文脈に基づくセキュリティモデルによる センサ情報へのアクセス制御とセンサ操作への権限制御

Access Control for Sensor Information and Authorization Control for Sensor Operation by Context-based Security Model

城戸 聡 †, 渡邊 泰成 ‡, 飯島 正 ‡
Satoshi KIDO†, Taisei WATANABE‡, and Tadashi IIJIMA‡
† 慶應義塾大学大学院　理工学研究科
‡ 慶應義塾大学　理工学部
†Graduate School of Science and Technology, Keio Univ.
‡Faculty of Science and Technology, Keio Univ.

要旨

文脈を意識したアクセス制御ポリシーの強制系の設計と，それを用いたセンサの制御例に関して報告する．このアクセス制御モデルではオブジェクト指向ペトリネットを文脈表現として採用している．オブジェクト指向ペトリネットには幾つかのバリエーションがあるが，本報告では，nets-within-nets 意味論と呼ばれる意味論に基づくものを採用する．特に，異なる管理組織にまたがったワークフローにおける動的なアクセス制御モデルに利用することを目的としている．このモデルによって，センサ情報へのアクセス制御とセンサの操作への権限制御を実現している．全体的なアーキテクチャは XACML の構成を参考にしており，セキュリティポリシーの記述には Scala による内部 DSL を利用している．

1. はじめに

文脈を意識したアクセス制御ポリシーの強制系の設計に関して報告する．このアクセス制御モデルではオブジェクト指向ペトリネットを文脈表現として採用している．特に，複数のデータ管理組織にまたがって分散されている情報にアクセスすることを想定している．従来，単一のデータベースにアクセスするのであれば，その Facade としてのデータベース管理システムにアクセス制御モデルを組み込んでおくことで，アクセス制御を実現していた．しかし，情報資源が分散・管理される近年では，それを前提とすることが必ずしも有効ではなくなってきている．そこで本報告では，分散した複数の情報資源へのアクセス制御モデルを抽象化し外部化することで，共通的なフレームワークを構築することを試みている．もちろん，このモデルではアプリケーションソフトウェア側だけでアクセス制御ポリシーを強制することはできないので，情報資源側とアプリケーションソフトウェア側で協調してポリシーを順守するアクセス制御を成立させることになる．

本報告で利用するオブジェクト指向ペトリネットは，オブジェクト指向の考え方に基づくモジュール性を取り入れており，組織ごとのワークフロー，組織にまたがるワークフロー，情報にアクセスする主体 (アクター) ごとの状態遷移，アクセスされる資源の状態遷移などのそれぞれに対して，個別にプロセス表現を与えて協調動作させることができるモデルである．この点が，前述した「複数のデータ管理組織にまたがって分散されている情報にアクセスする」際の文脈表現という目的に合致していることから，本報告では，このオブジェクト指向ペトリネットを文脈表現に採用する．

次節では，本報告で想定している具体的事例から，文脈を意識したアクセス制御 (Context-Aware Access Control: CxAC と略す) [1][2]，更には文脈を意識した権限制御 (Context-Aware Authenticaiton Control) の必要性を示す [1]．続く第 3 節では，nets-within-nets 意味論に基づくオブジェクト指向ペトリネットについて解説する．それを踏まえ，第 4 節では，CxAC の一つのサブクラスとして，ロールに基づくアクセス制御モデル (Role-Based Access Control: RBAC) [3] に文脈をとり入れた文脈指向ロールベースアクセス制御 (権限制御) モデルを示す．さらに，第 5 節においてポリシー強制系の設計に関して述べる．

2. 具体的な想定事例

この節では，状況に応じた動的なアクセス権限の変更事例に関して検討する．一つ目の想定事例として，センサ情報へのアクセス制御を取り上げる．近年，監視カメラによる映像や交通系 IC カードから読

[1] ここでは，情報へのアクセス制御と，通常，より一般性が高いとされる権限制御の間で特に区別を設けることはしない．機器の操作権限を有することを機器へのアクセスが許可されていると表現することもある

み取られるデータなどのセンサ情報に対し，PPDM(Privacy Preserving Data Mining) というようなプライバシ保護を加えたデータ分析の必要性が認識されてきている．実際に，認知症高齢者介護施設 (グループホーム) において，監視カメラは介護活動を支援する一手段としての有効性が認められてはいるが，その監視によってプライバシが侵害されているとする意見も存在する．監視カメラに限らずセンサネットワークが普及すれば，セキュリティとプライバシに関する同様の議論が起こることは避けられない．現在でもプライバシ情報の利用と保護のバランスを取る方法の模索が続けられている [4]．

こうしたセンサ情報のプライバシ保護策として「情報の加工」が考えられる．例えば，監視カメラの生映像をそのまま表示，録画するのではなく，映像から人の存在を感知し，在室している人がいるかどうかという情報のみをモニタ画面に表示したり，IC カードから取得した個人情報を性別と年齢層のみに制限し，データマイニング目的で保存しておくなど，プライバシ侵害に関わる情報を削除した加工された情報にのみアクセスできるようにする．これにより寝室や浴室といった高度なプライバシ保護が必要な環境であっても，監視カメラの実用が可能となる (図 1)．

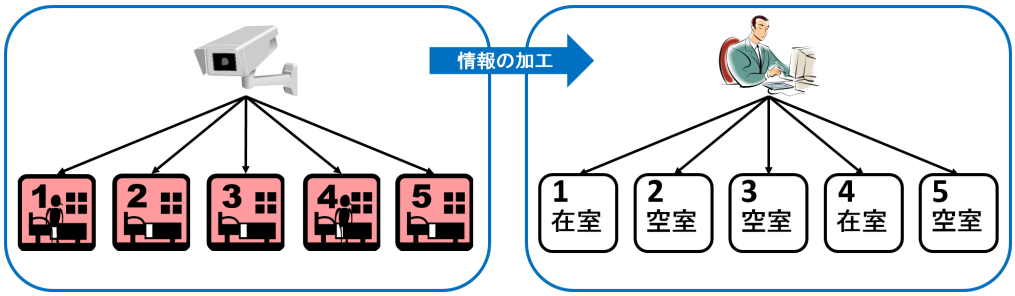


図 1: 監視カメラ映像の情報加工例

しかし，こうした「情報の加工」により，情報の有用性に悪影響が生じる可能性もある．上記の例で言えば，認知症高齢者介護施設での介護側にとって，緊急時の被介護者救助の際に監視カメラを活用できなくなってしまう．認知症高齢者介護施設で想定される緊急時には以下の 2 点がある．

- 被介護者が何らかの状況により他人の救助を必要とする場合
- 災害，事故等により施設からの緊急避難を要する場合

こうした状況に陥った場合，被介護者の救助，ないしは避難を迅速に進めるため，介護側は加工されていない情報，つまり監視カメラの生映像にアクセスできるようにしなければならない．従って，プライバシ保護のための「情報の加工」を行う場合，同時に，加工されていない情報へのアクセスも考慮しなければならない．つまり，これまでの多くモデルが対象としてきた電子文書へのアクセス制御とは異なり，本報告で扱うセンサ情報へのアクセス制御では，情報の加工度合いを考慮に入れた，状況を意識したアクセス制御を整備する必要がある．

別の想定事例として，センサ操作の権限制御を取り上げる．センサのような機器の操作権限の制御では，これまで提案されてきた多くのモデルが対象とする電子文書情報へのアクセス制御とは異なり，Readable，Writable 以外の権限も制御する必要がある．例えば，監視カメラの映像であれば，読取，編集，ズームイン，ズームアウト等が挙げられる．しかし，これらの操作権限が常に利用者に付与されることは必ずしも有効ではない．例えば，コンビニエンスストアに設置されている監視カメラを考えてみる．コンビニエンスストアの従業員側に，店内ではなく，店外を映す監視カメラの全ての操作権限を与えてしまうと，その監視カメラを利用して店外を歩く人のプライバシを侵害する行為が可能になってしまう．そのため，従業員側には監視カメラの全ての操作権限が与えられるべきではない．一方，犯罪捜査等の緊急時には監視カメラのリアルタイムの映像を解析しなければならない状況もありうる．

このように，実際には状況に応じてセンサ操作への権限が適切に制御される必要がある．しかし，従来の RBAC のようなアクセス制御モデルを単純に適用した場合，情報へのアクセス可能かどうか (Readable かどうか) の判定しかできないため，操作権限の制御という点において不十分である．従って，センサ操作の権限制御に対しても，状況を意識したアクセス制御を実現しなければならない．

本報告は，電子文書のための状況を意識したアクセス制御モデル Context-Aware Access Control (CxAC) [1][2] を，さらにセンサー情報に関連した情報の加工操作や機器制御の権限制御 (Authorization) にまで拡大する提案 (Context-Aware Authorization Control) である．

3. オブジェクト指向ペトリネット

3.1. オブジェクト指向ペトリネットとは

本研究では，文脈表現としてオブジェクト指向ペトリネットを用いている．オブジェクト指向ペトリネットとはオブジェクト指向概念に基づいてモジュール性を取り入れたペトリネットの拡張モデルであり，多くのものが提案されている．ここでは，nets-within-nets 意味論に基づく参照ネット (Reference Net) [5] の考え方を採用したオブジェクト指向ペトリネットを採用している．

参照ネットにおいては，トークンには 2 通りある．一つは，通常の P/T ネット (Place/ Transition ネット) におけるトークンである単純トークン (Black Token) である．プレースに存在する単純トークンは，トークン数を示す．もう一つは，別のサブシステムを表現するサブネットへの参照 (reference) に相当する参照トークン (Reference Token) である．プレースは，単純トークン用の単純プレースと，参照トークン用の参照プレースに分類できる．

- 単純トークン (Black Token)[2]
- 参照トークン (Reference Token)

ここで，参照トークンが参照しているサブネットは，オブジェクト・ネットと呼ばれる．また，一つの参照トークンがトランジションの発火によってコピーされることもある．その際には，あくまで共通のオブジェクト・ネットへの参照がコピーされる点に注意されたい．

3.2. オブジェクト指向ペトリネット OPeN

以下では，具体的なオブジェクト指向ペトリネットのモデルとして，著者らの研究プロジェクトで継続的に開発を行ってきたオブジェクト指向ペトリネット OPeN ファミリー (the Object-oriented Petri Net family) の中でも，特に人と連携して行う協調作業や，SOA におけるサービス間連携を記述するためのビジネスモデル記述 (ワークフロー) に特化した OPeN/WF を用いている．

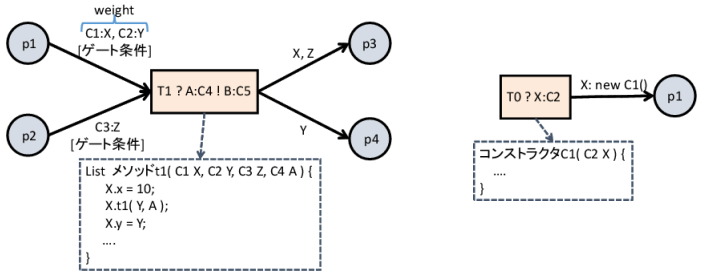


図 2: OPeN におけるアークとトランジション

OPeN では，個々のオブジェクト・ネットは，クラス記述から生成されるインスタンスに相当する．クラス記述は，オブジェクトを定義するものであり，opn(もしくは従来からの慣習から pnmlx) という拡張子を持つ XML ファイルとして記述する．クラスが定義するオブジェクト・ネットは，そのオブジェクトの振る舞い (ライフサイクル) を記述する一つのペトリネット，そのオブジェクトが持つプロパティ (インスタンス変数) 群と，ライフサイクルを遷移する際に使われるメソッド群である．メソッドは，主に，トランジションの発火条件であるゲート条件の記述や，トランジション発火時に起動されてプロパティの値の更新に使われる．図 2 に OPeN におけるアークとトランジションのアノテーションを示す．

トランジション t_1 の発火によって，同じ名前をもつメソッド t1 が起動される [3]．現時点のプロトタイプ仕様では，プロパティの型やメソッドの定義のための言語仕様は，厳密には定義していないが，図 2 に準拠し，基本的なリフレクション機能を備えた一般的なオブジェクト指向言語 (たとえば Java や Scala) のクラス定義 (ネットと同名のクラス定義) に外部化しており，相互に対応付けている．

複数のオブジェクト・ネットから構成されるシステムは階層的に表現されるが，ここでは，まず分かり易さを優先し，2 階層の構造に限定してその意味論を説明する．これは，[5] における EOS(初等オブジェクトシステム) に相当する．2 階層の構造では，全体的な振る舞いを統制するオブジェクト (ペトリネット) と，それに規定された振る舞いを行うサブオブジェクト (ペトリネット) 群から構成される．ここでは，全体的な統制を記述する方を EOS の慣習に従いシステム・ネットと呼び，統制されたサブオブ

[2] 慣習的にブラック・トークンと呼ばれるが，ここでは分かり易さのために単純トークンと呼ぶ

[3] 後述するオブジェクト指向ペトリネットの発火則「Interaction(相互作用) 」により，t_1 と同期するトランジション t_{s1} が，このペトリネットのサブネット中に存在し，かつそれが発火可能であれば，その t_{s1} も同時に発火する．t_{s1} が発火可能でなければ t_1 も発火できない．トランジションの入力アークがメソッドの入力パラメータに対応する．

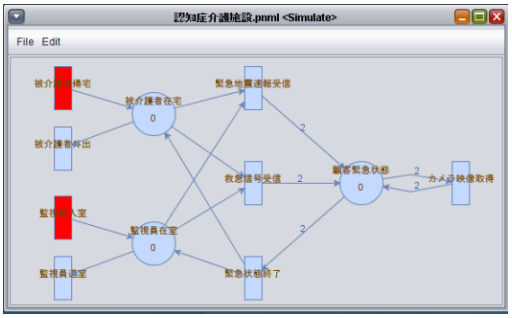
ジェクトの側をオブジェクト・ネットと呼ぶことにする [4]．システム・ネットとオブジェクト・ネットの間では，一部のトランジションの発火が同期的に行われる．システム・ネットとオブジェクト・ネットの同期関係は，双方のトランジションの対の集合として表現される．システム・ネットにおいて，あるトランジション T が発火するためには，以下の三条件が成り立たねばならない [5]．

(a) システム・ネットにおいて，単純トークン B_i に関して発火条件を満たしていること，
(b) システム・ネットにおいて，参照トークン R_j に関して発火条件を満たしていること，
(c) T の発火に寄与している参照トークン R_j が参照しているオブジェクト・ネット中で，T と同期関係にあるトランジションが発火可能であること．

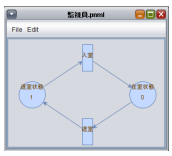
以上が，nets-within-nets 意味論 (ないし参照意味論)，およびその拡張に基づくオブジェクト指向ペトリネットの発火のメカニズムである．

4. 文脈指向ロールベースアクセス制御モデル

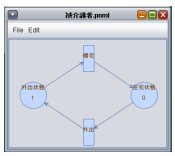
文脈を意識したアクセス制御モデル (CxAC) には，そのベースとするアクセス制御モデルと，具体的にどのような状況を取り扱うかで，2 つの観点からサブクラスが考えられる．ここでは，ベースとするアクセス制御モデルに RBAC(Role-Based Access Control) を採用し，データに対する操作を従来の「情報の読み書き」をベースにしたものから,「情報の加工」や「機器の制御」まで拡大した権限制御モデルである，文脈指向ロールベースアクセス制御モデルを提案する．



(a) 介護施設のシステム・ネット



(b) 監視員の振る舞い



(c) 被介護者の振る舞い

図 3: オブジェクト指向ペトリネットによる文脈表現

一般に，アクセス制御（権限制御）は，主体–操作–対象の三つ組のうち，許可されている組の集合で表現できる．ロールベースアクセス制御モデルは，情報にアクセス主体ごとにアクセス権限を定義するのではなく，ロールすなわち役割に基づいて主体を分類し，ロールごとにアクセス権限を定義するものであり，現在，広く使われている．ロールに分類することで大幅にアクセス権限の定義数を減らすことができ，またロールへ継承階層を導入することで管理の容易性を一層進めることができる．これにもとづき，介護施設の監視カメラの場合には，介護職員の一人一人にアクセスと操作の権限を定義するのではなく，介護福祉士，監視員，パートといった役割ごとに，アクセス権限を定義するものとする．

また，一般にこうしたロール割り当ては，アドホックに変更されるものではない．すなわち，介護福祉士や監視員といった役割はそう簡単に相互に変更されるものではなく，役職に関しても変更はありうるが，アドホックな変更とは言い難い．従って，緊急時のみのアクセス権限の拡大といった状況に応じた動的変更機構は，通常は，スパンの長いロール割り当てとは別に扱われる．

ロールは主体の表現の一形態であるが，ここでは操作の表現も重要である．電子文書情報の「読み書き」をベースとしたアクセス制御だけではなく，情報の加工操作や機器操作を含めて拡大しているので，多様な操作の体系化や，それを組み合わせた複合操作の定義も重要な要素であり，引き続き取り組んでいかねばならない．

更に，CxAC ではアクセス者と情報資源 (もしくはそのサーバ) のそれぞれの振る舞い (ワークフロー) を文脈とし，それぞれオブジェクト指向ペトリネットによって表現している．協調動作を行う場合はトランジション間の同期によって表現する．

[4]階層的なシステムの場合には，ここで言うオブジェクト・ネットもまた，より下位のサブペトリネットから見れば相対的にシステム・ネットに相当する．OPeN は EOS の概念を多階層に拡張しているが，混乱のない限り，最上位ネットに限らず，隣り合う階層のネット間の関係を，システム・ネットとオブジェクト・ネットというように称することとする

[5]但し，OPeN の場合には，2 階層の EOS ではなく，多階層を許容しているので，条件 (c) は再帰的に，より下位層にあるオブジェクト・ネットに対して適用されていくことになる．

以下に簡単な例を示す．この例では，認知症高齢者介護施設において，監視員が監視カメラで施設内を監視している際，高度なプライバシ保護が必要な部屋 (例えば浴室・寝室など) のカメラ映像に対するアクセス制御のポリシーを与えている．現行では，軽犯罪法第一条二十三項に「正当な理由がなくて人の住居、浴場、更衣場、便所その他人が通常衣服をつけないでいるような場所をひそかにのぞき見た者」[6] は違法とあり，被介護者の安全のための監視や緊急時の救助支援という「正当な理由」がある場合は設置することも可能である．そこで，浴室・寝室に設置された監視カメラの映像へのアクセス制御として，気象庁から緊急地震速報を受信した場合，または被介護者から救急信号を受信した場合に，緊急時と判断する文脈をオブジェクト指向ペトリネットで表現した (図 3)．

オブジェクト指向ペトリネットでは，オブジェクト・ネットで個別の要素の振る舞いを表現し，システム・ネットでそれらの協調の様子を表現することができる．2 階層以上を持つことによって複数組織にまたがったワークフローと，個別の組織のみならず，その中を動くアクター (情報にアクセスする主体) と，情報資源 (アクセスされる対象) の個別の振る舞い (状態遷移) を表現することができる．ペトリネットにおいて状態はマーキングとして扱うことができる．したがって，状態の時系列すなわち状況は，指定したいマーキングの時系列を作り出すペトリネット (ワークフローのサブネット) で表現することができる．そこで，本報告では，トランジションの発火系列で文脈を表現するものとする．文脈はワークフローのサブペトリネットで可能な発火系列の集合で表現し，これを文脈パターンと呼ぶ [7]．ある発火系列が文脈パターンにマッチするとは，それが，そのサブペトリネットで可能な発火系列の集合の要素である場合に限る．

5. アーキテクチャ

本報告で提案する CxAC のアーキテクチャとして，セキュリティポリシー記述方法と，ポリシー判定エンジンの構成に関して説明する．本モデルでは，セキュリティポリシーの記述に，基本的にプログラミング言語 Scala の内部 DSL 表現を利用し自然言語表現からポリシー判定エンジンの入力とする XML 表現に変換することを可能にしている (図 4)．ポリシー判定エンジンのアーキテクチャは，XACML(eXtensible Access Control Markup Language) のアーキテクチャをベースにしており，ポリシーの表現も XACML に準じている．

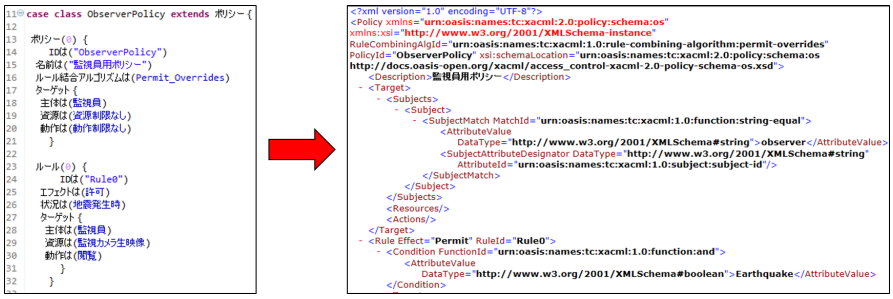


図 4: ポリシーの DSL 表現と XML 表現への変換

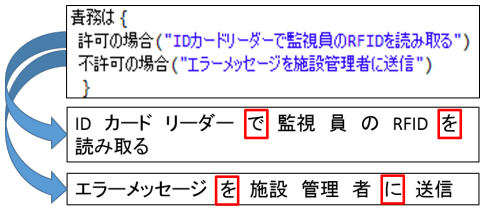


図 5: 責務の日本語 DSL による記述とその解析

但し，日本語によるポリシー記述では，日本語文章の解析手段として形態素解析を適用している．日本語は格助詞 (「が」「の」「を」「に」等) の前にその格要素が配置される後置表現になっていることに加え，格構造の単位で語順に柔軟性がある．英語をベースにしたプログラミング言語 (Java や Scala) における内部 DSL では，前置表現に向いているため日本語と相性が悪い．日本語文を内部 DSL として表現しようとしても，一般には，単語だけなら簡単に扱えても，文章となると不自然になりがちである．

[6] 軽犯罪法は昭和二十三年五月二日から施行されている．最終改正は昭和四八年一〇月一日法律第一〇五号．http://law.e-gov.go.jp/htmldata/S23/S23HO039.html

[7] あるペトリネット n のサブペトリネット $sub(n)$ は n のトランジション集合，ペトリネット集合，アーク集合の部分集合で構成できるペトリネットである

そこで，日本語 DSL 部分は外部 DSL とし，その構文解析手段として形態素解析を用いることとした．これにより日本語文章を意味のある単語ごとに区切ることができることができ，日本語独特の後置表現に対応することができるだけでなく，格構造を単位とする語順の自由度にも対応し易くなっている．本報告では，日本語形態素解析ツールとして，MeCab-IPADIC 辞書をサポートしており，独自辞書の編集が可能である kuromoji[8] を採用している．XACML におけるセキュリティポリシーの一部である責務 (obligation) の日本語 DSL による記述とその解析例を図 5 に示す．これによって，プログラミングの知識がない現場の業務担当者であっても，ポリシーの編集やポリシー内容の確認がより容易となっている．こうしたセキュリティポリシーの可読性の高さも重要な指針の一つとなる．

CxAC のポリシー判定エンジンの構成は，OASIS により標準化作業が進められている XACML(eXtensible Access Control Markup Language) [6] のアーキテクチャの拡張である．Sun Microsystems の XACML 実装 [7] を元に実装を行った．本報告におけるアクセス制御モデルのアーキテクチャを図 6 に示す．

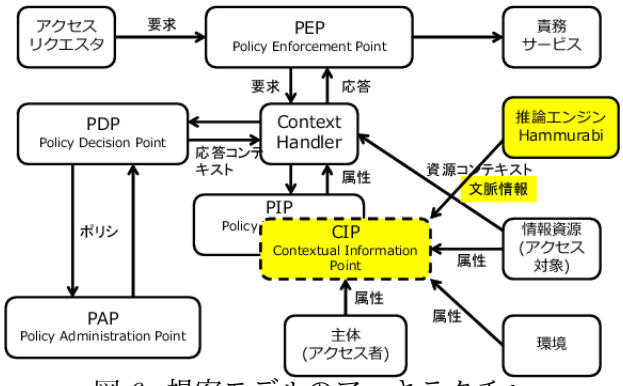


図 6: 提案モデルのアーキテクチャ

また，本研究では，文脈推論にルールエンジン Hammurabi [8] を用いている．図 6 のように，Hammurabi で推論された文脈情報を基にパーミッションを与えている．Hammurabi はプログラミング言語 Scala で表現可能であり，Java との親和性が高いというだけでなく，DSL 記述にも適しているため，このルールエンジンを文脈推論エンジンとして採用している．

6. まとめ

文脈を意識したアクセス制御ポリシーの強制系の設計とそれを用いたセンサ情報の制御例に関して報告した．このアクセス制御モデルではオブジェクト指向ペトリネットを文脈表現として採用している．全体的なアーキテクチャは XACML の構成を参考にしており，セキュリティポリシーの記述には Scala による内部 DSL を利用している．但し，事例の実装に関しては暫定的なものであり，今後，より現実環境に近い実装を行い，加えて他の想定事例への適用も進めていく予定である．

謝辞

当研究室・Sec(セキュリティ) 班の卒業生の皆さん (塩田 哲哉 (2012 年度学部卒)，山上 燦 (2011 年度学部卒)，雨宮 美奈帆 (2010 年度学部卒)，山本 由香里 (2009 年度学部卒)，奥村 恭平 (2008 年度学部卒)) に感謝します．

参考文献

[1] 飯島 正, 城戸 聡: “文脈に基づくセキュリティモデルの強制系の実現と評価,” 第 9 回 全国大会・研究発表大会, 情報システム学会, 2013.
[2] Tadashi Iijima, Satoshi Kido: “Design and Implementation of a Context-Based Security Model,” Proc. of 11th Joint Conference of Knowledge-based Software Engineering 2014, Sept.17-20, 2014. CCIS-Vol.486, pp.356-370, Springer.
[3] R. S. Sandhu, E. J. Coyne, et al.: “Role-based access control models”, IEEE Computer, vol.29, no.2, pp.38–47, 1996.
[4] 杉原 太郎, 藤波 努, 中川 健一: “カメラとモニタ導入に伴うグループホーム介護者の負担感に関する研究”, 電子情報通信学会技術研究報告 WIT, 福祉情報工学, 107(555) 号, pp.57-62, 2008.
[5] Rüdiger Valk: “Object Petri nets? Using the nets-within-nets paradigm”, LNCS 3098, pp.819–848, Springer, 2004.
[6] OASIS Standard: “eXtensible Access Control Markup Language (XACML) 3.0”, 22 January 2013, http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf
[7] Sun Microsystems, Inc. (当時): “Sun’s XACML Implementation”, 16 July 2004, http://sunxacml.sourceforge.net/
[8] M. Fusco: “Hammurabi - a Scala rule engine”, In Scala Days 2011, Stanford University, California, 2011.

[8] kuromoji はアティリカ株式会社によって開発された Java ベースのオープンソースの日本語形態素解析エンジンである．http://www.atilika.com/ja/products/kuromoji.html