

# 主体当前敏感标记动态确定方案研究

石文昌, 梁洪亮, 孙玉芳

(中国科学院软件研究所, 北京 100080; 中科红旗软件技术有限公司, 北京 100080)

**摘 要:** Bell & LaPadula (BLP) 模型是实现安全操作系统多级安全性 (MLS) 的基础, 根据主体当前敏感标记对主体访问客体的请求作判断是 BLP 模型的关键. 本文讨论一个动态确定主体当前敏感标记的方案, 首先, 作为进一步探讨的基础, 对 BLP 模型进行简要描述, 接着, 以 BLP 模型为基础构造所讨论的方案, 最后, 给出构成完整方案的限定条件, 并证明该限定条件的必要性和正确性.

**关键词:** 计算机安全; 安全操作系统; 安全模型; 敏感标记; 主体

**中图分类号:** TP309; TP316 **文献标识码:** A **文章编号:** 0372-2112 (2001) 08-1046-04

## On Scheme for Dynamic Determination of Subject's Current Sensitivity Label

SHI Wen-chang, LIANG Hong-liang, SUN Yu-fang

(Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

**Abstract:** Bell & LaPadula (BLP) model is the ground for the implementation of multilevel security (MLS) in secure operating systems. The key spirit of BLP model is to make decision on subject's request to access objects according to subject's current sensitivity label. This paper discusses a scheme for the dynamic determination of subject's current sensitivity label. First, as the basis of further exploration, the BLP model is briefly described. Then, the scheme in question is constructed based on the BLP model. Finally, the qualifying condition for the scheme is presented and its necessity and correctness are demonstrated.

**Key words:** computer security; secure operating system; security model; sensitivity label; subject

### 1 引言

Bell & LaPadula (BLP) 模型<sup>[1]</sup>是实现安全操作系统多级安全性 (MLS) 的基础<sup>[2,7]</sup>, 很多安全操作系统实现对 BLP 模型的支持<sup>[1,3-8]</sup>. 根据主体当前敏感标记对主体访问客体的请求作判断是 BLP 模型的关键, 安全操作系统中的活动主体是进程. 以前有关这方面的讨论主要支持静态当前敏感标记<sup>[1,3]</sup>, 本文讨论一个动态当前敏感标记的方案, 并证明它的限定条件.

进程发出访问请求时, BLP 模型对比进程和客体的敏感标记, 根据安全公理的条件作判断, 决定授权或拒绝访问. 为了讨论动态当前敏感标记的支持方案, 把判断空间划分为两个部分, 如图 1 所示.

在外层判断空间所作的判断是常规判断, 相当于静态当前敏感标记方案中的判断. 在内层判断空间所作的判断基于当前敏感标记的动态确定. 进程发出访问请求时, 首先在外层判断空间根据

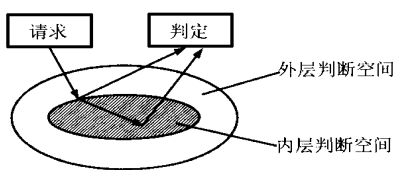


图 1 判断空间

BLP 公理进行判断, 不符合访问条件时, 进入内层判断空间, 合理地调整进程的当前敏感标记, 并作判断. 本文描述内层判断的规则, 并重点讨论外层判断对内层判断的影响.

### 2 BLP 模型简述

作为本文讨论的基础, 下面给出 BLP 模型的简要描述, 完整的模型及应用说明参见文 [1].

**约定 1** 设  $X$  和  $Y$  是任意集合, 记  $PX$  为  $X$  的幂集, 记  $X^Y := \{f | f: Y \rightarrow X\}$ .

**约定 2**  $S$  是主体集合,  $O$  是客体集合,  $A$  是访问属性集合,  $C$  是等级分类集合,  $K$  是非等级类别集合,  $S_T$  是可信主体集合,  $S = S - S_T$ .

**定义 1** 当前访问集合  $B := P(S \times O \times A)$ .

访问矩阵集合  $M := \{M | M \text{ 是矩阵 } M \text{ 中元素 } M_{ij} \text{ 是主体 } S_i \text{ 对客体 } O_j \text{ 的访问属性集}\}$ .

敏感标记集合  $L := \{(C, K) | C \in C, K \in K\}$ .

$[L_1 = (C_1, K_1) \quad L_2 = (C_2, K_2) \quad L] [(L_1 \geq L_2) := (C_1 \geq C_2 \quad K_1 \supseteq K_2)]$ .

敏感标记函数集合  $F :=$

$\{(f_s, f_o, f_c) | f_s: S \rightarrow L^S, f_o: O \rightarrow L^S, f_c: S \rightarrow L^S (\forall S(S(f_s(S) \geq f_c(S))))\}$ ,  $f_s$  称为主体敏感标记函数,  $f_o$  称为客

收稿日期: 2000-10-12; 修回日期: 2001-04-10

基金项目: 国家 863 高科技项目 (No. 863-306-ZD12-14-2); 国家自然科学基金项目 (No. 60073022); 中国科学院知识创新工程项目 (No. KGCX1-09)

体敏感标记函数,  $f_c$  称为主体当前敏感标记函数.

客体层次关系集合  $H := \{ H \mid H (PO)^O \}$  性质 1 性质 2), 性质定义略.

状态集合  $V :=$

$\{ (b, M, f, H) \mid b \in B, M \in M, f \in F, H \in H \}$ .

约定 3  $b(S : x, y, \dots, z) :=$

$\{ O \mid (S, O, x) \in b, (S, O, y) \in b, \dots, (S, O, z) \in b \}$ .

约定 4  $x$  为可读不可写属性,  $a$  为可写不可读属性,  $w$  为可读且可写属性,  $e$  为不可读且不可写(可执行)属性.

公理 1 ( $ss$ -特性) 状态  $v = (b, M, f, H)$  满足简单安全特性( $ss$ -特性) :=

$S \leq S' \Rightarrow [ (O \in b(S : x, w)) \Rightarrow (f_S(S) \geq f_O(O)) ]$ .

公理 2 ( $*$ -特性) 状态  $v = (b, M, f, H)$  满足相对于  $S$  的  $*$ -特性 :=

$(S \leq S' \Rightarrow [ (O \in b(S : a,)) \Rightarrow (f_O(O) \geq f_C(S)) ])$  (1)

$(S \leq S' \Rightarrow [ (O \in b(S : w,)) \Rightarrow (f_O(O) = f_C(S)) ])$  (2)

$(S \leq S' \Rightarrow [ (O \in b(S : x,)) \Rightarrow (f_C(S) \geq f_O(O)) ])$  (3)

公理 3 ( $ds$ -特性) 状态  $v = (b, M, f, H)$  满足自主安全特性( $ds$ -特性) :=

$(S_i, O_j, x) \in b \Rightarrow x \in M_{ij}$ .

### 3 当前敏感标记动态确定方案

#### 3.1 动态方案的理论框架

定义 2  $L$  是敏感标记集合, 设  $L_1 = (C_1, K_1)$ ,  $L, L_2 = (C_2, K_2)$ ,  $L, L_3 = (C_3, K_3)$ ,  $L$ , 函数  $\gamma : L \times L \rightarrow L$  定义为:  $L_3 = \gamma(L_1, L_2)$  由以下原则确定:

if  $(C_1 \leq C_2)$  then  $C_3 = C_2$  else  $C_3 = C_1$ ;

if  $(K_1 \subseteq K_2)$  then  $K_3 = K_1 \cap K_2$  else  $K_3 = K_1$ .

推论 1  $(L_1, L_2)$  是  $L_1$  和  $L_2$  的最大下界. (由定义 2 直接得出.)

定义 3  $L$  是敏感标记集合, 设  $L_1 = (C_1, K_1)$ ,  $L, L_2 = (C_2, K_2)$ ,  $L, L_3 = (C_3, K_3)$ ,  $L$ , 函数  $\gamma : L \times L \rightarrow L$  定义为:  $L_3 = \gamma(L_1, L_2)$  由以下原则确定:

if  $(C_1 \geq C_2)$  then  $C_3 = C_2$  else  $C_3 = C_1$ ;

if  $(K_1 \supseteq K_2)$  then  $K_3 = K_1 \cap K_2$  else  $K_3 = K_1$ .

推论 2  $(L_1, L_2)$  是  $L_1$  和  $L_2$  的最小上界. (由定义 3 直接得出.)

约定 5  $rq(S_i, O_j, x)$  表示主体  $S_i$  对客体  $O_j$  的  $x$  访问请求.

约定 6  $L_{RH}$  和  $L_{WL}$  表示在一个进程的生存期间使用的两个判断参考量,  $L_{RH}$  的初值为系统最小敏感标记值,  $L_{WL}$  的初值为系统最大敏感标记值.

规则 1 在  $(b, M, f, H)$  状态,  $x \in M_{ij}$ ,  $M, f_c(S_i) \leq f_o(O_j)$  时, 对  $rq(S_i, O_j, x)$  的处理:

if  $(f_s(S_i) \geq f_o(O_j) \text{ and } f_o(O_j) \leq L_{WL})$  then

构造  $f^*$ , 使得  $f_s^* = f_s, f_o^* = f_o, f_c^*(S_i) = (f_c(S_i), f_o(O_j))$ , 当  $S = S_i$  时,  $f_c^*(S) = f_c(S)$ .

$L_{RH} = (L_{RH}, f_o(O_j))$ .

对  $rq(S_i, O_j, x)$  授权, 构造  $b^*$ , 使得  $b^* = \{ (S_i, O_j, x) \}$

$b$ , 进入  $(b^*, M, f^*, H)$  状态.

else 拒绝  $rq(S_i, O_j, x)$ .

规则 2 在  $(b, M, f, H)$  状态,  $a \in M_{ij}$ ,  $M, f_c(S_i) \leq f_o(O_j)$  时, 对  $rq(S_i, O_j, a)$  的处理:

if  $(f_o(O_j) \geq L_{RH})$  then

构造  $f^*$ , 使得  $f_s^* = f_s, f_o^* = f_o, f_c^*(S_i) = (f_c(S_i), f_o(O_j))$ , 当  $S = S_i$  时,  $f_c^*(S) = f_c(S)$ .

$L_{WL} = (L_{WL}, f_o(O_j))$ .

对  $rq(S_i, O_j, a)$  授权, 构造  $b^*$ , 使得  $b^* = \{ (S_i, O_j, a) \}$ ,  $b$ , 进入  $(b^*, M, f^*, H)$  状态.

else 拒绝  $rq(S_i, O_j, a)$ .

规则 3 在  $(b, M, f, H)$  状态,  $w \in M_{ij}$ ,  $M, f_c(S_i) \leq f_o(O_j)$  时, 对  $rq(S_i, O_j, w)$  的处理:

if  $(f_s(S_i) \geq f_o(O_j) \text{ and } f_o(O_j) \leq L_{WL} \text{ and } f_o(O_j) \geq L_{RH})$  then

构造  $f^*$ , 使得  $f_s^* = f_s, f_o^* = f_o, f_c^*(S_i) = f_o(O_j)$ , 当  $S = S_i$  时,  $f_c^*(S) = f_c(S)$ .

$L_{RH} = (L_{RH}, f_o(O_j))$ ,  $L_{WL} = (L_{WL}, f_o(O_j))$ .

对  $rq(S_i, O_j, w)$  授权, 构造  $b^*$ , 使得  $b^* = \{ (S_i, O_j, w) \}$ ,  $b$ , 进入  $(b^*, M, f^*, H)$  状态.

else 拒绝  $rq(S_i, O_j, w)$ .

规则 1、2、3 构成了主体当前敏感标记动态确定方案的主要内容.

#### 3.2 动态方案的应用说明

动态方案假设的前提条件是: 公理 3 得到满足, 公理 2 得不到满足. 这是由问题的性质决定的.

问题的核心是在满足适当条件的情况下调整主体的当前敏感标记, 使得新的敏感标记满足公理 2 中的相应要求. 所谓适当条件, 一是公理 1 的条件得到满足, 二是主体的历史访问行为能确保敏感标记的调整不违反安全原则.

历史访问行为由约定 6 中给出的  $L_{RH}$  和  $L_{WL}$  表示, 分别表示已经读过的客体敏感标记的最大值和已经写过的客体敏感标记的最小值.

规则 1 是“读”规则, 它把主体的当前敏感标记  $f_c(S_i)$  调整为  $f_c(S_i)$  与客体敏感标记  $f_o(O_j)$  的最小上界, 把  $L_{RH}$  调整为  $L_{RH}$  与  $f_o(O_j)$  的最小上界.

规则 2 是“写”规则, 它把主体的当前敏感标记  $f_c(S_i)$  调整为  $f_c(S_i)$  与  $f_o(O_j)$  的最大下界, 把  $L_{WL}$  调整为  $L_{WL}$  与  $f_o(O_j)$  的最大下界.

规则 3 是“读写”规则, 它把主体的当前敏感标记  $f_c(S_i)$  调整为客体敏感标记  $f_o(O_j)$ , 把  $L_{RH}$  调整为  $L_{RH}$  与  $f_o(O_j)$  的最小上界, 把  $L_{WL}$  调整为  $L_{WL}$  与  $f_o(O_j)$  的最大下界.

### 4 动态当前敏感标记方案的限定条件

#### 4.1 限定条件的理论框架

当前敏感标记的动态确定是在内层判断空间按照规则 1、2、3 完成的, 但会受到外层判断空间中的活动的影响. 规则 4 给出了限定条件.

规则 4 在外层判断空间对访问请求进行授权时, 必须完成以下的相应操作:

- (a) 对  $rq(S_i, O_j, \underline{x})$  授权时, 要执行 .  
 (b) 对  $rq(S_i, O_j, \underline{a})$  授权时, 要执行 .  
 (c) 对  $rq(S_i, O_j, \underline{w})$  授权时, 要执行 .

**定理 1** 遵守规则 4 是规则 1 和 2 和 3 满足 BLP 公理的必要条件.

**证明** 采用反证法.

( ) 设新进程  $S_i$  欲请求对文件  $O_k$  和  $O_j$  进行访问, 请求次序是:

- (1)  $rq(S_i, O_k, \underline{x})$ ; (2)  $rq(S_i, O_j, \underline{a})$ .

选择:  $f_C(S_i) = f_O(O_k)$

$$f_O(O_k)|_k = f_O(O_j)|_k$$

$$f_O(O_k)|_c = f_O(O_j)|_{c+1}$$

(注: 设  $f_O(O_k) = (C_k, K_k)$ , 则

$$f_O(O_k)|_c \text{ 表示 } C_k, f_O(O_k)|_k \text{ 表示 } K_k).$$

处理  $rq(S_i, O_k, \underline{x})$  时, 因为

$$f_C(S_i) = f_O(O_k) \Rightarrow f_C(S_i) \leq f_O(O_k)$$

在外层判断空间可以直接对  $rq(S_i, O_k, \underline{x})$  授权. 假设在这种情形下不执行 .

处理  $rq(S_i, O_j, \underline{a})$  时, 因为

$$f_C(S_i) = f_O(O_k) > f_O(O_j) \Rightarrow f_C(S_i) \not\leq f_O(O_j)$$

所以需要进入内层判断空间(运用规则 2).

因为  $S_i$  是新进程, 此时  $L_{RH}$  为最小值, 显然有

$$f_O(O_j) \geq L_{RH}$$

执行 后, 有

$$f_C^*(S_i)|_k = f_O(O_j)|_k = f_O(O_k)|_k = f_O^*(O_k)|_k$$

$$f_C^*(S_i)|_c = f_O(O_j)|_c = f_O(O_k)|_c - 1 = f_O^*(O_k)|_c - 1$$

即  $f_C^*(S_i) < f_O^*(O_k)$

但此时有  $O_k \ b^*(S_i; \underline{x})$

违反公理 2 的 (3).

所以, 若不遵守规则 4(a), 则规则 2 不满足公理 2.

( ) 设新进程  $S_i$  欲请求对文件  $O_k$  和  $O_j$  进行访问, 请求次序是:

- (1)  $rq(S_i, O_k, \underline{a})$ ; (2)  $rq(S_i, O_j, \underline{x})$ .

选择:  $f_C(S_i) = f_O(O_k)$

$$f_O(O_j)|_k = f_O(O_k)|_k$$

$$f_O(O_j)|_c = f_O(O_k)|_{c+1}$$

$$f_S(S_i) = f_O(O_j)$$

处理  $rq(S_i, O_k, \underline{a})$  时, 因为

$$f_C(S_i) = f_O(O_k) \Rightarrow f_C(S_i) \leq f_O(O_k)$$

在外层判断空间可以直接对  $rq(S_i, O_k, \underline{a})$  授权. 假设在这种情形下不执行 .

处理  $rq(S_i, O_j, \underline{x})$  时, 因为

$$f_C(S_i) = f_O(O_k) < f_O(O_j) \Rightarrow f_C(S_i) \not\geq f_O(O_j)$$

所以需要进入内层判断空间(运用规则 1).

因为  $S_i$  是新进程, 此时  $L_{WL}$  为最大值, 显然有

$$f_S(S_i) = f_O(O_j) \geq f_O(O_j) \text{ and } f_O(O_j) \leq L_{WL}$$

执行 后, 有

$$f_C^*(S_i)|_k = f_O(O_j)|_k = f_O(O_k)|_k = f_C^*(O_k)|_k$$

$$f_C^*(S_i)|_c = f_O(O_j)|_c = f_O(O_k)|_{c+1} = f_O^*(O_k)|_{c+1}$$

即  $f_C^*(S_i) > f_O^*(O_k)$

但此时有  $O_k \ b^*(S_i; \underline{a})$

违反公理 2 的 (1).

所以, 若不遵守规则 4(b), 则规则 1 不满足公理 2.

( ) 设新进程  $S_i$  欲请求对文件  $O_k$  和  $O_j$  进行访问, 请求次序是:

- (1)  $rq(S_i, O_k, \underline{w})$ ; (2)  $rq(S_i, O_j, \underline{w})$ .

选择:  $f_C(S_i) = f_O(O_k)$

$$f_O(O_j)|_k = f_O(O_k)|_k$$

$$f_O(O_j)|_c = f_O(O_k)|_{c+1}$$

$$f_S(S_i) = f_O(O_j)$$

处理  $rq(S_i, O_k, \underline{w})$  时, 因为

$$f_C(S_i) = f_O(O_k)$$

在外层判断空间可以直接对  $rq(S_i, O_k, \underline{w})$  授权. 假设在这种情形下不执行 .

处理  $rq(S_i, O_j, \underline{w})$  时, 因为

$$f_C(S_i) = f_O(O_k) < f_O(O_j) \Rightarrow f_C(S_i) \leq f_O(O_j)$$

所以需要进入内层判断空间(运用规则 3).

因为  $S_i$  是新进程, 此时  $L_{WL}$  为最大值,  $L_{RH}$  为最小值, 显然有

$$f_S(S_i) = f_O(O_j) \geq f_O(O_j) \text{ and } f_O(O_j) \leq L_{WL} \text{ and } f_O(O_j) \geq L_{RH}$$

执行 后, 有

$$f_C^*(S_i)|_k = f_O(O_j)|_k = f_O(O_k)|_k = f_O^*(O_k)|_k$$

$$f_C^*(S_i)|_c = f_O(O_j)|_c = f_O(O_k)|_{c+1} = f_O^*(O_k)|_{c+1}$$

即  $f_C^*(S_i) > f_O^*(O_k)$

但此时有  $O_k \ b^*(S_i; \underline{w})$

违反公理 2 的 (2).

所以, 若不遵守规则 4(c), 则规则 3 不满足公理 2.

综上所述, 得证.

#### 4.2 限定条件的应用说明

限定条件要求在外层判断空间中进行授权时, 必须象在内层判断空间中进行授权一样, 以相同的方式调整  $L_{RH}$  和  $L_{WL}$  的值.

#### 5 结束语

本文通过规则 1、2、3 给出了在内层判断空间动态调整主体当前敏感标记以及作判断的方法, 通过规则 4 给出了外层判断空间的限定条件, 并证明了其存在的必要性. 在规则 4 的限定下, 可以证明规则 1 和 2 和 3 满足 BLP 公理, 但这已超出了本文的范围. 本文讨论的方案在红旗安全操作系统 RS-Linux 的开发中已付诸实施.

**参考文献:**

- [ 1 ] D E Bell, L J LaPadula. Secure computer system: unified exposition and MULTICS interpretation [R]. MTR-2997 Rev. 1, The MITRE Corporation, Bedford, MA 01730, Mar 1976.  
 [ 2 ] T Y Lin. Bell and LaPadula axioms: a "New" paradigm for an "Old"

# 短波宽带天线匹配网络的设计

朱义胜,郑紫微

(大连海事大学信息工程学院,大连 116026)

## Design of Matching Network for Short-Wave Antenna

ZHU Yi-sheng, ZHENG Zi-wei

(Dalian Maritime University, Dalian 116026)

**摘 要:**本文基于统计学习理论构造了一种快速自适应随机搜索算法,证明了算法的收敛性。给出了一种简易实用的宽带天线匹配设计新方法。应用该自适应算法进行天线匹配设计,不仅算法简单,易于编程实现;而且能够快速设计出具有较好性能的匹配网络,非常适用于各种短波、超短波天线的匹配设计问题。

如图 1 所示的短波天线阻抗匹配网络的设计问题,一直受到人们的广泛重视<sup>[1]</sup>。图中  $R_0 = 50$  表示信源内阻,  $Z_L(s)$  是天线阻抗。

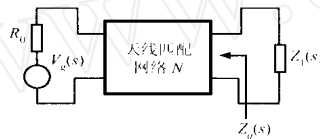


图 1 天线阻抗匹配问题

用 HP8753D 矢量网络分析仪进行测量的丹东康乃达电器有限公司生产的 KD-12 宽带天线阻抗 (27.5MHz ~ 39.5MHz) 的实部和虚部如图 2 所示。

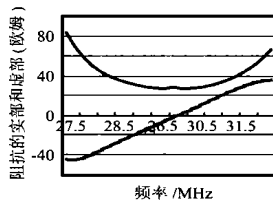


图 2 KD-12 天线阻抗特性

本文利用统计高斯-牛顿法<sup>[2]</sup>设计短波天线阻抗匹配网络获得满意的结果。

设待设计的天线匹配网络

由负载端口向匹配网络看去的策动点阻抗为  $Z_q(s)$ ,

$$\begin{aligned} Z_q(j) &= R_q(\omega) + jX_q(\omega) \\ Z_L(j) &= R_L(\omega) + jX_L(\omega) \end{aligned} \quad (1)$$

网络的功率传输特性可以写做<sup>[3]</sup>

$$G(\omega) = \frac{4R_q(\omega)R_L(\omega)}{[R_q(\omega) + R_L(\omega)]^2 + [X_q(\omega) + X_L(\omega)]^2} \quad (2)$$

假定匹配网络采用 T 型或  $\pi$  型结构,  $R_q(\omega)$  可以表示为  $R_q(\omega, \mathbf{V})$ ,  $X_q(\omega)$  可以表示为  $X_q(\omega, \mathbf{V})$ , 其中

$$\mathbf{V} = [C_1, \dots, C_i, L_1, \dots, L_j, m_1, \dots, n_k] \quad (3)$$

是待定的变量,即组成匹配网络的  $i$  个电容、 $j$  个电感和  $k$  个变压器。向量  $\mathbf{V}$  的维数为  $M = i + j + k$ 。因此,匹配设计问题可转化为寻找合适的  $\mathbf{v}$  使得式 (4) 最小。

$$E(\mathbf{V}) = \frac{1}{2n} \sum_{i=1}^n [G_0 - G(\omega_i, \mathbf{V})]^2 \quad (4)$$

其中  $G_0$  为某一常数,它代表某一平坦传输功率增益。  $G(\omega, \mathbf{V})$  是向量  $\mathbf{V}$  的组成元素的非线性函数。

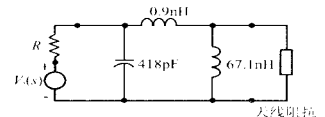


图 3 三元件天线阻抗匹配网络

(下转第 1075 页)

model [A]. Proc. of 1992 ACM SIGSAC New Security Paradigms Workshop [C], 1992.

[ 3 ] V D Gigor, E L Burch, et al. On the design and the implementation of secure xenix workstations [A]. Proc. of the 1986 IEEE Symposium on Security and Privacy [C], Apr 1986.

[ 4 ] C W Flink II, J D Weiss. System V/MLS labeling and mandatory policy alternatives [J]. AT&T Technical Journal, May/ Jun 1988.

[ 5 ] G L Grenier, R C Holt, M Funkenhauser. Policy vs mechanism in the secure TUNIS operating system [A]. Proc. of 1989 IEEE Symposium on Security and Privacy [C], 1989: 84 - 93.

[ 6 ] N A Waldhart. The army secure operating system [A]. Proc. of 1990 IEEE Computer Society Symposium on Research in Security and Privacy [C], 1990: 50 - 60.

[ 7 ] Secure computing corporation. DTOS generalized security policy specification [R]. DTOS CDRL A019, Secure Computing Corporation, 2675 Long Lake Road, Roseville, Minnesota 55113, Jun 1997.

[ 8 ] R Spencer, S Smalley, et al. The Flask security architecture: system support for diverse security policies [A]. Proc. of the 8th USENIX Security Symposium [C], Washington, DC, USA, Aug 1999.

### 作者简介:



石文昌 男, 1964 年 11 月生于广西钦州, 1987 年毕业于北京大学计算机系, 获学士学位, 1990 年于中科院软件所获硕士学位, 现为中科院软件所博士生, 主要研究方向为系统软件安全性。



梁洪亮 男, 1972 年生于山东济南, 1999 年毕业于中国石油大学计算机应用专业, 获硕士学位, 现为中科院软件所博士生, 主要研究方向为系统软件安全性。