bigcup_{li\in LOGITEMs}(SingLog(Map_{LogItem-SysData}(li)))\rhd$$

在系统日志函数中,对于日志项目集中的所有日志项,首先完成从日志项目到系统数据项目的对应工作,然后再将其拷贝成为日志数据.

4. 审计监视: $AuditMointor:AUDITTARGETs\times 2^{LOGDATAs}\times 2^{POLICYITEMs}\rightarrow RESULTs$. 这里,$RESULT$={TURE, FALSE},TRUE 表示合法状态,FALSE 则为异常状态.当评估结果为合法时,评估值为 TRUE,否则为 FALSE.TRUE 和 FALSE 是布尔值,要求作用于其上的运算符合布尔运算法则.系统审计监视功能根据审计目标(4-1)参照相关安全策略(4-2)和与策略系统状态相关部分对应的日志项(4-3),读取该系统状态相关部分所规定项目的日志数据(4-4),然后根据上述信息决定系统所处的状态(4-5).

定义策略关系评估函数: $Evalute:POLICYITEMs\times 2^{LOGDATAs}\rightarrow RESULTs$.

系统审计功能规范为

$$AuditMonitor(decision:RESULT)\lhd$$
$$at:AuditTarget;p:PolicyItem;$$
$$decision=\bigwedge_{at\in AUDITTARGETs}\left(\bigwedge_{p\in Map_{AuditTarget-POLICYITEMs}(at)}Evaluate\left(p,Map_{PolicyItem-SysSRP}(p)\right)\right)\rhd$$

在系统审计监视功能的规范中,首先针对一个审计目标,利用策略评估函数分别评估对当前审计目标所涉及的每一个策略的满足情况,得出该审计目标的满足情况,依次如法对系统中的所有审计目标进行评估,从而获得当前所有审计目标的满足情况.当所有审计目标都满足时,表明系统状态处于合法状态(完整状态或一致状态),否则说明系统处于非一致状态,于是就报警或调用预定义的响应程序.显然,通过把安全策略转化为相应的关系模式,然后利用由这些关系模式所生成的形式化二值(真/假)关系(断言),可以很直观地在审计功能的决策行为中使用安全策略.

3 模型分析

3.1 CW-SMM完整性命题

命题 1. 在基于 Clark-Wilson 完整性策略的安全系统中,CW-SMM 实现了针对该完整性策略的安全监视控制.

证明:CW-SMM 中的安全审计目标都是依据 Clark-Wilson 完整性策略条目的关系模式来构造的.如果 CW-SMM 要基于 Clark-Wilson 完整性策略的安全系统实现完全的安全监视控制,必定要求每一个 Clark-Wilson 完整性策略条目都可以在 Clark-Wilson 完整性策略条目关系模式集中找到相应的功能条目.

Clark-Wilson 完整性策略中的 C1,C3,E1,E2 和 E3 条目在 Clark-Wilson 完整性策略的关系模式集中都有对应的条目;C2 和 C5 已被合并为 C2&C5 条,合并的必要性已在本文第 3 节中做了必要的陈述;对于 C4,基于审计的安全监视策略无论在深度还是广度上都突破了 C4 所要求的仅仅进行变换过程的操作进行重构的要求,实现了对各种验证和执行完整性策略的完全覆盖,因此 C4 的功能要求已由 CS-SMM 的整体功能体现了,所以不再单独列出;对于原 Clark-Wilson 完整性策略条目集中的 E4 要求,已经由身份分类函数 *IdentityClass* 通用对验证函数 *Cert* 和变换过程执行函数 *Exec* 的约束带到了对各条完整性策略的形式化描述中,所以在 CW-SMM 中就没有必要出现针对 E4 的条目.因此,CW-SMM 实现了完全的针对 Clark-Wilson 完整性策略系统的安全监视. □

3.2 性能分析

针对传统审计系统中的日志数据冗余问题,Picciotto 提出应对所收集的日志数据进行约简,以使得异常线索易于被发现[10];Markantonakis 在其博士学位论文[11]中指出,日志数据项的选择主要受系统安全策略、可供日志文件所使用的存储空间及日志文件转移频率等因素影响,但他并没有对这些因素的具体作用继续做深入的

讨论.针对异常检测中的时延问题,Mayfield 提出使用监视和报警等实时手段来克服传统审计系统所固有的审计时间延迟问题[12].这一思想在各种安全监视方案中得到了广泛的支持.

相对地,本文主要借鉴了 Bishop[7,13,14]的工作成果.在文献[13]中,他提出使用系统状态相关部分作为审计系统自动约简日志数据的依据,使用错误反射机制(error-rejection mechanism)作为实施实时安全监视以及对异常进行甄别并报警的手段,但他并未对如何把系统状态相关部分和实际系统组件相关联做必要的说明,从而只能主要依赖系统设计者和管理员的经验来确定系统状态相关部分的具体内容;另一方面,由于系统状态相关部分主要用于对系统日志数据的化简,所以还是无法彻底解决冗余数据的日志所带来的系统性能问题.在文献[7]中 Bishop 提出,系统的安全策略和审计子系统的实现机制决定了安全监视的目标和内容,但他仍没有进一步指出安全监视目标和日志内容与安全策略之间的具体关联方式.随后,他在文献[14]中正式讨论了系统安全策略和日志内容以及安全监视目标三者之间的关系,提出了面向目标的审计和日志模型.但由于采用了极为抽象的图灵机来模拟实际系统,致使从安全策略到具体审计目标的映射仍旧是一个十分难以把握的过程.

与上面所述的 Picciotto,Mayfield 等人,特别是 Bishop 的工作相比,CW-SMM 从提高日志数据的利用效率和审计分析的自动化两方面着手,有效地解决了上述问题.在 CW-SMM 中,通过确定日志项目集从而得到了最小日志项目集,从根本上提高了日志数据的利用效率.这里,最小日志项目集是指系统所选择的日志项目足以供一定目的的审计分析之需,但同时要求除此之外没有其他项目被冗余日志.通过把 Clark-Wilson 完整性策略条目改写为策略关系模式,从而可以直接把 Clark-Wilson 完整性策略应用于对日志数据的自动审计中,解决了系统异常侦测时延过大的问题.

4 结 论

安全监视功能是系统审计的一个主要目标,安全监视目标主要由系统的安全策略决定.Clark-Wilson 完整性策略作为一种最有影响力的完整性策略,研究在其支持下的基于审计活动的安全监视机制具有重要的理论和实践意义.因此,本文在完成对 Clark-Wilson 完整策略条目的形式化关系模式转化工作的基础上,提出了基于 Clark-Wilson 完整性策略的安全监视模型(CW-SMM).该安全监视模型与传统的安全监视机制相比,可以有效地控制日志数据冗余和审计时间延迟.

References:

[1] Seiden, KF, Melanson JP. The auditing facility for a VMM security kernel. In: IEEE Symp. on Security and Privacy. New York: IEEE Computer Society Press, 1990. 262~277.
[2] Simone FH. IT-Security and Privacy. Berlin: Springer-Verlag, 2001. 35~104.
[3] Bishop M. A model of security monitoring. In: IEEE 5th Annual Computer Security Applications Conf. New York: IEEE Computer Society Press, 1990. 46~52.
[4] National Computer Security Center. A guide to understanding audit in trusted systems, Version 2. Technical Report, NCSC-TG-001, Fort Meade: National Computer Security Center, 1988.
[5] Clark DD, Wilson DR. A comparison of commercial and military computer security policies. In: IEEE Symp. on Security and Privacy. New York: IEEE Computer Society Press, 1987. 184~194.
[6] Qing SH, Liu WQ, Wen HZ, Liu HF. Operation System Security. Beijing: Tsinghua University Press, 2004. 73~114 (in Chinese).
[7] Özsu MT, Valduriez P. Principle of Distributed Database Systems. 2nd ed., Upper Saddle River: Prentice Hall, 1989. 25~51.
[8] Denning DE, Lunt TF. A multilevel relational data model. In: IEEE Symp. on Security and Privacy. New York: IEEE Computer Society Press, 1990. 220~234.
[9] Woodcock J, Davies J. Using Z. Upper Saddle River: Prentice Hall, 1996.
[10] Picciotto J. The design of an effective auditing subsystem. In: IEEE Symp. on Security and Privacy. New York: IEEE Computer Society Press, 1987. 13~22.
[11] Markantonakis C. Secure logging mechanisms for smart card [Ph.D. Thesis]. Egham: University of London, 1999.
[12] Mayfield T, Roskos JE, Welke SR, Boone JM. Integrity in automated information systems. Technical Report, 79-91, Fort Meade, 1991.
[13] Bishop M. A standard audit trail format. Technical Report, Department of Computer Science, University of California at Davis, 1995.
[14] Bishop M, Wee C, Frank J. Goal-Oriented auditing and logging. 1996. http://seclab.cs.ucdavis.edu/papers/tocs-96.pdf

附中文参考文献:

[6] 卿斯汉,刘文清,温红子,刘海峰.操作系统安全.北京:清华大学出版社,2004.73~114.