基于可信状态的多级安全模型及其应用研究

张晓菲[1],许 访[2],沈昌祥[3]
(1. 中国科学院研究生院信息安全国家重点实验室,北京 100049;2. 海军装备研究院指挥自动化研究所,北京 100036;
3. 北京工业大学计算机学院,北京 100022)

摘 要: 本文提出了一种基于可信状态的多级安全模型,它以 BLP 模型为基础,引入可信度和可信状态测量函数,利用可信计算平台的完整性测量、存储和报告功能,检测进程和被访问对象的可信状态,并针对不同类型访问对象,动态调节进程访问范围,提高模型的抗攻击能力. 文中说明了模型的基本设计思想,形式化描述和可信状态转换过程,证明执行新规则后系统仍然处于安全状态. 最后,本文还介绍了模型在操作系统内核的实施框架,及其实现性能分析.

关键词: 安全操作系统; 可信计算; 安全模型; 访问控制
中图分类号: TN309 **文献标识码**: A **文章编号**: 0372-2112 (2007) 08-1511-05

Research on Multilevel Security Model Based on Trustworthy State and Its Application

ZHANG Xiao-fei[1], XU Fang[2], SHEN Chang-xiang[3]
(1. *State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, Beijing* 100049, *China*;
2. *Automatization of Command Institute, Academy of Navy Equipment, Beijing* 100036, *China*;
3. *Institute of Computer Science and Technology, Beijing University of Technology, Beijing* 100022, *China*)

Abstract: This paper proposes a multilevel security model based on trustworthy state, which introduces the concept of reliability and the function of integrity measurement. It enforces diverse security rules to different kinds of objects. To prevent running intrusions, the access ranges of subjects are adjusted according to their reliabilities. The formal description of the model and the transform of its trustworthy states are given in the paper. It is proved that the system remains in a secure state after performing the security rules of the new model. Moreover, its realizing framework in operating system kernel is described, and its performance is analyzed.

Key words: secure operating system; trusted computing; security model; access control

1 引言

可信计算以可信平台模块(Trusted Platform Module, TPM)为支撑,增强终端安全体系. 安全操作系统防止非授权访问,弥补了可信计算对信息流控制的不足. 可信计算保证操作系统正确加载,反映运行环境的可信性,增强了系统自身的防护能力. 因此,构建以可信计算为基础的安全操作系统是解决终端安全的新途径.

目前,基于可信计算的安全操作系统局限于以可信计算提供物理保护,在信息流控制中仍然采用 BLP[3] 等模型. 现有 BLP 模型存在以下问题. 首先,主体安全属性继承于启动用户,高安全级用户启动的进程并不一定可信,这些进程可能访问不可信对象,因此主、客体可信度描述是安全模型不可忽视的要素. 其次,可信主体权限过大[4,5],运行时,可信主体不可信,将给系统造成严重破坏,保证可信主体确实可信是新模型应解决的问题. 第三,不同客体作用不同,使用统一策略,将会影响系统可用性,所以不同对象应采用不同安全策略.

本文把可信计算完整性测量、存储和报告功能运用于安全模型设计,提出了一种基于可信状态的多级安全模型(Multilevel Security Model Based on Trustworthy State, MSMBTS),不同类型客体使用不同安全规则,检测主、客体可信状态,调节主体访问范围. 文中给出了模型定义和可信状态转换过程,证明执行新规则后系统仍然处于安全状态. 最后,介绍了模型的实现框架,并就其性能进行了分析.

2 相关工作

作为新一代安全计算平台,NGSCB[1] (Next Generation Secure Computing Base) 具有进程隔离,数据封装存储,安全输入/输出通道和远程验证四个主要安全功能. IBM Watson 研究中心在可信计算平台上基于 Linux2.6 内核开发了可信 Linux 客户端[2] (Trusted Linux Client, TLC). TLC 使用可信引导保证操作系统及安全内核的正

收稿日期:2006-08-25;修回日期:2007-04-16
基金项目:国家 863 高技术研究发展计划(No. 2006AA01Z440)

确加载,安全内核验证运行文件的完整性,由于采用 BLP 和 Biba 模型控制信息流,TLC 仍然面临传统模型的安全问题.

BLP 可信扩展研究分为非等级和等级模型两种.扩展 BLP 安全模型[6]中的可信主体没有程度区分,主体获取访问权除满足安全公理外,还必须是可信的,文中没有说明可信主体的衡量标准,可操作性差.BLP 二维标识模型[7]的主、客体都有一个与安全级类似的可信度,它起着完整性保护的作用,并未反映运行时主、客体的可信状态.

3 基于可信状态的多级安全模型

本节将介绍 MSMBTS 设计思想,描述模型组成、安全规则和可信状态转换过程,证明规则的安全性.

3.1 MSMBTS 设计

可信度是指主体或客体可以信赖的程度.主体可信度包括可信度值和可信状态.客体可信度由创建该客体的主体可信度值和可信状态组成.

系统客体分为两类,一类是内容相对固定运行时必须访问的资源,其可信状态必须被检测,另一类是内容不固定可信状态不易检测的用户数据.当主、客体均处于非不可信状态时,不同类型客体对应不同安全规则.

处于可信状态的主体不仅可以访问满足安全级要求的固定内容客体,通过可信代理,还能访问不满足安全级要求的这类客体.可信代理是一种在访问固定内容客体且不符合安全级要求时才启动的可信主体,它保证了系统的可用性.与文献[8]相比,本文根据可信主体只需满足简单安全规则的特性,取消安全级调整,按照操作规则[3]描述并证明每个状态,而文献[8]中创建/删除客体,授予访问权,调整客体安全级等均缺乏相应状态描述及证明.此外,由于未考虑 BLP 和 Biba 模型保密级和完整级范畴间的关系,模型规则和定理证明均存在缺陷.

对非固定内容客体,当主体处于可信状态,调整其尽可能访问高密级客体,且操作完成后,主体状态变为“不可判定”;当主体处于不可判定状态,则调整其尽可能访问低密级客体.

3.2 MSMBTS 描述

本节将给出模型定义和可信状态转换过程,并证明规则的安全性.

3.2.1 MSMBTS 定义

与文献[3]相同,S 为主体集,S^T 为可信主体集,$S' = S - S^T$ 为不可信主体集,O 为客体集,$A = \{r, w, a, e\}$ 表示主体请求的访问方式.$TState = \{trusty, untrusty, unchecked\}$ 为可信状态集.请求类型集 $RA = \{g, r\}$,g 表示获得访问权,r 表示释放访问权.请求集 $RQ = (RA \times S \times O \times A)$,$\forall rq \in RQ$ 表示主体请求获得或释放对客体的某种操作.

定义 1 主体可信状态测量函数:$im\_s : S \to TState$,$\forall s \in S$,$im\_s(s) = trusty$,表示 s 处于可信状态,$im\_s(s) = untrusty$ 或 $im\_s(s) = unchecked$,分别表示 s 处于不可信或不可判定状态.

定义 2 可信度值:$D = \{d_1, d_2, \cdots d_n\}$,$\forall d \in D$ 为非负整数,表示可以置信的程度.$\forall d, d' \in D$,$|d - d'|$ 表示 d 和 d' 之间的可信度值之差.

定义 3 主体可信度:$\forall s \in S$,$r_s = (d_s, im\_s_s)$ 表示 s 当前可信度为 r_s,其中,可信度值为 d_s,当前可信状态为 im_s_s.若 $s \in S^T$,d_s 为系统最大可信度值 $System\_High_t$,$im\_s_s = trusty$ 表示可信主体运行时可信.

定义 4 客体可信状态测量函数:$im\_o : O \to TState$,$\forall o \in O$,$im\_o(o) = trusty$ 表示 o 处于可信状态,$im\_o(o) = untrusty$ 或 $im\_o(o) = unchecked$,分别表示 o 处于不可信或不可判定状态.

定义 5 客体可信度:$\forall o \in O$,$r_o = (d_o, im\_o_o)$ 表示 o 当前可信度为 r_o,其中,$d_o \in D$ 继承于创建客体的主体的可信度值;当前可信状态为 im_o_o.

模型客体分为两类,固定内容客体 O_{fixed},$\forall o \in O_{fixed}$,$im\_o(o) = trusty$ 或 $im\_o(o) = untrusty$;可变内容客体 $O_{unfixed}$,$O_{unfixed} = O - O_{fixed}$,$\forall o \in O_{unfixed}$,$im\_o(o) = unchecked$.可信主体访问处于不可判定状态的客体后,进入不可判定状态.

定义 6 安全级集合:$L = \{(c, k), c \in C_f, k \subseteq K\}$,其中,$C_f$ 为正整数保密级集合,$\forall m, n \in C_f$,$m \succ n$ 表示保密级 m 高于保密级 n.$System\_High_f$ 为系统最高保密级,$System\_Low_f$ 为系统最低保密级.$K = \{k_1, k_2, \cdots, k_r\}$ 为非等级范畴集,$\forall k_1, k_2 \in K$,$k_1 \cap k_2 = \Phi$.设 $l_1 = (c_1, k_1)$,$l_2 = (c_2, k_2)$,$l_1 \geq l_2$ 当且仅当 $c_1 \geq c_2$,且 $k_1 \supseteq k_2$.

定义 7 安全认定函数:$cm : O_{fixed} \to \{accept, reject\}$,$\forall o \in O_{fixed}$,$cm(o) = accept$ 或 $cm(o) = reject$,分别表示客体通过或未通过安全认定.

定义 8 系统状态:$\forall v \in V = (B \times M \times F \times T \times H)$ 为一个系统状态.其中,$B = P(S \times O \times A)$,$\forall b \in B$ 记录当前主体对客体的访问操作;M 为访问控制矩阵;$F = (f_s, f_c, f_o)$,$f_s(s)$ 为 s 的最大安全级;$f_c(s)$ 为 s 的当前安全级,且 $f_s(s) \geq f_c(s)$;$f_o(o)$ 为 o 的安全级;$T = (t_s, t_o)$,$t_s(s)$ 为 s 的可信度;$t_o(o)$ 为的可信度;H 为当前客体间的层次结构.

定义 9 保密级调整函数:$\delta : D \to C_f$,$\forall s \in S$,$o \in O$,$\delta(d_s, d_o)$ 表示当 s 的可信度值为 d_s,o 的可信度值为 d_o 时,对应调节保密级 c_f,且满足 $c_{fc(s)} - c_f \geq System\_$

Low_f, $c_{fc(s)} + c_f \leq System\_High_f$. c_f 与 $|d_s - d_0|$ 成正比,即 $|d_s - d_0|$ 值越大, c_f 值越大.

3.2.2 MSMBTS 安全规则

可信自主安全规则:一个系统状态 $v = (b \times M \times f \times t \times H)$ 满足可信自主安全规则,当且仅当 $(s_i, o_j, x) \in (b \Rightarrow x \in M_{ij}$,且 $im\_s(s) \neq untrusty$,且 $im\_o(o) \neq untrusty$.

可信简单安全规则:一个系统状态 $v = (b \times M \times f \times t \times H)$ 对主体集 S, $s \in S$, $im\_s(s) = trusty$,客体集 O_{fixed}, $o \in O_{fixed}$, $im\_o(o) = trusty$ 满足可信简单安全规则,当且仅当 $(s_i, o_j, x) \in b \Rightarrow$

$x = e$ 或 a;

或 $x = r$ 或 w,且 $f(s) \geq f_o(o)$.

可信安全规则:一个系统状态 $v = (b \times M \times f \times t \times H)$ 对主体集 S', $s \in S$, $im\_s(s) = trusty$,客体集 O_{fixed}, $o \in O_{fixed}$, $im\_o(o) = trusty$,满足可信安全规则,当且仅当

$(s, o, r) \in b \Rightarrow f_c(s) \geq f_o(o)$;或 $f_c(s) \prec f_o(o)$,且满足可信读规则; (1)

$(s, o, w) \in b \Rightarrow f_c(s) = f_o(o)$;或 $f_c(s) \prec f_o(o)$,且满足可信读规则;或 $f_c(s) \succ f_o(o)$,且满足可信追加写规则;

$(s, o, a) \in b \Rightarrow f_c(s) \leq f_o(o)$;或 $f_c(s) \succ f_o(o)$,且满足可信追加写规则. (2)

受控可信安全规则 A:一个系统状态 $v = (b \times M \times f \times t \times H)$ 对于主体集 S, $s \in S$, $im\_s(s) = trusty$,客体集 $O_{unfixed}$,满足受控可信安全规则 A,当且仅当

$(s, o, r) \in b \Rightarrow \delta(d_s, d_0) \leq c_{fo(o)} \leq c_{fc(s)}$,且 $k_{fo(o)} \subseteq k_{fc(s)}$,且 $v' = (b', M, f, t', H)$,其中, $b' = b \cup (s, o, r)$, $t' = (t_s', t_o)$, $im\_s'(s) = unchecked$,即 $ts'(s) = (d_s, unchecked)$;

$(s, o, w) \in b \Rightarrow f_c(s) = f_o(o)$,且 $v' = (b', M, f, t', H)$,其中, $b' = b \cup (s, o, w)$, $t' = (t_s', t_o)$, $im\_s'(s) = unchecked$,即 $t_s'(s) = (d_s, unchecked)$;

$(s, o, a) \in b \Rightarrow c_{fc(s)} + \delta(d_s, d_o) \leq c_{fo(o)} \leq System\_High_f$,且 $k_{fc(s)} \subseteq k_{fo(o)}$,且 $v' = (b', M, f, t', H)$,其中, $b' = b \cup (s, o, a)$, $t' = (t_s', t_o)$, $im\_s'(s) = unchecked$,即 $t_s'(s) = (d_s, unchecked)$.

满足受控可信安全规则 A 的主体的读和追加写范围如图 1 所示,此时,处于可信状态的主体将尽可能读或追加写高保密级客体.

受控可信安全规则 B:一个系统状态 $v = (b \times M \times f \times t \times H)$ 对于主体集 S, $s \in S$, $im\_s(s) = unchecked$,客体集 $O_{unfixed}$,满足受控可信安全规则 B,当且仅当

$(s, o, r) \in b \Rightarrow c_{fc(s)} - c_{fo(o)} \geq \delta(d_s, d_o)$,且 $k_{fo(o)} \subseteq k_{fc(s)}$;

$(s, o, w) \in b \Rightarrow f_c(s) = f_o(o)$;

$(s, o, a) \in b \Rightarrow c_{fc(s)} \leq c_{fo(o)} \leq System\_High_f - \delta(d_s, d_o)$,且 $k_{fc(s)} \subseteq k_{fo(o)}$.

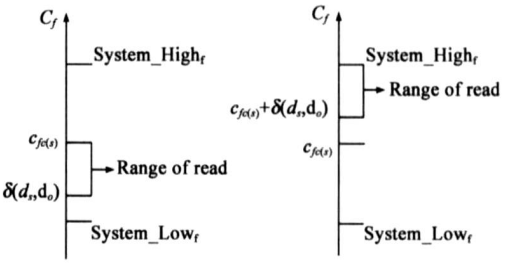

图 1　受控可信安全规则 A 中主体的读、追加写操作范围

满足受控可信安全规则 B 的主体的读和追加写范围如图 2 所示,此时,处于不可判定状态的主体将尽可能读或追加写低保密级客体.

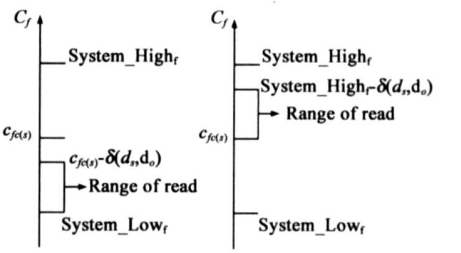

图 2　受控可信安全规则 B 中主体的读、追加写操作范围

可信读规则:在可信安全规则条件(1)下,若 $(s, o, r) \in M$,对请求 (g, s, o, r) 采用以下步骤处理:

(1) 启动可信代理 s_a, $s_a \in S^T$,构造 f^1,使得 $f_s^1 = (System\_High_f, k_{fo(o)})$, $f_c^1 = f_c$, $f_o^1 = f_o$;构造 t^1,使得 $t_s^1 = (System\_High_t, trusty)$, $t_0^1 = t_0$; $H^1 = H$. 若 $(s_a, o, r) \in M$,则 $M^1 = M$;否则对 (s_a, o, r) 授权,使得 $M^1 = M \cup (s_a, o, r)$,执行请求 (g, s_a, o, r) 后, $b^1 = b \cup (s_a, o, r)$,系统进入状态 $v^1 = (b^1, M^1, f^1, t^1, H^1)$.

(2) 可信代理 s_a 创建客体 o',构造 f^2,使得 $f_s^2 = f_s^1$, $f_c^2 = f_c^1$, $f_o^2 = f_c$;构造 t^2,使得 $t_s^2 = t_s^1$, $t_o^2 = (System\_High_t, trusty)$;构造 H^2,使得 $H^2 = H^1 \cup (o')$,系统进入状态 $v^2 = (b^2, M^2, f^2, t^2, H^2)$,其中, $b^2 = b^1$, $M^2 = M^1$.

(3) 对 (s_a, o', a) 授权,使得 $M^3 = M^2 \cup (s_a, o', a)$,执行请求 (g, s_a, o', a) 后, $b^3 = b^2 \cup (s_a, o', a)$,可信代理把从 o 中读出的内容写入 o',系统进入状态 $v^3 = (b^3, M^3, f^3, t^3, H^3)$,其中, $f^3 = f^2$, $t^3 = t^2$, $H^3 = H^2$.

(4) 对 (s, o', r) 授权,使得 $M^4 = M^3 \cup (s, o', r)$,执行请求 (g, s, o', r) 后, $b^4 = b^3 \cup (s, o', r)$,系统进入状态 $v^4 = (b^4, M^4, f^4, t^4, H^4)$,其中, f^4 满足 $f_s^4 = f_s$, $f_c^4 = f_c$, $f_o^4 = f_o^2$, t^4 满足 $t_s^4 = t_s$, $t_o^4 = t_o^2$; $H^4 = H^3$.

(5) s_a 删除 o',系统进入状态 $v^5 = (b^5, M^5, f^5, t^5, H^5)$,其中, $b^5 = b^4 - \{(s_a, o', a), (s, o', r)\}$, $M^5 = M^4 - \{(s_a, o', a), (s, o', r)\}$, $f^5 = f^2$, $t^5 = t^2$, $H^5 = H^4 - (o')$.

可信追加写规则:在可信安全规则条件(2)下,若$(s,o,a)\in M$,对(g,s,o,a)采用以下步骤处理:

(1)启动可信代理$s_a, s_a\in S^T$,创建客体o',构造f^1,使得$f_s^1=(\text{System_High}_f, k_{fc(s)})$,$f_c^1=f_c, f_o^1=f_c$;构造$t^1$,使得$t_s^1=(\text{System_High}_t, trusty)$,$t_o^1=(\text{System\_High}_t, trusty)$;$b^1=b$;$M^1=M$;$H^1=H\cup(o')$;系统进入状态$v^1=(b^1,M^1,f^1,t^1,H^1)$.

(2)对(s,o',a)授权,使得$M^2=M^1\cup(s,o',a)$,执行请求(g,s,o',a)后,$b^2=b^1\cup(s,o',a)$,系统进入状态$v^2=(b^2,M^2,f^2,t^2,H^2)$.其中,$f^2$满足$f_s^2=f_s, f_c^2=f_c, f_o^2=f_o^1, t^2$满足$t_s^2=t_s^1, t_o^2=t_o^1, H^2=H^1$.

(3)对(s_a,o',r)授权,使得$M^3=M^2\cup(s_a,o',r)$,执行请求(g,s_a,o',r)后,$b^3=b^2\cup(s_a,o',r)$,系统进入状态$v^3=(b^3,M^3,f^3,t^3,H^2)$.其中,$f^3$满足$f_s^3=f_s^1, f_c^3=f_c^1, f_o^3=f_o^1, t^3$满足$t_s^3=t_s^1, t_o^3=t_o^1, H^3=H^2$.

(4)若$cm(o')=accept$,进入下一步.否则,s_a删除o',系统进入状态$v^{5\prime}=(b^{5\prime},M^{5\prime},f^{5\prime},t^{5\prime},H^{5\prime})$,其中,$b^{5\prime}=b^3-\{(s,o',a),(s_a,o',r)\}$;$M^{5\prime}=M^3-\{(s,o',a),(s_a,o',r)\}$;$f^{5\prime}=f^1, t^{5\prime}=t^1, H^{5\prime}=H^3-(o')$.拒绝$(g,s,o,a)$.

(5)若$(s_a,o,a)\in M$,则$M^4=M^3$;否则对(s_a,o,a)授权,使得$M^4=M^3\cup(s_a,o,a)$,执行请求(g,s_a,o,a)后,$b^4=b^3\cup(s_a,o,a)$,系统进入状态$v^4=(b^4,M^4,f^4,t^4,H^4)$,其中,$f^4$满足$f_s^4=f_s^3, f_c^4=f_c^3, f_o^4=f_o, t^4$满足$t_s^4=t_s^3, t_o^4=t_o, H^4=H^3$.

(6)s_a删除o',系统进入状态$v^5=(b^5,M^5,f^5,t^5,H^5)$,其中,$b^5=b^4-\{(s,o',a),(s_a,o',r)\}$,$M^5=M^4-\{(s,o',a),(s_a,o',r)\}$,$f^5=f^1, t^5=t^1, H^5=H^4-(o')$.

3.2.3 MSMBTS 可信状态转换

模型可信状态转换过程如图3所示.主体从一个经认证的可信状态开始运行,记作S_{trusty}.访问固定内容客体O_{fixed}时,由可信简单安全规则和可信安全规则,操作完成后,仍然处于可信状态.由受控可信安全规则A,访问非固定内容客体$O_{unfixed}$后,S_{trusty}进入不可判定状态,记作$S_{unchecked}$.

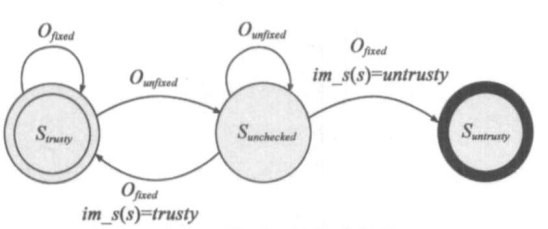

图3 模型可信状态转换

若$S_{unchecked}$请求O_{fixed},由可信简单安全规则和可信安全规则,测量其当前可信状态,若$im_s(s)=trusty$,主体重新返回可信状态,并执行授权操作.否则,主体可信状态标记为"$untrusty$",终止运行.若$S_{unchecked}$请求操作$O_{unfixed}$,由受控可信安全规则B,主体可信状态保持不变.

3.2.4 MSMBTS 定理证明

定理1 若$v=(b\times M\times f\times t\times H)$是一个满足BLP模型的安全状态,则在可信安全规则条件(1)下,系统仍然进入安全状态.

反证法:状态是一个安全状态[3],当且仅当v满足ds-property,ss-property,并且对S满足*-propetry.

假设在可信安全规则条件(1)下,系统不能进入安全状态.即v为安全状态,由可信读规则,v^1至v^5中至少存在一个不安全状态.

若v^1为不安全状态,由可信读规则(1),$(s_a,o,r)\notin b$,$(s_a,o,r)\in M^1$,满足ds-property;$s_a\in S^T$,$f_s^1=(\text{System\_High}_f, k_{fo(o)})$,$f_s^1\geqslant f_o$,满足ss-property;$v$为安全状态,对$S$满足*-property,而可信读规则(1)不存在非可信主体操作,v^1对S满足*-property.故v^1为安全状态.

若v^2为不安全状态,由可信读规则(2),$b^2=b^1$,且$\forall s\in S, \forall x\in A, (s,o',x)\notin b^1, M^2=M^1, f_c^2=f_c^1=f_c, f_o^2=f_c$,$v^1$为安全状态,创建客体$o$后,$v^2$仍然满足三条安全规则,故为安全状态.

若v^3为不安全状态,由可信读规则(3),$(s_a,o',a)\notin b^2$,$(s_a,o',a)\in M^3$,满足ds-property;$s_a\in S^T$,满足ss-property;v^2为安全状态,对S满足*-$property$,步骤(3)不存在非可信主体操作,所以,v^3对S满足*-$property$.故v^3安全.

若v^4为不安全状态,由可信读规则(4),$(s,o',r)\notin b^3$,$(s,o',r)\in M^4$,满足ds-property;$s\in S'$,且$f_c^4=f_c, f_o^4=f_o^2=f_c$,满足*-property;$v^3$为安全状态,满足ss-property,所以,$v^4$满足ss-property.故$v^4$为安全状态.

若v^5为不安全状态,由于v^4为安全状态,根据可信读规则(5),若$(s_i,o_j,x)\in b^5$,则$x\in M_{ij}$,删除客体o后,v^5仍然满足三条安全规则,故为安全状态.

综上所述,v^1至v^5均为安全状态,假设不成立,命题正确.

定理2 若$v=(b\times M\times f\times t\times H)$是一个满足BLP模型的安全状态,则在可信安全规则条件(2)下,系统仍然进入安全状态.

证明方法与定理1类似,不再详述.

4 可信状态多级安全模型实施框架

可信访问控制内核模块(Trusted Access Control Kernel Model,TACKM)构建在含有TPM的可信平台上,可信引导功能保证内核和TACKM的正确加载.运行时,TACKM利用LSM截获访问请求,获取主、客体安全属性,通过可信软件栈中可信内核服务(Trusted Core Service,TCS)提供的接口,判定被测对象的可信性,依据规则控制系统信息流.TACKM总体结构如图4所示,处理流程和接口概述如下.

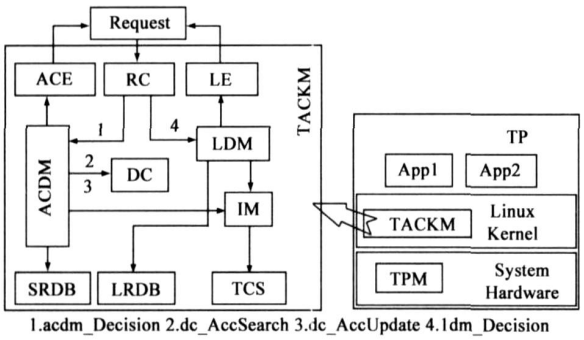

图 4　系统总体结构及 TACKM 结构

访问请求由请求分类处理(Request Classifier,RC)调用 acdm_Decision 送入访问控制判决(Access Control Decision-maker,ACDM).获取相关安全信息后,调用 dc_AccSearch 查找决策缓存(Decision Catch,DC)中是否存在相应决策,存在则返回结果.否则依据安全规则库(Security Rule Database,SRDB)中的策略,根据需要由完整性测量(Integrity Measurement,IM)调用 TCS 检测对象可信状态,最后判定结果由判决实施(Access Control Enforcer,ACE)返回.根据使用频率,ACDM 调用 dc_AccUpdate 更新 DC 中的决策项.

标记请求由 RC 调用 ldm_Decision 发送给标记判决(Label Decision-maker,LDM),判定对象可信状态后,根据标记规则库(Label Rule Database,LRDB)中的规则给出标记结果,经标记实施(Label Enforcer,LE)返回.

5　性能分析

系统使用 SHA-1 度量进程和系统文件,依据规则实施访问控制,由此给系统带来额外开销.进程加载时,首先验证其可信状态,并判定是否执行当前操作请求.运行中,TACKM 对非固定内容客体的访问请求直接做出决策,对固定内容客体的访问请求验证主、客体可信状态后再判决.

系统中,程序的首次运行耗时最长,启动程序越大,系统性能影响越明显.启动后,运行中访问控制的实施使性能平均降低较少,可信状态检测和访问控制决策对程序性能影响较首次运行明显减小.这是因为进程初次加载时,需要通过 IO 操作从磁盘读取数据,载入后再次检验时,进程已位于内存,IO 操作减少,验证时耗随之下降.

6　总结

本文提出的 MSMBTS 使用可信度和可信状态测量函数,反映运行时主、客体可信状态,不同类型客体依照不同安全规则调节主体访问范围.文中详细介绍了模型定义,证明了其安全特性,并就其实现框架和性能进行了说明.在今后的研究中,将改进现有模型的特权管理,完善模型安全性分析.

参考文献:

[1] Micorsoft. Security model for the next generation secure computing base[EB/OL]. http://www.microsoft.com/resources/ngscb/archive.mspx,2004-08-14.

[2] David S, Mimi Z. A trusted Linux Client[EB/OL]. http://www.acsaadmin.org/2004/workshop/David-Safford.pdf, 2005-11-14.

[3] Bell D E, LaPadula L J. Secure computer system: unified exposition and multics interpretation[R]. Mitre Report, MTR-2997 Rev 1, 1976.

[4] Bell D E. Security policy modeling for the next-generation packet switch[A]. Proceeding of the IEEE 1988 Symposium on Security and Privacy[C]. Oakland: IEEE Computer Society Press, 1988. 212-216.

[5] Schell R R, Tao T F, et al. Designing the GEMSOS security kernel for security and performance[A]. Proceeding of the 8th National Computer Security conference[C]. Maryland: IEEE Computer Society Press, 1985. 108-119.

[6] Kang J M, Shin W, et al. Extended BLP security model based on process reliability for secure Linux kernel[A]. Proceedings of the 2001 Pacific Rim International Symposium on Dependable Computing[C]. Washington: IEEE Computer Society Press, 2001. 299-303.

[7] 蔡谊,郑志蓉,等.基于多级安全策略的二维标识模型[J].计算机学报,2004,27(5):619-624.
Cai yi, Zheng Zhirong et al. A planar attributes model based on multi level security policy[J]. Chinese Journal of Computers, 2004,27(5):619-624. (in Chinese)

[8] 郑志蓉,蔡谊,等.操作系统安全结构框架中应用类通信安全模型的研究[J].计算机研究与发展,2005,42(2):322-328.
Zheng Zhirong, Cai yi, et al. Operating system application class BLP model Biba model B/S application[J]. Journal of Computer Research and Development, 2005, 42(2): 322-328. (in Chinese)

作者简介:

张晓菲　女,1974 年生于湖北武汉.中国科学院研究生院信息安全国家重点实验室博士.研究方向为信息安全、可信计算.
E-mail: salro@sina.com

许　访　男,1976 年生于江苏.博士,研究方向为信息安全、指挥自动化.

沈昌祥　男,1940 年生于浙江.院士,博导,研究方向为信息安全.