多域环境下基于代数系统的工作流访问控制技术研究

嵌入式系统及网络实验室

导师：李仁发 教授

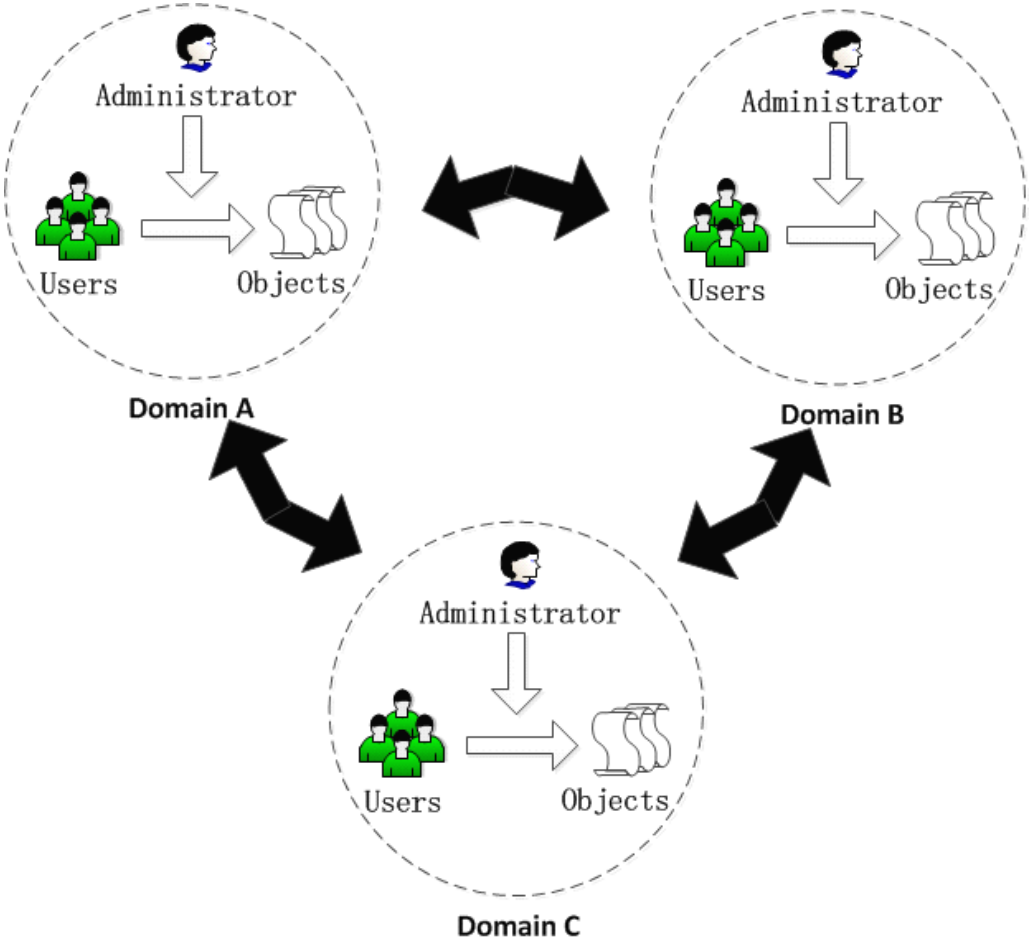
学生：唐　　鹭

2012-5-26

目录

本文主要工作

研究背景和意义



- 一个工作流任务通过执行若干个访问策略而完成的；
- 不同自治域的计算模式和安全策略是不同的；
- 各自治域中的局部策略之间呈现错综复杂的关系。

请求策略格模型

1、引入请求策略

➢请求策略*rp*（*Request Policy*）：

$$rp = (u, req) \times U \quad REQ$$

其中*u*表示用户，*req*表示访问请求，$req = S = \{s_1, s_2, s_3, \ldots, s_m\}$。

2、风险偏序关系

➢若用户安全等级不同而请求的服务集相同，则用户安全等级较低的请求策略，给系统带来的风险较大。

若$Safe(u_1) \leqslant Safe(u_2)$，$req_1 = req_2$，则$rp_2 \leqslant rp_1$。

➢若用户安全等级相同而请求的服务集不同，则假设请求的服务集req_1包含在请求的服务集req_2中，则请求服务集req_2的策略rp_2，所带来的风险较大。

请求策略格模型

格的推导

1 $(RP, \leqslant)$是偏序集

2 存在上确界

3 存在下确界

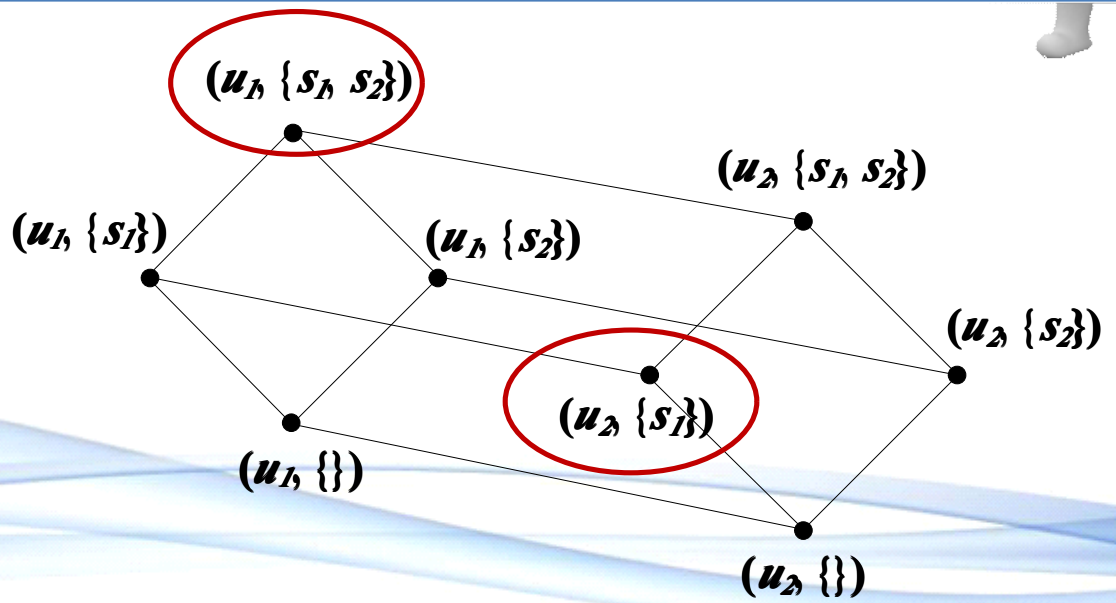
➢利用本文提出的$(RP, \leqslant)$格模型可有效制定多级安全策略。

u_1：程序设计人员

u_2：项目经理

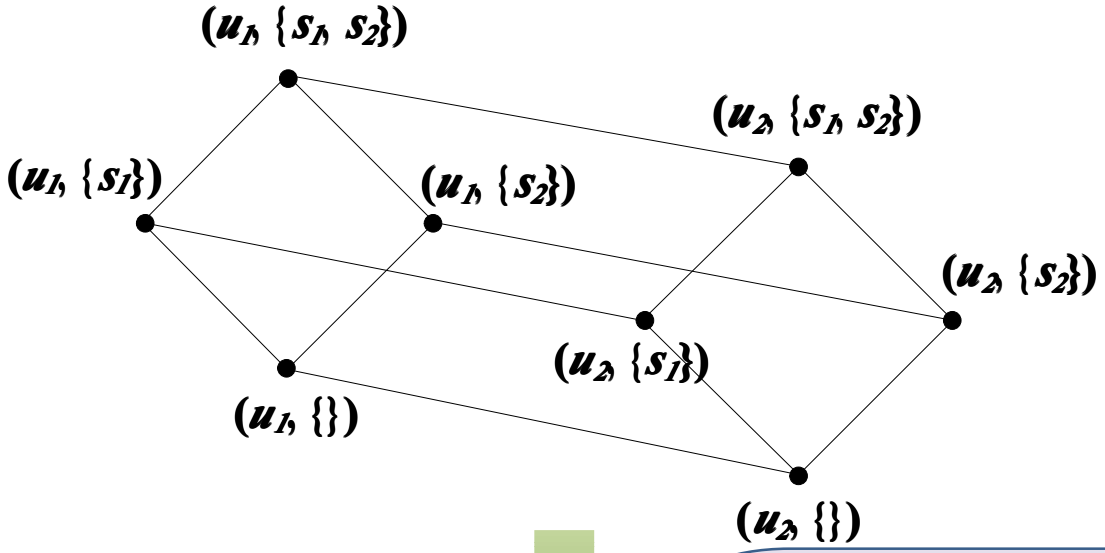
s_1：读程序文件

s_2：修改程序文件



格模型优化机制

t_1: $Safe(u_1) \leqslant Safe(u_2)$

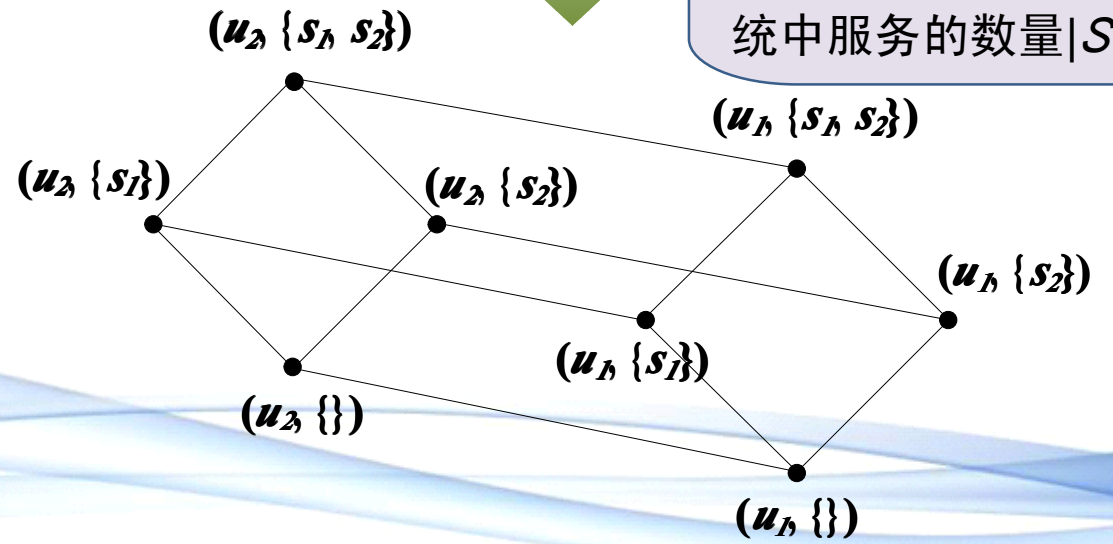


u_1负重减轻

u_2负重加重

需更新的节点数为$n \times 2^m$（n为安全等级偏序关系发生变化的用户数，m为系统中服务的数量$|S|$）

t_2: $Safe(u_2) \leqslant Safe(u_1)$



格模型优化机制

t_1: $Safe(\text{Bob}) \leqslant Safe(\text{Ella})$

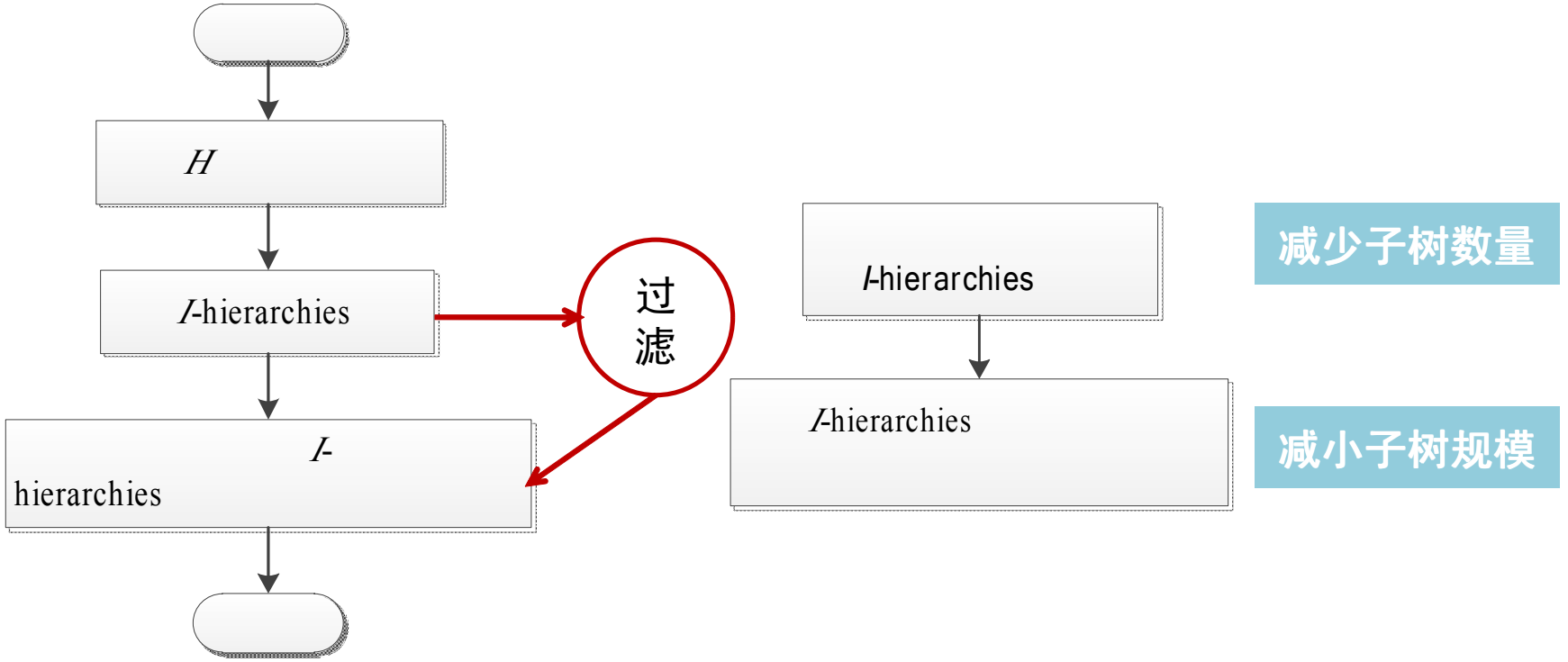
Bob负重减轻

Ella负重加重

t_2: $Safe(\text{Ella}) \leqslant Safe(\text{Bob})$

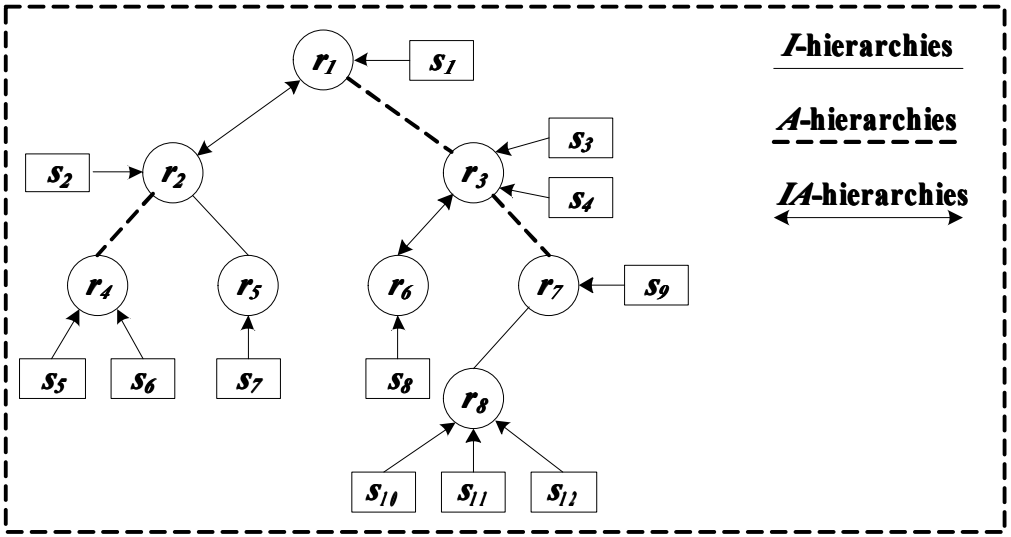
只需更新用户配置表中相应的n项内容，显然$n << n \times 2^m$ 。

MUR优化算法



- 计算整个系统所有可能的最小角色集，浪费资源。

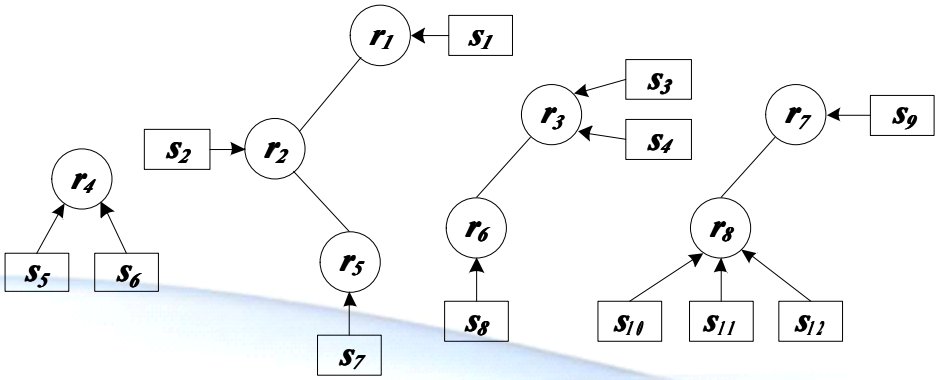
MUR优化算法



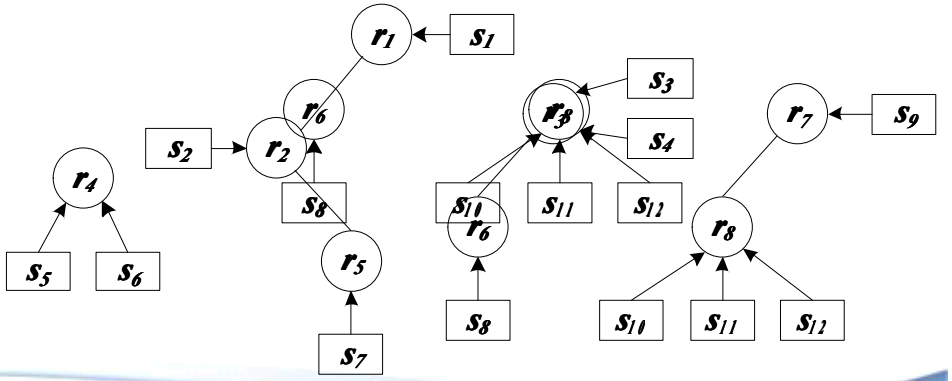
最坏情况： $range(q_1) = \{s_1, s_8, s_2, s_{10}, s_3, s_{11}, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}\}$

MUR

MUR优化算法

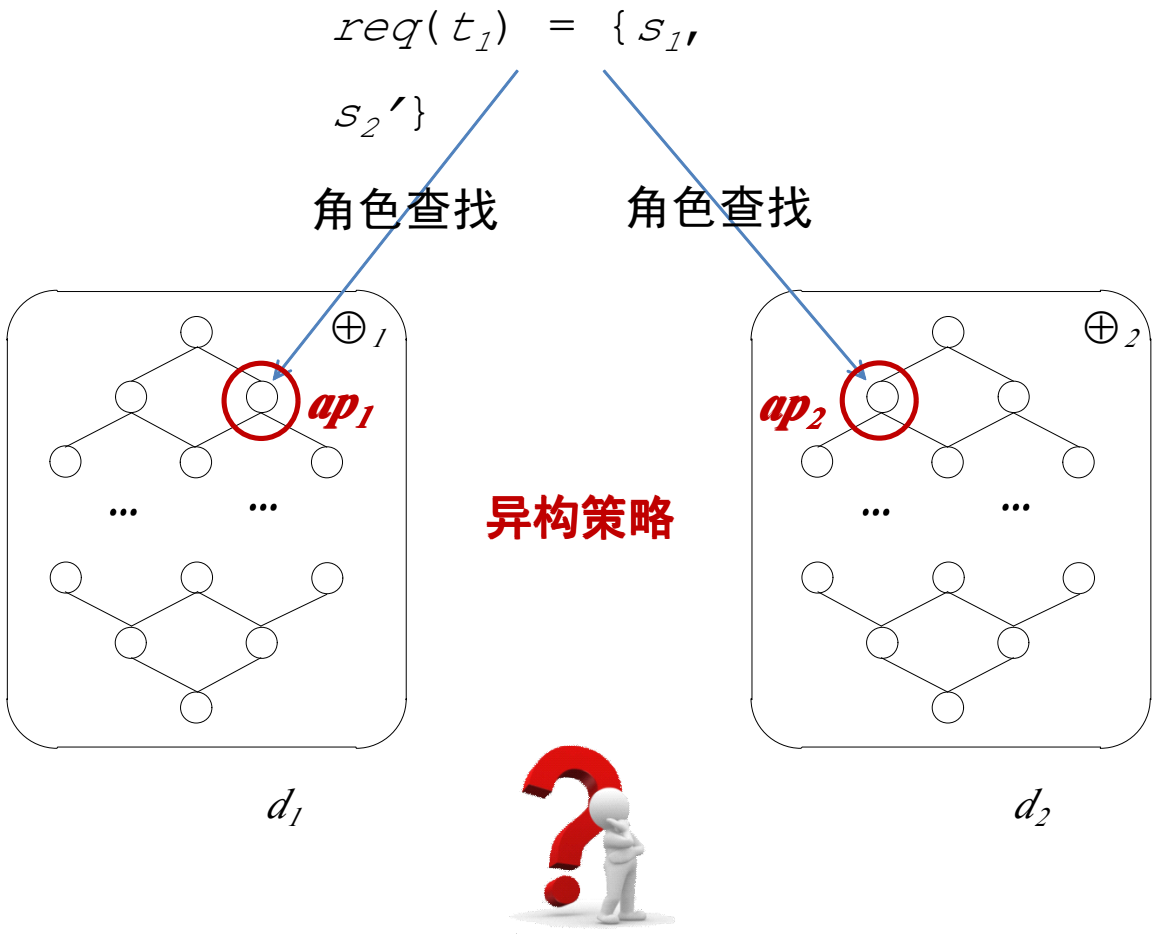


|MUR|=71



|MUR|=31

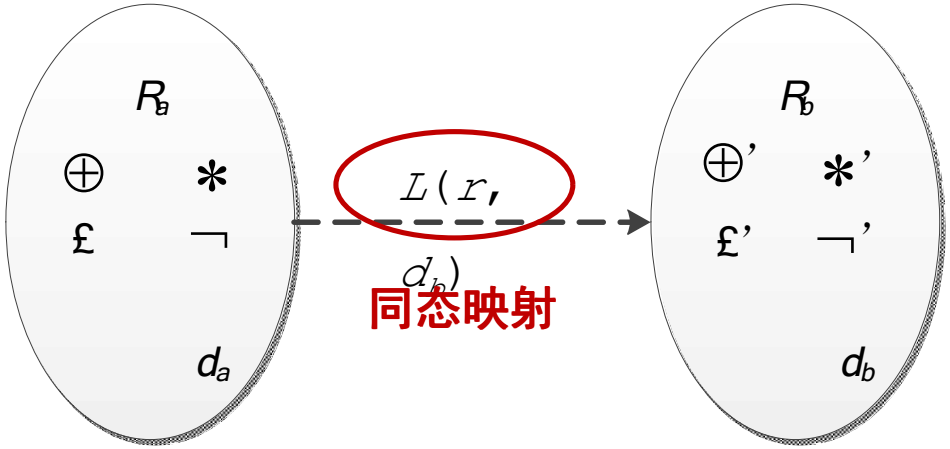
策略组合方案



策略组合方案

1、域间映射

➢根据代数系统的定义，一个自治域中的角色及角色间的运算 ⊕，*，£，¬ 就构成了一个代数系统，记为<R, ⊕，*，£, ¬ >。

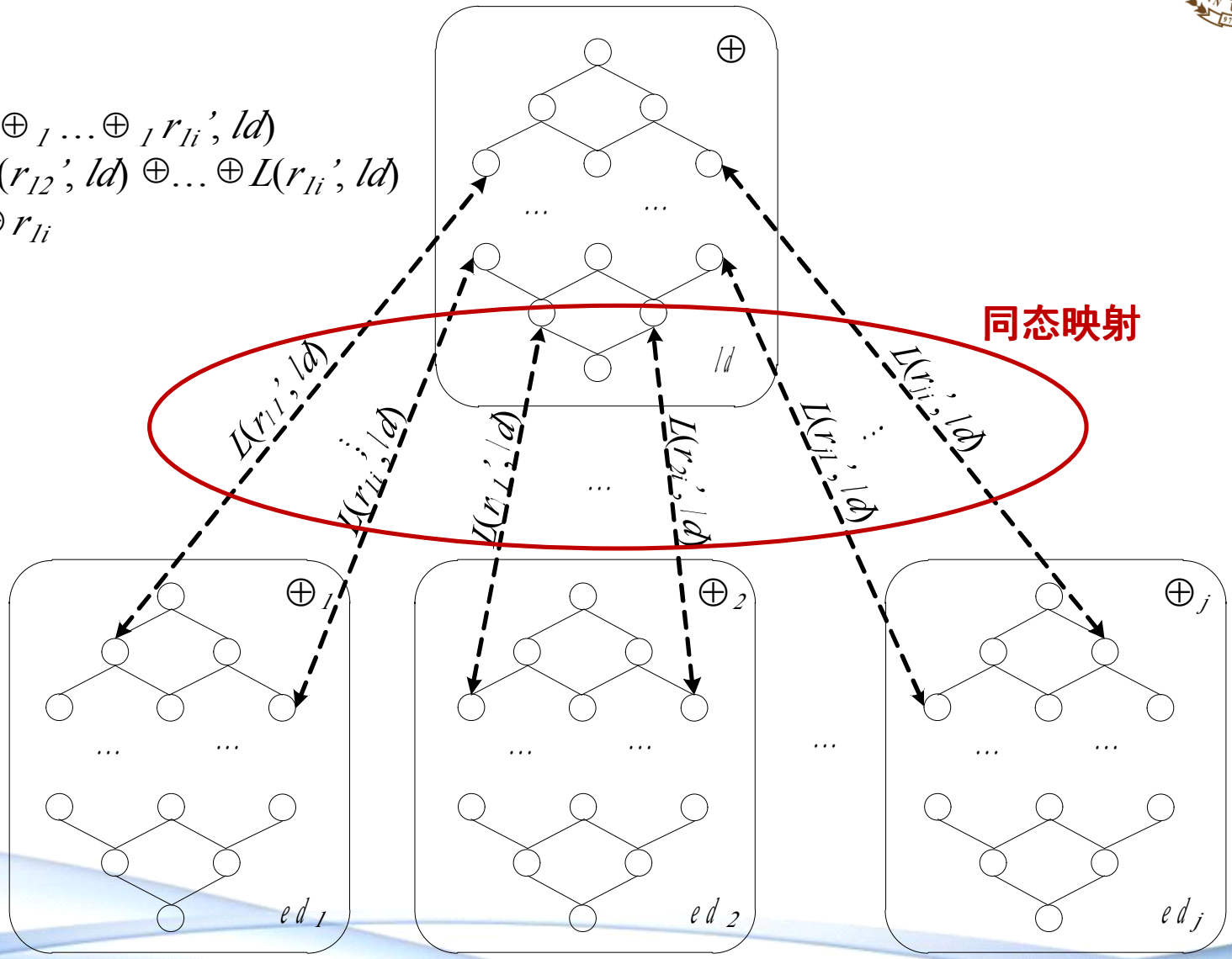


代数系统 代数系统

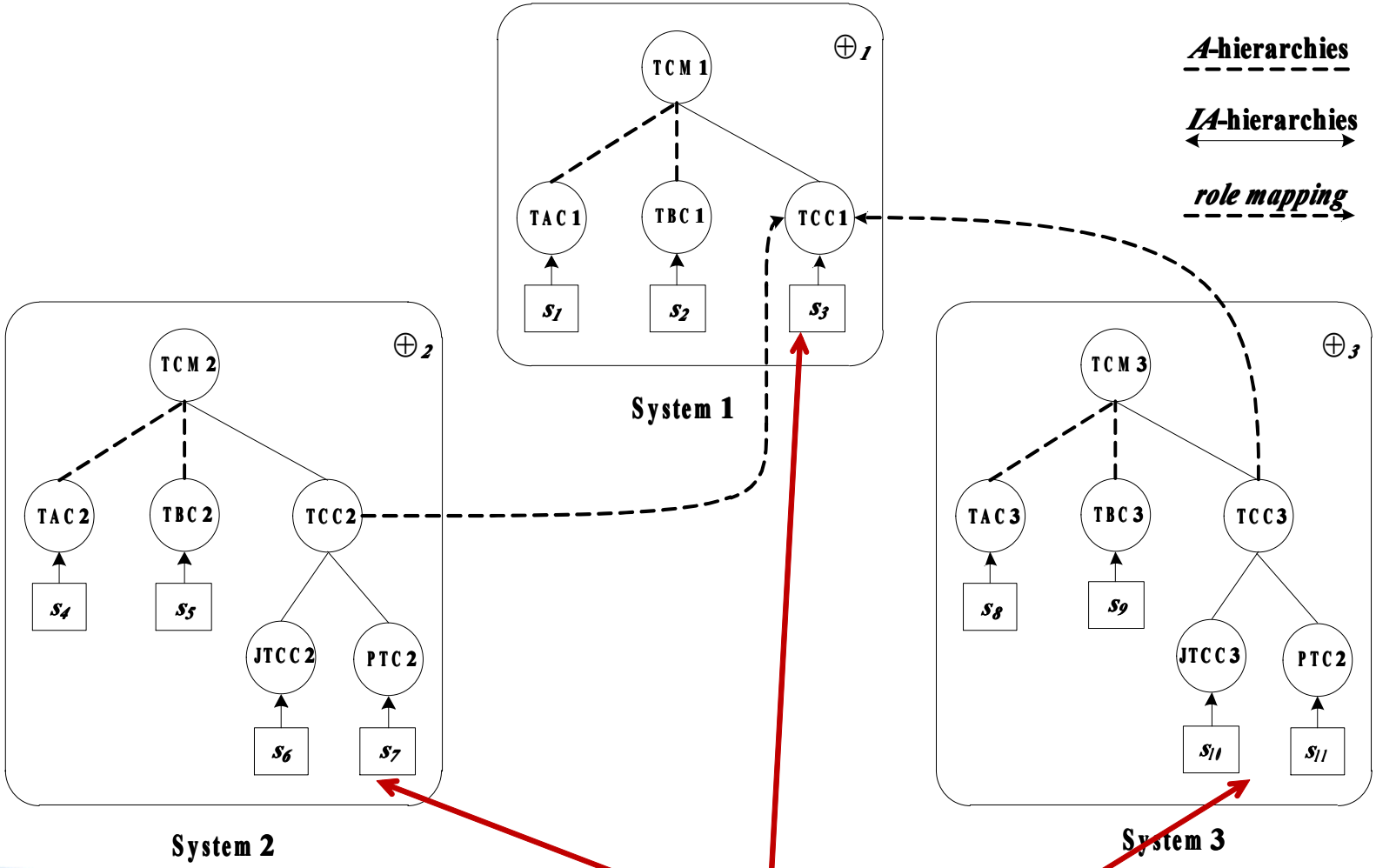
V_a = <R_a, ⊕，*，£, ¬ > V_b = <R_b, ⊕，*，£, ¬ >

策略组合方案

$$L(r_{11}' \oplus_1 r_{12}' \oplus_1 \ldots \oplus_1 r_{1i}', ld)$$
$$= L(r_{11}', ld) \oplus L(r_{12}', ld) \oplus \ldots \oplus L(r_{1i}', ld)$$
$$= r_{11} \oplus r_{12} \oplus \ldots \oplus r_{1i}$$



实例分析

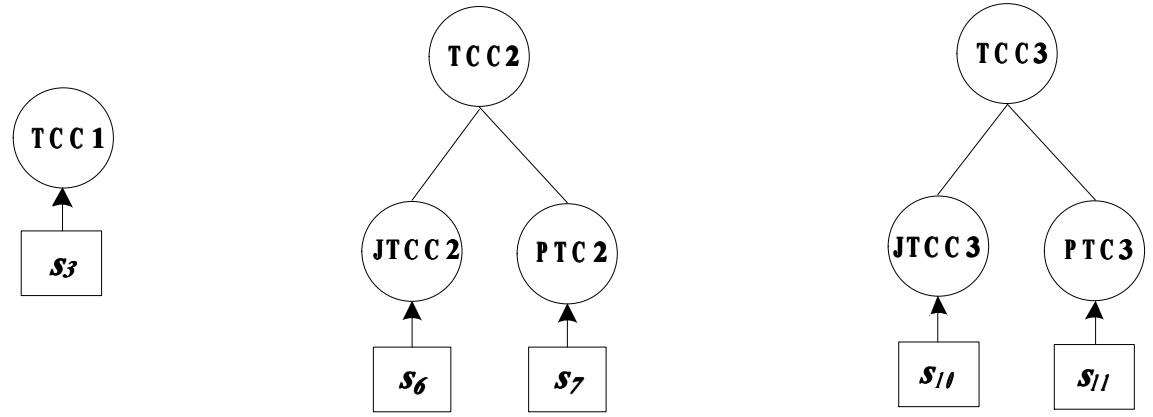


➢税收经理TCM1的用户u_1在时刻t，

提交访问申请$req(t) = \{s_3, s_6, s_7, s_{10}, s_{11}\}$

实例分析

➢ *I*-hierarchies子树



自治域	访问请求	角色集	原子策略
System1	s_3	{TCC1}	ap_1=(TCC1, {s_3})
System2	s_6, s_7	{TCC2}	ap_2=(TCC2, {s_6, s_7})
System3	s_{10}, s_{11}	{TCC3}	ap_3=(TCC3, {s_{10}, s_{11}})

实例分析

➢利用同态映射关系，可将不同自治域内的原子策略都映射到一个域内进行组合。

各域内的原子策略	映射到System1中的原子策略
ap_1=(TCC1, {s_3})	ap_1=(TCC1, {s_3})
ap_2=(TCC2, {s_6, s_7})	ap_4=(TCC1, {s_6, s_7})
ap_3=(TCC3, {s_{10}, s_{11}})	ap_5=(TCC1, {s_{10}, s_{11}})

➢组合策略$cp = \Theta AP = ap_1 \oplus ap_4 \oplus ap_5$= (TCC1, {$s_3$, s_6, s_7, s_{10}, s_{11}})。

有效避免多域环境下访问策略爆炸的情形。

结论

1、多级安全策略

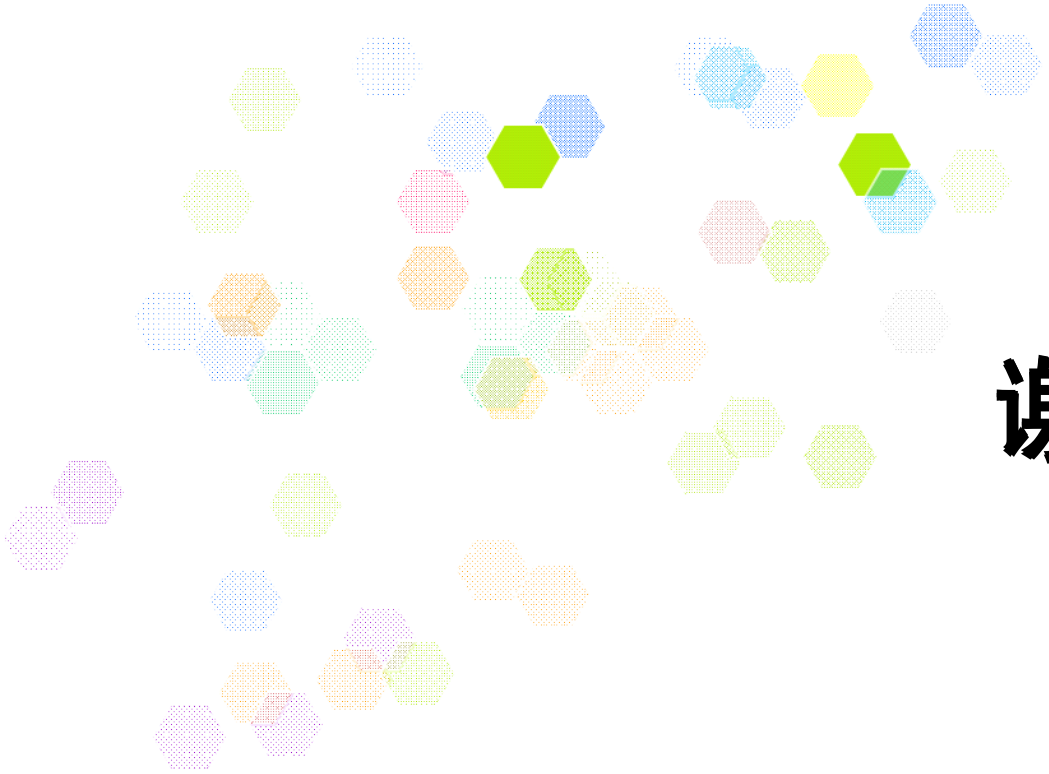
➢在请求策略之间建立基于安全性的偏序关系，并基于这种策略间的偏序关系在数学上构建格代数模型，以能有效地实施多级安全策略。

2、MUR优化算法

➢根据访问请求对*I*-hierarchies子树进行过滤，缩小角色查找的范围，提高了求解效率。

3、异构策略组合

➢证明自治域间的映射是一个同态映射。通过这种同态关系，将不同自治域内的异构策略映射到同一代数系统中进行组合。



谢谢

Email：tanglu1010@sina.com