37. 保護システムの安全性判定法†

宮　地　利　雄††

1. まえがき

計算機システム内の複数の利用者の間でプログラムやデータの共有を行いながら，意図しない利用者からのアクセスによって起る情報の漏洩や攪乱あるいは破壊を防ぐように情報の共有を制御している環境が保護システムである．保護システムの提供者は，幅広い種類の情報の共有関係を利用者が設定できるような基本機能を提供することに加えて，その機構により情報が正しく保護され続ける事をあいまいさの無い形式的な方法で証明して見せることが是非ともできなくてはならない．

各利用者を物理的に分離すれば安全ではあるが情報を共有できなくなり，また単純パスワード・システムでは，たとえば参照だけを許したい場合にも全権利を与えねばならないので安全性について十分でないことから，capability[1)-3)]の概念が Dennis と Van Horn によって提案された．**Capability** とは，アクセスされる情報の番地と，アクセス動作の種類に対応して与えられる権限（たとえば，読出し，書込等）の集合との対である．保護システムは，共有情報ごとに capability を作り出して利用者に与え，利用者が共有情報を操作する際に必ずこれを用いて操作対象を指定するよう義務づけることによって情報の共有と保護を実現する．**図-1** は Cambridge CAP 計算機内で各手続に主記憶セグメントへのアクセスのために与えられる capability の構造を示している．

保護システムの安全性を考える時に重要な点は，各利用者が持っている capability が固定されたものではなく，規則に従いながら利用者が他の利用者などの持っている capability のコピーを獲得したり逆に自分の持っている capability のコピーを他に与えるなどの行為を通して時間的に変化するという点である．このようなシステム内で与えられた規則を使って情報を共有する関係を実際に作りうるか，あるいはどのような条件下で情報漏洩が発生しうるかといった問に答えるのが保護システムに関する安全性の判定問題である．

PRL へのポインタ
相対的なベース番地
セグメントの大きさ
アクセス権限

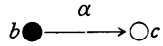
PRL: Process Resource List アクセス権限はデータ用 capability で使う読み，書き，実行の 3bit と capability 用 Capability で使う capability 読み，capability 書きの 2bit との合計 5bit で定義される．

図-1 Cambridge CAP 計算機における capability の構造．

以降の議論の準備として次節で保護システムのモデル化と安全性判定問題の形式化を行う．また，3節と4節で代表的なモデルである Take Grant モデルと HRU モデルを紹介し，これらの上で知られている結果について述べる．なお，この分野のすぐれた解説として Snyder による文献[9)]があり，本稿の解説も主としてこれによった．更に詳しい議論については同文献を参照されたい．

2. 準　備

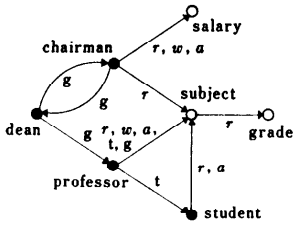
保護システムのモデル化のため，capability の2つの成分である番地と権限集合を次のような有向グラフで表す：



このグラフで，b は利用者を，c は情報を，また b から c への有向枝は b が持っている c に対する capability を表し，ラベル α はその権限集合である．グラフの節点は人間を表す●とデータを表す○の2種類があって，前者で利用者の能動性を後者で情報ファイル

† Safety check of protection system by Toshio MIYACHI (Dept. of Computer Science, Faculty of Engineering, Tokyo Institute of Technology).

†† 東京工業大学工学部情報工学科



r: read w: write a: append
e: execute t: take g: grant

図-2 保護グラフの例

の受動性を抽象化し，それぞれを能動節点および受動節点と呼ぶ．特に区別しない場合には⊗により表現する．このような情報の共有に関連する関係のすべてを有向枝として書き挙げた2種類の色の節点上の有向有限ラベルつきグラフにより保護システムの状態を表現することができる．これを**保護グラフ**と呼ぶ．**図-2**にその一例を示す．これによりシステムのある時点での状態が決まるが，「salary の情報が盗まれうるだろうか」という疑問に答えるにはさらに以下の形式化を与えておく必要がある．

一定の規則に従って利用者が capability の変更を行うことによって保護システムの状態を変えていく行為をモデル化して，保護グラフの書換えを行う規則の集合を導入する．書換え規則は基本的には $\alpha \Rightarrow \beta$ の形式で与えられ，α が保護グラフ G の部分グラフにマッチする時には，その規則を G に適用して α に対応する部分グラフを β で置換えた新しい保護グラフ G' を作り出すことができる．G に書換え規則 r を適用して G' を得る操作を $G \vdash rG'$ と書き，また r を特に指定しない場合には $G \vdash G'$ と書く．次は書換え規則の一例である：

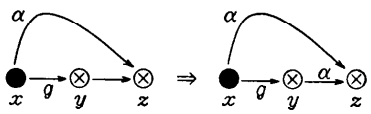


図-2 の保護グラフを例にとり，たとえば x に chairperson を y に dean を z に salary を対応させて上の規則を適用すると **図-3** の保護グラフが得られ，dean が新しく salary に対する capability を獲得する．

ここで我々の問題を次のように一般化できる：

保護グラフ G と書換え規則の集合 $\mathcal{R}$ が与えられた時，次の3条件を満す保護グラフの系列 $G_1, G_2, \cdots G_n$ が存在するかどうかを判定せよ．

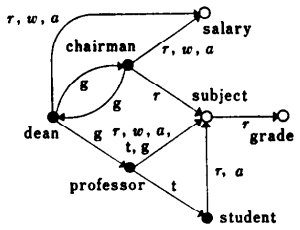


図-3 書換え規則により図2から得られた保護グラフ

(条件1) $G = G_1$,

(条件2) G_n は性質 X を持つ,

(条件3) すべての i に対して，それぞれ $r_i \in \mathcal{R}$ が存在して $G_i \vdash r_i G_{i+1}$.

ここで述語 X は情報保護に関するある事態の発生を表明するもので，換言すればそのような事態を含む状態が到達可能であるか否かを判定するのがこの問題である．また述語 X は「節点 p から q へのラベル α を持った枝が存在する」といったように表現されたことがしばしばであり，この観点に立てば推移閉包の問題の一般化として考えることもできる．注意すべき点は，一般には書換え規則が新しい節点を追加する操作を含むことから，保護グラフが限りなく大きくなる可能性を持っていることである．

3. Take-Grant モデル

3.1 定 義

Take-Grant モデル[4]と呼ばれる保護システムのモデルは，**図-4** に示した4個の書換え規則により特徴づけられる．それぞれの直観的意味は次のとおり．

(i) Grant… y に対して権限 g を持った能動節点 x が，z に対して持っている権限 α を y にも与える．

(ii) Take… y に対して権限 t を持った能動節点 x が，y が持っている z に対する権限 α を獲得する．

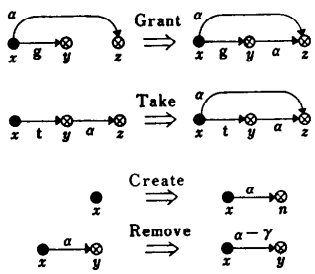


図-4 Take-Grant モデルの書換え規則

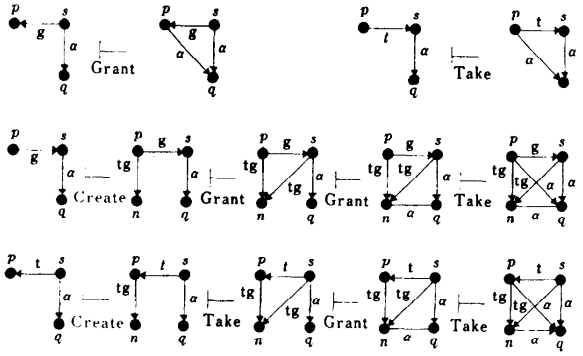


図-5 tg-連結による capability の伝播

(iii) Create…能動節点が新しい節点（子関係の利用者や自分用のファイル）を作り，これに対する任意の権限を得る．

(vi) Remove…能動節点が自分の持っている権限の全部または一部を取消す．

以上の4規則の他に Call と呼ばれる規則（利用者が x プログラム z に引数情報 y を与えて動作させるためにプロセス n を作り出す）を加えることがあるが，本質的ではないので除外して議論を進める．

本節の以降では，この Take-Grant モデルの上で実用的に興味ある2つの判定問題に関する結果を紹介する．

3.2 情報共有の実現性の判定

情報共有を実現するには，共有をする利用者 p が情報 q に対する正当なアクセス権 α を含む capability を獲得できなければならない．最初に与えられる保護グラフを G とする時，この条件を can•share (α, p, q, G) と書く．形式的には：

［定義］ can•share $(\alpha, p, q, G_0) \overset{\text{def}}{\Longleftrightarrow}$

$$^{\exists}G_1, \cdots, G_n \text{ s.t. } G_i \vdash G_{i+1}\ (0 \leq i < n) \text{ かつ } p \xrightarrow[G_n]{\alpha} q.$$

ここで $p \xrightarrow[G_n]{\alpha} q$ は，G_n 中に p から q へのラベル β（ただし $\alpha \subseteq \beta$）を持つ枝が存在することを表す．

Take-Gran モデルでは，権限 t と g を含む枝が書換え規則の中で大きな役割を持っているが，枝の向きを無視して t または g を含んだラベルを持つ枝だけからなるパスで結ばれている2節点を tg-連結といい，そのパスが1本の枝のみの時には tg-直結と言うことにしよう．受動節点を含まないような保護グラフに限ると次の定理が成立する．

［定理］ G_0 が能動節点のみからなる保護グラフなら，

can•share $(\alpha, p, q, G_0) \Longleftrightarrow$

（条件1） 節点 $s_1, s_2, \cdots, s_u$ が存在し，各 $i(1 \leq i \leq u)$ に対して $s_i \xrightarrow[G_0]{\gamma_i} q$

かつ $\alpha = \overset{u}{\underset{i=1}{\cup}} \gamma_i$.

（条件2） p は $s_1, \cdots, s_u$ に tg-連結．

証明の詳細は文献[5]に譲り，tg-連結の節点を経由して capability が伝播していく様子を図-5 に示すにとどめる．この伝播は枝の向きと無関係に双方向に可能だが，文献[12]では一方向の伝播のみを許すモデルを提案し論じている．

次に受動節点も含んだ保護グラフに議論を拡張するが，その前にいくつかの概念を導入しておく．島とは tg-連結な能動節点だけからなる極大部分グラフである．p と q を能動節点，$x_1, \cdots, x_n (n \geq 1)$ を受動節点とし，p と x_1，x_i と $x_{i+1} (1 \leq i < n)$，x_n と q がそれぞれ tg-連結である時，$p, x, \cdots, x_n$ を tg-半経路，また $p, x_1, \cdots, x_n, q$ を tg-経路と呼ぶ．枝の向きと権限により tg-経路や tg-半経路に対して，アルファベット $\{\vec{t}, \vec{g}, \overleftarrow{t}, \overleftarrow{g}\}$ 上の語を対応づけることができる．たとえば，●$\xrightarrow{t}$○$\xleftarrow{rg}$● なら $\vec{t}\,\overleftarrow{g}$，●$\xrightarrow{g}$○$\xrightarrow{tg}$○ ならば $\vec{g}\,\vec{t}$ と $\vec{g}\,\vec{g}$ が対応する．正規表現 $E_b = \vec{t}^{\,*} \cup \overleftarrow{t}^{\,*} \cup \vec{t}^{\,*}\vec{g}\,\vec{t}^{\,*} \cup \vec{t}^{\,*}\overleftarrow{g}\,\overleftarrow{t}^{\,*}$ により生成される言語に属する語が対応するような tg-経路を橋と呼ぶ．また，$E_i = \vec{t}^{\,*}\vec{g}$，$E_t = \vec{t}^{\,*}$ の語が対応する tg-半経路をそれぞれ始端部分橋，終端部分橋と呼ぶ．橋と部分橋は，tg-連結でない節点間で capability を伝播させうる唯一の路であり，この事から次の定理を得る．（図-6 を参照）

［定理］ can•share $(\alpha, p, q, G_0) \Longleftrightarrow$

（条件3） 節点 $s_1, \cdots, s_v$ が存在し，各 $i(1 \leq i \leq v)$ に対して $s_i \xrightarrow[G_0]{\gamma_i} q$ かつ $\alpha = \overset{v}{\underset{i=1}{\cup}} \gamma_i$.

（条件4） 次を満たす能動節点 $p', s_1', \cdots, s_v'$ が存在：

(i) $p = p'$ または p' が始端部分橋により p と tg-連結．

(ii) $s_i = s_i'$ または s_i' が終端部分橋により s_i と tg-連結．

（条件5） (p', s_i') の各対 $(1 \leq i \leq v)$ に対して，それぞれ島の集合 $\{I_1, \cdots, I_u\}$ $(u \geq 1)$ が存在して $p' \in I_1$，$s_i' \in I_u$，かつ I_j と $I_{j+1} (1 \leq j < u)$ の間を結ぶ橋がある．

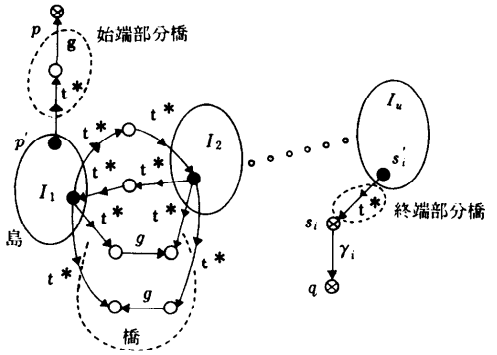


図-6 Can-share $(\alpha, p, q\ G)$ の条件

これより can・share の判定を連結性の判定に帰着させることができ，次が結論される．

[系] 与えられる保護グラフのサイズに比例した時間で can・share を判定するアルゴリズムが存在する[5]．

3.3 情報漏洩の可能性の判定

最初に capability を持っていた利用者による積極的な権限の譲渡 (Grant) なしに他の利用者が情報のアクセス権を得る情報漏洩の可能性を述語 can・steal (α, p, q, G) により表現し，次のように定義する：

[定義] can・steal $(\alpha, p, q, G_0) \overset{\text{def}}{\Longleftrightarrow}$

$\neg(p \xrightarrow[G_0]{\alpha} q)$ かつ次の条件を満足する保護グラフの列 $G_1, \cdots, G_n$ が存在する：

(i) $G_{i-1} \vdash \rho_i G_i$, $\rho_i \in \mathcal{R}$ $(1 \leq i \leq n)$,

(ii) $p \xrightarrow[G_n]{\alpha} q$,

(iii) $s \xrightarrow[G_0]{\alpha} q$ なる s に対しては，どの ρ_i も q に対する権利 α を s が譲渡する Grant 規則でない．

明らかに can・steal $(\alpha, p, q, G) \Rightarrow$ can・share (α, p, q, G) であるが，さらに次の定理が成立する．

[定理] can・steal $(\alpha, p, q, G_0) \Longleftrightarrow$

(i) $\neg(p \xrightarrow[G_0]{\alpha} q)$,

(ii) 能動節点 p' が存在して $p = p'$ または p' から p への始端部分橋が存在する．

(iii) 節点 s が存在し，$s \xrightarrow[G_0]{\alpha} q$ かつ can・share ("t", p', s, G_0) が真となる．

この定理と can・share (α, p, q, G) が線型時間で判定できたことからただちに次が得られる．

[系] 与えられる保護グラフのサイズに比例した時間でcan・steal を判定するアルゴリズムが存在する．

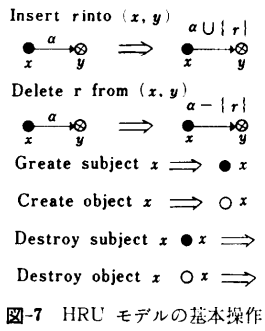


図-7 HRU モデルの基本操作

4. HRU モデル

Take-Grant モデルでは特定の4個の書換え規則だけが許されていたのに対し，Harrison と Ruzzo, Ullman により提案された **HRU モデル**は[7]，枝の始点が能動節点でなければならないが，種々の変化に富んだ書換え規則を含むことができ，保護グラフ中のループ枝の存在も許している．書換え規則は次の図式を使って定義される規則の有限集合として与えられる：

Rule-name $(x_1, x_2, \cdots, x_s)$:

if $p_1 \wedge p_2 \wedge \cdots \wedge p_t$ then $q_1; q_2; \cdots; q_u$;

ここで各 p_i は $x_{v_i} \bullet \xrightarrow{r} \otimes x_{w_i}$ $(1 \leq v_i, w_i \leq s)$ の形式で，x_{w_i} に対する権限 r を含む capability を x_{v_i} が持つ時に真となる述語を表す，また q_i は**図-7** の6種の基本操作のいずれかで，その中の節点名を $\{x_1, \cdots, x_s\}$ から選んで当てたものである．このように定義された規則は，保護グラフの節点を仮引数 $x_1, \cdots, x_s$ に割当て述語 $p_1, \cdots, p_t$ をすべて真にできた時に基本操作 $q_1, \cdots, q_u$ を順に実行するものとして解釈される．たとえば，前節の Grant 規則は次のようにして与えることができる：

Grant (x, y, z):

if $\bullet \xrightarrow[x\ y]{g} \bullet \wedge \bullet \xrightarrow[x\ z]{r} \otimes$ then $\bullet \xrightarrow[y\ z]{\alpha} \otimes \Rightarrow \bullet \xrightarrow[y\ z]{\alpha \cup \{r\}} \otimes$;

与えられた保護グラフを G_0，書換え規則の集合を $\{C_1, \cdots, C_k\}$ としよう．G_0 から書換え規則を適用していくことにより G_i が得られ $x \xrightarrow[G_i]{r} y$ かつ $\neg(x \xrightarrow[G_0]{r} y)$ の時，G_0 から権限 r が漏れたといい，保護システムが r と G_0 に対して安全でないと言う．そうでない場合，安全であると言い safe $(r, G_0, \{C_1, \cdots, C_K\})$ で表す．

ところが，HRU モデルの上では述語 safe は決定

不能であり，また can•share$(\alpha, p, q, G_0, \{C_1, \ldots, C_K\})$ も can•steal$(\alpha, p, q, G_0, \{C_1, \cdots, C_K\})$ も決定不能となって，判定アルゴリズムが存在しない事が知られている．さらに，safe$(r, G_0, \{\bar{C}_1, \cdots, \bar{C}_k\})$ を決定不能にする G_0 と $\{\bar{C}_1, \cdots, \bar{C}_K\}$ が存在し，このモデルが非常に強力な書換え規則を定義することを許している事を示している．

書換え規則のクラスを制限した HRU モデルの議論については文献[8]を参照されたい．

5. まとめ

Take-Grant モデルと HRU モデルを中心に保護システムの形式的取扱いと安全性判定について，主として Snyder[9] に従って述べた．より全般的な解説として文献[13]がある．興味ある形式文法モデル[10]や，安全性を論ずる場合の共謀者[11]の問題に関しては文献を掲げるにとどめる．

参考文献

1) J. B. Dennis & E. C. Van Horn : Programming semantics for multi-programmed computations, C. ACM, Vol. 9 (1966).
2) R. S. Fabry Capability-based addressing, C. ACM, Vol. 17, No. 7 (1966).
3) R. M. Needham & R. D. H. Walker : The Cambridge CAP computer and its protection system, Proc. of 6th ACM Symp. on OS Principles (1977).
4) A. K. Jones, R. J. Lipton & L. Snyder : A linear time algorithm for deciding security, Proc. 17th Symp. on Foundation of Comp. Sci. (1976).
5) R. J. Lipton & L. Snyder : A Linear Time Algorithm for Deciding Subject Security, J. ACM, Vol. 24, No. 3 (1977).
6) L. Snyder : On the Synthesis and Analysis of Protection System, Proc. of 6th ACM Symp. on OS Principles (1977).
7) M. A. Harrison, W. L. Ruzzo & J. D. Ullman : Protection in Operating Systems, C. ACM Vol. 19, No. 8 (1976).
8) M. A. Harrison & W. L. Ruzzo : Monotonic protection Systems, Foundation of Secure Computation, Academic Press (1978).
9) L. Snyder : Formal Models of Capability-Based Protection Systems, IEEE Tr. on Comp., Vol. C-30 No. 3 (1981).
10) T. Budd & R. J. Lipton : On classes of protection systems, Foundation of Secure Computation, Academic Press (1978).
11) L. Snyder : Theft and conspiracy in the Take-Grant Model, Dept. of Comp. Sci., Yale Univ. TR-147 (1978).
12) A. Lockman & N. Minsky : Unidirectional Transport of Rights and Take-Grant Control. IEEE Tr. on S. E., Vol. SE-8 No. 6 (1982).
13) C. E. Landwehr : Formal Models for Computer Security, ACM Computing Surveys, Vol. 13, No. 3 (1981).

(昭和57年12月3日受付)