『アクタ関係表に基づくセキュリティ要求分析手法（ＳＡＲＭ）の提案』

金子　朋子†　山本　修一郎‡　田中　英彦†

mgs085510@iisec.ac.jp　yamamotosui@nttdata.co.jp　tanaka@iisec.ac.jp

あらまし　代表的なゴール指向要求工学手法である i*フレームワークのＳＤ図に変換可能で、ｉ＊の表記の複雑さを解消した表記方法として、アクタ関係行列が提唱されている。これを拡張し、アクタ関係表に基づくセキュリティ要求分析手法（ＳＡＲＭ）を提案する。本手法は攻撃と通常のシステム機能との間のセキュリティ上の関係を分析するための要求分析手法であり、開発現場における要求分析の利便性を向上させ、セキュアなシステム開発を実現することを目的としている。攻撃者をアクタに追加し、セキュリティ機能の国際標準であるコモンクライテリア（ISO/IEC15408)を利用して、セキュリティ機能を統一的に表記するものである。

キーワード：セキュリティ要求分析、ｉ＊、ゴール指向、アクタ関係行列、ＣＣ

Proposal of a Security Requirements Analysis Method based on "Actor Relationship Matrix"

Kaneko Tomoko†　Yamamoto Shuichiro‡　Tanaka Hidehiko†

mgs085510@iisec.ac.jp　yamamotosui@nttdata.co.jp　tanaka@iisec.ac.jp

Abstract. The ARM(Actor Relationship Matrix)is a descriptive method which is able to convert SD diagram, and decrease comprehension of the i* framework that is a representative goal-oriented approach. By extension of the ARM, we propose a Security requirements analysis method based on the Actor Relationship Matrix (SARM). This is a method to analyze relationship between attacks and normal system functions. The purpose of this method is to improve requirements analysis at system development, then to realize secure system development by adding attackers for actors and by using common criteria (ISO/IEC15408)which is International standard of security function.

Keywords： Security Analysis, i*, Goal oriented, Actor relationship Matrix、ＣＣ

1．本研究の背景

1-1．セキュリティ要求の特徴

ソフトウェアのシステム開発において、お客様の要求を適切に把握し、実現させることは非常に大切なことである。しかし、上流工程における要求分析が不十分であるためにシステム開発に重大な影響を及ぼすことは多い。一般の機能要求でもお客様と開発側には、システムへの要求や視点が異なり、両者のギャップにより、開発上の問題は起きやすい。つまりお客様は早く、安く、品質の良いシステムを求めるのに対し、開発者側は利害関係者の合意をとり、ＩＴ技術方式を決定し、スキルのある人材を集めなければならないといった課題を抱え、お客様の要求を変換するときにギャップが生じるからである。セキュリティ要求の場合、このギャップはより大きくなる。セキュリティ要求は、一般機能要求より具体的な要求を受けないのにも関わらず、できていて当然とされる非機能要求だからである。「情報漏洩は困る、内部統制も一緒に考えてほしい。」といった漠然とした要望はあるもののセキュリティ自体の知識にとぼしいのでセキュリティ機能は開発者に任せるというお客様が一般的である。また、開発者側でも、セキュリティの専門家でない限り、「リスクの洗い出しに漏れがないかは不明、リスク対策の費用をお客様は負担していただけない。」などの問題を抱えることが多い。

† 情報セキュリティ大学院大学　情報セキュリティ研究科　‡(株)NTT ﾃﾞｰﾀ ｼｽﾃﾑ科学研究所

† Institute of Information Security　‡NTT Data Corporation Reseach Institute for System Science

こうした問題を抱えているにも関わらず、セキュリティ要求に関しては手順も手法も普遍的に確立されたものはないのが実情である。ただし、様々な研究はなされており、その事例をいくつか紹介したい。

1-2. ゴール指向のセキュリティ要求分析手法と事例

セキュリティ要求分析の多くは、一般的な要求分析技術に基づくユースケース、ゴール指向モデリングであり、ミスユースケース、ｉ＊、ＫＡＯＳ、ＮＦＲフレームワークなどが例としてあげられる。

ゴール分析とは開発対象システムの要求が達成すべき目標を分析し定義するための手法である。要求の目標を明確にして顧客と合意することで、設計の妥当性が機能仕様に対して確認できるように、機能要求の妥当性もゴールに対して確認できるようになる特徴をもつ。

文献[1]によるとTondelらは、セキュリティ要求手法に共通する最も基本的な要素として、セキュリティ目標、資産抽出、脅威分析が共通するとしてセキュリティ要求工学の留意点をまとめている。

セキュリティ目標では、①顧客が望む要求であること ②法制度、標準、ポリシーに適合することといった条件を満足するような上位レベルのセキュリティ・ゴールを識別する。資産抽出では、顧客、システム所有者、攻撃者の視点から資産の価値とリスクを判断するために資産ごとに機密性、一貫性、可用性について視点ごとに優先度を判断する。このときリスクを発生確率と影響度で定義することができる。しかしリスクを完全に定量化できないので、高、中、低のような簡易な影響度の評価値も有効である。脅威分析では、最も重要な資産に注力することと脅威カタログを事前に準備することが重要である。

1-3. ｉ＊セキュリティ要求分析

ｉ＊セキュリティ要求分析プロセスでは、セキュリティ上の脅威としての攻撃方法を攻撃者のタスクとして特定し、攻撃者の意図(悪意)をゴールとして特定し、攻撃方法に対する対策を対象となるアクタのタスクとして特定する。以上により攻撃者モデル、悪意モデル、対策モデルなど、ドメイン要求分析で得られたドメインモデルとは別のモデルを作成する[2]。

1-4. ＣＣ(Common Criteria)とＳＴの概要

ＣＣ (Common Criteria)とは国際標準「ISO/IEC 15408 情報技術セキュリティ評価基準(Common Criteria for Information Technology Security Evaluation)」のことで 1999 年 6 月、国際標準に制定された。情報技術に関連した製品のセキュリティの度合いを系統的に評価できるようにするための国際規格である。2000 年 7 月には JIS 標準(JIS X 5070)として制定されている。ユーザにとってはセキュリティレベルに応じた適正なコストで、安全な製品やシステムを導入可能な指標であり、開発者にとっては、安全な製品やシステムを開発するための指標である。

ＣＣはパート１「概説と一般モデル」、パート２「セキュリティ機能要件」、パート３「セキュリティ保証要件」の３つのパートより構成され、機能要件、保障要件がカタログ化されている。共通評価手法（ＣＥＭ）にはＥＡＬ５までの評価手法が定義されている。ＣＣパート２のセキュリティ機能要件（ＳＴ）は 11 クラスに分類されたセキュリティ機能要件のカタログがあり、アクセス制御・情報フロー制御、暗号化、識別認証、監査等の「脅威に対抗するためのセキュリティ機能の要件」が列挙されている。

1-5. i*とＣＣの手法の組み合わせ i*(Liu 手法)による ST 作成

ｉ＊(Liu 手法)により作成したモデルを、CC (Common Criteria)のＳＴ（セキュリティ目標）との対応関係に従ってマッピングし、自然言語により説明する事例を紹介する。ゴール指向ベースのセキュリティ要求分析手法により、ＳＴのほぼ全体を作成することができる。そこでセキュリティ機能要件に当たるゴールやタスクは、ＣＣパー

ト２のレベルまで詳細化して記述する手法が提案されている[3]。

クロスサイトスクリプティング（ＸＳＳ）の事例を本手法で作成した。ＸＳＳとはソフトウェアのセキュリティホールの一つで、Web サイトの訪問者の入力をそのまま画面に表示する掲示板などのプログラムが、悪意のあるコードを訪問者のブラウザに送ってしまう脆弱性のことである。

本事例は、利用者ＡＬＩＣＥが脆弱性のあるサイトＢＯＢにアクセスし、ログインした後に、攻撃者ＥＶＥのサイトをクリックしてしまうと、ＢＯＢのサイトのＸＳＳに対する脆弱性が起因となり、ＡＬＩＣＥはＥＶＥに自分のＣＯＯＫＩＥ情報を盗まれてしまう。そして攻撃者ＥＶＥはＡＬＩＣＥになりすましてＢＯＢのサイトに入ることができるというケースである。

ｉ＊フレームワークとＣＣの組み合わせ手法によって ＸＳＳ（クロスサイトスクリプティング）を記述した例を付図１に示す。

1-7. ゴール指向とｉ＊の課題

文献[4]ではｉ＊について以下の課題を指摘している。

(1)アクタの数が多い場合、アクタ間の関係が複雑で網羅性がｉ＊フレームワークの図だけでは効率的に確認できない。

(2)アクタの置かれたシーンによってアクタが抱える問題やそれに伴うゴール、ソフトゴール、資源が変化する可能性がある。すなわちある特定のシーンにおけるアクタと意図の関係だけを表現するｉ＊フレームワークではこれらの変化を記述できないことを指摘している。

事例を作成し、ｉ＊に対するこれらの課題を確認した結果、①図が複雑でわかりづらい②要求の洗い出しの完全性が検証できない③セキュリティ上の経験知が必要④記述が冗長という問題点を発見した。①に関して、この事例は脆弱性のあるサイトにアクセスをするだけの比較的簡単な事例であるにも関わらず、一般アクタと攻撃者、ゴール、ソフトゴール、リソース、タスクが１つの図の中に複雑に表されることになった。②に関して、アクタ間の相互関係もゴールもソフトゴールも資源も作成者が思いついた限りを列挙するだけで、これで全て抽出されているのか不明である。③に関して、この手法では対策として必要な機能をＣＣのＳＴで表現しているが、モデルの作成者にセキュリティ知識がない場合、攻撃者に対する対策はたてられない。④に関して、ｉ＊の記法では同じゴール等であっても他のアクタとの関係を示すゴール等をアクタ内のゴール等とは別に記載する必要があり、冗長性があり、図を複雑にしている。

２．アクタ関係行列

2-1. アクタ関係行列ＡＲＭとは

ｉ＊フレームワークはアクタ間のゴール、ソフトゴール、資源、タスクの依存関係を表現する有効な手法だが、既述の課題をはらんでいる。そこでアクタが置かれる問題状況ごとにアクタ間の関係を行列で定義する方法を用いて、網羅的にｉ＊のＳＤ図を作成する方法として、アクタ関係行列ＡＲＭが提唱されている[4] [5] [6]。ＡＲＭはゴールと意図としてのソフトゴール、資源、タスクを整理したものである。ＡＲＭ(Ai,Bj)には受益者 Ai から提供者 Bj への意図Ｉij を記入する、また対角要素ＡＲＭ(Ai,Aj)にはアクタの内部ゴール Gi を記入する。

2-2. ＡＲＭの有用性

ＡＲＭには次の３つの有用性がある。

(1) アクタ間の関係の網羅性を確認できる。
(2) アクタの関連性の類似性がわかる。
(3) 登場しつつも他のアクタと関連性がないアクタを行列の疎の部分として見極めることができる。

３．アクタ関係表に基づくセキュリティ要求分析手法（ＳＡＲＭ）の提案

3-1. ＳＡＲＭのねらい

従来、設計レベルでセキュリティの脆弱性を議論していることが多かった。しかし、

非機能要求として、お客様と開発側にギャプが生じやすいセキュリティ機能は要求分析段階から検討をすることが必要である。セキュリティ要求分析に際し、前述のようにアクタ関係表はｉ＊などの従来の要求分析手法より優れた特長を備えており、これをセキュリティ要求分析に適用した『アクタ関係表に基づくセキュリティ要求分析手法（ＳＡＲＭ)』を提案する。

ＳＡＲＭのねらいを以下に示す。

(1)ＡＲＭの網羅性の高いアクタ分析に攻撃者の存在を追加して検討することにより、網羅性の高いセキュリティ要求分析を実施する。

(2)攻撃シーンと守るべき資源（アセット）との組み合わせを検討することにより、セキュアなシステムを実現する。攻撃にはある程度既知のパターンがあるのでWebシステムの攻撃パターンなどを適用してある程度定形的な分析が可能になる。

(3)セキュリティに関して専門知識が必要とされるため、一般の要求分析工程、設計工程は一般のシステム開発者が担当し、脅威分析とセキュリティ機能分析は専門家が別途実施する方式が現実的である。そこで、一定の様式上で通常機能と攻撃を分析できる手順とする。

3-2. 対象者、メリット

ＳＡＲＭはシステム開発の上流工程において設計者が作成し、ユーザとの要求仕様の洗い出しに使用することを想定している。

ＳＡＲＭのメリットは、セキュリティ要求分析を攻撃シーンごとに、攻撃者を含むアクタ間のゴール、意図としてのソフトゴール、資源、タスクの依存関係を高い網羅性をもって、利便性の高い表形式で実施できることである。

3-3. ＳＡＲＭの作成方法

ＳＡＲＭには、(1)ＡＡ（Asset×Attack）表(2) ＳＡＲＭ＿Ａ表が存在する。

(1)ＡＡ（Asset×Attack）表の作成方法

ＡＡ表を用いて攻撃シーンごとに守るべきアセットを特定する表である。アセットと攻撃の組み合わせに応じて、よりセキュアなシステムを実現できる。表１にＡＡ表を示す。

表１． ＡＡ表

	ｱｾｯﾄA	ｱｾｯﾄB	ｱｾｯﾄC
攻撃１	○		○
攻撃２		○	○
攻撃３	○		
攻撃ｎ	○	○	

→SARM作成単位

(2) ＳＡＲＭ＿Ａ(Attack)表の作成方法

ＳＡＲＭ＿Ａ表はＡＡ表で抽出した攻撃シーン単位で作成し、一般アクタに、攻撃者を追加したアクタ間係行列である。ＳＡＲＭ＿Ａ表では、一般アクタのマークを□：資源情報、○：期待としてのゴール、☆：ソフトゴール、◇:タスク、機能で表現し、攻撃者のマークを■：攻撃者の資源情報、●：攻撃者の期待としてのゴール、★：攻撃者のソフトゴール、◆:攻撃者のタスク、機能で表現する。表２にＳＡＲＭ＿Ａ表を示す。

表２． ＳＡＲＭ＿Ａ表

受益者＼提供者	一般者A	一般者B	攻撃者C
一般者A	○Gab,Gac	○Iab	○Iac
一般者B	○Iba	○Gba,Gbc	○Ibc
攻撃者C	●Ica	●Icb	●Gca,Gcb

Gij:アクタｉのｊに対する内部ゴール

Iij:アクタｉのｊに対するゴール、タスク、資源

ＳＡＲＭの作成手順を以下に示す。

①一般の各アクタ間の要求分析をアクタ関係表で実施（通常のシステム開発者が担当）

②ＡＡ（Asset×Attack）表で守るべき資源と攻撃パターンを洗い出す。(セキュリティ専門家が実施)

②攻撃者を加えたセキュリティの要求分析をＳＡＲＭ＿Ａ表で実施する。(セキュリティ専門家が追記)

③セキュリティ機能設計分析をＣＣのＳＴ単位で実施する。(セキュリティ専門家が実

施）

表３に図１で示したＸＳＳに対応するｉ＊フレームワークとＣＣの組み合わせ手法の事例に対応するＳＡＲＭ__Ａ表を示す。

４．考察

ｉ＊の事例で列挙した問題点を、ＳＡＲＭでは、以下のように解決できると考える。①図が複雑でわかりづらい問題に関しては、ＡＡ表もＳＡＲＭ__Ａ表も単なる表であり、直交するアクタ間の関係をマークで示しているだけなので関係性の把握は容易である。また、一般アクタのマークを白、攻撃者のマークを黒で統一しているので、一般機能要求とセキュリティ要求の区別がしやすい。②要求の洗い出しの完全性が検証できない問題に関しては、ＡＡ表でアセットと攻撃の組み合わせ検証を繰り返すごとに網羅性は向上するので、必要なレベルに応じて守るべきアセットの完全性は確保できる。また、ＳＡＲＭ__Ａ表では、攻撃者を含めたアクタ間の関係を表形式で埋めていくため、関係が疎な欄にも着目しやすく、完全性は向上する。③セキュリティ上の経験知がないと作成できない問題点に関しては、現在のＳＡＲＭでもセキュリティの知識が不可欠であり、同様な問題を抱えている。ただし、提案したように一般の要求分析工程、設計工程は一般のシステム開発者が担当し、脅威分析とセキュリティ機能分析は専門家が別途実施する方式をとりやすい工夫をしているため、分業による問題解決が可能である。④記述が冗長という問題点に関しては、ＳＡＲＭは同じゴール等を複数、記述する必要はないので、冗長性を省ける。

５.今後の課題

ＣＣ（コモンクライテリア）のＳＴは要求レベルの記述であり、ＣＣに定義されたセキュリティ機能とＳＡＲＭ__Ａ 表の各項目との対応付けをスムーズに行う手法の検討が必要である。今後はＣＣに定義されたセキュリティ機能ごとの要求分析を実施する方法を検討し、設計工程へつなぐことができる手法に拡張したいと考えている。

表３．　ＳＡＲＭ__Ａ表　ＸＳＳ（ｸﾛｽｻｲﾄｽｸﾘﾌﾟﾃｨﾝｸﾞ）具体例

	利用者 ALICE	攻撃者 EVE	脆弱性のあるシステムＢＯＢ
利用者 ALICE	○利用者情報を取得する ☆色々なサイトを閲覧したい □COOKIE 情報	☆EVE のサイトを閲覧したい ◇BOB のサイトにログインしたままＥＶＥのサイトをクリックする	☆BOB のサイトを閲覧したい ◇BOB のサイトのトップページにアクセスする ◇ログインをする
攻撃者 EVE	★ALICE に EVE のサイトをクリックさせたい ◆利用者 ALICE に EVE のサイトをクリックさせる	★ALICE のｾｯｼｮﾝ ID をＧＥＴしたい ◆Cookie 情報窃盗スクリプトを起動 ◆Cookie 情報内のｾｯｼｮﾝ ID を利用する □COOKIE 情報	★ALICE になりすまして BOB のシステムにアクセスしたい ◆窃盗した ALICE のｾｯｼｮﾝ ID で不正アクセスする
脆弱性のあるシステム BOB	☆正当な利用者にアクセスさせる	☆正当ではない利用者にはアクセスさせない ◇偽造された認証データの使用を検知または拒否する	○利用者情報を保護する ○利用者ごとの管理をする ◇許可された利用者に情報へのアクセスを許可する ◇ALICE にセッションＩＤを割り当て COOKIE 情報を通知する ◇COOKIE 情報を管理する ◆ＸＳＳの脆弱性により ALICE の Cookie 情報をＥＶＥに送信 □利用者情報 □COOKIE 情報 □認証データ ■ＸＳＳの脆弱性

一般アクタのマーク　□：資源情報、○：期待としてのゴール、☆：ソフトゴール、◇：タスク、機能

攻撃者のマーク　　■：攻撃者または脆弱性のある資源情報、●：攻撃者の期待としてのゴール、

★：攻撃者のソフトゴール、◆：攻撃者のタスク、機能

参考文献

[1]エンタープライズ ICT 総合誌　月刊ビジネスコミュニケーション　山本修一郎 連載第 41 回セキュリティ要求工学 http://www.bcm.co.jp/site/youkyu/youkyu41.html

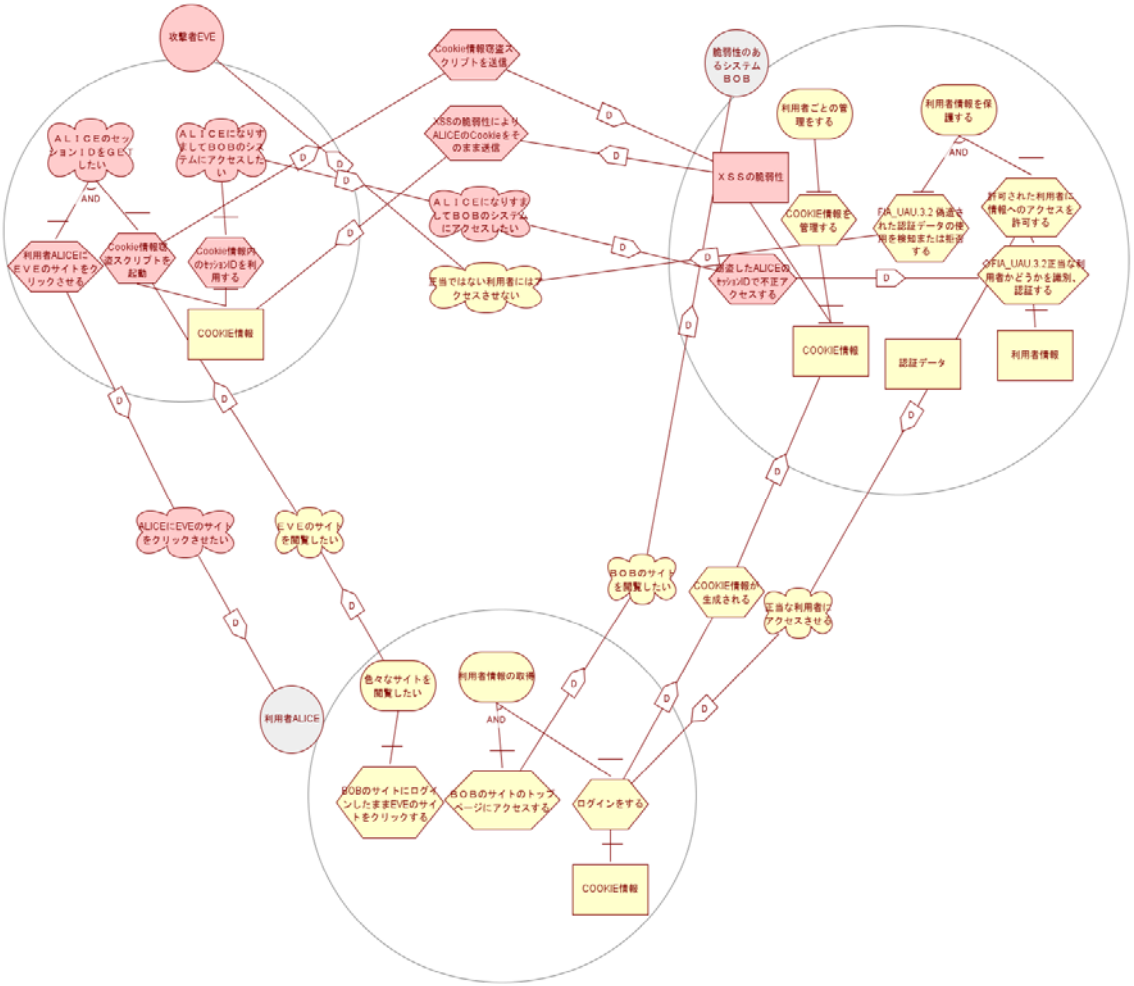
[2] Liu 他, “Security and Privacy Requirements Analysis within a Social Setting”, RE2003

[3] K. Taguchi, Y. Tahara, “Curriculum Design and Methodologies for Security Requirements Analysis”, Progress in Informatics, No. 5, pp. 19-34,(2008)

[4]アクタ関係行列を用いた i スターフレームワーク作成方法の提案　井部己文, 山本修一郎, 佐藤友合子

[5]エンタープライズ ICT 総合誌　月刊ビジネスコミュニケーション　山本修一郎 連載第 39 回アクタ関係分析 //www.bcm.co.jp/site/youkyu/youkyu39.html

[6] Actor Relationship Analysis for the i＊ Framework, Shuichiro Yamamoto1, Komon Ibe1, June Verner2, Karl Cox2, and Steven Bleistein2



付図１．ｉ＊フレームワークとＣＣの組み合わせ手法　ＸＳＳ（ｸﾛｽｻｲﾄｽｸﾘﾌﾟﾃｨﾝｸﾞ）具体例