情報処理学会論文誌 Vol.55 No.9 2081–2091 (Sep. 2014)

素性構造に基づいたアクセス制御モデルの提案

藤田 邦彦[1,a)] 塚田 恭章[1]

受付日 2013年11月18日, 採録日 2014年6月17日

概要：近年の情報システムの複雑化やネットワーク接続の常態化にともない，アクセス制御の必要性と複雑性も増大している．アクセス制御は，情報やコンテンツなどのオブジェクトの利用を許諾あるいは拒否する仕組みであり，諾否の基準がアクセス制御ポリシとして設定される．アクセス制御ポリシの記法は，個々のシステムや言語に特化しており，管理者はそれぞれの記法に習熟していないと，設定ミスによる情報漏洩や不正行為を引き起こす可能性がある．また，アクセス制御の結果は，従来は許諾/拒否の二値を返すものとして定義されてきたが，アクセス要求の特定の部分に限定して許諾する，というような諾否判定結果の多様性に対する需要が高まりつつある．そこで本論文では，自然言語処理の構文解析の分野で長く使われてきた素性構造を用い，記述性に優れたアクセス制御ポリシの記法を持つモデルを提案する．本提案モデルでは，アクセスの諾否判定プロセスは素性構造の単一化の操作によって実現され，許諾/拒否の二値だけでなく一部許諾も返すことができる．また，提案手法を P3P によるアクセスの諾否判定に適用した例を示すことで，本提案モデルの有効性を示す．

キーワード：アクセス制御，素性構造，P3P，ポリシ

An Access Control Model Based on Feature Structure

Kunihiko Fujita[1,a)] Yasuyuki Tsukada[1]

Received: November 18, 2013, Accepted: June 17, 2014

Abstract: New security and privacy enhancing technologies are demanded in the new information and communication environments where a huge number of computers interact with each other in a distributed and ad hoc manner to access various resources. In this paper, we focus on access control because this is the underlying core technology to enforce security and privacy. Access control decides permit or deny according to access control policies. Since notations of policies are specialized in each system, it is difficult to ensure consistency of policies that are stated in different notations. In this paper, we propose a descriptive notation for policies by adopting the concept of feature structures, which has mainly been used for parsing in natural language processing. Our proposed notation is also logically well-founded, which guarantees strict access control decisions, and expressive in that it returns not only a binary value of permit or deny but also various result values through the application of partial order relations of the security risk level. We illustrate the effectiveness of our proposed method using examples from P3P.

Keywords: access control, feature structure, P3P, policy

1. はじめに

アクセス制御は，主体（例：人，プログラムなど）の，オブジェクト（例：情報，コンテンツ，システムなど）に対する，権利（例：読み，書きなど）の実行の許諾または拒否の仕組みであり，諾否の基準がアクセス制御ポリシ（以下「ポリシ」という）として設定される．近年の情報システムの複雑化やネットワーク接続の常態化にともない，アクセス制御の必要性とポリシの複雑性も増大している [1]．

アクセス制御はポリシや主体の属性などに基づいてアクセスの諾否の判定を行う．このため，ポリシの設定ミスは，不適切なアクセスを誤って許諾し，情報漏洩などの事故を

[1] 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所
NTT Communication Science Laboratories, NTT Corporation, Atsugi, Kanagawa 243–0198, Japan
[a)] fujita.kunihiko@lab.ntt.co.jp

招来する可能性がある．また，ポリシは，環境の変化（例：組織の変更，組織の構成員の転出入など）に応じて修正する必要があるが，可読性が低いと，修正に時間を要したり，誤った修正をする可能性が高くなる [2]．さらに，ポリシの記法は，個々のシステムや言語に特化しており，ポリシの管理者はそれぞれの記法に習熟する必要がある．以上のような課題から，記述性に優れたポリシの記法に対する需要が高まりつつある．

また，アクセス制御の結果は，従来は許諾/拒否の二値を返すものとして定義されてきた．これに対し，「許諾，但し必須処理の実行を義務づける」「一部許諾（主体のアクセス要求の特定の部分に限定して許諾）」など，諾否判定結果の多様性に対する需要が高まりつつある [3]．

本論文では，以上の 2 つの重要な要素である「記述性に優れた記法」「多様な諾否判定結果」を満たすため，自然言語処理の構文解析の分野で長く使われてきた素性構造を用いたポリシの記法を提案する．諾否判定の結果は，許諾/拒否の二値だけでなく，一部許諾も返すことができる．また，提案手法の有用性を，P3P への適用例を示すことで証明する．

本論文の構成は次のとおりである．2 章では素性構造について概説する．3 章では素性構造の記法をポリシに適用し，アクセスの諾否判定とアクセスの範囲の導出のプロセスを示す．4 章では，web サイトの個人情報収集規定の標準である P3P に，提案手法を適用することにより，その有用性を示す．5 章では，本提案モデルの記述性と，本提案モデルによる他のアクセス制御モデルの実装可能性について議論する．6 章では関連研究との比較を行う．7 章では結論ならびに今後の課題について述べる．

2. 素性構造

素性構造は，自然言語処理における構文解析の一手法である単一化文法で用いられるデータ構造である [4]．素性構造を用いることにより，膨大な文法規則を見通しよく記述でき，単一化と呼ばれる操作を施すことで構文解析を行うことが可能になる．素性構造は，数（単複）や人称，時制などを表す素性と，その値（素性値）の対の集合と定義される．通常は以下のような属性・属性値のペアの集合で表記される．

$$\begin{bmatrix} \mathit{person}: & \text{third} \\ \mathit{number}: & \text{plural} \end{bmatrix} \tag{1}$$

この例は，英語の三人称複数の名詞（例：“they”）の属性を表現しており，左の列にあるものが素性ラベルであり，その右にあるものが対応する素性値である．また，1 行目は，素性ラベル *person*（人称）の値が third（三人称）であることを示している．同様に 2 行目は，素性ラベル *number*（数）の値が plural（複数）であることを示している．

また，次の例にあるように，素性値はアトミックなシンボルだけでなく，素性構造であってもよい．

$$\begin{bmatrix} \mathit{tense}: & \text{present} \\ \mathit{agreement}: & \begin{bmatrix} \mathit{person}: & \text{third} \\ \mathit{number}: & \text{plural} \end{bmatrix} \end{bmatrix}$$

この例の 1 行目は，素性 *tense*（時制）の値が present（現在）であることを示している．2 行目は，素性 *agreement*（呼応）の値が，式 (1) の素性構造であることを示している．また，以下のようにアトミックなシンボルを要素とする集合を素性値とすることも可能である．

$$\begin{bmatrix} \mathit{person}: & \{\text{first}, \text{second}, \text{third}\} \\ \mathit{number}: & \{\text{singular}, \text{plural}\} \\ \mathit{tense}: & \text{past} \end{bmatrix}$$

この例は，英語の過去形の動詞（例：“walked”）の属性を表現している．1 行目は，人称が一人称・二人称・三人称のいずれも取り得ることを示している．2 行目は，数が単数・複数のいずれも取り得ることを示している．3 行目は，時制が過去であることを示している．英語の三人称単数現在の動詞（例：“walks”）がの属性を素性構造で表現すると以下のようになる．

$$\begin{bmatrix} \mathit{person}: & \text{third} \\ \mathit{number}: & \text{singular} \\ \mathit{tense}: & \text{present} \end{bmatrix}$$

文の構成要素どうしを集めてさらに大きな構成要素を作る際に用いられる演算が，素性構造の単一化である．単一化の演算子を $\otimes$ とし，以下に例をあげる．

$$\begin{bmatrix} \mathit{person}: & \text{third} \\ \mathit{number}: & \text{plural} \end{bmatrix} \otimes \begin{bmatrix} \mathit{person}: & \{\text{first}, \text{second}, \text{third}\} \\ \mathit{number}: & \{\text{singular}, \text{plural}\} \\ \mathit{tense}: & \text{past} \end{bmatrix} = \begin{bmatrix} \mathit{person}: & \text{third} \\ \mathit{number}: & \text{plural} \\ \mathit{tense}: & \text{past} \end{bmatrix}$$

この例が示すとおり，単一化の対象となる 2 つの素性構造の双方に同じ素性ラベルが存在する場合は，対応する素性値どうしの積を結果とし，片方にしか存在しない素性ラベルについてはそのまま結果とする．この例は，単一化が成功したことにより，文 “They walked” が文法的に正しいことを示すとともに，単一化の結果が当該文の属性を素性構造で表現したものとなっている．

また，単一化が失敗する例を以下に示す．*FAIL* は単一化の失敗を表す．

$$\begin{bmatrix} person: & \text{third} \\ number: & \text{plural} \end{bmatrix}$$
$$\otimes \begin{bmatrix} person: & \text{third} \\ number: & \text{singular} \\ tense: & \text{present} \end{bmatrix}$$
$$= FAIL$$

この例では，単一化の対象となる 2 つの素性構造の双方に存在する素性ラベル *number* について，対応する素性値どうし（plural と singular）の積が存在しないため，単一化に失敗する例である．この例は，単一化が失敗したことにより，文 “They walks” が文法的に誤っていることを示す．

3. 素性構造のポリシへの適用

本章では，まず最初に，一般的なアクセス制御の手順について説明する．次に，ポリシとアクセス要求の素性構造による表現について説明する．次に，アクセス可能なセキュリティリスクレベルの導出について定義する．次に，ポリシとアクセス要求の素性構造による表現の体系 FSP（Feature Structures for Policy）の構文規則と単一化を定義する．最後に，単一化によるアクセスの諾否判定方法について説明する．

3.1 アクセス制御の手順

一般的なアクセス制御の手順を図 **1** に従い以下に示す．

(1) オブジェクトに関する権利の保有者（権利保有者）は，当該オブジェクトについてアクセスを許諾する条件をアクセス制御ポリシ $\mathbf{P}$ として記述する．

(2) オブジェクトの利用者たる主体は，オブジェクトへのアクセス要求 $\mathbf{Q}$ をリファレンスモニタに送る．$\mathbf{Q}$ は，アクセスの諾否を判定するために必要な事実や主体の属性などの情報を含む．

(3) リファレンスモニタは，$\mathbf{Q}$ が $\mathbf{P}$ を満たすか否か，すなわちアクセスを許諾するか拒否するか，を判定する．

(4) リファレンスモニタは，(3) においてアクセスを許諾するという判定が出た場合のみ，主体によるオブジェクトへのアクセス実行を媒介する．

3.2 ポリシとアクセス要求の素性構造による表現

ポリシとは，オブジェクトを操作（例：読み，書きなど）する権利を，当該権利を保有する者が他者に対して許諾する内容を記述したものである．これを整理すると，ポリシには以下の種類の要素が含まれる．

- 権利保有者（author）
- 主体（subject）
- オブジェクト（object）
- 権利（right）
- 条件（condition）

リファレンスモニタがアクセスの諾否を判定するためには，アクセス要求とポリシの各要素を突合する必要がある．このため，アクセス要求に含まれる要素の種類は，上述のポリシのそれと等しい必要がある．以上のことを素性構造で表すこととする．ポリシを素性構造で表した $\mathbf{P}^{\mathsf{FSP}}$ と，アクセス要求を素性構造で表した $\mathbf{Q}^{\mathsf{FSP}}$ の内容は，以下のとおりである．

$$\begin{bmatrix} auth: & \mathrm{v}^{auth} \\ subj: & \mathrm{v}^{subj} \\ obj: & \mathrm{v}^{obj} \\ right: & \mathrm{v}^{right} \\ cond: & \mathrm{v}^{cond} \end{bmatrix}$$

auth は権利保有者，*subj* は主体，*obj* はオブジェクト，*right* は権利，*cond* は条件の素性を表す．v^{auth}，v^{subj}，v^{obj}，v^{right}，v^{cond} はそれぞれの素性に対応する値で，アトミックな値か集合か素性構造のいずれかであるとする．

3.3 アクセス可能なセキュリティリスクレベルの導出

本節では，まず最初にセキュリティリスクの高低を示すセキュリティリスクレベルについて定義する．次に，セキュリティリスクレベルを勘案した場合のアクセスの可能性について定義する．

セキュリティリスクレベルとは，主体やオブジェクト，アクセスの目的などポリシを構成するドメインに含まれる要素間でのセキュリティリスクの高低を示す基準のことである．

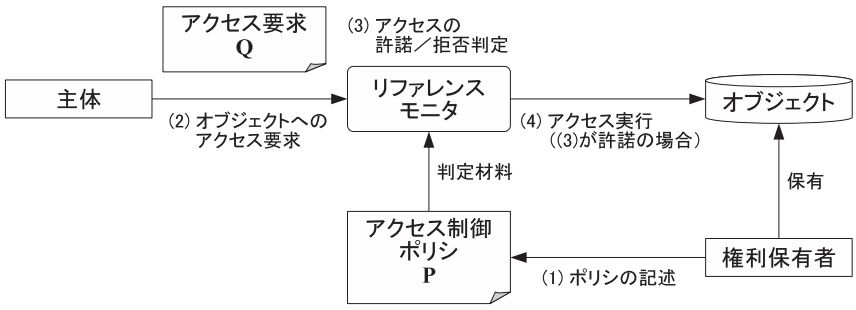

図 1 アクセス制御の手順（文献 [5] 図 4.1 を参考にして作成）

Fig. 1 A simple access control mechanism (We refer to Fig. 4.1 of Ref. [5] to depict it).

定義 3.1（セキュリティリスクレベル） s_i, s_j をポリシを構成するドメイン D の要素とする．s_j が s_i よりもセキュリティリスクの高い概念を示すとき，$s_i \leq_D s_j$ と表すこととする． □

例として，オブジェクトにアクセスを許諾する期間をドメイン T とし，これに属する要素として「無期限」「必要最低限の期間」「法律の定める期間」「業務慣行に従った期間」の 4 つを仮定する．オブジェクトの権利保有者の観点では，アクセスを許諾する期間が長ければ長いほど，情報漏洩などセキュリティを侵害される事案が発生する可能性が高まる．このセキュリティ侵害の可能性の高低のことを，セキュリティリスクレベルと定義する．これを前提とすると，ドメイン T のセキュリティリスクレベルの関係 $\leq_T$ は，「無期限」が最高，「必要最低限の期間」が最低となり，「法律の定める期間」と「業務慣行に従った期間」の両者は，このままでは（具体的な期間が明示されない限り）相互に高低の比較はできないが，「無期限」よりは低く「必要最低限の期間」よりは高い．したがって，以下の関係が成り立つ．

$$
\begin{aligned}
\text{無期限} &\leq_T \text{無期限}\\
\text{必要最低限の期間} &\leq_T \text{必要最低限の期間}\\
\text{法律の定める期間} &\leq_T \text{法律の定める期間}\\
\text{業務慣行に従った期間} &\leq_T \text{業務慣行に従った期間}\\
\text{法律の定める期間} &\leq_T \text{無期限}\\
\text{業務慣行に従った期間} &\leq_T \text{無期限}\\
\text{必要最低限の期間} &\leq_T \text{法律の定める期間}\\
\text{必要最低限の期間} &\leq_T \text{業務慣行に従った期間}\\
\text{必要最低限の期間} &\leq_T \text{無期限}
\end{aligned}
$$

この例の場合，ドメイン T は関係 $\leq_T$ について反射的で反対称的かつ推移的であるため，半順序集合である．さらに，任意の 2 要素について最小上界と最大下界を持つためラティスでもある．

以下，本提案モデルにおいては，ポリシを構成するドメイン $(D, \leq_D)$ はラティスであると仮定する．3.2 節に示すポリシを構成するドメインのうち，*auth*（権利保有者）や *subj*（主体），*obj*（オブジェクト）など，ラティスを形成しないであろうものについては，この仮定に対応するため，該当するドメイン内で，上界 *ANY* と下界 *NULL* を付加することにより，ラティス化を行う（**図 2**）．

これは，LBAC（Lattice-Based Access Control）モデル [6], [7] に基礎をおくものである．$\lambda(s)$ と $\lambda(o)$ を，それぞれ主体 s とオブジェクト o のセキュリティレベルを示すものとし，セキュリティレベルはラティスを形成するものとする．LBAC モデルでは，$\lambda(s) \geq \lambda(o)$ のとき，オペレーションを許諾する．この考え方に基づくと，セキュリティリスクレベルに関しては，主体のアクセス要求は，たかだかオブジェクトのポリシのセキュリティリスクレベルにおいて，許諾される．上例を用いて説明すると，主体が「無期限」のアクセス要求をしたのに対し，オブジェクトは「法律に定める期間」でのアクセスを許諾するとポリシに記述されていた場合，主体は「法律に定める期間」でのアクセスが許諾される，ということである．このことを以下に定義する．

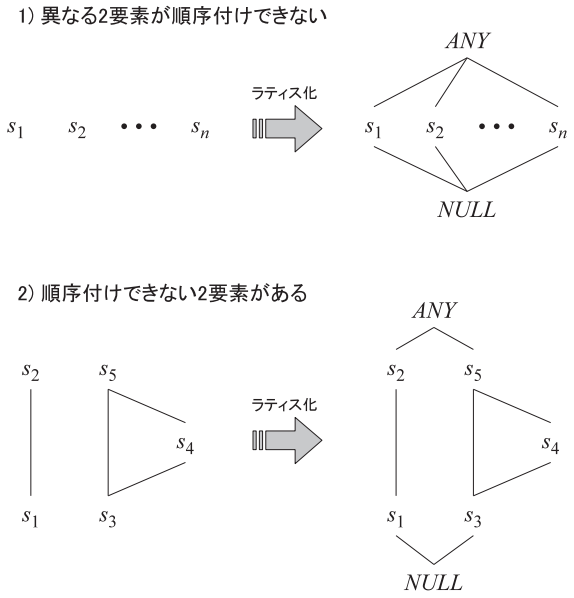

図 2 ラティス化のイメージ
Fig. 2 Illustration of converting elements into latice.

定義 3.2（アクセス可能なセキュリティリスクレベル） A, B をポリシを構成するドメイン D の部分集合とし，A が主体，B がオブジェクトに対応することとする．このとき，主体がオブジェクトにアクセス可能なセキュリティリスクレベルは $\nabla(A,B) = \{c \mid \exists a \in A, \exists b \in B\, [c \leq_D a \wedge b]\} \setminus \{NULL\}$ によって与えられる．ただし，上式の $\wedge$ はラティスにおける交わり（下限）を求める演算子である．また，$\nabla(A,B) = \emptyset$ のときは，アクセス可能なセキュリティリスクレベルは存在しないものとする． □

3.4 FSP の構文規則と単一化

ポリシとアクセス要求を素性構造で表したとき，主体のオブジェクトに対するアクセスの許諾/拒否の判定は，素性構造の単一化の成功/失敗にそれぞれ対応する．本節では，ポリシとアクセス要求の素性構造の構文規則と単一化を定義する．

3.4.1 FSP の構文規則

FSP の構文規則は，**図 3** に示すとおりである．α は FSP を，*NIL* は何も情報がないことを表す．L は素性ラベルの有限集合である．l : v は素性ラベル l について素性値 v を持つことを表す．

a はアトミックな素性値を表す．FSP では，素性値がア

$$
\begin{array}{lcll}
\alpha & ::= & \begin{bmatrix} l_1: & \mathrm{v}_1 \\ \vdots & \vdots \\ l_n: & \mathrm{v}_n \end{bmatrix} & \text{where } l_i \in L \\
\mathrm{v} & ::= & \mathit{NIL} & \\
 & | & \{a_1, \cdots, a_m\} & \text{where } \exists k \forall i\, [a_i \in D_k] \\
 & | & \alpha &
\end{array}
$$

図 3 FSP の構文規則
Fig. 3 Syntax for FSP.

トミックの場合は単集合で表すこととし，略記として $\{a\}$ を a と表すことを許す．また，$(D_1, \leq_{D_1}), \cdots, (D_l, \leq_{D_l})$ はドメインの列であり，任意の異なる i, j に対し D_i, D_j は互いに素である．

3.4.2 FSP の正規化

単一化の前処理としての正規化について説明する．ここでの正規化とは，2 つの FSP に属する素性構造について，一方に存在し他方に存在しない素性ラベルと素性値のペアがあった場合，存在しない側に当該素性ラベルと素性値 *NIL* のペアを追加し整列する処理である．

具体的な FSP の正規化の手順について以下に示す．$\alpha, \beta \in \mathsf{FSP}$ を単一化しようとしている．関数 $label(\alpha)$ は α に含まれる素性ラベルを集合として返す．関数 $add(\alpha,\, l : \mathrm{v})$ は，α に素性ラベル l と素性値 v のペアを追加した結果を返す．このとき，α の β に対する正規化は，以下の操作により行われ，結果が α' に格納される．

for each $l_x \in label(\beta) \setminus label(\alpha)$ **do**
  $\alpha' \leftarrow add(\alpha,\, l_x : \mathit{NIL})$

同様に，β の α に対する正規化は，以下の操作により行われ，結果が β' に格納される．

for each $l_y \in label(\alpha) \setminus label(\beta)$ **do**
  $\beta' \leftarrow add(\beta,\, l_y : \mathit{NIL})$

最後に，α' と β' について，素性ラベルと素性値のペアの順番を整列し揃えることとする．

3.4.3 FSP の単一化

α，$\beta \in \mathsf{FSP}$ とし，$\otimes$ を素性構造の単一化の二項演算子とし，$\bar{\otimes}$ を素性値の単一化の二項演算子とする．また，単一化の対象となる FSP に属する素性構造は，前処理として正規化を行うものとする．このとき，$\alpha \otimes \beta$ を以下のように定義する．

$$
\alpha \otimes \beta = \begin{bmatrix} l_1: & \mathrm{v}_1 \\ \vdots & \vdots \\ l_n: & \mathrm{v}_n \end{bmatrix} \otimes \begin{bmatrix} l_1: & \mathrm{v}'_1 \\ \vdots & \vdots \\ l_n: & \mathrm{v}'_n \end{bmatrix}
$$

$$
= \begin{bmatrix} l_1: & \mathrm{v}_1 \mathbin{\bar{\otimes}} \mathrm{v}'_1 \\ \vdots & \vdots \\ l_n: & \mathrm{v}_n \mathbin{\bar{\otimes}} \mathrm{v}'_n \end{bmatrix}
$$

$$
\mathrm{v_i} \mathbin{\bar{\otimes}} \mathit{NIL} = \mathrm{v_i}
$$

$$
\mathit{NIL} \mathbin{\bar{\otimes}} \mathrm{v_i} = \mathrm{v_i}
$$

$$
\mathrm{v}_i \mathbin{\bar{\otimes}} \mathrm{v}'_i = \begin{cases} \mathrm{v}_i \otimes \mathrm{v}'_i & \text{if } \mathrm{v}_i, \mathrm{v}'_i \in \mathsf{FSP} \\ \nabla(\mathrm{v}_i, \mathrm{v}'_i) & \text{if } \exists k\, [\mathrm{v}_i, \mathrm{v}'_i \subseteq D_k] \\ \emptyset & \text{otherwise} \end{cases}
$$

$$
\begin{bmatrix} l_1: & \mathrm{v}_1 \\ \vdots & \vdots \\ l_i: & \emptyset \\ \vdots & \vdots \\ l_n: & \mathrm{v}_n \end{bmatrix} = \emptyset
$$

3.5 アクセスの諾否判定方法

アクセスの諾否判定とアクセス許諾範囲の導出について説明する．ポリシを FSP で表現したものを $\mathbf{P}^{\mathsf{FSP}}$，アクセス要求を FSP で表現したものを $\mathbf{Q}^{\mathsf{FSP}}$ とする．$\mathbf{P}^{\mathsf{FSP}} \otimes \mathbf{Q}^{\mathsf{FSP}} = \emptyset$ が導出されたとき，$\mathbf{P}^{\mathsf{FSP}}$ と $\mathbf{Q}^{\mathsf{FSP}}$ は単一化の過程で矛盾が発生したため，単一化できないことを表す．これは，アクセス要求 $\mathbf{Q}$ がポリシ $\mathbf{P}$ のあげる条件を満たさず，アクセスが許諾されないことを示す．$\mathbf{P}^{\mathsf{FSP}} \otimes \mathbf{Q}^{\mathsf{FSP}} = \mathbf{R}^{\mathsf{FSP}}$ のとき，単一化は成功し，その結果が $\mathbf{R}^{\mathsf{FSP}}$ であることを表す．これは，$\mathbf{R}^{\mathsf{FSP}}$ に記述されている条件でのアクセスが許諾されたことを示す．

4. FSP の P3P への適用

本章では，ポリシに素性構造を適用することの有用性を示すため，FSP を P3P（Platform for Privacy Preferences）に適用した例を示す．まず最初に P3P について説明し，次に，P3P ポリシを素性構造で表現したうえで，アクセスの諾否判定やアクセス許諾範囲の導出が行えることを，具体例を用いて示す．

4.1 P3P

P3P とは，web サイトにおける個人情報の取扱いに関する規定を，標準化された XML フォーマットで発行することを可能とする技術仕様である [8]．P3P ポリシで最も重要な要素として，“Statement” エレメントがあげられる [9]．このエレメントは，以下のサブエレメントを含む．

- Data-Group (D)：当該 web サイトで収集しようとしている個人情報の集合．たとえば，「名前」，「電話番号」，「性別」などが要素 (d) となる．
- Purpose (P)：当該 web サイトで個人情報を収集する目的の集合．以下の項目のうちの 1 つ以上を含まなければならない．末尾のアルファベット 3 文字は略称で

ある．

<current/> データが提供された活動の遂行とサポート CUR

<admin/> web サイトとシステムの管理 ADM

<develop/> 調査と開発 DEV

<tailoring/> その場限りの web サイトの編成 TAI

<pseudo-analysis/> ペンネーム分析 PSA

<pseudo-decision/> ペンネーム決定 PSD

<individual-analysis/> 個人分析 IVA

<individual-decision/> 個人決定 IVD

<contact/> サービスまたは商品のマーケティングのためにサイト訪問者と連絡をとる CON

<historical/> 過去の出来事を保存 HIS

<telemarketing/> 電話を通じてのサービスと商品のマーケティングのために訪問者に連絡 TEL

<other-purpose> string </other-purpose> その他の利用 OPT

- Recipient (R)：当該 web サイトで収集する個人情報の受領者の集合．以下の項目のうちの 1 つ以上を含まなければならない．

  <ours> 当組織および/または当組織の業務委託先として業務を行っている法人または当社が業務委託先として業務をしている法人 OUR

  <delivery> 当組織とは異なるプラクティスに従う可能性のある配送サービス DEL

  <same> 当組織のプラクティスに従う合法組織 SAM

  <other-recipient> 当組織とは異なるプラクティスに従う合法組織 OTR

  <unrelated> 当組織と無関係な第三者 UNR

  <public> 公のフォーラム PUB

- Retention (T)：当該 web サイトで収集する個人情報の保持期間．以下の項目のうちの 1 つを含まなければならない．

  <no-retention/> 情報は，オンラインでの 1 回のインタラクションにおいてその情報を利用するのに必要最低限の時間以上は保有されない NOR

  <stated-purpose/> 情報は言明された目的をかなえるために保有される STP

  <legal-requirement/> 情報は，言明された目的をかなえるために保有されるが，その保有期間は，法律上の要求または責務によってそれよりも長い場合がある LEG

  <business-practices/> 情報は，サービス提供者の言明した業務慣行に従って保有される BUS

  <indefinitely/> 無期限 IND

P3P のポリシ例を**表 1** にあげる．

P，R，T の各ドメインの項目を，セキュリティリスクレベルのラティスで表現することについて検討する．P は個人情報を収集する目的について分類した項目であり，セキュリティリスクレベルを基準に順序付けすることは困難である [10]．このため，3.3 節で定義した上界 ANY と下界 $NULL$ を付加する方法でラティス化を行う（**図 4**）．一方，R については，収集した個人情報の受領者（組織）について分類した項目であり，受領者は（自組織だけ/他組織も含む/無制限）という区別と，個人情報の取扱いは自組織のポリシに（従う/従わない）という区別で各要素を分ける下表のように整理することができる．

	自/他/無制限	自ポリシに従う？
OUR	自	○
DEL	他	×
SAM	他	○
OTR	他	×
UNR	他	×
PUB	無制限	×

たとえば，要素 SAM は，他組織に個人情報を提供するが管理は自組織のポリシに従うため，同じく他組織に個人情報

表 1 P3P ステートメントの例
Table 1 Example P3P statement snippets.

web サイト A のアクセス要求

```
<STATEMENT>
    <DATA-GROUP>
        <DATA REF="#USER.MAIL-ADDRESS.GIVEN"/>
    </DATA-GROUP>
    <PURPOSE>
        <TAILORING/> <CONTACT/>
    </PURPOSE>
    <RECIPIENT>
        <UNRELATED/> <SAME/>
    </RECIPIENT>
    <RETENTION>
        <BUSINESS-PRACTICES/>
    </RETENTION>
</STATEMENT>
```

web サイト B のアクセス要求

```
<STATEMENT>
    <DATA-GROUP>
        <DATA REF="#USER.MAIL-ADDRESS.GIVEN"/>
    </DATA-GROUP>
    <PURPOSE>
        <TAILORING/> <PSEUDO-ANALYSIS/>
    </PURPOSE>
    <RECIPIENT>
        <DELIVERY/>
    </RECIPIENT>
    <RETENTION>
        <NO-RETENTION/>
    </RETENTION>
</STATEMENT>
```

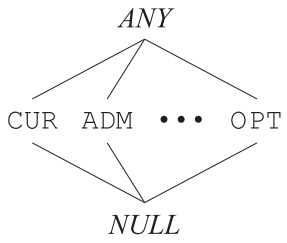

図 4 $(P, \leq_P)$ のハッセ図
Fig. 4 Hasse Diagram of $(P, \leq_P)$.

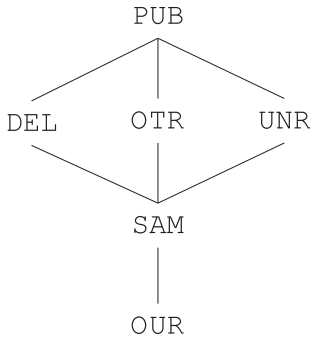

図 5 $(R, \leq_R)$ のハッセ図
Fig. 5 Hasse Diagram of $(R, \leq_R)$.

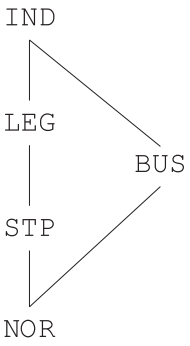

図 6 $(T, \leq_T)$ のハッセ図
Fig. 6 Hasse Diagram of $(T, \leq_T)$.

を提供するが管理は自組織のポリシに従わない DEL，OTR，UNR の各要素よりも，セキュリティリスクレベルが低いといえる．これを整理して順序構造を図示したのが，**図 5** である．また，T は，収集した個人情報を保持する期間について分類した項目である．NOR，LEG，BUS，IND の各要素間のセキュリティリスクレベルについての関係は，3.3 節にあげた例のとおりである．目的達成のために必要な期間保持するという要素 STP を加えて順序構造を図示すると，**図 6** のようになる．

4.2 提案モデルの P3P ポリシへの適用

本節では，P3P ポリシを素性構造で表現し，アクセス制御の判定を行う方法を定義する．P3P では，web サイト側が訪問者の個人情報の利用許諾を得ようとするので，訪問者（visitor）が権利保有者で，オブジェクトが訪問者の個人情報（private_info）となり，web サイト（web_site）が主体である．権利は個人情報の利用（use）である．また，条件として，目的（P），受領者（R），保持期間（T）があげられる．権利保有者（=訪問者，$\mathbf{P}$）のポリシと主体（=web サイト，$\mathbf{Q}$）のアクセス要求を素性構造で表現すると，以下のようになる

$$\mathbf{P}^{\mathsf{FSP}} = \begin{bmatrix} \mathit{auth}: & \text{visitor} \\ \mathit{subj}: & \mathit{NIL} \\ \mathit{obj}: & \begin{bmatrix} d_1: & \text{private\_info}_1 \\ \vdots & \vdots \\ d_n: & \text{private\_info}_n \end{bmatrix} \\ \mathit{right}: & \text{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \mathrm{v}^P \\ R: & \mathrm{v}^R \\ T: & \mathrm{v}^T \end{bmatrix} \end{bmatrix}$$

$$\mathbf{Q}^{\mathsf{FSP}} = \begin{bmatrix} \mathit{auth}: & \mathit{NIL} \\ \mathit{subj}: & \text{website} \\ \mathit{obj}: & \begin{bmatrix} d_1: & \text{private\_info}_1 \\ \vdots & \vdots \\ d_n: & \text{private\_info}_n \end{bmatrix} \\ \mathit{right}: & \text{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \mathrm{v}^P \\ R: & \mathrm{v}^R \\ T: & \mathrm{v}^T \end{bmatrix} \end{bmatrix}$$

$\mathbf{P}^{\mathsf{FSP}}$ の素性ラベル subj の素性値が NIL なのは，このポリシをすべての web サイトに適用することを意味する．また，$\mathbf{Q}^{\mathsf{FSP}}$ の素性ラベル auth の素性値が NIL がなのは，このアクセス要求をすべての訪問者に適用することを意味する．

表 1 にあげた web サイト A，B のアクセス要求の，素性構造による表現例を以下に示す．

$$\mathbf{Q}^{\mathsf{FSP}}_{\text{website}_\text{A}} = \begin{bmatrix} \mathit{auth}: & \mathit{NIL} \\ \mathit{subj}: & \text{website}_\text{A} \\ \mathit{obj}: & \begin{bmatrix} d_1: & \mathit{NIL} \end{bmatrix} \\ \mathit{right}: & \text{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \{\texttt{TAI}, \texttt{CON}\} \\ R: & \{\texttt{UNR}, \texttt{SAM}\} \\ T: & \texttt{BUS} \end{bmatrix} \end{bmatrix}$$

$$\mathbf{Q}^{\mathsf{FSP}}_{\text{website}_\text{B}} = \begin{bmatrix} \mathit{auth}: & \mathit{NIL} \\ \mathit{subj}: & \text{website}_\text{B} \\ \mathit{obj}: & \begin{bmatrix} d_1: & \mathit{NIL} \end{bmatrix} \\ \mathit{right}: & \text{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \{\texttt{TAI}, \texttt{PSA}\} \\ R: & \texttt{DEL} \\ T: & \texttt{NOR} \end{bmatrix} \end{bmatrix}$$

ここで，訪問者 Alice のポリシの，素性構造による表現例を以下のように仮定する．

$$\mathbf{P}^{\mathsf{FSP}}_{\mathrm{Alice}} = \begin{bmatrix} \mathit{auth}: & \mathrm{Alice} \\ \mathit{subj}: & \mathit{NIL} \\ \mathit{obj}: & \left[d_1: \ \mathrm{alice@foo.bar.jp} \right] \\ \mathit{right}: & \mathrm{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \{\mathtt{CON}, \mathtt{TEL}\} \\ R: & \{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\} \\ T: & \mathtt{LEG} \end{bmatrix} \end{bmatrix}$$

以上の仮定の下で，まず，$\mathbf{P}^{\mathsf{FSP}}_{\mathrm{Alice}} \otimes \mathbf{Q}^{\mathsf{FSP}}_{\mathrm{website_A}}$ について，各々の素性ラベルと素性値のペアの単一化の過程を示す．

$\mathit{auth}: \mathrm{Alice} \ \bar{\otimes} \ \mathit{NIL} = \mathit{auth}: \mathrm{Alice}$

$\mathit{subj}: \mathit{NIL} \ \bar{\otimes} \ \mathrm{website_A} = \mathit{subj}: \mathrm{website_A}$

$$\begin{aligned} &\mathit{obj}: \left[d_1: \mathrm{alice@foo.bar.jp} \right] \bar{\otimes} \left[d_1: \mathit{NIL} \right] \\ &= \mathit{obj}: \left[d_1: \mathrm{alice@foo.bar.jp} \ \bar{\otimes} \ \mathit{NIL} \right] \\ &= \mathit{obj}: \left[d_1: \mathrm{alice@foo.bar.jp} \right] \end{aligned}$$

$\mathit{right}: \mathrm{use} \ \bar{\otimes} \ \mathrm{use} = \mathit{right}: \mathrm{use}$

$$\begin{aligned} &\mathit{cond}: \begin{bmatrix} P: & \{\mathtt{CON}, \mathtt{TEL}\} \\ R: & \{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\} \\ T: & \mathtt{LEG} \end{bmatrix} \otimes \begin{bmatrix} P: & \{\mathtt{TAI}, \mathtt{CON}\} \\ R: & \{\mathtt{UNR}, \mathtt{SAM}\} \\ T: & \mathtt{BUS} \end{bmatrix} \\ &= \mathit{cond}: \begin{bmatrix} P: & \{\mathtt{CON}, \mathtt{TEL}\} \ \bar{\otimes} \ \{\mathtt{TAI}, \mathtt{CON}\} \\ R: & \{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\} \ \bar{\otimes} \ \{\mathtt{UNR}, \mathtt{SAM}\} \\ T: & \mathtt{LEG} \ \bar{\otimes} \ \mathtt{BUS} \end{bmatrix} \end{aligned}$$

$$\begin{aligned} &P: \{\mathtt{CON}, \mathtt{TEL}\} \ \bar{\otimes} \ \{\mathtt{TAI}, \mathtt{CON}\} \\ &= P: \nabla(\{\mathtt{CON}, \mathtt{TEL}\}, \{\mathtt{TAI}, \mathtt{CON}\}) \\ &= P: \mathtt{CON} \end{aligned}$$

$$\begin{aligned} &R: \{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\} \ \bar{\otimes} \ \{\mathtt{UNR}, \mathtt{SAM}\} \\ &= R: \nabla(\{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\}, \{\mathtt{UNR}, \mathtt{SAM}\}) \\ &= R: \{\mathtt{UNR}, \mathtt{SAM}, \mathtt{OUR}\} \end{aligned}$$

$$\begin{aligned} &T: \mathtt{LEG} \ \bar{\otimes} \ \mathtt{BUS} \\ &= T: \nabla(\mathtt{LEG}, \mathtt{BUS}) \\ &= T: \mathtt{NOR} \end{aligned}$$

したがって，以下の結果が導出される．

$$\mathbf{P}^{\mathsf{FSP}}_{\mathrm{Alice}} \otimes \mathbf{Q}^{\mathsf{FSP}}_{\mathrm{website_A}} = \begin{bmatrix} \mathit{auth}: & \mathrm{Alice} \\ \mathit{subj}: & \mathrm{website_A} \\ \mathit{obj}: & \left[d_1: \ \mathrm{alice@foo.bar.jp} \right] \\ \mathit{right}: & \mathrm{use} \\ \mathit{cond}: & \begin{bmatrix} P: & \mathtt{CON} \\ R: & \{\mathtt{UNR}, \mathtt{SAM}, \mathtt{OUR}\} \\ T: & \mathtt{NOR} \end{bmatrix} \end{bmatrix}$$

また，$\mathbf{P}^{\mathsf{FSP}}_{\mathrm{Alice}} \otimes \mathbf{Q}^{\mathsf{FSP}}_{\mathrm{website_B}}$ については，*cond* 内の素性ラベルと素性値のペアの単一化の過程のみを示す．

$$\begin{aligned} &P: \{\mathtt{CON}, \mathtt{TEL}\} \ \bar{\otimes} \ \{\mathtt{TAI}, \mathtt{PSA}\} \\ &= P: \nabla(\{\mathtt{CON}, \mathtt{TEL}\}, \{\mathtt{TAI}, \mathtt{PSA}\}) \\ &= P: \emptyset \end{aligned}$$

$$\begin{aligned} &R: \{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\} \ \bar{\otimes} \ \mathtt{DEL} \\ &= R: \nabla(\{\mathtt{OTR}, \mathtt{UNR}, \mathtt{SAM}\}, \mathtt{DEL}) \\ &= R: \{\mathtt{SAM}, \mathtt{OUR}\} \end{aligned}$$

$$\begin{aligned} &T: \mathtt{LEG} \ \bar{\otimes} \ \mathtt{NOR} \\ &= T: \nabla(\mathtt{LEG}, \mathtt{NOR}) \\ &= T: \mathtt{NOR} \end{aligned}$$

FSP の単一化の定義により，$P: \emptyset$ のため $\mathit{cond}: \emptyset$ である．したがって，以下の結果が導出される．

$$\mathbf{P}^{\mathsf{FSP}}_{\mathrm{Alice}} \otimes \mathbf{Q}^{\mathsf{FSP}}_{\mathrm{website_B}} = \emptyset$$

5. 議論

5.1 記述性の比較

本提案モデルの記述性について議論する．4.2 節では，P3P による記述（表 1）を FSP で記述した例を示した．本節では表 1 の web サイト A の例の，XACML [15] での記述を試みる（**表 2**）[*1]．

XACML ではアクセスを要求する側は，主体やオブジェクト，アクションなどは設定できるが，P3P に “Statement” エレメントとして組み込まれている，アクセスする目的や情報の受領者，情報の保持期間などの条件を設定することはできない．このため，P3P の例（表 1）を XACML で完全に表現することはできず，表 2 は主体・オブジェクト・アクションのみを記述している．

P3P や XACML は XML で実装されるコンピュータ言語であるため，人間にとっては複雑で期待するアクセス諾否の条件を記述するのは困難である．P3P と XACML

[*1] 簡単のため，ここで示した例は，AttributeID など様々な設定を省略している．

表 2 XACML のアクセス要求の記述例
Table 2 Example XACML request snippets.

web サイト A のアクセス要求

```
<Request>
    <Attributes Category="access-subject">
        <Attribute>
            <AttributeValue>websiteA</AttributeValue>
        </Attribute>
    </Attributes>
    <Attributes Category="resource">
        <Attribute>
            <AnyResource/>
        </Attribute>
    </Attributes>
    <Attributes Category="action">
        <Attribute>
            <AttributeValue>use</AttributeValue>
        </Attribute>
    </Attributes>
</Request>
```

は，元々記述性を配慮されたものではなく，何らかのインタフェースを介してポリシを記述することが前提となっている．一方，本提案モデルは素性構造を用いてアクセス諾否の条件をコンパクトに記述できるように設計している．このため，本提案モデルがインタフェースとなり，P3P や XACML で記述されたポリシを生成するような実装も考えられる．

5.2 他のアクセス制御モデルの実装

本提案モデルによる様々なアクセス制御モデルの実装可能性について議論する．本提案モデルでは，Bell-LaPadula モデル [7] における no-read-up（$\lambda(s) \geqq \lambda(o)$ ならば，s は o に対する Read アクセスを許諾される）や no-write-down（$\lambda(s) \leqq \lambda(o)$ ならば，s は o に対する Write アクセスを許諾される）といった，オペレーションごとにアクセスの諾否が決定されるという性質は組み込まれていない．しかし，以下に示す方法により同様の性質を実現することが可能である．

たとえば，Top Secret, Secret, Confidential, Unclassified（以下ではそれぞれ TS，S，C，U と略記する）の 4 つのセキュリティクラスを仮定する．セキュリティの強度を降順に TS，S，C，U とすると，このセキュリティクラスは線形束を形成する．このとき，no-read-up と no-write-down の両性質を兼ね備えたアクセス制御モデルでは，たとえば，オブジェクトが S，主体が C のとき，Write アクセスは許諾されるが，Read アクセスは許諾されない．また，オブジェクトが U，主体が C のときは，Read アクセスは許諾されるが，Write アクセスは許諾されない．

本提案モデルで no-read-up と no-write-down の両性質を実現する手順は，以下のとおりである．

(1) セキュリティクラスの集合について，その部分集合全体の包含関係を順序とする順序集合を D_{sc} とする．この順序集合 $(D_{sc}, \subseteq)$ はラティスを形成する．

(2) オペレーションごとに，オブジェクトと主体それぞれのポリシーを以下のように設定する．

- **Read**
  オブジェクト
  - 属性 *right* に属性値 read を設定する．
  - 属性 *cond* 内に，属性 *SC* を設定する．
  - 属性 *SC* に，属性値として，Read アクセスを許諾する主体のセキュリティクラスを要素とする集合を設定する．

  主体
  - 属性 *right* に属性値 read を設定する．
  - 属性 *cond* 内に，属性 *SC* を設定する．
  - 属性 *SC* に，属性値として，主体のセキュリティクラスを要素とする集合を設定する．
- **Write**
  オブジェクト
  - 属性 *right* に属性値 write を設定する．
  - 属性 *cond* 内に，属性 *SC* を設定する．
  - 属性 *SC* に，属性値として，Write アクセスを許諾する主体のセキュリティクラスを要素とする集合を設定する．

  主体
  - 属性 *right* に属性値 write を設定する．
  - 属性 *cond* 内に，属性 *SC* を設定する．
  - 属性 *SC* に，属性値として，主体のセキュリティクラスを要素とする集合を設定する．

この手順を，上例を用いて説明する．オブジェクトと主体のセキュリティレベルがそれぞれ S と C の場合，Read アクセス用のポリシ（属性 *right* に read が設定されている）については，オブジェクトのポリシの属性 *SC* に $\{\mathrm{TS}, \mathrm{S}\} \in D_{sc}$ を設定し，主体のポリシの属性 *SC* に $\{\mathrm{C}\} \in D_{sc}$ を設定する．この場合，$\nabla(\{\mathrm{TS}, \mathrm{S}\}, \{\mathrm{C}\}) = \emptyset$ となり，アクセスは拒否される．また，Write アクセス用のポリシ（属性 *right* に write が設定されている）については，オブジェクトのポリシの属性 *SC* に $\{\mathrm{S}, \mathrm{C}, \mathrm{U}\} \in D_{sc}$ を設定し，主体のポリシの属性 *SC* に $\{\mathrm{C}\} \in D_{sc}$ を設定する．この場合，$\nabla(\{\mathrm{S}, \mathrm{C}, \mathrm{U}\}, \{\mathrm{C}\}) = \{\mathrm{C}\}$ となり，アクセスは許諾される．また，オブジェクトと主体のセキュリティレベルがそれぞれ U と C の場合，Read アクセス用のポリシについては，オブジェクトのポリシの属性 *SC* に $\{\mathrm{TS}, \mathrm{S}, \mathrm{C}, \mathrm{U}\} \in D_{sc}$ を設定し，主体のポリシの属性 *SC* に $\{\mathrm{C}\} \in D_{sc}$ を設定する．この場合，$\nabla(\{\mathrm{TS}, \mathrm{S}, \mathrm{C}, \mathrm{U}\}, \{\mathrm{C}\}) = \{\mathrm{C}\}$ となり，アクセスは許諾される．また，Write アクセス用のポリシについては，オブジェクトのポリシの属性 *SC* に $\{\mathrm{U}\} \in D_{sc}$ を

設定し，主体のポリシの属性 SC に $\{\mathrm{C}\} \in D_{sc}$ を設定する．この場合，$\nabla(\{\mathrm{U}\},\{\mathrm{C}\}) = \emptyset$ となり，アクセスは拒否される．

以上のように，本提案モデルは，アクセス制御に関する様々なモデルを追加できる柔軟性を備えている．ライブラリという形で提供すれば，設定ミスなども回避することが可能である．

6. 関連研究

May らはプロセス計算を適用し，複数のポリシ間の類似度の尺度を提案し，P3P を用いた例を示した [11], [12]．これによりたとえば，あるポリシを修正した場合に，意図したとおりに修正されたかどうかを確認するため，修正前後の 2 つのポリシの類似度と相違点を容易に確認できる．本論文で定義したセキュリティリスクレベルというコンセプトは，彼らの提唱したポリシ間の類似度というコンセプトに刺激されたものである．しかし，本研究の提案モデルでのメリットの 1 つであるポリシの可読性については，彼らの研究では対象外となっており，この点で異なる．

P3P などのプライバシーポリシは，web サイトの条件とユーザの条件に合致しない場合は，web サイトの提供するサービスはいっさい使えないという，いわゆる “take it or leave it” アプローチをとるのが通常である．プライバシーを守るために「安全側に倒す」という考え方がその根本にある．Walker らは，“Or Best Offer” と呼ばれる交渉戦略に基づいたプライバシーポリシの交渉プロトコルを提案している [13]．“Or Best Offer” とは，「金額は提示するが，交渉にも応じる」といった意味であり，プライバシーポリシの文脈に敷衍すると，ユーザ側がプライバシーポリシを提示して web サイトの条件と合わなかった場合でも，そこから交渉の落とし所を探るという方法である．換言すると，アクセス制御の結果について，許諾/拒否以外に「主体のアクセス要求の特定の部分に限定して許諾」など多様な結果を返すことができる技術であり，この点で我々の成果と類似している．しかし，彼らの手法では，クライアントとサーバは交渉戦略をポリシとは別に定めなければならず，ポリシの管理者の負荷の増大は免れ得ない．本論文の提案手法では，ポリシ保守に関しての管理者の負荷を低減することが目的の 1 つであり，我々はシンプルなアプローチを採用した．

Ni らは，複雑なプライバシーポリシを記述するため，RBAC（Role-based access control）モデルを拡張した P-RBAC（Privacy-aware RBAC）モデルを提案した [14]．P-RBAC は，アクセス制御を構成する主体・主体の属するロール・オブジェクトなどの階層化に対応できる．この階層化の考え方は，セキュリティリスクレベルに基づき要素が半順序集合を構成するという本論文の仮定に影響を与えている．しかし，本研究の提案モデルでのメリットの 1 つであるポリシの記述性については，彼らの研究では対象外となっており，この点で異なる．

7. おわりに

ポリシの記述性向上のため，我々は素性構造を用いたアクセス制御モデルを提案した．アクセス制御の結果は，許諾と拒否の二値だけでなく，アクセス要求の一部のみ許可，といった多様な値を取りうる．提案手法の有効性を確認するため，P3P に提案モデルを適用した．

今後の課題について述べる．XACML [15] は，ポリシの配下に複数のサブポリシを含むことができ，あるアクセス要求に対する諾否判定は，アクセス要求に対する各サブポリシの諾否判定結果を，特定のアルゴリズムに従い統合した結果となる．XACML のように複数のポリシを統合するようなアルゴリズムを開発して提案モデルに組み込むことにより，フェースブックのプライバシーポリシのような複雑なポリシにも対応できるようにしたい．

また，本提案モデルの，概念学習の一手法であるバージョン空間探索法 [16] における概念の学習プロセスへの適用可能性についても追求したい．バージョン空間探索法においては，与えられたすべての具体例から作られる概念仮説の集合をバージョン空間と定義する．このバージョン空間は，最も特殊な概念を最大元，最も一般的な概念を最小元とするラティスを形成する[*2]．バージョン空間探索法は，個別の訓練例 i を参照しながら，特殊な概念の下界の集合 S と一般的な概念の上界の集合 G を挟み込んでいき，最適な概念を探索するプロセスである．i が正例の場合に S を更新する際は，S の要素が i と一致しない場合，i を含む概念まで当該 S の要素をバージョン空間内で下にたどるという手続きをとる．この手続きを本提案手法の素性構造の演算 ∇ を用いて形式的に記述すると，次のとおりとなる．あるバージョン空間 D_v について，正例 $i \in D_v$ かつ S の要素 $s_j \in D_v$ とし，$i \neq s_j$ とする．このとき更新手続きは $s'_j = \nabla(i, s_j)$ と表すことができる．また，素性構造は再帰的な記述（素性値として素性構造を持つことができる）が可能なため，複雑な概念についても記述能力を有することが期待できる．

また，今回は型なし素性構造を用いてアクセス制御のモデル化を行ったが，型付き素性構造 [17] を用いることにより，セキュリティリスクレベルを型階層を用いてよりシンプルに表現できる可能性がある．したがってアクセス制御を型付き素性構造を用いてモデル化することも今後の課題としたい．また，合わせて，型付き素性構造はそれを処理できるシステムである LiLFeS [18] や ALE [19] などが開発されているため，それらを用いてモデル化の検証を行いたい．

[*2] 逆に，最も一般的な概念を最大元，最も特殊な概念を最小元とする説明もあるが，本論文では文献 [16] に従うこととする．

また，本論文で我々は，記述性の高い記法として素性構造を導入したが，高い記述性は，高い可読性を生むことが期待できる．可読性の評価については未実施であるため，今後の課題としたい．評価方法案として，提案手法を含む様々な記法で表現されたポリシとアクセス要求を被験者に提示し，被験者にアクセスの諾否を判定してもらう，という方法を検討中である．これは，可読性の高い記法ならば，正しい諾否の判定を導出しやすいだろうという仮説に基づく．

参考文献

[1] Capretta, V., Stepien, B., Felty, A. and Matwin, S.: Formal correctness of conflict detection for firewalls, *FMSE '07: Proc. 2007 ACM Workshop on Formal Methods in Security Engineering*, pp.22–30 (2007).

[2] 横川 晃，品川高廣，加藤和彦：クラス階層型セキュリティポリシーによるアクセス制御，情報処理学会研究報告—システムソフトウェアとオペレーティング・システム，Vol.2009-OS-112, No.9, pp.1–8 (2009).

[3] Kudo, M. and Hada, S.: Access Control Model with Provisional Actions, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E84-A, No.1, pp.295–302 (2001).

[4] 今村 誠：素性構造の単一化，情報処理，Vol.32, No.10, pp.1070–1078 (1991).

[5] Chin, S.-K. and Older, S.: *Access Control, Security, and Trust: A Logical Approach*, CRC Press (2010).

[6] Denning, D.E.: A lattice model of secure information flow, *Comm. ACM*, Vol.19, No.5, pp.236–243 (1976).

[7] Sandhu, R.S.: Lattice-Based Access Control Models, *Computer*, Vol.26, No.11, pp.9–19 (1993).

[8] World Wide Web Consortium (W3C): P3P: The Platform for Privacy Preferences (online), available from ⟨http://www.w3.org/P3P/⟩.

[9] Cranor, L.: P3P: Making Privacy Policies More Useful, *Security & Privacy*, Vol.1, No.6, pp.50–55, IEEE (2003).

[10] Karjoth, G., Schunter, M., Herreweghen, E.V. and Waidner, M.: Amending P3P for Clearer Privacy Promises, *DEXA '03: Proc. 14th International Workshop on Database and Expert Systems Applications*, pp.445–449 (2003).

[11] May, M.J., Gunter, C.A., Lee, I. and Zdancewic, S.: Strong and Weak Policy Relations, *POLICY '09: Proc. 2009 IEEE International Symposium on Policies for Distributed Systems and Networks*, pp.33–36 (2009).

[12] May, M.J., Gunter, C.A., Lee, I. and Zdancewic, S.: Strong and Weak Policy Relations, Technical Report, MS-CIS-09-10, University of Pennsylvania (2009).

[13] Walker, D.D., Mercer, E.G. and Seamons, K.E.: Or Best Offer: A Privacy Policy Negotiation Protocol, *POLICY'08: Proc. 2008 IEEE International Symposium on Policies for Distributed Systems and Networks*, pp.173–180 (2008).

[14] Ni, Q., Trombetta, A., Bertino, E. and Lobo, J.: Privacy-aware role based access control, *SACMAT '07: Proc. 12th ACM Symposium on Access Control Models and Technologies*, pp.41–50 (2007).

[15] Organization for the Advancement of Structured Information Standards (OASIS): Extensible Access Control Markup Language (XACML) (online), available from ⟨http://xml.coverpages.org/xacml.html⟩.

[16] Mitchell, T.M.: Generalization as Search, *Artif. Intell.*, Vol.18, No.2, pp.203–226 (1982).

[17] Carpenter, B.: *The logic of typed feature structures*, Cambridge University Press (1992).

[18] Makino, T., Yoshida, M., Torisawa, K. and Tsujii, J.: LiLFes - Towards a Practical HPSG Parser, *COLING-ACL*, pp.807–811 (1998).

[19] Gerald Penn: The ALE Homepage (online), available from ⟨http://www.cs.toronto.edu/~gpenn/ale.html⟩.

藤田 邦彦 （正会員）

1999 年北陸先端科学技術大学院大学情報科学研究科情報処理学専攻博士後期課程修了．同年日本電信電話株式会社入社．現在，コミュニケーション科学基礎研究所研究主任．アクセス制御，フォーマルメソッド，デジタル著作権管理の研究に従事．博士（情報科学）．人工知能学会，電子情報通信学会各会員．

塚田 恭章 （正会員）

1990 年東京工業大学大学院理工学研究科情報科学専攻修士課程修了．同年日本電信電話株式会社入社．現在，コミュニケーション科学基礎研究所主幹研究員．型理論とロジカルフレームワーク，論理的手法に基づくソフトウェアの安全性検証の研究に従事．博士（工学）．日本ソフトウェア科学会，日本応用数理学会，ACM 各会員．