混合多策略视图安全模型*)

夏 磊 黄 皓 于淑英

(南京大学软件新技术国家重点实验室 计算机科学与技术系 南京 210093)

摘 要 传统的MLS策略侧重于信息机密性保护,却很少考虑完整性,也无法有效实施信道控制策略,在解决不同安全级别信息流动问题时采用的可信主体也存在安全隐患。同时,应用环境的多样性导致了安全需求的多样化,而当前的安全模型都只侧重于其中一种或几种安全需求。本文给出的混合多策略模型—MPVSM模型有机组合了BLP,Biba,DTE和RBAC等安全模型的属性和功能,消除了MLS模型的缺陷,提高了信道控制能力和权限分配的灵活度,对可信主体的权限也进行了有力的控制和约束,同时为实现多安全策略视图提供了一个框架。文中给出了MPVSM模型的描述和形式化系统,并给出了几种典型策略的配置实例。

关键词 安全模型,多安全策略,多级安全策略,最小特权,机密性保护,完整性保护

Hybrid Multiple Policy Views Security Model

XIA Lei HUANG Hao YU Shu-ying

(State Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Nanjing 210093, Chian)

Abstract MLS security policy aims at only confidentiality assurance, with less consideration on integrity assurance and weakness in channel control policy. And its way to handle the trusted subjects has many security shortcomings. Moreover, increasing diversity of the computing environments results in various security requirements. However, current mainstream security models are aiming at only one or few requirements of them each. The Multi-Policy Views Security Model is presented, which is based on the MLS model, combining the domain and type attributes to the model, to enforce the expression power in channel control policies, make permission management more fine-grained and enhance the ability of confining the permission of the trusted subjects. MPVSM is also able to enforce multi-policy views in operating system in a flexible way.

Keywords Security model, Multiple security policies, Multilevel security policy, Least privilege, Confidentiality assurance, Integrity assurance

1 概述

Internet和信息技术的发展及大规模应用使得实际应用的安全需求也越来越多样化。为了满足不同的安全需求,几十年来,产生了多种安全策略和安全模型。实施一个好的安全策略模型,可以有效保证系统应用软件的完整性和机密性,限制攻击者的非法行为,阻止攻击者对系统信息的随意破坏和任意泄露。

多级安全策略MLS(Multilevel Security)[1]是目前各种安全系统应用最为广泛的一类安全策略。然而,MLS策略的目的是防止高密级信息泄漏给低密级的用户,它能很好地防止信息的非授权泄漏,保护信息的机密性,但是它不允许主体同时处理多种密级的信息。因为在MLS中,任何主体都是不可信任的,即使它是处理高密级信息的用户,这使得不违背MLS很多功能无法正常实现。如防火墙软件,若不违背MLS的规则,就不能完成其正常的连接内外网络的功能。而从信息处理的角度看,一些信息处理程序必须同时处理多种安全等级的信息,如一个信息管理系统可以综合多种密级的信息提供给用户一个综合信息(如平均值),但不允许用户知道具体的各项秘密信息。

为了解决这个问题,很多实现MLS的安全系统采用的方法是将一些安全核外的主体视为可信主体,并把这些主体排除在MLS控制的范围以外,给它们特权,使它们可以绕过系统的访问控制机制来直接访问各种资源,如HP-UX/CMW[16]等。而这些能绕过访问控制机制的可信主体,往往获得了比实际需要大得多的权限,是各种攻击者的主要攻击对象,成为系统的安全隐患。

MLS的另一缺点是它没有考虑信息的完整性,而完整性保护同样非常重要[2,7,14]。Biba模型[6]采用与MLS相似的方法通过标记限制信息的流动来维护信息的完整性要求。Biba模型在数学上与MLS等价,都是基于格的模型。而基于格的模型都无法实现可信信道控制[17]等重要安全问题。本质原因在于基于格的信息流策略是传递的,即若信息能从A流入B,从B流入C,则信息也能从A流入C;而不能描述信息只能从A经B控制后流入C这样的安全策略。而一个防火墙系统实际上是一个信道控制系统,图1就是一个简单防火墙系统的信道控制功能。防火墙的外连模块不能和内连模块直接通信,必须经过防火墙的访问控制模块,在该模块的控制下外连模块和内连模块进行通信。而这样的信道控制策略则很难用多级安全模型来实现。

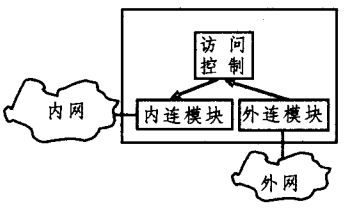

图1 防火墙系统的信道控制功能示意图

同时,计算环境的多样性和复杂性显著地增加,导致实际应用越来越多样化。操作系统应用多样化带来了多样化的安

*)本课题得到国家自然科学基金(60473093)资助。夏 磊 硕士研究生,研究方向为操作系统安全;黄 皓 教授,博士生导师,研究领域为计算机网络安全;于淑英 硕士研究生,研究方向为操作系统安全。

全需求，不同应用有着不同的安全需求。例如，在一个典型的系统中，普通用户最大的安全需求是机密性和完整性保障。而对于特权用户进程，职责分离和最小特权原则也很重要。对于大多数网络服务进程，如防火墙或网关，信道控制和用户信息的隔离更重要。

上述这些安全需求通常情况下是共存于同一个系统中的。而早期的操作系统多采用了单一的安全策略模型来实施强制访问控制，如 Multics 系统[3]采用 BLP 模型来实现多级安全策略。当前的每个安全策略模型都只侧重于其中某一种或几种安全需求，如 Bell-LaPadula(BLP)模型[1]解决的本质问题是机密性保护，没有考虑信息的完整性保护和信道控制[5]，也不能很好地支持职责隔离和最小特权原则的实现。Biba[6]主要针对信息完整性的安全要求。DTE[8,9]可以有效地实现信道控制，保证信息流动的非传递性。RBAC[12,13]能够很好地体现最小特权和职责分离的安全原则。这些安全需求无法通过单个主流的安全策略模型来实现，系统中应该存在一种多策略视图，即针对不同的应用能灵活地采用不同的策略模型。

作为策略中的模型，RBAC[13,14]被看作是替代传统的自主访问控制和强制访问控制的一种新的模型。基于角色的访问控制模型便于权限管理和实施最小特权原则。但是由于 RBAC 的结构，用 RBAC 来直接实现自主访问控制、多级安全策略[15]非常复杂，在实际系统中没有应用价值。同时，目前文献中还没有发现一个用 RBAC 实现的安全系统，它具有类似 BLP 模型那样的形式模型。

本文综合了 MLS，DTE 和 RBAC 等安全模型的思想，构建了一个混合多策略模型—MPVSM(Multiple Policy Views Security Model)。MPVSM 模型通过对多种安全模型属性的有机组合，消除了 MLS 模型的缺陷，可以有效地实施信道控制，细化了权限的分配，提高了策略表达能力，保障了可信主体的最小特权原则的实现，并综合了多个安全模型的特点，为实现多安全策略视图提供了一个灵活的框架。

MPVSM 基于多级安全策略模型，它将 MLS 和 Biba 这两个格叠加起来，形成一个统一的格，同时实现系统的机密性策略和完整性策略。在多级安全策略的基础上添加域属性，把处于同一安全级别中的主体划分到不同的域，利用 DTE 模型的控制机制来实现同一安全级别中的权限分配的进一步细化和权限分配的灵活性，控制处于同一安全级别下不同职责的主体的分权。另一方面，利用 RBAC 模型，为主体设置角色，为角色分配权限，这些权限是独立于多级安全策略 MLS 的。我们利用这种角色授权机制为特定的可信主体设置策略，这样可以避免特定的可信主体受到 MLS 策略影响而无法完成需要的功能，又使得分配给该主体的权限能够遵循最小特权原则，有效地限制可信主体的权限范围。由于 RBAC 的角色授权机制具有很强的策略表达能力和策略配置的方便性，能够提高有效实现最小特权原则的能力，大大增强权限分配和策略表达能力。

MPVSM 模型有机结合了 BLP，Biba，DTE 和 RBAC 等多种安全模型的属性和功能，可以从不同侧面实现 BLP，Biba，DTE 和 RBAC 等安全模型的功能，并对这些模型的特点进行有机的综合，细化了权限分配，增强了权限分配的灵活性和策略的表达能力，有效地解决了特定可信主体的权限的管理问题，在很大程度上满足了各种应用的不同安全需求。

本文第 2 节给出了 MPVSM 模型的描述以及基于该模型的形式化安全系统，第 3 节介绍利用 MPVSM 配置的实例以及实现多策略视图的方法，最后是文章的总结。

2 MPVSM 模型描述

2.1 模型概述

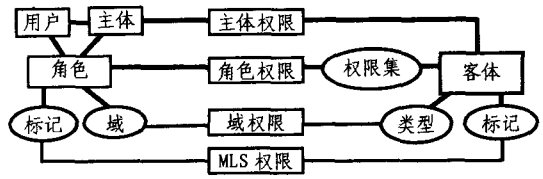

图 2 MPVSM 模型之元素-属性-关系示意图

MPVSM 模型包含主体、用户、角色、域和客体等元素，包含用户-主体关系、用户-角色分配关系、主体-角色关系和角色-域授权关系。用户-主体关系给出运行主体当前所代表的用户；主体-角色关系给出主体当前的运行角色，多个角色可以被分配给一个用户，但主体每次只能代表一个角色运行；角色-权限关系指定角色拥有的权限；角色-域授权关系给出了角色允许进入的所有域。角色还拥有一个安全标记属性，代表了该角色的安全级别。每个客体具有一个类型属性，指出该客体当前的类型，客体还拥有一个安全标记属性，给出该客体的安全级别。

主体拥有的权限从三个方面得出：1)MLS 权限。根据主体的安全级别(包括机密性级别和完整性级别)和被访问客体的安全级别，遵守多级安全访问控制规则获得的基础权限(读类权限或写类权限)。2)域权限。主体当前所处的域对某类型的客体的访问权限。3)角色权限。主体拥有的这些通过角色-权限关系分配给其运行角色的权限。

我们将 MLS 访问权限作为粗粒度的基础权限，指出主体具有读类权限或写类权限，而域访问权限是在 MLS 基础权限上对主体拥有的权限的进一步细化，指定各种细粒度的权限。角色访问权限是直接分配给角色的权限，它独立于前两种权限。这种独立的权限定义方法便于我们设计“特权程序”，并有效地对它的权限进行控制。

与权限相对应，MPVSM 模型具有三个方面的访问控制视图：1)多级访问控制：每个角色拥有一个安全标记，代表了该角色的安全级别，主体的安全级别由其当前的运行角色的安全级别决定。主体拥有的权限由主体的安全级别和被访问客体的安全级别决定。2)域访问控制视图：主体在不同域中运行，处在某个域的主体对某个类型的客体具有的权限由域-类型访问控制矩阵定义。3)角色控制视图：角色分配给用户，每个用户可以拥有多个角色，每个角色拥有不同的权限。主体代表着当前用户的某个角色运行，主体拥有该角色的所有权限。

多级访问控制和域访问控制是互补的关系。多级访问控制把系统中的权限进行粗粒度的划分，而域访问控制则对权限和权限划分进行细化。这样，既利用了多级访问控制的简洁性，又有域访问控制对权限分配的灵活性。角色访问控制独立于前两种视图，它分配的权限独立于其它权限。角色访问控制用来处理那些特定的可信主体的权限控制问题，利用灵活的角色权限授权机制来实现可信主体的最小特权。

2.2 模型中的基本元素

MPVSM 模型结构如图 2 所示，模型中包含的元素、属性和关系定义如下：

定义 2.1(基本元素集合)

用户集 U

主体集 S

客体集 O

角色集 R

域集 D

类型集 T

机密性标记集 C

完整性标记集 I

安全标记集 $SL \subseteq C \times I$，安全标记由机密性标记和完整性标记组成。

访问模式集 P，包括两个子集：读类模式集 $RP=\{read(r), execute(e), getattr(g) \cdots\}$；写类模式集 $WP=\{write(w), append(a), create(c), delete(d), setattr(s) \cdots\}$。

域交互模式 $transfer$，$transfer$ 表示主体从一个域转换到另外一个域的操作。

权限集 $CAP \subseteq P \times O$，$(p, o) \in CAP$ 的含义是：以方式 p 访问对象 o 的权限。

定义 2.2 $US \subseteq U \times S$，用户-主体关系，一对多的关系，每个用户拥有多个主体，而每个主体只能代表一个用户运行。

$user: S \to U$，主体-用户映射，单射，即每个主体只代表一个用户运行。$user(s)=u$ 当且仅当 $(u, s) \in US$。

定义 2.3 $UA \subseteq U \times R$，用户-角色分配关系，多对多关系，每个用户可以分配多个角色，每个角色可以被分配给多个用户。

$UR: U \to 2^R$，用户到其所拥有的角色集合的映射，$UR(u)=\{r \in R \mid (u, r) \in UA\}$。

$SR: S \to R$，主体到角色的映射，单射。任何时刻主体只以一个角色运行，该角色是已经分配给主体当前代表的用户的角色：$SR(s) \in UR(user(s))$。

定义 2.4 $RL: R \to SL$，角色到其安全标记的映射，单射。每个角色有且只有一个安全标记。

$Ssl: S \to SL$，主体到其安全标记的映射，单射。主体安全标记为主体当前运行角色的安全标记：$Ssl(s)=RL(SR(s))$。

定义 2.5 $RD \subseteq R \times D$，角色-域授权关系，多对多关系。多个角色可以进入同一个域，一个角色也可以进入多个域。

$RDom: R \to 2^D$，角色到其所授权的域集合的映射，$RDom(r)=\{d \in D \mid (r, d) \in RD\}$。

$SDom: S \to D$，主体到其当前所在的域的映射，单射，每个主体同一时刻只能在一个域中运行，并且该域已经授权给主体当前的运行角色：$SDom(s) \in RDom(SR(s))$。

定义 2.6 $RCAP \subseteq R \times CAP$，角色-权限分配关系，角色拥有的角色权限。

$Rolecap: R \to 2^{CAP}$，每个角色拥有的所有角色权限的集合。$Rolecap(r)=\{cap \in CAP \mid (r, cap) \in RCAP\}$。

定义 2.7（客体安全属性）

$OT: O \to T$，客体的类型属性，该映射为单射，每个客体只能属于一种类型。

$OL: O \to SL$，客体的安全标记，该映射为单射，每个客体只有一个安全标记。

定义 2.8（域-类型/域-域访问控制矩阵）

$DTM: D \times T \to 2^P$，域-类型访问控制矩阵。$p \in DTM(d, t)$ 的含义是：域 d 中的主体能以方式 p 访问具有类型 t 的客体。

$DDI: D \times D \to \{\Phi, \{transfer\}\}$，域-域交互控制矩阵，$transfer \in DDI(d_1, d_2)$ 表示域 d_1 中的主体可以转换到域 d_2。

定义 2.9（多级安全访问控制规则 MLS_rule） $SL \times SL \to 2^P$，$a \in MLS\_rule(ls, lo)$ 的含义是：具有安全标记 ls 的主体对具有安全标记 lo 的客体具有 a 访问模式。该规则函数组合了 BLP 机密性和 Biba 完整性两个格。令 $ls=(cs, is)$，$lo=(co, io)$。

当 $cs \geqslant co$，允许读一类操作，即 $RP \subseteq MLS\_rule(ls, lo)$。

当 $cs < co$，禁止读一类操作，即 $\forall p \in RP, p \notin MLS\_rule(ls, lo)$。

当 $is \geqslant io$ 时，允许写一类操作，即 $WP \subseteq MLS\_rule(ls, lo)$。

当 $is < io$ 时，禁止写一类操作，即 $\forall p \in WP, p \notin MLS\_rule(ls, lo)$。

2.3 权限规则

定义 2.10（主体-客体访问权限）

MLS 权限：主体 s 对客体 o 的 MLS 权限 $mlp(s, o)=\{p \mid p \in MLS\_rule(Ssl(s), OL(o))\}$。

域权限：主体 s 对客体 o 的域权限 $dp(s, o)=\{p \mid p \in DTM(SDom(s), OT(o))\}$。

角色权限：主体 s 对客体 o 的角色权限 $rp(s, o)=\{p \mid (p, o) \in Rolecap(SR(s))\}$。

定义 2.11（主体 s 对客体 o 拥有的访问权限） $ap(s, o)=(mlp(s, o) \cap dp(s, o)) \cup rp(s, o)$。

2.4 模型的限制条件

限制条件 1：任何状态下，客体都有且仅有一个类型属性、安全标记属性，即

$dom(OT)=O, dom(OL)=O$

限制条件 2：任何状态下，主体都只能处于一个域当中，并且该域是主体当前角色拥有的域，即

$dom(SDom)=S, SDom(s) \in RDom(SR(s))$

限制条件 3：任何状态下，主体都只代表一个用户，并且以一个角色执行，该角色必须是已经分配给用户的角色，即

$dom(user)=S, dom(SR)=S, SR(s) \in UR(user(s))$

限制条件 4：任何状态下，主体都只有一个安全标记，该安全标记为主体当前角色的安全标记，即 $dom(Ssl)=S, Ssl(s)=RL(SR(s))$

2.5 形式化系统描述

上节给出了 MPVSM 模型中的定义。本节给出基于 MPVSM 模型的形式化系统的状态、状态的安全属性及系统规则、安全系统的定义，最后还给出系统的安全定理。

定义 2.12（系统状态集 V） 系统状态 $v \in V$ 是一个四元组 (b, m, e, f)，其中 $b \in P(S \times O \times P)$，表示主体 s 可以以方式 p 来访问客体 o；m 为 (DTM, DDI)，其中 DTM 和 DDI 分别对应该状态下系统的域-类型访问控制矩阵和域-域交互控制矩阵；e 为 (U, S, O, R, D, T)，分别对应该状态下系统的用户集、主体集、客体集、角色集、域集和类型集；f 为 $(US, UA, SR, RD, SDom, RL, Ssl, OT, OL, RCAP)$，分别对应该状态下系统的各种关系和映射。

定义 2.13（系统状态的各种安全属性）

一个状态 (b, m, e, f) 满足多级安全属性当且仅当 $\forall (s, o, p) \in b, p \in MLS\_rule(SL(s), OL(o))$。

一个状态 (b, m, e, f) 满足域安全属性当且仅当 $\forall (s, o, p) \in b, p \in DTM(SDom(s), OT(o))$。

一个状态 (b, m, e, f) 满足角色安全属性当且仅当 $\forall (s, o, p) \in b, (p, o) \in Rolecap(SR(s))$。

定义 2.14 一个状态是安全的当且仅当该状态同时满足以下三个条件：

1）满足模型限制条件 1-4；

2）满足域安全属性和多级安全属性或满足 3）；

3)满足角色安全属性。

定义 2.15 状态(b, m, e, f)到状态(b^*, m^*, e^*, f^*)的转换满足域转换安全属性当且仅当满足以下条件：

1)$\forall s\in(S^*\cap S)$，如果$SDom(s)\neq SDom^*(s)$，则$transfer\in DDI(SDom(s), SDom^*(s))$。

定义 2.16 RA:请求集合，DE:判断集合。对请求作出的判定结果，包括 YES(允许)、NO(拒绝)、ILLEGAL(非法请求)、ERROR(错误)。系统动作集$W\subseteq RA\times DE\times V\times V$，表示一个请求被发出，针对该请求作出的决策，以及引起的系统状态的转换。

设定 N 为自然数集合。$X=RA^N$，该集合的元素 x 为一个请求的序列。$Y=DE^N$，该集合的元素 y 为一个决策的序列。$Z=V^N$，该集合的元素 z 为一个状态的序列。x, y, z 的第 i 个元素表示成x_i, y_i, z_i.

定义 2.17 状态序列$z\in Z$是一个安全状态序列当且仅当

1)$\forall i\in N, z_i$ 是安全的状态；

2)$\forall i\in N, z_{i+1}$到z_i的状态转换满足域转换安全属性。

定义 2.18 系统定义为$\sum(RA, DE, W, z_0)\subseteq X\times Y\times Z$，$z_0$ 是系统初始状态。$(x, y, z)\in\sum(RA, DE, W, z_0)$当且仅当，对于所有$i\in N$，$(x_i, y_i, z_i, z_{i+1})\in W$。

定义 2.19 $(x,y,z)\in\sum(RA, DE, W, z_0)$是安全的当且仅当 z 是一个安全状态序列。一个系统$\sum(RA, DE, W, z_0)$是安全的当且仅当$\forall(x,y,z)\in\sum(RA, DE, W, z_0)$是安全的。

定义 2.20 规则是一个函数$\rho: RA\times V\to DE\times V$。给定一个状态和请求，规则给出对该状态的判定结果和转换后的状态。$(ra, de, v, v^*)\in RA\times DE\times V\times V$是系统$\sum(RA, DE, W, z_0)$的一个规则当且仅当，存在$n\in N$，$(ra, de, v, v^*)=(x_n, y_n, z_n, z_{n-1})$。

定义 2.21 一个规则ρ是安全的当且仅当

1)对于所有$\rho(ra, v)=(de, v^*)$，如果 v 是一个安全状态，那么 v^* 也是一个安全状态；

2)对于所有$\rho(ra, v)=(de, v^*)$，v到v^*的状态转换满足域转换安全属性。

下面给出系统安全定理。

定理 2.1 如果z_0是一个安全的状态并且 W 中的所有规则都是安全的，则$\sum(RA, DE, W, z_0)$是一个安全的系统。

证明：假设$\sum(RA, DE, W, z_0)$是不安全的。令$(x, y, z)\in\sum(RA, DE, W, z_0)$是不安全的，则 z 为不安全的状态序列，因此 z 中存在不安全状态或是 z 中存在状态转换不满足域转换安全属性。

1)设z_t是 z 中不安全的状态中 t 最小的一个状态。因为z_0是安全的，故$t>0$。根据我们的假设，z_{t-1}是安全的，并且根据$\sum(R,D,W,z0)$的定义，$(x_t, y_t, z_t, z_{t-1})\in W$。并且存在一个唯一的规则$\rho\in\omega$，$\rho(x_t, z_{t-1})=(y_t, z_t)$，因为$\rho$是安全的，$z_{t-1}$是安全的，根据定义，$z_t$也是安全的，与前提假设相矛盾。

2)假设z_{t-1}到z_t的转换不满足域转换安全属性，同样存在一个唯一的规则$\rho\in\omega$，$\rho(x_t, z_{t-1})=(y_t, z_t)$，因为$\rho$是安全的，所以$z_{t-1}$到$z_t$的转换满足域转换安全属性，所以假设不成立。

综上，假设不成立，因此$\sum(RA, DE, W, z_0)$是安全的。

3 策略配置实例

3.1 信道控制策略配置实例

我们设计了一个防火墙系统信道控制策略模型。该防火墙系统的安全策略要求从外网进入系统的数据必须通过访问控制模块的安全检查才能进入内网，反之亦然。在模型中该策略的解释为：所有从外连模块流入内连模块的信息和所有从内连模块流入外连模块的信息，也必须经过访问控制模块。除此之外，系统还要求所有模块可以读取系统配置信息，但不能对其修改；所有模块可以追加日志信息，但不能读取该信息。

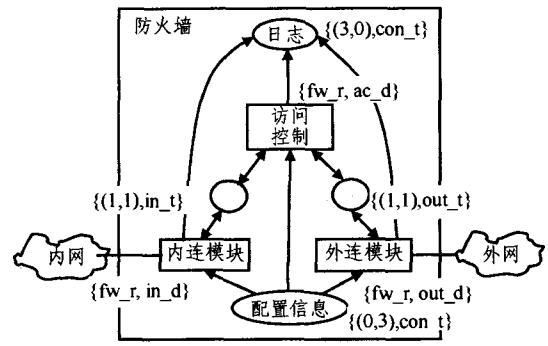

图 3 防火墙策略配置实例图

为实现以上信息流策略，我们为每个模块分配一个安全属性$\{role, domain\}$。$role$ 表示运行角色，$domain$ 表示运行域。为每个存储对象设置相应的安全属性，安全属性$\{(c, i), t\}$表示对象的机密性级别为 c，完整性级别为 i，类型为 t。配置如图 3 所示。其中的域-类型和域-域控制矩阵如表 3 所示。其中，$Rolecap(fw\_r)=\Phi$，角色 fw_r 没有角色权限。角色 fw_r 的安全级别为(1,1)，授权域为$\{ac_d, in_d, out_d\}$，这三个域之间不能相互转换。

表 3 DTM 和 DDI 矩阵

	in_t	out_t	con_t	in_d	out_d	ac_d
in_d	r,w		r,a			
out_d		r,w	r,a			
ac_d	r,w	r,w	r,a			

通过提高日志信息的机密级别使得主体不能读取日志，提高配置信息的完整级别使得配置信息不能被修改。通过把同一安全级别的主体划分到不同的域，而同一安全级别的客体划分到不同的类型，并通过域-类型访问控制矩阵细化访问权限，以此实现内外网信息流动通道的控制。

3.2 多级安全信息交互配置实例

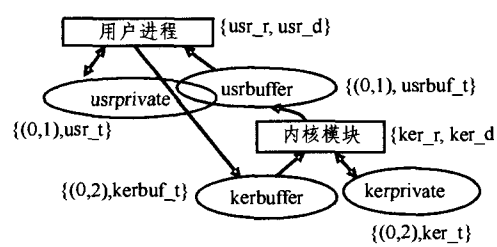

图 4 用户进程-操作系统交互模型

我们设计了一个用户进程与操作系统的交互控制策略实例。如图 4 所示，用户进程与操作系统的数据交互通过各自的缓冲数据存储对象来完成。内核数据对象的完整性高于用户数据对象。但是用户必须能够向内核发送数据，即低完整级别数据又需要在受控制的方式下流向高完整级别的数据对象。用户发送数据给操作系统时，将这些数据写入操作系统的缓冲区域 kerbuffer；操作系统回复用户进程时，将回复数

据写入用户进程的缓冲区域 usrbuffer。不允许用户进程写操作系统的其它数据区域。

表 2 *DTM* 矩阵

	ker_t	kerbuf_t	usr_t	usrbuf_t
ker_d	r,w	r		w
usr_d			r,w	r

为实现以上策略，我们的配置如图 4 所示。其中，$Rolecap(usr\_r)=\{(w, kerbuffer)\}$，表示角色 *usr_r* 拥有写 *kerbuffer* 数据对象的权限，角色 *usr_r* 的安全级别为(0,1)，授权域为{*usr_d*}。$Rolecap(ker\_r)=\Phi$，*ker_r* 没有角色权限，角色 *ker_r* 的安全级别为(0,2)，授权域为{*ker_d*}。其中的域-类型和域-域控制矩阵如表 2 所示。

通过不同的安全级别把内核数据空间和用户数据空间隔离开来，同一安全级别的 usrbuffer 和 usrprivate 通过划分到不同类型，使得 User process 对两个数据空间的权限不同。而根据安全级别限制，User process 无法对 kerbuffer 进行写操作，而这个操作又是功能需要，因此赋予用户角色 usr_r 单独的写 kerbuffer 的权限。该权限独立于 MLS 和域权限，既满足了写多安全级别信息的需要，又不用担心 User process 获得过多的权限而带来安全隐患。

3.3 多策略视图的实现

如前所述，多安全策略视图在很多系统中是需要的。假设系统中的所有用户分为三个部分：*Grpa*，*Grpb*，*Grpc*。每个部分的用户所见到的安全策略是不同的，如 *Grpa* 的用户认为系统的安全策略是多级安全策略，而 *Grpb* 的用户则认为系统是采用 RBAC 安全策略，*Grpc* 的用户则认为系统的安全策略是 DTE。下面给出了用 MPVSM 模型实现这种多策略视图的方法：

(1)$U=Grpa\cup Grpb\cup Grpc$，并且 *Grpa*，*Grpb*，*Grpc* 两两互不相交。

(2)$R=mls\_rs\cup rbac\_rs\cup\{dte\_r\}$。*mts_rs* 为 MLS 模型对应的角色集，*rbac_rs* 为 RBAC 策略对应的角色集。*dte_r* 为 DTE 策略对应的角色。

(3)$D=\{mls\_d\}\cup\{rbac\_d\}\cup dte\_ds$。

(4)$(u,r)\in UA\wedge(u,r')\cup A$，其中 $u\in Grpa$，$r\in mls\_rs$，$r'mls\_rs$。即 *Grpa* 中用户仅仅指派角色集 *mls_rs* 中的角色。$(u,r)\in\cup A\wedge(u,r')UA$，其中 $u\in Grpb$，$r\in rbac\_rs$，$r'\notin rbac\_rs$。即 *Grpb* 中用户仅仅指派角色集 *rbac_rs* 中的角色。同样，$(u,dte\_r)\in\cup A\wedge(u,r)UA$，其中 $u\in Grpc$，$r\neq dte\_r$。*Grpc* 中所有用户都仅指派角色 *dte_r*。

(5)$|mls\_rs|=|SL|$，即 *mls_rs* 中的角色数量等于系统中的安全标记数。*mls_rs* 中的角色和 *SL* 中的安全标记一一对应，每个角色对应一个安全标记。*rbac_rs* 集合中的角色对应的安全标记为所有标记中最低级别的。即：$\forall r\in rbac\_rs$，$RL(r)=(cs, is)$，其中，$\forall c\in C, cs\leqslant c$；$\forall i\in I, is\leqslant i$。角色 *dte_r* 对应的安全标记为所有标记中最高级别，即 $RL(dte\_r)=(cs, is)$，其中，$\forall c\in C, cs\geqslant c$；$\forall i\in I, is\geqslant i$。

(6)$(r, mls\_d)\in RD\wedge(r, d)\notin RD$，其中 $r\in mls\_rs$，$d\neq mls\_d$，即 *mls_rs* 中的角色的授权域仅仅为 *mls_d*。$(dte\_r, d)\in RD\wedge(r', d)\notin RD$，其中 $r'\neq dte\_r$，$d\in dte\_ds$，即 *dte_r* 角色的授权域为整个 *dte_ds* 集合。同样，$(r, rbac\_d)\in RD\wedge(r, d)\notin RD$，其中 $r\in rbac\_rs$，$d\neq rbac\_d$，即 *rbac_rs* 中的角色的授权域为 *rbac_d*。

(7)$(mls\_d,t,p)\in DTM$，$t\in T$，$p\in P$，即 *mls_d* 域中的主体对任何类型的客体都具有所有的域访问权限。$DDI(mls\_d, d)=\Phi$，$d\in D$，*mls_d* 域中的主体不能转换到其他任何域中去。

(8)$(rbac\_d,t,p)DTM$，$t\in T$，$p\in P$，即 *rbac_d* 域中的主体对任何类型的客体不具有任何域访问权限。$DDI(rbac\_d, d)=\Phi$，$d\in D$，*rbac_d* 域的主体不能转换到其他任何域中去。

(9)$(r, c)\notin RCAP$，$r\in mls\_rsU\{dte\_r\}$，*mls_rs* 中的角色和角色 *dte_r* 的角色权限为空。

使用以上规则之后，呈现给用户群 *Grpa* 的安全策略是基于机密性格和完整性格的混合多级安全策略，呈现给用户群 *Grpb* 的是 RBAC 安全模型，呈现给用户群 *Grpc* 的则是 DTE 安全模型。

结束语 MLS 安全策略在实际系统中存在诸多缺陷，如多安全级别信息处理、信道控制等问题。同时，为了满足实际应用系统多样化的安全需求，操作系统必须能够支持多安全策略。本文构建的 MPVSM 模型通过对多种安全模型属性的有机组合，消除了 MLS 策略模型的缺陷，综合了各个模型的优点，增加了权限的分配表达能力，为安全操作系统灵活高效地实施多安全策略视图提供了灵活的框架。

参 考 文 献

[1] Bell D, LaPadula L. Secure Computer Systems: Mathematical Foundations. Technical Report, MTR-2547. Vol I, MTR-2997 Rev. 1. MITRE Corporation, Bedford, MA. Mar. 1975

[2] Amoroso E, Nguyen T, Weiss J, et al. Towards an Approach to Measuring Software Trust// Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy. 1991:198-218

[3] Organick E. The MULTICS System: An Examination of Its Structure. Cambridge, MA: The MIT Press, 1972

[4] Zanin G, Mancini L V. Towards a Formal Model for Security Policies Specification and Validation in the SELinux System// Preceedings of the 9th ACM Symposium on Access Control models and technologies. 2004

[5] Rushby J. Noninterference, Transitivity, and Channel-Control Security Policies. Technical Report, CSL-92-02. Computer Science Lab, SRI International, 1992

[6] Biba K. Integrity Considerations for Secure Computer Systems. Technical Report, MTR-3153. MITRE Corporation, Bedford, MA, 1977

[7] Eswaran K, Chamberlin D. Functional Specifications of Subsystem for Database Integrity// Proceedings of the International Conference on Very Large Data Bases. 1975:48-68

[8] Walker K M, Sterne D F, Badger L, et al. Confining Root Programs with Domain and Type Enforcement (DTE)//Proceedings of the 6th USENIX UNIX Security Symposium. 1996

[9] Badger L, Sterne D F, Sherman D L, et al. A Domain and Type Enforcement UNIX Prototype // Proceedings of the Fifth USENIX UNIX Security Symposium. 1995

[10] Loscocco P, Smalley S. Integrating Flexible Support for Security Policies into the Linux Operating System. Technical report. NAI Labs, 2001

[11] Spencer R, Smalley S, Hibler M, et al. The Flask Security Architecture: System Support for Diverse Security Policies// Proceedings of the 8th USENIX Security Symposium. 1999:123-139

[12] Sandhu R. Role-based Access Control. Advances in Computers. 1998,46: 237-286

[13] Sandhu R, Ferraiolo D, Kuhn R. The NIST Model for Role-Based Access Control: Towards A Unified Standard// Proceedings of the 5th ACM Workshop on Role-Based Access Control. 2000

[14] Lipner S. Non-discretionary Controls for Commercial Applications//Proceedings of the 1982 Symposium on Privacy and Security. 1982: 2-10

[15] Osborn S, Sandhu R, Munawer Q. Configuring Role-based Access Control to Enforce Mandatory and Discretionary Access Control Policies. ACM Transactions on Information and System Security, 2000, 3(2): 85-106

[16] Berger J L, Picciotto J, Woodward J P L, et al. Compartmented mode workstation: Prototype highlights. IEEE Transactions on Software Engineering, 1990, 16(6):608-618

[17] Bell D, LaPadula L. Secure Computer System: Unified Exposition and Multics Interpretation. Technical Report, MTR-2997. Rev1. MITRE Corporation, Bedford, MA, July 1975