平成20年度 卒業論文

論文題目

SNSにおける情報漏洩を防止するための情報フィルタの適用

神奈川大学 工学部 電気電子情報工学科

学籍番号　200402735

内野 雄策

指導担当者　木下宏揚　教授

目 次

図 目 次

第1章　序論

近年、ネットワークの拡大と共に、mixi [2] を始めとしたSNS(Social Networking Service) [1][5][9] が人々に急速に浸透してきた。SNS の楽しさやメリットを知った人々の中には，従来のSNS のユーザ数が爆発的に増加に伴い、情報過多も重なり、特定のテーマなどに特化したSNS が派生し、それは様々な遷移を遂げ，現段階でSNS にその形態を移してきた。ウェブコミュニティーが普及してからウェブ日記，掲示板やブログと同じようにSNS も時間の経過と共に、荒らしや情報漏洩、情報改竄などの問題が発生し、それらの対策が益々重要になっている。情報セキュリティ事故の増加において、セキュリティ対策の重要度は増しているが、データベースへの不正なアクセスによる情報漏洩、データベース構造や情報内容の破壊・改竄、アクセス権の変更など、外部・内部からのアクセスによる危険性が指摘されている。そのため、データベースへのアクセスに対しては、アクセス制御技術を適用する事が一般的である。しかし、顧客ごとにパミッションを割り振るだけでは、CovertChannel[14][17] と言う情報漏洩が発生することがあり得る。従って、セキュリティポリシーをセキュリティモデル [15][19] によって記述し、CovertChannel を分析・制御するシステムが必要になる。コミュニティをベースとしたセキュリティモデルについて、CovertChannel 分析・制御のアルゴリズム [12][20] によって情報漏洩を防ぐ。SNS の安全性, 信頼性を高めるため、最も適切なセキュリティモデルはCommunity Based Access Control Model[16][18] と考えられる。このモデルは、コミュニティ・主体・客体の属性を定義し、その属性に基づいた主体、客体関係を記述する。これによって、Covert Channel の分析、制御による情報漏洩の防止はより安全なネットワークシステムの提供に貢献する。CovertChannel が発生する際、コミュニティでは、通信の際に本来アクセスする権限の無いユーザーが第三者を通じて Object にアクセス可能となり情報漏洩が発生する。[13] この Covert Channel に問題に対する、SNS の対応を、コミュニティー間の CovertChannel に関する限定的な例をもとに情報漏洩の流れを考察する。

以下はウェブコミュニティー普及と移り変わりである。ネット提示版といったコミュニティを最初に、インターネットが爆発的に普及した1990 年代以降、老若男女が、インターネットに触れる機会が増えた。それに伴い、オンライン上に、ウェブコミュニティーという巨大な仮想社会が構築されていった。 それは、人々がリアルで

生活するように、ネットで日常的にコミュニケーションを取り合うために必要な要素であったことと、インターネットの匿名性は、年齢、権力問わず自由な発言ができ、多くの人と知り会える機会を手に入れることができたため、異常な速度で、社会に浸透していった。ウェブ日記やネット掲示板システムもウェブコミュニティーのそのひとつである。だが、匿名性を悪利用したいわゆる「荒らし」や「なりすまし」、またスパムなどの存在が障壁となり、 インターネット上で見ず知らずのもの同士が親交を深めることや、現実社会以上に本当に信頼できる情報を得ることは難しいという声は消えなかった。それにつれてネットコミュニティは変化を遂げた。ウェブ日記や掲示板に代わって普及したのはブログである。2001 年頃から広く普及し、ネット上での仮想の名前でも、ある程度の非匿名性を備えていた。しかし、ウェブ日記やネット掲示板に代わる巨大なウェブコミュニティーとして発展したブログでさえも、匿名性を原則とするネット社会の法則に縛られ、ネット上において信頼できる情報を求めることは困難だった。 そのため注目されたのがソーシャルソフトウェアという概念である。代表的なソーシャルソフトウェアとして、wiki と SNS が考えられる。SNS の主な特徴である招待制や実名登録といったものは、ネット上においても、ブログに勝る非匿名性を備え、信頼できる情報を手に入れることが可能となり、現実社会での友人関係をそのまま持ち込める SNS は瞬く間に広がっていった。しかし、非匿名性を備えた SNS であっても、「荒らし」といった悪質なユーザーと無縁ではなかった。そのため、SNS の安全性、信頼性を高めるためのセキュリティーの重要性が望まれていた。故に本論文では SNS に着目して、安全性の向上を目指した SNS のより良い利用法を模索する。SNS 上での情報流出を防ぐ、情報フィルタの適用法を考案し、オープンソースの SNS エンジン OpenPNE[6][11] に適用することを目的とする。

本論文の流れを以下に示す。

第 2 章では、SNS と CovertChannel に関する知識について述べる。

第 3 章では、SNS における Covert Channel について、コミュニティー間の情報漏洩を防止する方法を提案する。

第 4 章では、OpenPNE における covert channel の対応について考察する。

第 5 章では、本研究のまとめ，今後の課題について述べる。

第2章　基礎知識

2.1　SNS

SNS(ソーシャルネットワーキングサービス) とは、広義的には、社会的ネットワークの構築の出来るサービスや Web サイトであれば、SNS(ソーシャルネットワーキングサービスまたはソーシャルネットワーキングサイト) と定義される。狭義的には、人と人とのつながりを促進・サポートする、コミュニティ型の会員制のサービスと定義される。あるいはそういったサービスを提供する Web サイトも含まれる。友人・知人間のコミュニケーションを促進する手段や場を提供したり、趣味や嗜好、居住地域、出身校、あるいは「友人の友人」といったつながりを通じて新たな人間関係を構築する場を提供することを目的とする。人の繋がりを重視して「既存の参加者からの招待がないと参加できない」というシステムになっているサービスが多いが、最近では誰でも自由に登録できるサービスも増えている。

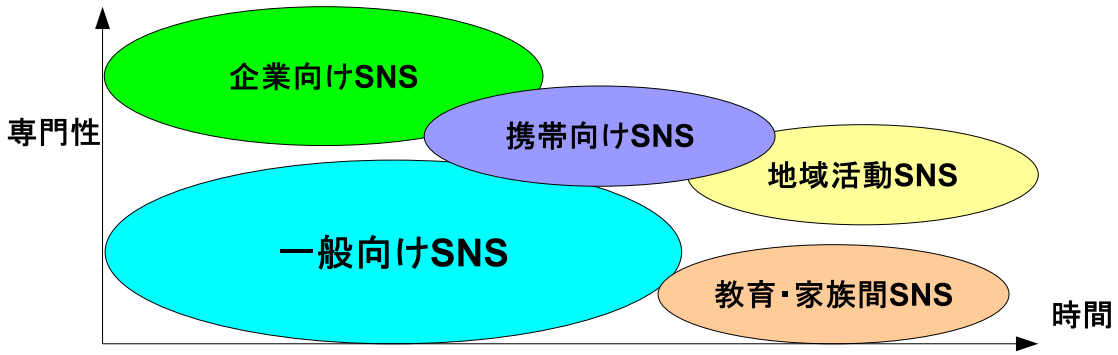


図 2.1: SNS の変化

2.1.1　SNS の種類と特徴

SNS の中には、既存の参加者から招待されると参加できるもの (紹介制、招待制)、SNS に登録することによって初めて参加が可能となるもの (登録制) などがある。

招待制とは、誰かから招待してもらわないと参加できないシステムであり、SNS に参加している友人からの招待というシステムは、荒らしなどの悪質なユーザーが発生しにくい特徴を備える。その安心感は、SNS の利用拡大に大きく貢献し、現在の mixi といった巨大な SNS を誕生する要因となった。しかし、閉鎖社会になりやすいため、まだ利用したことのないユーザーには、不便さを感じさせる。

登録制とは、既存ユーザーからの招待状が無くても誰でも自由に登録できるシステムであり、ある程度の匿名性があるためテーマや表現を自由に発言できる。また、話す機会のあまりない人とコミュニケーションをとることができるといった利点をもつ。しかし、招待制に比べ、悪質なユーザーが非常に多い。

世界的には招待制 SNS より登録制 SNS のほうが多いが、日本国内では招待制の mixi がシェアをほぼ独占している。ただし、2009 年春より招待制を廃止し、登録制に移行予定となっている。

SNS の機能として、

・自分のプロフィールや写真を公開して自己紹介する機能

・知人、友人を登録するアドレス帳

・同じ趣味や感性など共通する話題を持った人が掲示板などで交流できるコミュニティ機能

・予定書き込めるカレンダー機能

・互いにメールアドレスを知られること無く別の会員にメッセージを送る機能

・友人に別の友人を紹介する機能

・マイページを訪問したユーザーの履歴を参照できる機能

・公開範囲を制限できる日記帳

などで構成される。

悪質ユーザーの特定や閲覧制限を行うことによって、情報漏洩のリスクを抑えることができ、友人からの紹介やユーザーの履歴から、ユーザは安心して SNS を利用することができる。インターネット上での仮想的な名前でも利用制限のある SNS では非匿名制となりうるため、SNS では 2ch などの掲示板と比較して悪質なユーザーが発生しにくい。また、有料のサービスもあるが、多くは無料のサービスとなっており、多くの人が利用できる。そういったユーザーによってコミュニティの文化が作られていくが、ユーザー同士の諍いも発生しやすい。基本的にルールは存在しないが、一般的なマナーは存在するため、注意が必要である。

2.1.2　SNS の歴史と現状

2003 年頃に米国で成立した SNS は、2004 年頃に日本で登場し始めた。代表的な SNS として世界最大の会員数を持つ MySpace、国内においては日本最大の会員数を持つ mixi を筆頭に GREE[7]、Yahoo! Days[8] などがある。また、2004 年頃より大手企業各社でも社内コミュニケーションや内定者囲い込みにも使われ始め、多種多様な SNS が登場してきている。

国内最大の SNS である mixi では 2008 年 7 月 13 日に登録ユーザが 1500 万人を超えたことを発表した。[3] その中には、規約違反の多重登録者や利用をやめて放置された ID も含まれるが、利用者が 800 万人を超えた (2008 年 12 月 31 日)GREE に 2 倍以上の差をつけている。しかし、mixi は 2009 年春より招待制から登録制に移行するため、ユーザーに大きな影響を与えることが予想される。

2.1.3　SNS の抱える問題点

SNS の問題点として、インターネット上のサービスの問題点である個人情報の流出や不正ユーザによるスパム配信やフィッシング詐欺、迷惑メール、不正な書き込みによる出会いサイトへの未成年のアクセス、猥褻な画像や動画の開示などに加え、SNS 固有の問題点である情報公開によるプライバシーの侵害や悪質なユーザーによるコミュニティ書き込みや日記の情報漏洩、プロフィール詐称、複数アカウントによる荒らしなどがある。

解決策としては SNS ユーザーを複数のユーザーが評価することによって安全性を確認し悪質ユーザーを特定する、もしくは情報漏洩の流れを分析し、それによってセキュリティの強化をするといったことが考えられる。

2.1.4　SNS の将来

国内では mixi の普及により、多くの人が SNS を利用をしている。招待制における友人のつながりは、リアルな友人関係を SNS 上で再現でき、多くのコミュニティを生み出した。友人の紹介という安心感が利用を活性化させ、利用者は年々増えていった。こういった SNS の特徴を生かしたサービスを作ろうとする動きが広まっている。

2.2　SNS の関連技術

2.2.1　OpenPNE による SNS サーバ

今回の関連的な技術として、SNS エンジンの OpenPNE を使用する。OpenPNE は、株式会社手嶋屋が中心となって、オープンソース方式で開発を行ってきた SNS エンジンであり、PC とモバイルのハイブリッドなプラットフォームに、ウェブサイト、ポータルサイトの構築、管理など豊富な SNS 機能を搭載し、多様なサーバー環境で、誰もが無料で自由に利用できるオープンソースの SNS ソフトウエアである。社内 SNS やサークル、ソーシャルメディアやファンサイトなど、現在、30,000 以上の組織が OpenPNE を利用している多様な環境・組織に対応する柔軟性が高いソフトウェアである。本研究では OpenPNE を使用し、SNS サーバーの構築をする。ウェブページを作成し、運用するにあたって、HTML ファイルや、それを保存するディレクトリについての知識が必要だが、専門的な知識がなくてもウェブによる情報発信が簡単にできるように工夫されている。

2.2.2　OpenPNE の実装

図 2.1 は OpenPNE を実装した時のログイン画面である。管理者の権限でログインすると管理者として様々な設定をすることができる。図 2.2 はその管理者設定画面である。管理者設定画面により様々な設定が可能である。本論文では OpenPNE のコンテンツ管理機能を応用し、SNS の構築により発生するコミュニティーで、Covert Channel を制御する。

図 2.2: ログイン画面

図 2.3: 管理者設定画面

2.3　アクセス行列

アクセス行列とは主体 (Subject) と客体 (Object) の関係を表した行列のことで，主体と客体の関係にはR(Read:読み込み可)、W(Write:書き込み可能)、RW(Read+Write:読み書き可能)、φ (Phi:読み書き不可) の 4 種類の権限がある。

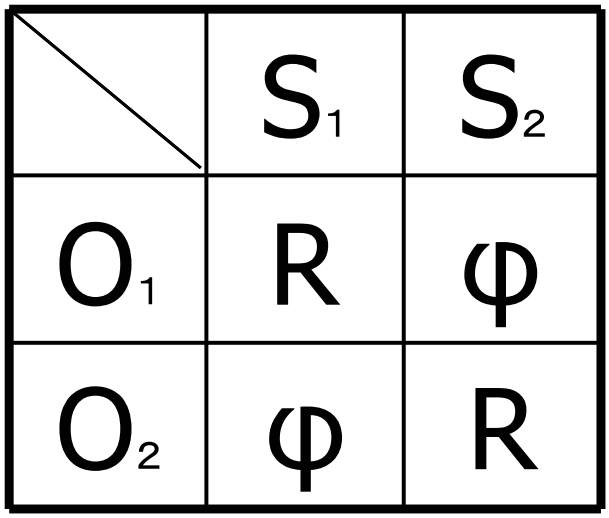


図 2.4: アクセス行列

2.3.1　客体 (Object)

客体はデータベース内で管理されている情報であり、ファイルに相当する。

- 名前 客体の名前
- 競合 管理している主体のコミュニティの情報
- 階層 コミュニティ内で指定されたセキュリティレベル
- 所有 管理している主体の情報
- プライベート 管理している主体の情報

2.3.2　主体 (Subject)

主体とはデータベース内で管理されている客体にアクセスする行為者であり、ユーザーに相当する。

- 名前 主体の名前
- 競合 管理しているコミュニティの情報
- 階層 コミュニティ内で指定されたセキュリティレベル
- 役割 客体の権限を決定する種類

2.3.3　コミュニティ (Community)

コミュニティとは、コミュニティの属性、コミュニティに属する主体、および、コミュニティが管理する客体とその属性の集まりから成る社会システムに相当する。コミュニティ (Community) 同士には利害関係があり、管理している主体には組織的に階層レベルや役割を割り振られる。コミュニティにも様々な種類があるが、インターネット上のコミュニティを考えてみると主体の役割や利害関係、或いはプライベートな情報（個人情報）が複雑に絡み合っている。現在、求められているのはこのように複雑に絡み合ったコミュニティにおいて実現するセキュリティモデルである。それが実現されたのが Community Based Access Control Model である。

2.4　セキュリティモデル

セキュリティモデルは、アクセス制御システムを構築する上で、セキュリティポリシーを具体的な論理的形式で表現したものである。そこには制御したいサービスや組織構造が反映される。最も単純な型では、パミッション (read,write, ¬ read, ¬ write) であり、主体 (Subject)、客体（object）を含めた３つをシステムで如何扱うかによってアクセス制御が行われる。従来のセキュリティモデルには様々な種類があり、使用される状況に応じて使い分けたり、複数使用する等しているが今回の実験ではアクセストリプルの構造や Covert Channel の検出・訂正を考慮したモデルとして Community Based Access Control Model を改良する。このモデルには幾つかのモデルには無い機能を持ち、他のモデルと併用することで Covert Channel の検出機能及び今回追加する訂正機能を実行することができる。

2.4.1　Community Based Access Control Model

Community Based Access Control Model は Covert Channel の制御を実現するためのセキュリティモデルの１つである。Access Control Agent System がこのモデルには組み込まれていて、コミュニティを用いた Covert Channel 分析が行なえる。まずユーザ数とファイル数を抑制し, 整理するために共通する属性を持った小規模なユーザの集合を Community と定義する。各コミュニティではそれぞれ内部で Covert Channel 分析を行い、Covert Channel がおきないように制御する。外部コミュニティと通信した場合の Covert Channel 分析は自コミュニティ、アクセス要求者、その要求について全て分析する。Covert Channel 分析は、Subject や Object の関係を以下のようなアクセス行列で表現し、Covert Channel の検出をする。属性はアクセス制御や Covert Channel との関連性から競合、所有、階層、役割、プライベートの５つを使用する。このとき、アクセス行列から図 1 のような 2 × 2 行列のパターンを全て取り出し、Covert Channel 分析を行なう。このやり方により、2 × 2 行列全てのパターンだけでなくそれらを組み合わせることで 3 × 3 やそれ以上の場合も Covert Channel を全て検出することが出来る。このようにこのモデルは Covert Channel を防ぐのに元々適したモデルであるので、この推論機能をベースとして改良し、新たな処理のアクセス制御を行なう。

2.5 Covert Channel

2.5.1 間接情報フロー

Covert Channel とはある客体 (Object) にアクセスする権限のない主体 (Subject) が第三者の協力によりアクセスできるようになる不正経路のことを示す。Covert Channel には間接情報フローと間接情報改竄の 2 つのパターンがある。この場合、CovertChannel はアクセス禁止のパミッションに矛盾する情報フロー（アクセス禁止のものもこのフローを使えばアクセス禁止の内容を閲覧したり、間接的に情報を改竄できてしまうこと）ともいう。（各 S=Subject、各 O=Object、R=読みこみ可能権限、W=書き込み可能権限）図 2.4 の場合、矢印の流れで Subject1 が本来読めないはずの Object1 を読めてしまう。Covert Channel 流出の流れは以下のようになっている。これは間接情報フローとも呼ばれる。

- **始点（Subject1・Object1）**Subject1 が Object1 を読み込む
- **中間点１（Subject1・Object2）**Subject1 が Object2 に Object1 で読んだ内容を Object2 に書き込む
- **中間点２（Subject2・Object2）**Subject2 が Object2 を読む
- **終点（Object1・Subject2）**Covert Channel により間接的に Object1 の内容を読めてしまう

このように不正な情報流出が発生してしまうため、アクセス制御を行う推論エンジンとしては出来るだけ発生を抑制し、検出と訂正を的確に行えるようにするのが情報フィルタに必要な機能である。

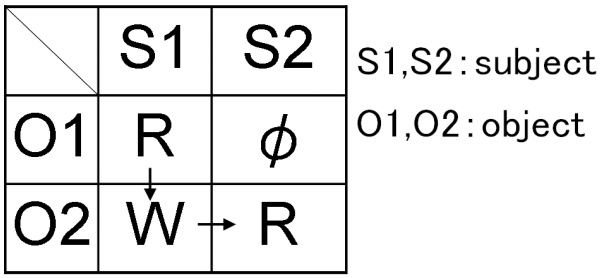


図 2.5: Covert Channel(間接情報フロー)

2.5.2　フローレベル

Covert Channel において、ある Subject から Subject へと情報が流出する回数をフローレベルと定義する。このレベルが多いほど漏洩や改竄の範囲が広くなる。関わる主体の数によって、Covert Channel の程度は決まり、フローレベル 2 が最小単位となり、フローレベル 2 の行列同士の連結がフローレベル 3 となる。フローレベルはフローレベル 2 が基となっていて, 図で表すと図 2.5 のように示される。この理由は図 2.5 から如何なるフローレベルにおいてもフローレベル 2 をベースとして生成される性質からである。この性質からフローレベル 2 を抑えれば, そこから発生しうる多くのフローレベルを防ぐことが出来、Covert Channel 分析と評価をシンプルにしている。

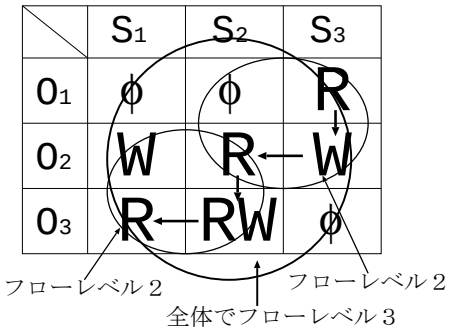


図 2.6: Covert Channel とフローレベル

2.5.3　Covert Channel の例

不正な情報経路である Covert Channel を全て塞いでしまえば安全なシステムを構築することが出来るように見えるが、単独では隠れチャンネル (Covert Channel) が存在しないようなコンピュータでもネットワークに接続されたコンピュータ群が協調することによって、隠れチャンネルを構成できてしまう。つまり、単独では安全なコンピュータでも、それがネットワークを構成すると安全ではなくなるような状況が簡単に存在し得るのである。このようなネットワーク構成機能の問題点が Covert Channel で利用される。例えば以下のような例が挙げられる。

- 会社の機密データを社外へ持ち出したり、社外の人間（社外の PC）でも見れるようにする。
- mixi 等の SNS の個人データが掲示板やブログ等不特定多数へ流出
- スパイウェア等, 個人 PC から情報を持ち出すためにこれを用いて通信を行い, 検知を困難とする。

WWW 等、不特定大多数が利用するネットワークでは意図しなくてもカバートチャンネルが発生してしまう恐れがあるのでそういった情報網では比較的安易に情報漏洩が起こりうる。このように Covert Channel は今のネットワーク社会にとって情報を安易に流出させてしまう存在なのである。

2.6　アクセスルール

主体、客体、パミッションから構成されるアクセス行列が与えられ、その中の間接的情報フローを分析した時、間接的情報フローが Covert Channel であるかどうかを判断する基準が必要である。それは実際、間接情報フローがあったとしても、その情報フローが脅威である場合とそうでない場合がある。たとえば、間接的情報フローが競合関係の間で生じる場合は Covert Channel であるが、間接的情報フローが連携関係（競合でない）の間で生じる場合は Covert Channel ではない。そこで、属性の定義を用いて Covert Channel を判定するルールを定義した。主体と客体の行列間において権限の制限させるためのアクセスルールを設定し、主体を客体に書き込まれた各属性によってアクセス権限の判断をする。属性には、以下の 5 種類があり、上から順番にルールを適用する.

- 競合属性

- 階層属性
- プライベート属性
- 所有属性
- 役割属性

情報のやり取りをする関係をアクセス制御に関係する属性別に分け。この定義に従ったアクセスの制御をおこなう。情報のやり取りで矛盾があった場合、Covert Channel と確定する。

2.7　情報フィルタ

情報フィルタとは Covert Channel 検出時にその Covert Channel が無くなるように特定の権限を変更することである。情報フィルタには 4 種類の方法があり、それぞれ一長一短がある。3 種類はフロー経路の権限を禁止して遮断するのに対し、Read 権限を許可する方法は情報共有の拡大の意味を持つ。以前は読めなかった客体が修正により普通に読めるようになれば、不正経路ではなくなるので Covert Channel 自体は無くすことができる。情報フィルタの具体的な処理を以下にまとめていく。図 2.4 のように Covert Channel が発生して検出された場合、以下、図 2.6 の (1)(2)(3)(4) のいずれかを適用すれば Covert Channel が解消される。

(1) (S1,O1) の READ 権限を削除。

(2) (S1,O2) の WRITE 権限を削除。

(3) (S2,O1) に READ 権限を添付。

(4) (S2,O2) の READ 権限を削除。

上記のどの情報フィルタを選択するかは各コミュニティのセキュリティポリシーや主体のアクセス履歴、ユーザがどういう方針で処理するか定めるユーザーポリシーを考慮して決定するが、(1) から (4) のどの場合でも Covert Channel は訂正できる。[10]

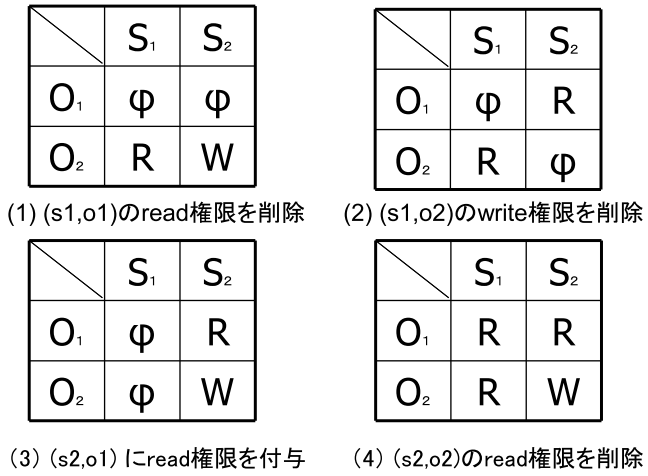


図 2.7: 情報フィルタによる Covert Channel 制御

2.7.1　競合属性

主体と客体の競合属性に書き込まれたコミュニティが一致しない場合は読み書きを禁止する。2 属性がそれぞれコミュニティ名属性と役割属性の組合せを含み、かつそれらが競合関係であるならば、その 2 属性間において競合属性間のアクセスが禁止される。下の図の場合は、主体である Subject1 と Subject2 はコミュニティ C1 に属しており，Subject3 は C3 に属している。同様に客体である Object1 と Object2 はコミュニティ C1、Object3 は C2 の管理下にある。Subject1 と Object3 ではコミュニティが違うのでアクセスできなくなる。

	S1[C1]	S2[C1]	S3[C2]
O1[C1]	RW	RW	RW
O2[C1]	RW	RW	RW
O3[C2]	RW	RW	RW

	S1[C1]	S2[C1]	S3[C2]
O1[C1]	RW	RW	
O2[C1]	RW	RW	
O3[C2]			RW

図 2.8: 競合属性によるアクセスルール

2.7.2　階層属性

主体と客体に刻まれた階層レベルに応じてアクセス権限を制御する。アクセスする客体のほうが低い場合は書き込みが禁止され、客体のほうが高い場合は読み込みが禁止される。一方が階層属性（セキュリティレベル）を含み、もう一方が階層属性を含まない時、かつセキュリティレベルが機密性を目的とするならば、階層属性

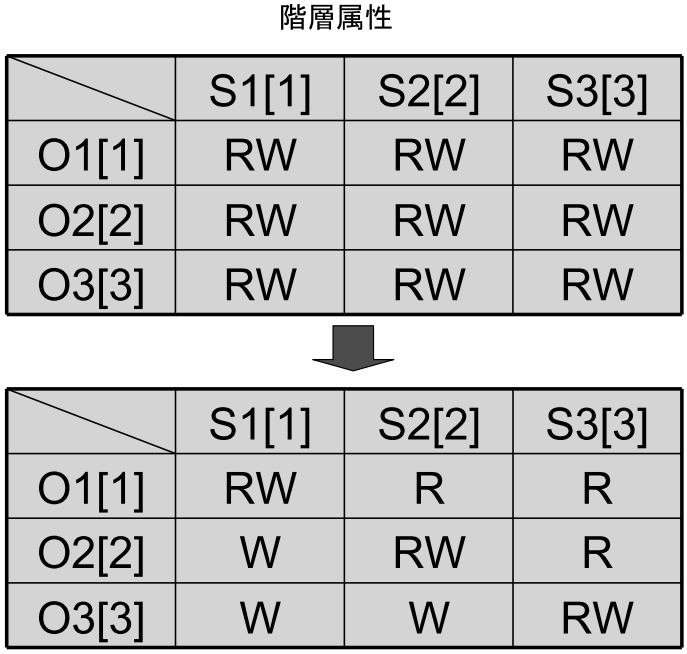


図 2.9: 階層属性によるアクセスルール

からの客体流出が禁止される。また、一方が階層属性（セキュリティレベル）を含み、もう一方が階層属性を含まない時、かつセキュリティレベルがインテグリティを目的とするならば、階層属性からの客体流入が禁止される。階層属性を持たない場合は階層レベルを最低値として扱う。

2.7.3　プライベート属性

客体の所有属性に書き込まれた主体以外の読み書きを禁止する。プライベート属性に客体情報がない場合はどの主体からの制限もかからない。一方が所有属性を含む時、かつプライベート属性にアクセス許可された属性の値をもう一方が持つ場合、2 属性間のアクセスが許可される。また、2 属性間において、同一のプライベート属性の値を持つ時、2 属性間のアクセスが許可される。

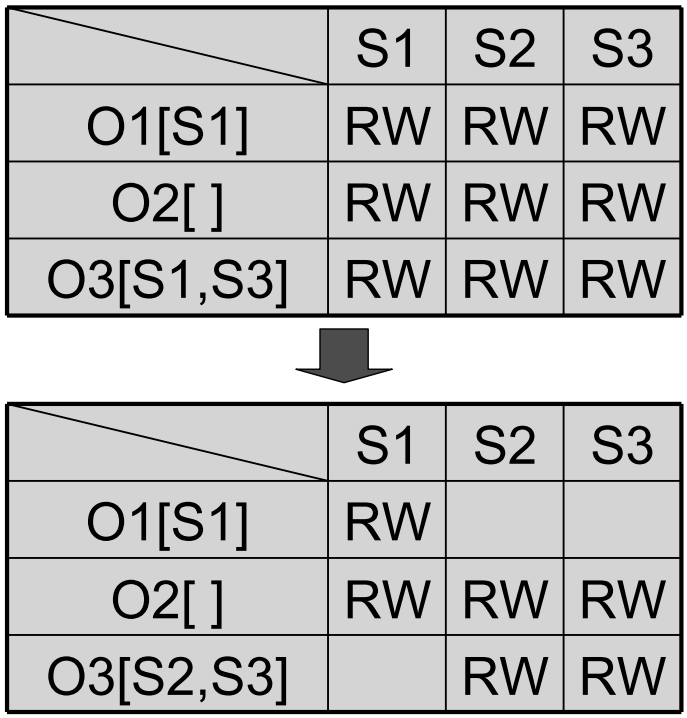


図 2.10: プライベート属性によるアクセスルール

2.7.4　所有属性

一方が所有属性を含む時、所有属性を含まない方からの WRITE が禁止される。所有属性に客体情報がない場合はどの主体からの制限もかからないため、一方が所有属性を含み、所有関係に矛盾がない時、WRITE 可能である。このとき、所有属性に対する READ のアクセスは、競合属性、階層属性に従う。

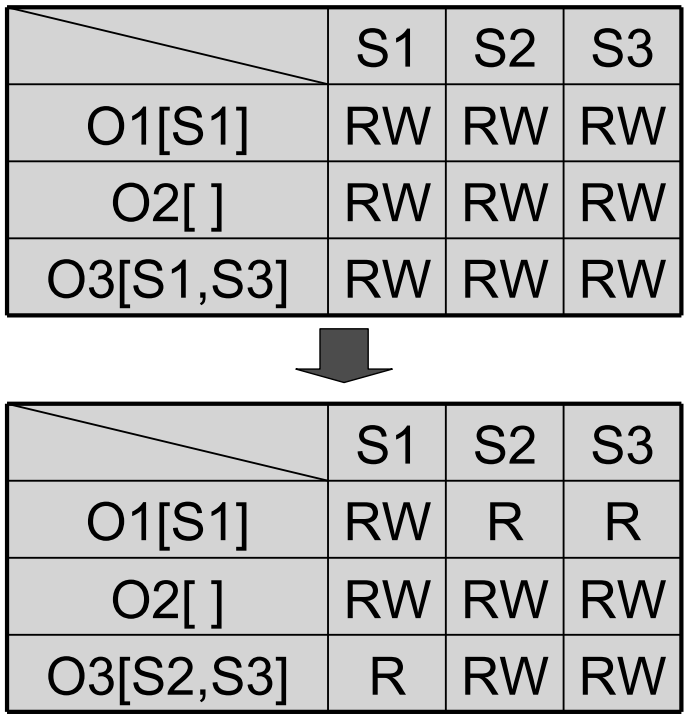


図 2.11: 所有属性によるアクセスルール

2.7.5　役割属性

主体の役割属性に書き込まれたロール (役割) をコミュニティが設定した各ロールの客体へのアクセス設定によって客体へのアクセス権限を変更する。主体の役割が決められていない場合は全て客体への読み書きが許可される。

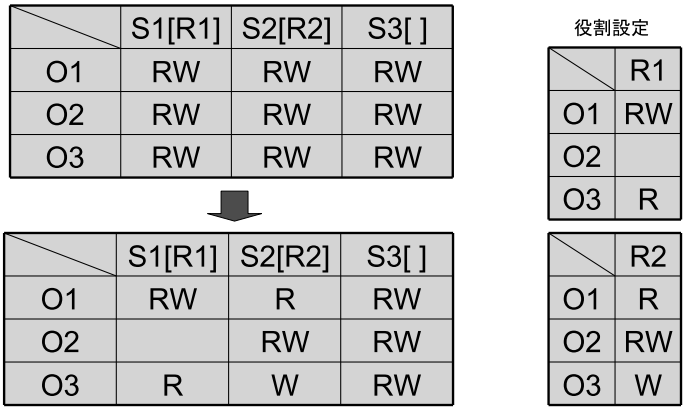


図 2.12: 役割属性によるアクセスルール

第3章　提案方式

3.1　コミュニティー同士の情報漏洩

SNS が構築されコミュニティーが生成されると、次のような情報漏洩が起きる可能性が出てくるものと思われる。図 3.1 において、ユーザーを主体、ユーザの集合をコミュニティ (Community) 、コミュニティ内のプライバシー情報を客体と定義する。ある 2 つの Community 1、2 において、2 つの Community に所属しているユーザー B からコミュニティのプライバシー情報が流出することがある。この場合、ユーザー B が 2 つのコミュニティの、アクセス権を持っているため発生する。この様に、コミュニティー同士でのセキュリティー問題が発生し、何らかの対策が必要となる。この Covert Channel を解消しようとした場合、「Community1 へのアクセスをユーザー A、B 共に禁止する。」とするとき、図 2.6(1) の情報フィルタを適用することでアクセス制御する。また、「Community1、2 に同時に所属できなくする。」とすると、図 2.6(2)、「2 つの Community に対するユーザー A、B どちらかのアクセスを禁止にする。」とすると、図 2.6(3)、「Community にプライバシー情報を書き込まないことを前提に、ユーザー A、B のアクセスを許可する」とすると、図 2.6(4) を適用する。

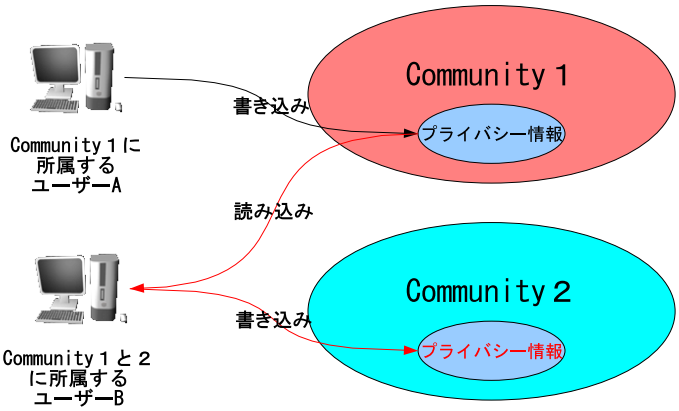
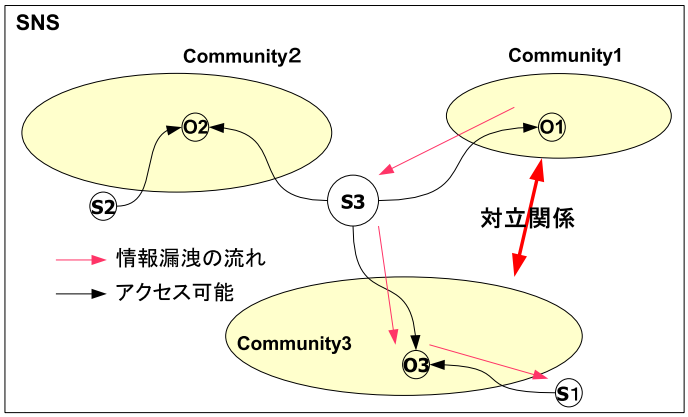


図 3.1: コミュニティ間の情報漏洩

3.2　Covert Channel におけるコミュニティ関係

図 3.2 において、Object をコミュニティの情報、Subject を各コミュニティに所属するユーザーと仮定すると、O1 はコミュニティー 1 のプライバシー情報、O2 はコミュニティー 2 の公開情報、O3 はコミュニティー 3 の公開情報となり S1 はコミュニティー 3 のユーザー、S2 はコミュニティー 2 のユーザー、S3 はコミュニティー 1 と 2 と 3 のユーザーとなる。このとき、コミュニティ 1 と 3 が対立関係にあるとすると、O1 はコミュニティー 1 のプライバシー情報であるため、コミュニティー 3 の S1 には伝わらない。しかし、S3 がコミュニティー 1 での情報を得て，コミュニティー 3 にその情報を漏洩させる可能性がある。このように複数のコミュニティに所属するユーザーから情報漏洩が発生する可能性がある。この場合、S3 への信頼度によって適切な情報フィルタを選択する。



	S_1	S_2	S_3
O_1	φ	φ	R
O_2	φ	R	RW
O_3	R	φ	W

図 3.2: コミュニティ間の関係

3.3　SNS での Covert Channel の制御

ユーザー、情報、コミュニティから構成されるアクセス行列から、その中のフローレベル 2 の Covert Channel を分析した時、それをどう防止するかと言う事を判断する情報フィルタルールによって、SNS の Covert Channel を制御する。以下にそのルールを示す。

[情報フィルタルール]

・S1 と S2 が競合の場合 (3) を選択する。

・S1、S2 の役割によるアクセス制限のある場合、また、役割関係から O1，O2 が S2 によって産出される場合、(3) を選択する。

・役割関係から O1，O2 が S2 によって選択され、かつ、S1 が S2 を Read する必要がある場合、(1) を選択し、更に、S2”を仮想的に追加設定する。S2”は O1 を Read、S2 は O1 をφとする。ただし、S2 と S2”は同一主体であるとする。

・O1 が S2 のプライベート情報の場合、(3) を選択する。

・O1 が S2 の所有の刻印がしてあるならば、他の Object に書き込んではならない。かつその Object に書き込んではならない。

[セキュリティレベル]

・S1 ＜ S2 のとき (3) を選択する。

・S1 ＞ S2 のとき (4) を選択する。

・O1 ＜ O2 のとき (3) を選択する。もしくは (2) を選択する。

・O1 ＞ O2 のとき (2) を選択する。もしくは (3) を選択する。

・S1 ＜ S2 かつ O1 ＜ O2 のとき (3) を選択する。または (4) を選択する。（O2 にアクセス出来るなら O1 にもアクセス可能）

・S1 ＜ S2 かつ O1 ＞ O2 のとき (3) を選択する。

・S1 ＞ S2 かつ O1 ＜ O2 のとき (4) を選択する。（S2 がアクセス出来るなら S1 もアクセス可能、O2 にアクセス出来るなら O1 にもアクセス可能）

・S1 ＞ S2 かつ O1 ＞ O2 のとき (4) を選択する。（S2 がアクセス出来るなら S1 もアクセス可能）

・いずれにも該当しなければ (4) を選択する

[SNS での情報フィルタ]

複数のコミュニティに所属するユーザーからコミュニティー内のプライバシー情報を漏洩しないようにするため、コミュニティの情報フィルタによるアクセス制御を行う。閲覧者を制限した場合、情報フィルタは、情報がプライバシー情報であるか判定し、オープンな状態では見れない様に遮断する。一定の制約条件を設定し、それを情報フィルタが権限を与えた人のみにその情報を閲覧できるようにする。情報フィルタが信頼できるユーザーに対してのみ閲覧を許可する。もしくはコミュニティの管理者が、それを判定する。

3.4　アクセスルールの適用

各コミュニティにおいて、それぞれ内部で Covert Channel 分析を行い、Covert Channel が起きないように制御する。外部のコミュニティとの通信時における Covert Channel 分析は、自コミュニティ と、アクセスしたユーザーおよびその要求について分析を行う。分析は、主体や客体の関係をアクセス行列によって表現し、Covet Channel の検出をする。このとき、アクセス行列の中から 2 × 2 の行列のパターンを全て取り出し、フローレベル 2 の Covert Channel を分析する。

第4章　適用結果

4.1　適用例

SNS において Community Based Access Control をアクセス制御システムとして構築するにあたり、情報フィルタに於ける Covert Channel を制御する。図 4.1 はこの実験で用いた主体 (Subject)、客体 (Object)、コミュニティ (Community) の関係である。

4.2　OpenPNE による CovertChannel 制御

情報フィルタにより、２段階の制御を行う。図 4.1 の場合、Subject をユーザー、Object を情報と設定する。最初に各コミュニティおける信頼度を情報フィルタにて評価し、ユーザーを信頼度の高い順から A、B、C に場合分けを行う。次に、このコミュニティにおけるフレンドかどうか判定する

(1)A ならば個人情報、公開情報共に閲覧可能。

(2)B ならば公開情報のみ閲覧可能。

(3)C ならばアクセス制限を行う。

3 通りの場合分けを行いアクセス制御を行う。また、このコミュニティにおける管理者は個人情報及び公開情報の設定を変えることにより、2 つの情報を自由に公開可能。

次に (1) 及び管理者による個人情報を公開したユーザーに対し、CovertChannel 制御を行う。Subject によって刻印された Object は情報フィルタによって属性を付加される。このコミュニティとユーザー及びユーザーの所属するコミュニティの関係から、フローレベル 2 の Covert Channel を分析し、情報フィルタによって制限する。

・コミュニティ間が競合関係にあるとき、 ユーザーはコミュニティへのアクセスを禁止される。

・付加された属性によって、アクセス制限を実行する。

図 4.2 の場合、管理者によって刻印された情報 (個人情報）は情報フィルタによってプライベート属性を与えられ、管理者以外のアクセスを制限する。同様に、情報

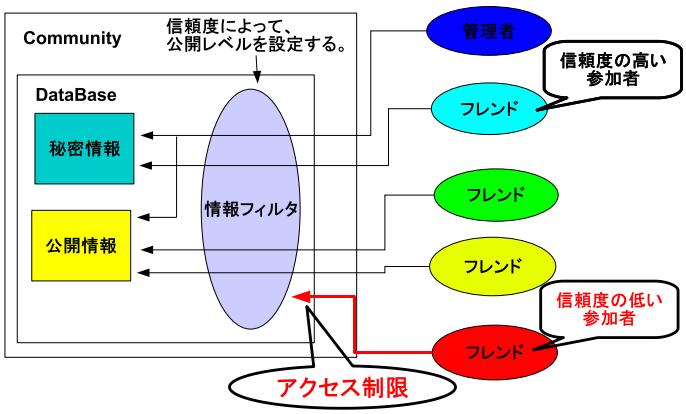


図 4.1: OpenPNE における情報フィルタ 1

(Web 日記) は、管理者の所有属性を付加されているため、書き込みはできないが、読み込みは実行できる。また、情報 (提示板) は属性が付加されてないため、誰でも自由に読み込み、書き込みができる。

以上のアクセスルールにより SNS の Covert Channel を制御した。ただし、これらの操作はコミュニティの管理者によって、その制限を解除できる。

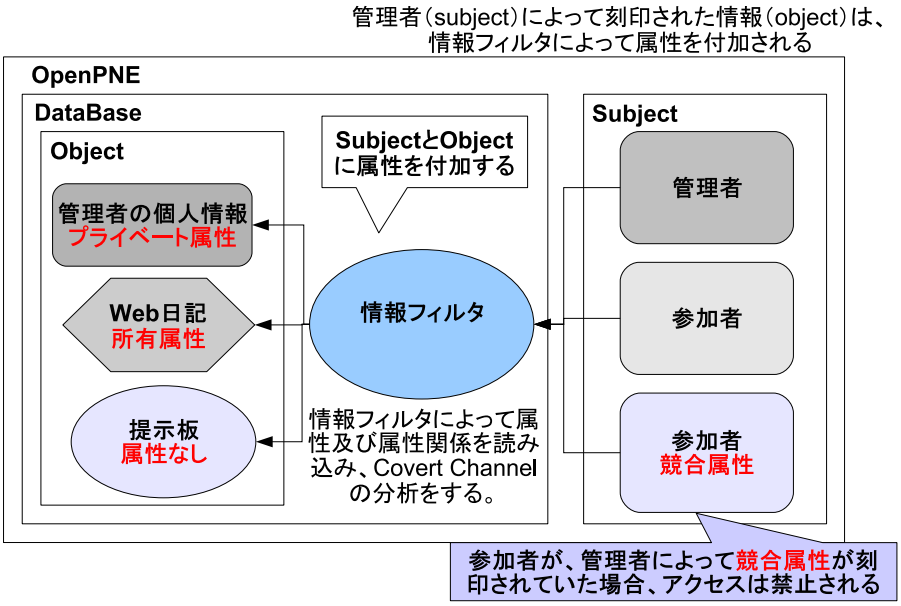


図 4.2: OpenPNE における情報フィルタ 2

第5章　結論

本研究ではSNS コミュニティーにおける限定的な情報漏洩の状況を想定しOpenPNE の情報フィルタにおける Cover Channel 制御を提案した。また、信頼度による情報フィルタの実装と、主体と客体の属性に基づく情報フィルタの提案をした。これによって、SNS の Covert Channel の情報流出を防止する情報フィルタに近づいたといえる。しかし、多くの実験を行ったわけではなく、あくまで本研究は SNS における Community Based Access Control Model の適用を提案しただけである。また、問題があった場合は修正し完成を目指す。しかしながら、過度にユーザーの制限を行うと SNS の自由さが失われ、成り立たなくなる可能性がある。

謝辞

本研究を行なうにあたり，終始熱心に御指導していただいた木下宏揚教授に心から感謝致します．また，様々な面で数多くの有益な御助言をしていただいた東洋ネットワークシステムズ株式会社の森住哲也氏に深く感謝致します．さらに，公私にわたり良き研究生活を送らせていただいた木下研究室の方々に感謝致します．

参考文献

関連図書

[1] ”IT 用語辞典”＜ http://e-words.jp/w/SNS.html ＞

[2] ”株式会社ミクシィ”＜ http://mixi.jp/＞

[3] ”mixi、ユーザ数が1500万人を突破” ＜http://www.rbbtoday.com/news/20080714/52748.h
＞

[4] ”Wikipedia”＜ http://ja.wikipedia.org/wiki/Mixi ＞

[5] ”Wikipedia” ＜ http://ja.wikipedia.org/wiki/SNS ＞

[6] ”OpenPNE”＜ http://trac.openpne.jp/wiki/＞

[7] ”グリー株式会社”＜ http://www.gree.jp ＞

[8] ”Yahoo!Days”＜ http://days.yahoo.co.jp/＞

[9] 佐々木 俊尚,，原田 和英，保田 隆明，齊藤 和生，田口 和裕，平山亜佐子,「シナトラ千代子」管理人，松永 英明，園田 道夫，寺本 秀雄
”SNS の研究”，株式会社翔泳社 (2007)

[10] 小松充史:“ Covert Channel 分析制御のための推論を導入した情報フィルタに関する研究 ”,2006 年度神奈川大学修士論文.

[11] 小川晃夫, 南大沢ブロードバンド研究会 :”OpenPNE で作る！最強の SNS サイト” , 株式会社ソーテック社

[12] 森住哲也, 牛頭靖幸, 稲積泰宏, 木下宏揚 :“ Covert Channel 分析評価のための Access Control Agent System の提案 ”，日本セキュリティ・マネジメント学会誌，第 18 号， pp.30-43，(2005-3).

[13] 森住哲也, 辻井重男, 木下宏揚 :“ 直観主義論理によるダイナミックなアクセス制御の記述 ”,SCIS2008,

[14] 酒井剛典, 森住哲也, 畔上昭司, 小松充史, 稲積泰宏, 木下宏揚 :“ Covert Channel 分析メカニズムと EJB による情報フィルタの構築 ”,2006 年暗号と情報セキュリティシンポジウム, (2006).

[15] 森住哲也，木下宏揚 :“ インターネット社会の情報漏えいを防止するセキュリティモデルの提案（社会システム論と記号論からの着想）”，情報セキュリティ学際シンポジウム, 2005.11, (2005).

[16] 森住哲也, 牛頭靖幸, 畔上昭司, 酒井剛典, 稲積泰宏, 木下宏揚 :“ セマンティック Web システムに於ける“ Community Based Access Control Model ”の適用に関する一考察 ”，SCIS2005,3B1-4, (2005).

[17] 森住哲也, 木下宏揚 :“ 社会システムの中の Covert Channel について ”，技術と社会・倫理研究会，(2005).

[18] 小松充史, 森住哲也, 木下宏揚 :”推論機能を導入した Community Based Access Control の実現”, 2007 年暗号と情報セキュリティシンポジウム, SCIS2007,4E1-2(2007-1).

[19] 森住哲也, 牛頭靖幸, 畔上昭司, 稲積泰宏, 木下宏揚 :“ 機能文化的社会システムの属性に基づくセキュリティモデル ”,(2005).

[20] 森住哲也, 牛頭靖幸, 稲積泰宏, 木下宏揚: “ Covert Channel 分析制御のための Access Control Agent System の提案 ”, コンピュータセキュリティシンポジウム 2004

質疑応答

Q1:セキュリティモデルを適用するのはSNSだけなのか。他のシステムにも応用はしないのか。(豊嶋教授)
A1:本研究は、SNSにおいてのCvert Channelに着目し、SNSのみに効果のある情報フィルタの構築を実行するため、他のシステムには応用はいたしません。