情報処理学会論文誌 Vol. 52 No. 9 2775–2787 (Sep. 2011)

アクタ関係表に基づくセキュリティ要求分析手法（SARM）を用いたスパイラルレビューの提案

金子朋子[†1] 山本修一郎[†2] 田中英彦[†3]

代表的なゴール指向要求工学手法である i* 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案されている．本論文ではこれをセキュリティ対応に拡張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．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，攻撃者をアクタに加えることによって，セキュアなシステム開発を実現することを目的とする．本手法は i* に対してセキュリティ対応をしている i*-Liu 法で作成するモデルを表現可能であり，以下の利点を持つ．① 表形式で作成しやすい，② 表形式で関係性を検証していくのでアクタ間の関係を網羅しやすい，③ タスク間の関係を AND と OR の構造化で表現し，要求を論理的に表現できる．さらに SARM の網羅性の高さと i*-Liu 法の一覧性の良さを生かした効果的な使用方法として，SARM と i*-Liu 法の双方を用いたスパイラルレビューを提案する．

A Spiral Review Method for Security Requirements Based on “Actor Relationship Matrix”

Tomoko Kaneko,[†1] Shuichiro Yamamoto[†2] and Hidehiko Tanaka[†3]

Actor Relationship Matrix (ARM) is a descriptive method which is able to generate SD diagram, and increase understandability of the i* that is a representative goal-oriented approach. As an extension of the ARM, we propose a security requirements analysis method based on the Actor Relationship Matrix (SARM). The proposed method which analyzes the relationship between attacks and normal system functions is able to improve requirements analysis at system development, then to realize secure system development by considering attackers for actors. This method is also able to convert ARM into SD diagram of i* of Liu method for security, and has the following benefits: 1) Easy to create because of tabular form. 2) Easy to improve the integrity among related actors, by verifying the relationship in tabular form. 3) Easy to express logically by AND/OR relationship between tasks. We propose a spiral review process based on SARM and i* framework of Liu method as a combination of these methods as well.

1. はじめに

ソフトウェアのシステム開発において，顧客の要求を適切に把握して実現することが重要である．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，顧客と開発側では，システムへの要求や視点が異なり，開発上の問題は起きやすい．セキュリティはシステムの機能に明確には関係しないような非機能要求の 1 つとされる[1)]．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要求は開発者に任せるという顧客が一般的である．また，開発者側でもセキュリティの専門家ではないことが多いので，セキュリティ要求を仕様に入れるのは難しい．こうした問題をかかえているため，セキュリティ要求に関して様々な研究がなされている[2)]．しかしながら，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．

以下では，まず関連研究について 2 章で述べる．3 章でセキュリティ要求分析手法 SARM（Security requirements based on Actor Relationship Matrix）とそれに基づくスパイラルレビュー手法を提案する．4 章では SARM と i*-Liu 法[3)] の等価性を証明する．5 章では提案手法の評価を行い，6 章で全体を通した評価を述べ，最後に 7 章で結論と今後の課題をまとめる．

2. 関連研究

2.1 セキュリティ要求分析手法

ゴール分析に着目した一般的な要求分析手法には i*[4)]，KAOS[5)]，ARM（Actor Relationship Matrix）[6)–8)] がある．なかでも i* はステークホルダやシステムをアクタと

†1 株式会社 NTT データ
NTT DATA CORPORATION
†2 名古屋大学
Nagoya University
†3 情報セキュリティ大学院大学
Institute of Information Security

© 2011 Information Processing Society of Japan

してモデル化し，利害関係を，アクタ間の依存関係としてモデル化する有効な手法として期待されている．アクタ関係行列 ARM は，アクタがお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である．アクタの数が多い場合，アクタの関係が複雑で，要求を網羅しているかどうかを i* の図では効率的に確認できない，アクタの置かれたシーンによってアクタがかかえる問題や，それにともなうゴール，ソフトゴール，資源が変化する場合，i* では，これらの変化を記述できないという課題がある．これらの欠点を改善し，網羅的に i* の SD 図を作成する方法として，ARM が提唱されている．

セキュリティ要求分析では，一般的な要求分析に加えて攻撃者の存在を考慮しなければならない．セキュリティ要求はアセットに対する脅威とその対策の記述だからである．セキュリティ要求分析の代表的な手法にミスユースケース[9]，NFR フレームワーク[10]，Secure Tropos[11]，i*-Liu 法[3),12)]，問題フレーム[13]，Abuse Frames[14] などがある．いずれの手法もセキュリティを考慮した手法だが，明示されない非機能要求に関してあらゆる要件をつくすことは難しく，セキュリティ要求に関する網羅性は考慮されていないのが実情である．

2.2　i*-Liu　法

ゴール指向要求分析のための i* 手法をセキュリティ要求分析が実施できるように拡張した手法が i*-Liu 法である．i* の SD（Strategic Dependency）図では，ビジネスを構成するアクタ間の依存関係を現状分析（as-is）する．これに対して i* の SR（Strategic Rational）図は，各アクタ内部で，アクタがソフトゴールを達成するためのゴール，タスク，リソースとの階層的な関係を分析できる．

i*-Liu 法では**表 1** に示すように，i* の SD 図と SR 図の分析に加えて攻撃者，悪意，脆弱性，攻撃方法とその対策を分析できる．分析の方法は i* の図式要素に加えて，攻撃者をアクタとして識別する．攻撃者は対応するアクタと同じように，ゴール・ソフトゴール・タスク・リソースを持つ．なお，i*-Liu 法の具体的な記述方法は 4 章で説明する．

表 1　i* と i*-Liu 法の持つモデルの比較表（○：有，—：無）
Table 1　Model comparison between i* and i*-Liu method (○: presence, —: none).

モデル	i*	i*-Liu法
SD	○	○
SR	○	○
攻撃者	—	○　攻撃アクタ
悪意	—	○　悪意のソフトゴール
脆弱性	—	○　攻撃対象となるリソース
攻撃方法	—	○　攻撃タスク
対策	—	○　タスク

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

前述のように ARM は i* の課題を克服する手法である．しかしセキュリティ対応にはなっていない．そこで ARM 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[15)]．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）との組合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 STRIDE[16)] などの分類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の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セキュリティ担当者が別途実施する方式が現実的である[17)]．そこで，同じ表現形式を用いて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，利便性の高い表形式で，攻撃シーンごとに攻撃者を含むアクタ間の依存関係を網羅性高く分析できることである．網羅性とはあらゆる脅威を考えることだが，SARM においては，表 1 に示す ① AA 表によって抽出された攻撃パターン単位で攻撃者を加える場合の網羅性と，② 攻撃者を加えた後，他のアクタとの関係で場合を尽くす網羅性の 2 段階に分けて，網羅性を向上させている．また，セキュリティと機能要件の両方の分析を 1 つの様式で行えるメリットもあわせ持っている．

また ARM は i* の SD 図とは変換可能であるが，SR 図との対応は考慮していない．これに対して SARM はタスク分解，手段目的分解を表現可能であり，SD 図の範囲を超えて SR 図への対応ができる．

3.3　SARM の作成方法

SARM では，まず (1) 資源に対して想定される攻撃方法を AA（Asset × Attack）表を用いて記述し，次いで (2) AA 表で特定した各攻撃方法に基づくアクタへの攻撃状況をアクタ間の関係として SARM 状況表で記述する．

© 2011 Information Processing Society of Japan

3.4　AA（Asset × Attack）表

AA 表では攻撃シーンごとに守るべきアセットを特定することを目的として，守るべきアセットと攻撃の組合せを整理する．要求定義工程で利用するレベルに絞り込むため，AA 表を用いて STRIDE[16] 単位で攻撃と守るべきアセットを特定できるようにする必要がある．STRIDE モデルは Spoofing（なりすまし），Tampering（改ざん），Repudiation（否認），Information disclosure（情報の漏えい），Denial of service（サービス拒否），Elevation of privilege（特権の昇格）の 6 種にマイクロソフトが脅威を分類したものである．

一般にセキュリティ要件を検討するときにはどのような攻撃がありうるか不明なところからスタートする．網羅性の向上は要件定義者の知見によることが多いが，セキュリティ知見は専門家に限られる場合も多い．そこでセキュリティ上の網羅性を向上させることを目的として，AA 表では STRIDE モデルを用いて攻撃者による攻撃方法を列挙することとした．STRIDE 単位で SARM 状況表を作成することにより，網羅性が高くかつ細かすぎないセキュリティ要求分析ができる．さらに脅威の詳細化が必要になる場合には，IPA の脅威データベース[18] なども参考にすることができる．

以下では，まず AA 表を定義し，次いで具体例を示す．

[定義] AA 表

AA 表の各要素を次式で定義する．

AA[attack, asset] = もし attack が asset を攻撃するなら，○．
そうでないとき，—

AA[attack, 種別] = 脅威種別を表す S, T, R, I, D, E などの記号

AA[attack, 攻撃内容] = 脅威に対応する具体的な攻撃内容

AA[attack, 状況表] = 対応する SARM 状況表を識別するための ID

[具体例]

攻撃方法の集合 Attack = {なりすましによる不正注文，注文情報の改竄，商品注文への否認，情報の漏えい，システムへの DOS 攻撃など，管理人への権限昇格}

資源の集合 Asset = {COOKIE 情報，セッション ID，パスワード，個人情報} とすると，**表 2** のような AA 表を作成できる．たとえば，

AA[なりすましによる不正注文，COOKIE 情報] = ○

AA[注文情報の改竄，COOKIE 情報] = —

となる．なりすましによる不正注文の場合は，各攻撃ごとの脅威種別は S，対応する SARM 状況表を識別するための ID は，A_S を記入する．

表 2　AA（Asset × Attack）表の具体例
Table 2　Example of AA (Asset × Attack) table.

攻撃方法			攻撃対象資源			
種別	攻撃内容	状況表	COOKIE情報	セッションID	パスワード	個人情報
S	なりすましによる不正注文	A_S	○	○	○	○
T	注文情報の改竄	A_T	−	○	○	○
R	商品注文への否認	A_R	−	−	−	○
I	情報の漏えい	A_I	○	○	○	○
D	システムへのＤｏｓ攻撃等	A_D	○	○	−	−
E	管理人への権限昇格	A_E	○	−	−	−

3.5　SARM 状況表

SARM 状況表は攻撃者，悪意，攻撃方法，脆弱性の特定を行うことを目的として，AA 表で抽出した攻撃状況単位で作成する．SARM 状況表では一般アクタだけでなく，攻撃アクタを追加してアクタ関係行列を作成する．ARM は i* の SD 図を表構造を用いて記述する表記方法である．しかし ARM では，AND/OR などに基づくゴール分解構造を記述することはできなかった．このため，SARM では以下の定義で述べる階層ゴール式を導入することにより，SR 図の内容を表現できるように ARM を拡張した．

[定義] SARM 状況表

SARM[X, Y] = { e:階層ゴール式 }

アクタ X がアクタ Y に対して要求する，ゴール G，ソフトゴール S，タスク T，資源 R としての階層ゴール式の集合を SARM[X, Y] で表す．ゴール G，ソフトゴール S，タスク T，資源 R の各要素に識別番号（ID）を付与して，一意に ID を管理する欄を設定する．また依存先を（to 要素 ID）という形式で示す．ここで，X および Y は，一般アクタか攻撃アクタである．とくに，SARM[X, X] は，アクタ X 自身の目標としてのゴール，ソフトゴール，タスク，資源としての階層ゴール式の集合を表す．なお，＋は項目の選択を示し，“S” とは文字 S をそのまま記述することを示している．

[定義] 階層ゴール式

⟨階層ゴール式⟩::=⟨階層ゴール基本式⟩⟨改行⟩⟨字下げ⟩⟨階層ゴール式⟩ ＋
⟨階層ゴール基本式⟩⟨改行⟩⟨字下げ⟩⟨積階層ゴール式並び⟩ ＋
⟨階層ゴール基本式⟩⟨改行⟩⟨字下げ⟩⟨和階層ゴール式並び⟩ ＋
⟨階層ゴール基本式⟩

⟨積階層ゴール式並び⟩::= “&” ⟨階層ゴール式⟩ ＋
“&” ⟨階層ゴール式⟩⟨改行⟩⟨積階層ゴール式並び⟩

© 2011 Information Processing Society of Japan

⟨和階層ゴール式並び⟩::= “|” ⟨階層ゴール式⟩ +
　“|” ⟨階層ゴール式⟩⟨改行⟩⟨和階層ゴール式並び⟩
⟨階層ゴール基本式⟩::=
　⟨ゴール⟩ + ⟨ソフトゴール⟩ + ⟨タスク⟩ + ⟨資源⟩ + ⟨攻撃ゴール⟩ +
　⟨攻撃ソフトゴール⟩ + ⟨攻撃タスク⟩ + ⟨攻撃資源⟩
⟨ゴール⟩::= “◯” ⟨ゴール名⟩
⟨ソフトゴール⟩::= “☆” ⟨ソフトゴール名⟩
⟨タスク⟩::= “◇” ⟨タスク名⟩
⟨資源⟩::= “□” ⟨資源名⟩
⟨攻撃ゴール⟩::= “●” ⟨脆弱性⟩⟨攻撃ゴール名⟩
⟨攻撃ソフトゴール⟩::= “★” ⟨脆弱性⟩⟨攻撃ソフトゴール名⟩
⟨攻撃タスク⟩::= “◆” ⟨脆弱性⟩⟨攻撃タスク名⟩
⟨攻撃資源⟩::= “■” ⟨脆弱性⟩⟨攻撃資源名⟩
⟨脆弱性⟩::= “− −” + “−” + “ ”

この表記を用いて作成したなりすまし（A_S）に対する状況表を**表 3** に示す．たとえば，攻撃者 EVE のゴールが対角行列で表現されている．★ALICE になりすまし，商品を手に入れたいという S8 に依存する意図のもとに，&◆利用者 ALICE に EVE のサイトをクリックさせるという S1 に依存するタスクと &◆COOKIE 情報内のセッション ID を利用するという 2 つのタスクを実施するので，& が前置される．COOKIE 情報という資源が攻撃者のゴール，ソフトゴールに対してどの程度の脆弱性を持つかを i*-Liu 法に準じて − −，−，未記入からなる 3 段階の接頭辞によって明記する．

表 3　AA 表のなりすまし（A_S）を作成単位とした第 1 種 SARM 状況表の具体例
Table 3　Example of 1st type SARM table which represents AA table through spoofing (A_S) as the unit.

	利用者ALICE　A1		脆弱性のあるシステムBOB　A2		攻撃者EVE　A3	
利用者ALICE A1	S4	☆色々なサイトを閲覧したい (to S3),(to S5)	S5	☆BOBのサイトを閲覧したい(to A2)	S3	☆EVEのサイトを閲覧したい(to S2)
	G6	&◯必要な情報を入力する				
	T3	&◇ログインをする				
脆弱性のあるシステムBOB A2	S6	☆正当な利用者にアクセスさせる(to G6)	G1	○webページを生成する	S7	☆正当ではない利用者にはアクセスさせない(to S2)
			G2	○利用者情報に従って情報を提供する		
			G3	○利用者情報を保護する		
			R2	□利用者情報		
			G4	○利用者ごとの管理をする		
			T4	&◇許可された利用者に情報へのアクセスを許可する(to S6)(to S7)		
			R3	□認証データ		
			T5	&◇COOKIE情報を管理する		
			G5	○商品の販売をする		
			T6	◆ALICEになりすまして、商品の注文をする		
攻撃者EVE A3	S1	★ALICEにEVEのサイトをクリックさせたい(to A1)	S8	★BOBのシステムで商品を注文し、商品を手に入れたい(to T6)	S2	★ALICEになりすまし商品を手に入れたい(to S8)
					T1	&◆利用者ALICEにEVEのサイトをクリックさせる(to S1)
					T2	&◆COOKIE情報内のセッションIDを利用する
					R1	■COOKIE情報

また，SARM 状況表は第 1 種 SARM 状況表と第 2 種 SARM 状況表に分類することができる．この 2 種の定義を以下に示す．

[定義] 第 1 種 SARM 状況表
SARM[X, Y] 要素が（X ≠ Y）のとき，⟨階層ゴール基本式⟩ のみであるとき第 1 種 SARM 状況表という．

[定義] 第 2 種 SARM 状況表
SARM 状況表が第 1 種でないとき，第 2 種 SARM 状況表であるという．

このように，SARM 状況表を第 1 種と第 2 種に分けることによるメリットは，① SARM の作成方法を明確にする，② SARM と i*-Liu 法の等価性の証明の根拠を明確にする，③ 変換ツールを作成するときの基準を示すことである．

第 1 種 SARM 状況表の対角要素と非対角要素が SR 図の情報に対応する．第 2 種 SARM 状況表の対角要素は SR 図と同じ情報を持つ．第 2 種 SARM 状況表では，非対角要素として，階層ゴール基本式以外に，階層ゴール式も記述できることから，ゴール分解関係も記述できるという特徴がある．一方，SR 図ではアクタ関係に対してゴール分解することはできない．この点で第 2 種 SARM 状況表は SR 図よりも詳細なアクタ関係を記述できる．

またアクタ関係行列という表形式の手法を使用することにより，表のすべての欄を埋めるように検討を行うことができる点で SARM 状況表では網羅性を向上させることができる．定義から分かるように，SARM 状況表では，一般アクタと攻撃アクタのゴールとソフトゴール，タスク，資源に対する表現を階層的に記述できる．

このように階層ゴール式による攻撃表現の構造化では，① マークを ◯ ゴール，☆ソフトゴール，◇タスク，□ 資源のように前置すること，② 一般者は白抜きのマーク，攻撃者は黒塗りのマークを使用すること，③ タスク間の関係は &：論理積（AND）か |：論理和（OR）を前置し，構造化することにより分かりやすく表現できるようにした．

また，攻撃者の目標・意図・タスクを明確にするため，攻撃者欄を灰色に塗りつぶすこともできる．

表 3 に示したのは第 1 種 SARM 対応表の例である．

3.6　スパイラルレビュー手法

(1)　目的

SARM は網羅性が高いという利点があるので，1 人でも網羅性を持って分析することに向いており，i*-Liu 法は一覧性が高く，作成結果を理解しやすいという利点があるので，複

© 2011 Information Processing Society of Japan

数人での分析に向いていると考えられる．そこで，両手法の利点を活用し，より効果的な要求分析を実施するため，SARM と i*-Liu 法の双方を用いたスパイラルレビューを提案する．

(2)　スパイラルレビューの手順と事例

SARM と i*-Liu 法の双方を用いたスパイラルレビューの手順を以下に示す．

[手順 1] システム開発者が一般の各アクタ間の要求分析を i* で実施する．

[手順 2] セキュリティ担当者が AA（Asset × Attack）表で守るべき資源と攻撃パターンを洗い出す．

[手順 3] AA 表で洗い出した作成単位で，通常のシステム開発者とセキュリティ担当者がディスカッションツールとして，i*-Liu 法を使用し，結果を作図する．

[手順 4] セキュリティ担当者が，i*-Liu 法を SARM に変換したうえで，攻撃者を加えたセキュリティの要求分析を第 1 種 SARM 状況表で実施する．

3.7　詳細レビューによる攻撃者による具体的なタスク分析

(1)　目的

攻撃者による具体的なタスク分析として，第 2 種 SARM 状況表を用いたアクタ間の詳細レビューが実施可能である．第 2 種 SARM 状況表による詳細レビュー実施の目的は現状の i*-Liu 法では詳細に記述できないアクタ関係を SARM でレビューできるようにすることである．現状の i*-Liu 法の記述法や i*-Liu 法と等価性を持つ第 1 種 SARM 状況表では，アクタ間の関係を記述はできるが，この関係を論理的に分解して記述することができない．これに対して第 2 種 SARM 状況表は非対角欄にも対角欄と同様に論理式を記述できるので，アクタ間の関係をより詳細に分析できる．スパイラルレビューに加えて第 2 種 SARM 状況表による詳細レビューを実施する理由は，攻撃者の他のアクタに対する具体的な攻撃は非対角欄に記述することによって，その攻撃タスクが何であるかをより具体的にタスクレベルで抽出する必要があるからである．非対角欄こそが，攻撃者の他のアクタに対する具体的な意図が示される場所であり，詳細レビューの実施に際し，攻撃の具体的なタスクをあげて分析を実施すべき場所となる．

(2)　詳細レビューの手順と事例

第 2 種 SARM 状況表の記述能力の高さを生かすために，**図 1** のようにして i*-Liu 法と SARM のアクタ間のスパイラルレビュー後に第 2 種 SARM 状況表を用いて，アクタ間の詳細レビューを実施する．

第 2 種 SARM 状況表による詳細レビュー結果を**表 4** に示す．表 4 では，攻撃者 EVE の脆弱性のあるシステム BOB に対する具体的な攻撃内容が非対角要素に記載されている．

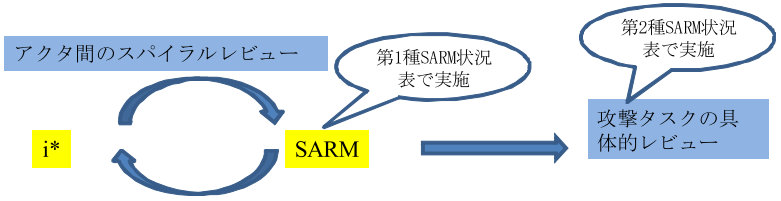

図 1　スパイラルレビューと詳細レビューの関係
Fig. 1　Relationship between spiral and detailed reviews.

表 4　第 2 種 SARM 状況表の詳細レビュー後の事例
Table 4　Example of the 2nd type SARM table after detailed review.

	利用者ALICE　A1		脆弱性のあるシステムBOB　A2		攻撃者EVE　A3	
利用者ALICE A1	S4	☆色々なサイトを閲覧したい(to S3)	S5	☆BOBのサイトを閲覧したい	S3	☆EVEのサイトを閲覧したい(to T2)
	G6	&○必要な情報を入力する		○webページをリクエストする		◇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
	T3	&◇ログインをする	T7	&◇BOBのサイトのトップページにアクセスする		
脆弱性のあるシステムBOB A2	S6	☆正当な利用者にアクセスさせる(to G2)	G1	○webページを生成する	S7	☆正当ではない利用者にはアクセスさせない(to S2)
	T8	◇ALICEにセッションIDを割り当てCOOKIE情報を通知する	G2	○利用者情報に従って情報を提供する		
			G3	○利用者情報を保護する		
			R2	□利用者情報	T10	◇偽造された認証データの使用を検知または拒否する
			G4	○利用者ごとの管理をする		
			T4	&◇許可された利用者に情報へのアクセスを許可する(to S6)(to S7)		
			R3	□認証データ		
			T5	&◇COOKIE情報を管理する		
			G5	○商品の販売をする		
			T6	◆ALICEになりすまして、商品の注文をする		
			T9	◆XSSの脆弱性によりALICEのCOOKIEをそのままEVEに送信(to T7)		
攻撃者EVE A3	S1	★ALICEにEVEのサイトをクリックさせたい(to A1)	S8	★BOBのシステムで商品を注文し、商品を手に入れたい(to T6)	S2	★ALICEになりすまし商品を手に入れたい
			T11	◆ALICEのｾｯｼｮﾝIDを盗む	T1	&◆利用者ALICEにEVEのサイトをクリックさせる(to S1)
			T12	| ◆Script　insertion	T15	&◆COOKIE情報窃盗スクリプトを起動(to T5)
			T13	| ◆HTTPレスポンス攻撃	T2	&◆COOKIE情報内のｾｯｼｮﾝIDを利用する
			T14	| ◆XSS	R1	■COOKIE情報

（表注）▭は攻撃タスクの具体的レビュー範囲を表す

たとえば，攻撃者 EVE の脆弱性のあるシステム BOB への意図を★BOB のシステムで商品を注文し，商品を手に入れたい（to T6）というソフトゴールのもとに◆ALICE のセッション ID を盗むというタスクがあり，そのタスクを実行するには | ◆Script Insertion か | ◆HTTP レスポンス攻撃か | ◆XSS という具体的な攻撃タスクをあげるといった詳細な記述ができることが分かる．

4.　SARM と i*-Liu 法のゴール等価性

4.1　ゴール関係

ゴール関係 GR を次のようにして定義する．

© 2011 Information Processing Society of Japan

[定義] ゴール関係

ゴール関係 $GR = \langle G, S, T, R, A, B, D, L \rangle$ は，ゴール集合 G，ソフトゴール集合 S，タスク集合 T，リソース集合 R，アクタ集合 A，アクタ所属関係 $B \subset (G \cup S \cup T \cup R) \times A$，依存関係 $D \subset A \times (G \cup S \cup T \cup R) \times A$，ゴール分解関係 $L \subset C \times (G \cup S \cup T \cup R) \times (G \cup S \cup T \cup R)$ からなる 8 項組である．

ここで，集合 G, S, T, R ならびに A は互いに素である．アクタ所属関係 B は，G, S, T, R の各要素がどのアクタに対応するかを示す G, S, T, R からアクタ集合 A への関係である．依存関係 D は，アクタ間に依存関係としてどのようなソフトゴール，ゴール，タスク，リソースがあるかを表しており，依存元，依存対象，依存先からなる 3 項関係である．したがって依存対象は集合 G, S, T, R の要素である．依存元と依存先は，アクタ集合 A の要素である．ゴール分解関係 L は，分解種別 $C = \{AND, OR\}$，親ゴール，子ゴールからなる 3 項関係である．ここで親ゴールならびに子ゴールは G, S, T, R の要素である．

[定義] SR 図

SR 図 P には，アクタ集合 Ap，各アクタ $a \in Ap$ についてのゴールのアクタ所属関係 $B_{Ap} = \{(g, a) \mid$ ゴール $g \in G \cup S \cup T \cup R$ がアクタ $a \in Ap$ に含まれる $\}$，各アクタ a のゴール分解 $La = \{(c, Gparent, Gchild) \mid c \in C,\ Gparent \in G \cup S \cup T \cup R,\ Gchild \in G \cup S \cup T \cup R,$ ここで Gparent と Gchild はアクタ $a \in Ap$ に含まれる$\}$ の集合，ゴール，ソフトゴール，タスク，リソースを依存対象とするアクタ $a \in Ap$ と他のアクタ $b \in Ap$ との依存関係 $DAp = \{(a, d, b) \mid a \in Ap, b \in Ap,\ d \in Gp \cup Sp \cup Tp \cup Rp,$ ここで $Gp \subset G$, $Sp \subset S$, $Tp \subset T$, $Rp \subset R$ は，それぞれ P に含まれるゴール，ソフトゴール，タスク，リソースの集合$\}$ の集合である．

このとき，$\langle Ap,\ \cup a \in Ap\, Ga,\ \cup a \in Ap\, Sa,\ \cup a \in Ap\, Ta,\ \cup a \in Ap\, Ra,\ BAp,\ DAp,\ \cup a \in Ap\, La \rangle$ を SR 図 P の表現といい，$\phi(P)$ で表す．

[命題] SR 図 P の表現 $\phi(P)$ はゴール関係である．

[証明] ゴール関係の定義から，SR 図 P の表現 $\phi(P) = \langle Ap, \cup a \in Ap\, Ga, \cup a \in Ap\, Sa, \cup a \in Ap\, Ta, \cup a \in Ap\, Ra, BAp, DAp, \cup a \in Ap\, La \rangle$ の各項がゴール関係の 8 項に対応することは明らかである．（証明終わり）

SARM 状況表 Z は $\{$e:階層ゴール式 $\mid e \in SARM[X, Y],\ X \in A,\ Y \in A\}$ で与えられる．したがってアクタ集合 A と，A の要素としてのアクタ間の階層ゴール式の集合 SARM[X, Y] が SARM 状況表 Z に含まれている．ここで，Z が含むアクタの集合 A はゴール関係の要素である．また非対角要素 SARM[X, Y]（$X \neq Y$）はアクタ間の関係 $D_A = \{(X, d, Y) \mid X \in A, Y \in A,\ d \in G \cup S \cup T \cup R\}$ を示している．第 1 種 SARM 状況表の場合，非対角要素は階層ゴール基本式から生成される構造に限定される．つまりゴール，ソフトゴール，タスク，資源もしくは攻撃ゴール，攻撃ソフトゴール，攻撃タスク，攻撃資源だけになり深さが 2 以上の階層構造を持つことはない．一方，対角要素 SARM[X, X] で与えられる階層ゴール式はアクタ X 内部のゴールの階層構造に対応しているので深さが 2 以上の階層構造を持つことがある．

もし Z が含む階層ゴール式の集合がゴール関係を構成することが示されれば，SARM 状況表からゴール関係を導くことができる．階層ゴール式は，複数の階層ゴール式が適用されることで階層構造を生成している．そこで，階層ゴール式の階層の深さについての帰納法を用いることにより，階層ゴール式からゴール関係を構成できることを示す．なお，以下の議論では，簡単のため攻撃ゴールなどの先頭に接頭辞として付与される脆弱性段階の扱いについては省略した．

[命題] 第 1 種 SARM 状況表からゴール関係を構成できる

[証明]

（段階 1） 階層の深さが 1 のとき階層ゴール式は，前述したように，ゴール，ソフトゴール，タスク，資源もしくは攻撃ゴール，攻撃ソフトゴール，攻撃タスク，攻撃資源によって表現される．したがって，深さ 1 の階層ゴール式から $\{G_X \mid X \in A\}$, $\{S_X \mid X \in A\}$, $\{T_X \mid X \in A\}$, $\{R_X \mid X \in A\}$ を構成できる．ここで，攻撃ゴール，攻撃ソフトゴール，攻撃タスク，攻撃資源をそれぞれ，ゴール，ソフトゴール，タスク，資源に含めて考えるものとする．

SARM 状況表の対角要素 SARM[X, X], $X \in A$ についても，階層の深さが 1 であるから，ゴール，ソフトゴール，タスク，資源もしくは攻撃ゴール，攻撃ソフトゴール，攻撃タスク，攻撃資源によって表現される．このとき，各アクタ $X \in A$ とこれらのゴール，ソフトゴール，タスク，資源のアクタ所属関係 $B_A = \{(g, X) \mid g \in G_X \cup S_X \cup T_X \cup R_X$ がアクタ $X \in A$ に含まれる$\}$ を構成できる．

対角要素に含まれるゴールには階層の深さが 1 であることから，親子関係を持つゴールはない．このため，SARM[X, X] の要素に対してゴール分解 $L_X = \{(\varepsilon, \varepsilon, G) \mid G \in G \cup S \cup T \cup R,$ G は SARM[X, X] に含まれる$\}$ を構成する．なお，ε 記号は，対応する AND 記号，OR 記号，親ゴールが存在しないことを表す．

また SARM 状況表の非対角要素 SARM[X, Y], $X \in A$, $Y \in A$（$X \neq Y$）についても，階層の深さが 1 であるから，ゴール，ソフトゴール，タスク，資源によって表現される．このとき，各アクタ X, $Y \in A$ とこれらのゴール，ソフトゴール，タスク，資源の関係から，依

© 2011 Information Processing Society of Japan

存関係 $D_A = \{(X, d, Y) \mid X \in A,\ Y \in A,\ d \in G_Z \cup S_Z \cup T_Z \cup R_Z$, ここで G_Z, S_Z, T_Z, R_Z は，それぞれ Z に含まれるゴール，ソフトゴール，タスク，リソースの集合$\}$ を構成できる．

以上の議論から階層の深さが 1 のときに第 1 種 SARM 状況表 {e:階層ゴール式 | e ∈ SARM[X, Y], X ∈ A, Y ∈ A} からゴール関係 $\langle A, G_Z, S_Z, T_Z, R_Z, B_A, D_A, L_X\rangle$ を構成できる．

（段階 2） 深さ N の階層ゴール式からゴール関係 $\langle A, G_Z(N), S_Z(N), T_Z(N), R_Z(N), B_A(N), D_A(N), L_X(N)\rangle$ を構成できるとき，深さ N + 1 の階層ゴール式からゴール関係 $\langle A, G_Z(N+1), S_Z(N+1), T_Z(N+1), R_Z(N+1), B_A(N+1), D_A(N+1), L_X(N+1)\rangle$ を構成できることを示す．ここで，第 1 種 SARM 状況表のアクタ集合 A はゴール階層と独立なので変化しない．

第 1 種 SARM 状況表では非対角要素には階層性がないので，段階 1 と同じである．したがって $D_A(N+1) = D_A(N)$ である．

対角要素 SARM[X, X] の深さ N + 1 の階層ゴール式 e は，階層ゴール式の構文規則から，① 深さ N の階層ゴール式，② 深さ N の積階層ゴール式並び，③ 深さ N の和階層ゴール式並びのいずれかに対して親ゴール t が追加される構造になっているはずである．そうでなければ深さが N + 1 であることに矛盾する．

場合 ① については，親子関係 (ε, t, g) が追加される．ここで t, g ∈ G ∪ S ∪ T ∪ R は SARM[X, X] に含まれる．また t は構文規則 ⟨階層ゴール基本式⟩⟨改行⟩⟨字下げ⟩⟨階層ゴール式並び⟩ によって追加される第 N + 1 階層の親ゴールである．g は，この同じ構文規則によって親が追加される子ゴールである．このとき，深さ N の階層ゴール式に対するゴール分解 $L_X(N)$ では親ゴールだった g に対するゴール分解 $(\varepsilon, \varepsilon, g)$ が深さ N + 1 のゴール関係のゴール分解 $L_X(N+1)$ では削除され，新しい親ゴール t と g のゴール分解 (ε, t, g) が追加される．この関係以外は $L_X(N)$ と $L_X(N+1)$ は変化しない．すなわち，t を階層分解する g に対して $L_X(N)$ から $(\varepsilon, \varepsilon, g)$ を削除し (ε, t, g) を追加することによって $L_X(N+1)$ を構成できる．

場合 ② については，親子関係 (AND, t, g) が追加される．ここで t, g ∈ G ∪ S ∪ T ∪ R は SARM[X, X] に含まれる．また t は構文規則 ⟨階層ゴール基本式 ⟩⟨改行⟩⟨字下げ⟩⟨積階層ゴール式並び⟩ によって追加される第 N + 1 階層の親ゴールである．g は，この同じ構文規則によって親が追加される子ゴールである．このとき，深さ N の階層ゴール式に対するゴール分解 $L_X(N)$ では親ゴールだった g に対するゴール分解 $(\varepsilon, \varepsilon, g)$ が深さ N + 1 のゴール関係のゴール分解 $L_X(N+1)$ では削除され，新しい親ゴール t と g のゴール分解 (AND, t, g) が追加される．この関係以外は $L_X(N)$ と $L_X(N+1)$ は変化しない．すなわち，t を AND 分解する g に対して $L_X(N)$ から $(\varepsilon, \varepsilon, g)$ を削除し (OR, t, g) を追加することによって $L_X(N+1)$ を構成できる．

場合 ③ については，親子関係 (OR, G_{N+1}, G_N) が追加される．ここで t, g ∈ G∪S∪T∪R は SARM[X, X] に含まれる．また t は構文規則 ⟨階層ゴール基本式⟩⟨改行⟩⟨字下げ⟩⟨和階層ゴール式並び⟩ によって追加される第 N + 1 階層の親ゴールである．g は，この同じ構文規則によって親が追加される子ゴールである．このとき，深さ N の階層ゴール式に対するゴール分解 $L_X(N)$ では親ゴールだった g に対するゴール分解 $(\varepsilon, \varepsilon, g)$ が深さ N + 1 のゴール関係のゴール分解 $L_X(N+1)$ では削除され，新しい親ゴール t と g のゴール分解 (OR, t, g) が追加される．この関係以外は $L_X(N)$ と $L_X(N+1)$ は変化しない．すなわち，t を OR 分解する g に対して $L_X(N)$ から $(\varepsilon, \varepsilon, g)$ を削除し (OR, t, g) を追加することによって $L_X(N+1)$ を構成できる．

以上の場合 ① ② ③ について，N+1 階層で追加される t はゴール階層の頂点となるゴールであるから G_Z か S_Z の要素である．t が G_Z の要素であるとき，$G_Z(N+1) = G_Z(N) \cup \{t\}$, $S_Z(N+1) = S_Z(N)$, $T_Z(N+1) = T_Z(N)$, $R_Z(N+1) = R_Z(N)$. t が S_Z の要素であるとき，$G_Z(N+1) = G_Z(N)$, $S_Z(N+1) = S_Z(N) \cup \{t\}$, $T_Z(N+1) = T_Z(N)$, $R_Z(N+1) = R_Z(N)$.

また以上の場合 ① ② ③ について，第 N + 1 階層の親ゴール t に対して $B_A(N)$ に (t, X) を追加することで $B_A(N+1)$ を構成できる．

以上から，深さ N の階層ゴール式からゴール関係 $\langle A, G_Z(N), S_Z(N), T_Z(N), R_Z(N), B_A(N), D_A(N), L_X(N)\rangle$ を構成できるとき，深さ N + 1 の階層ゴール式からゴール関係 $\langle A, G_Z(N+1), S_Z(N+1), T_Z(N+1), R_Z(N+1), B_A(N+1), D_A(N+1), L_X(N+1)\rangle$ を構成できることが明らかになった．

したがって任意の階層の階層ゴール式で記述された第 1 種 SARM 状況表からゴール関係を構成できることが証明された．（証明終わり）

[定義] SARM 状況表のゴール表現

SARM 状況表 Z = {e:階層ゴール式 | e ∈ SARM[X, Y], X ∈ A, Y ∈ A} に対して構成されるゴール関係 $\langle A, G_Z(N), S_Z(N), T_Z(N), R_Z(N), B_A(N), D_A(N), L_X(N)\rangle$ を Z のゴール表現 $\pi(Z)$ という．

4.2　ゴール関係に基づく等価性について

[定義] ゴール等価

2 つのゴール関係 GR と GR' に対して，GR = GR' となるとき，すなわち，それぞれの 8 項組の各集合が一致するとき，GR と GR' はゴール等価であるという．ここで，アクタ

© 2011 Information Processing Society of Japan

所属関係，依存関係，ゴール分解関係は複数の集合からなる直積集合であり，やはり集合であることに注意する．

[命題]

i*-Liu 法で記述された SR 図を P，第 1 種 SARM 状況表を Z = {e:階層ゴール式 | e ∈ SARM[X, Y], X ∈ A, Y ∈ A} とするとき，ϕ(P) と π(Z) のゴール等価性を判定できる．ここで，$\phi(\mathrm{P}) = \langle \mathrm{Ap}, \cup a \in \mathrm{Ap}\, \mathrm{Ga}, \cup a \in \mathrm{Ap}\, \mathrm{Sa}, \cup a \in \mathrm{Ap}\, \mathrm{Ta}, \cup a \in \mathrm{Ap}\, \mathrm{Ra}, \mathrm{BAp}, \mathrm{DAp}, \cup a \in \mathrm{Ap}\, \mathrm{La} \rangle$ $\pi(\mathrm{Z}) = \langle \mathrm{A}, \mathrm{G_Z(N)}, \mathrm{S_Z(N)}, \mathrm{T_Z(N)}, \mathrm{R_Z(N)}, \mathrm{B_A(N)}, \mathrm{D_A(N)}, \mathrm{L_X(N)} \rangle$

[証明]

ϕ(P) と π(Z) はゴール関係であること，ならびに，ゴール関係の 8 項組が有限集合から構成されることから，ゴール等価性を判定できる．（証明終わり）

一般には，同じセキュリティ要求に対して記述された P と Z からそれぞれ作成された Q と W が必ずしも一致するとは限らない．もし同じセキュリティ要求に対して記述された P と Z から作成された Q と W がゴール等価でなければ，P と Z のいずれかに抜けや誤りがあるか，ともに抜けや誤りがあることになる．したがって Q と W が一致するように P と Z を見直すことでセキュリティ要求分析の正確性を向上させることができることになる．この点が，i*-Liu 法と SARM を組み合わせたスパイラルレビューの有効性の根拠である．

4.3　ゴール関係の例

依存関係 D は依存元，依存対象，依存先を i*-Liu 法では線で結んでいる．これは SARM では依存先を（to 要素の ID）の形式で示すことで表現可能である．

アクタ所属関係 B は各 A すなわち各アクタがその円の中にある要素を所属させている．これを SARM 状況表では各アクタに所属する i*-Liu 法の要素を各アクタの対角欄に記載することで表現できる．

ゴール分解関係 L は i*-Liu 法ではゴール，ソフトゴール，タスクの各要素とその論理関係を分解種別，親ゴール，子ゴールとして図示する．一方 SARM 状況表では & と | の分解種別を各要素に前置して論理関係を示し，親ゴールから字下げをして子ゴールを記載することで表現可能である．**図 2** の i*-Liu 法に対するゴール関係 ⟨G, S, T, R, A, D, B, L⟩ を作成すると以下のようになる．なお互いに素である集合 G, S, T, R, A には通番を振り，表 3 に示した第 1 種 SARM 状況表に対する i*-Liu 法の表記例を図 2 に示した．表 3 では，第 1 種 SARM 状況表の項目ごとに G, S, T, R, A に対応する項番を振っている．この項目は図 2 の i*-Liu 法の項目にすべて一致させて記入することができる．また，i*-Liu 法からも第 1 種 SARM 状況表の項目にすべて一致させて記入することができる．

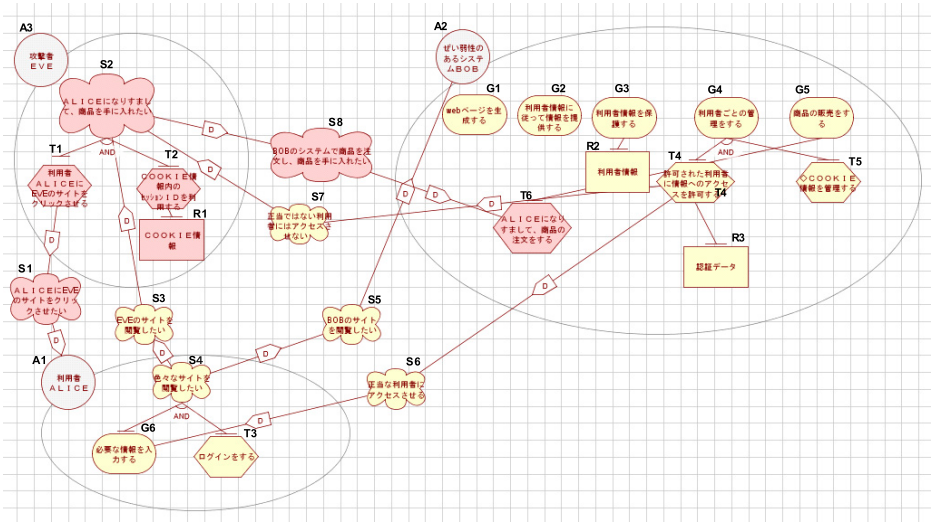

図 2　第 1 種 SARM 状況表の G, S, T, R, A に対応する i*-Liu 法の事例

Fig. 2　Example of the i*-Liu method corresponding to the G, S, T, R and A of the 1st type SARM table.

G = {G1:Web ページを生成する，G2:利用者情報に従って情報を提供する，G3:利用者情報を保護する，G4:利用者ごとの管理をする，G5:商品の販売をする，G6:必要な情報を入力する}

S = {S1:ALICE に EVE のサイトをクリックさせたい，S2:ALICE になりすまして商品を手に入れたい，S3:EVE のサイトを閲覧したい，S4:いろいろなサイトを閲覧したい，S5:BOB のサイトを閲覧したい，S6:正当な利用者にアクセスさせる，S7:正当ではない利用者にアクセスさせない，S8:BOB のシステムで商品を注文し商品を手に入れたい}

T = {T1:利用者 ALICE に EVE のサイトをクリックさせる，T2:COOKIE 情報内のセッション ID を利用する，T3:ログインをする，T4:許可された利用者に情報へのアクセスを許可する，T5:COOKIE 情報を管理する T6:ALICE になりすまして商品の注文をする}

R = {R1:COOKIE 情報，R2:利用者情報，R3:認証データ}

A = {A1:利用者 ALICE，A2:脆弱性のあるシステム BOB，A3:攻撃者 EVE}

アクタ間の関係を示す B, D, L の 3 種類については，紙面の都合上，B, L は EVE に関してのみ D は EVE と ALICE の関係のみを記述する．

B = {(ALICE になりすまし商品を手に入れたい，攻撃者 EVE)，(利用者 ALICE に EVE

© 2011 Information Processing Society of Japan

のサイトをクリックさせる，攻撃者 EVE)，(COOKIE 情報内のセッション ID を利用する，攻撃者 EVE)，(COOKIE 情報，攻撃者 EVE)}

D = {(利用者 ALICE に EVE のサイトをクリックさせる，ALICE に EVE のサイトをクリックさせたい，利用者 ALICE)，(いろいろなサイトを閲覧したい，EVE のサイトを閲覧したい，ALICE になりすまして商品を手に入れたい)，

L = {(ε, ε, ALICE になりすまして商品を手に入れたい)，(AND, ALICE になりすまして商品を手に入れたい，利用者 ALICE に EVE のサイトをクリックさせる)，(AND, ALICE になりすまして商品を手に入れたい，COOKIE 情報内のセッション ID を利用する)，(ε, COOKIE 情報内のセッション ID を利用する，COOKIE 情報)} ただし，ε 記号は，対応する AND 記号，OR 記号，親ゴールが存在しないことを表す．i*-Liu 法と同様に SARM 状況表からゴール関係を導出できるが紙面の都合で省略する．

この結果から，第 1 種 SARM 状況表と i*-Liu 法と互いに相互変換できることから，等価性を持つことが示された．ここでは，具体例に対して i* と第 1 種 SARM 状況表が相互に変換できることを確認しただけだが，変換ツールを作成することにより，一方の表現を他方の表現に変換することが期待できる．

5. 研究仮説の評価

本研究では，SARM と i*-Liu 法に関する以下の研究仮説を評価する．

A1. SARM のほうが i*-Liu 法よりも覚えやすい．
A2. SARM のほうが i*-Liu 法よりも書きやすい．
A4. SARM のほうが i*-Liu 法よりもセキュリティ要求の網羅性が高い．
A4. SARM のほうが i*-Liu 法よりも分析しやすい．
A5. SARM のほうが i*-Liu 法よりも理解しやすい．
A6. SARM と i*-Liu 法の表現能力は等価である．
A7. SARM と i*-Liu 法を相互変換することでセキュリティ要求をスパイラルにレビューできる．
A8. SARM で詳細レビューを実施することにより，攻撃の具体的なタスク分析が可能になる．

5.1 評 価 方 法

仮説 A1，A2，A3，A4，A5 については，アンケート結果に基づいて，5.2 ならびに 5.3 節で評価する．すなわち，仮説 A1 から A5 については，セキュリティ専門家に対するワークショップに基づくアンケートによって評価する．仮説 A6 については，すでに 4.1 と 4.2 節の議論から演繹的に得られる帰結ならびに，4.2 節で示したように表現能力の等価性を確認するための記述実験に基づいて確認した．仮説 A7 については，仮説 A1 から A6 までの評価結果に基づいて 5.4 節で議論する．仮説 A8 については，5.5 節でレビュー可能性を議論する．

5.2 ワークショップの開催とアンケートの実施

ワークショップを開催し，両手法の説明を行った後で，両手法を実際に作成してもらい，アンケートに回答する実験を実施した．

(1) 実験の目的

SARM を i*-Liu 法と比較することにより仮説 A1 から A5 を評価する．

(2) 実験対象としてのセキュリティ要求

i*-Liu 法の論文に記載されている医療現場におけるセキュリティ要求を題材とする．

(3) 実験の参加者

参加者は全員で 11 人であった．このうちセキュリティ専門家が 1 人で，他は情報セキュリティ専攻大学院生が 10 人であった．またセキュリティ設計の経験者はセキュリティ専門家に加えて 2 人の大学院生がいた．

(4) 実験仮説と検証手段

前述の A1，A2，A3，A4，A5 を二項検定で検証し，A3 のセキュリティ要求の網羅性はさらに抽出要素数の比較による t 検定を実施し，セキュリティ要求の内容の正しさを検証する．

(5) 実施内容

最初に i*-Liu 法と SARM の手法と作成方法の概要を説明し，i*-Liu 法作成後 SARM を作成するグループと SARM 記入後 i*-Liu 法を記入するグループに分け，実際に作図してもらった．作図するのは，医者，医療システム，患者，攻撃者をアクタとする i*-Liu 法と SARM である．記入するのは攻撃者にかかわる部分のみで他の部分はすでに記入ずみの様式を両手法とも提供し，未記入部分を埋めてもらった．作成条件を同じにするため，手書きで記入することとした．また，後に作成した図で気づいた点があっても前の図に反映させないこととした．

(6) 計測時間

両手法あわせて 20 分程度を作成時間とし，それぞれの図の作成にかかった時間を計測した．なお，最初の手法ばかりに時間がかかってしまわないように，実施すること自体の理解

© 2011 Information Processing Society of Japan

などの時間は計測時間にいれていない．ただし，各手法を個別に理解するのにかかった時間は各手法の所要時間にいれている．分析は 20 分程度で記入できる簡単な分析とした．

(7)　アンケート内容

アンケートでは i*-Liu 法作成にかかった時間と SARM 作成にかかった時間，両手法を比較し，覚えやすさ，書きやすさ，セキュリティ要求の網羅性，分析しやすさ，作成結果の理解しやすさの優れている方をあげてもらった．この 5 つの評価項目は，ミスユースケースによるセキュリティ要求分析の論文[19] での手法の評価の観点を参考に設定している．

(8)　実験設計上の考慮点

実施時間は両手法とも 20 分としていたが，i*-Liu と SARM のどちらを先に実施したかにより，最初の手法で抽出した内容を 2 回目の手法のほうでもまた記載すればいいという条件の差（慣れによるバイアス）などが生じる．そこで，i*-Liu を適用してから SARM を適用する第 1 グループと，SARM を適用してから i*-Liu を適用する第 2 グループにまず大学院生 4 人ずつを割り当てた．次に，セキュリティ設計経験のある 2 人の大学院生を第 1 グループに，そしてセキュリティ専門家を第 2 グループに割り当てることにした．このようにして 2 つのグループを構成することにより，できるだけ手法の適用順序の差が実験結果に影響しないように配慮した．

(9)　アンケートの実施結果

ワークショップで実施したアンケートの結果を**表 5** にまとめる．表 5 の二項検定の数字欄の値は優れていると回答した回答者数とそのパーセントである．

5.3　アンケートに基づく仮説の検証

アンケートで質問した結果を見ると，A1. 覚えやすさ，A2. 書きやすさ，A3. セキュリティ要求網羅性の高さ，A4. 分析しやすさ，A5. 作成結果を見て理解のしやすさの 5 つの観点から i*-Liu 法より SARM の評価のほうが全項目で高い．

表 5 で示したように SARM と i*-Liu 法のどちらが支持されたかを二項分布で検定したところ，有意水準 0.05 で，両者に差がないという帰無仮説は，A1 と A4 についてはともに 0.0107 で棄却される．よって A1 と A4 では SARM は i*-Liu 法より支持されたといえる．しかし A2，A3，A5 では，有意水準 0.05 では両者に差がないという帰無仮説は棄却される結果となったため，結論を保留する必要がある．

二項検定で結論を保留した評価項目のうち，③ セキュリティ要求の網羅性は SARM の特徴であるので，参加者に記述してもらった要素数で，網羅性をさらに評価する．正しく抽出されたセキュリティ要求の要素数を比較すると SARM の要素数合計は 87，i*-Liu 法の要素数合計は 43 である．t 検定（等分散を仮定した 2 標本による検定）で有意差があるかを検定すると，SARM と i*-Liu 法の抽出要素数に差がないという仮定は 0.0315 で棄却される．よって SARM の抽出要素数は i*-Liu 法よりも多く，セキュリティ要求の網羅性が高いのは SARM であるといえる．

表 5　アンケート結果
Table 5　Summary of questionnaire results.

		i*-Liu 法	SARM	不明	検定値	
二項検定	A1. 覚えやすい方はどちらか？	1(9%)	9(82%)	1(9%)	0.0107	5%有意
	A2. 書きやすい方はどちらか？	3(27%)	7(64%)	1(9%)	0.1719	結論を保留
	A3. セキュリティ要求の網羅性が高い方はどちらか？	1(9%)	6(55%)	4(36%)	0.0625	結論を保留
	A4. 分析しやすい方はどちらか？	1(9%)	9(82%)	1(9%)	0.0107	5%有意
	A5. 作成結果をみて，理解しやすい方はどちらか？	2(18%)	8(73%)	1(9%)	0.0547	結論を保留
t 検定	A3. セキュリティ要求の網羅性が高い方はどちらか？	要素数 43	要素数 87	-	0.0315	5%有意

なお，A3 の回答で不明が多い理由は，両手法の図作成時間が合わせて 20 分と短く，網羅性の検証まではできていなかったと被験者が感じているためと考えられる．SARM には，① 攻撃パターン単位で攻撃者を加えるという攻撃パターン網羅性と，② 他のアクタとの関係で場合を尽くす関係網羅性がある．現在の実験では，その一部しか評価の対象にしておらず，また，抽出した要素の内容妥当性をきちんと議論できる程正確なものでもない．したがって，網羅性については今後，それら 2 種の網羅性それぞれについて，要素条件の抽出数，抽出要素種別，抽出した要素内容の妥当性などをきちんと定義したうえで再評価する必要がある．

5.4　スパイラルレビューの仮説検証と初期評価

仮説 A7 については，仮説 A1 から A6 までの評価結果と 4.1.2 項ゴール関係に基づく等価性の証明で Q と W が一致するように P と Z を見直すことでセキュリティ要求分析の正確性を向上させることができることから，i*-Liu 法と第 1 種 SARM 状況表を組み合わせた

© 2011 Information Processing Society of Japan

スパイラルレビューの有効性を確認した．

さらに提案したスパイラルレビューの初期評価として各手順を以下で吟味する．

① 一般の各アクタ間の要求分析を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に分けることにより，一定の様式上で通常機能と攻撃を分析できる現実的な手順としている．

② AA 表を作成することによって，STRIDE 単位で各種攻撃のパターンと守るべき資源（アセット）との組合せを網羅することができる．これにより，よりセキュアなシステムを実現する．

③ i*-Liu 法は多人数でディスカッションしながら，要求分析をするときに分かりやすく，使いやすい．そこで，要求分析の初期段階で通常のシステム開発者とセキュリティ専門家同士が相互理解を図り，ディスカッションツールとして，使用することで特長を生かせると考える．

④ i*-Liu 法の図を第 1 種 SARM 状況表に変換し，レビューすることにより，セキュリティ要求分析の網羅性を向上させる．SARM 状況表は個人での要求分析や 1 つのノードの内部構造を詳細化する要求分析のときには，欄ごとに内容を詰めることが容易である．i* のように各種の図の中にテキストを記入し，関連性を線で結ぶのではなく，SARM 状況表はテキストにマークを前置するだけで内容を示すので，要素に分解したときの結果を簡潔に作成できる．また，表形式は設計に落としやすいので，議論して作った i*-Liu 法の SR 図をさらに細かく分析していくことに適している．i*-Liu 法の図を第 1 種 SARM 状況表に変換するのは，2 度手間にはなるが，表 5 のアンケート結果では，82%の人が SARM の方が覚えやすいとしており，i* を習得している人ならば，SARM を理解するのは簡単な作業であると考えられる．さらに 64%の人が SARM の方が書きやすいとしているので，SARM を書くこと自体は簡単にできる．また，ツールを用いた変換を支援することも考えられる．

5.5　詳細レビューの仮説検証

仮説 A8 については，3 章で示した第 2 種 SARM 状況表による詳細レビューの実施により，具体的なタスク分析ができることを確認した．アクタ間のセキュリティ要求分析において，攻撃者の具体的なタスクを分析することは，対策立案につながる作業であるため，重要である．

6. 全体を通した評価

6.1　研究の限界

本論文で小規模なアンケートに基づいて評価しているが，限定されたアンケート評価では仮説検証の一般性に限界がある．また，分析といっても 1 人による単独分析と複数人による共同分析では，分析のしやすさに差がある．i*-Liu 法では明示的に図で対象が表示されるので，複数人が図を見て議論を進めながら分析するのに適していると考えられる．一方 SARM は，① AA 表を作成することによって，STRIDE 単位で各種攻撃のパターンを網羅している ② 表として欄を埋めていくので，アクタ間の完全性が向上するという点で網羅性に優れている．アクタ数が多いときや 1 人で深く分析するときは，欄ごとにつめていけるので SARM には分析しやすいという特長もある．双方の長所をいかすためにスパイラルレビューを提案した．ただし，SARM は位置情報で関係性を示すために，間接攻撃など 2 者間の関係を，要素 ID を付ける形式で表現しているが，視覚的な分かりやすさに欠けるという問題点もあり，今後の改善への課題を残している．

6.2　SARM の効果

SARM の効果には，スパイラルレビューを可能にした点と，アクタ関係に対する詳細レビューを実施できる点の 2 つがある．スパイラルレビューが適切な理由は i*-Liu 法と SARM で共通する要素について，視点を変えて確認できることである．次に SARM による詳細レビューが効果的な理由は現状の i*-Liu 法では詳細に記述できないアクタ関係を第 2 種 SARM 状況表でレビューできることである．詳細レビューにより，具体的な攻撃タスクの抽出が可能となり，攻撃に対する対策の立案という次の段階にすすむことができる．ただし，対策の立案の具体的な手法は今後の課題である．

7. 結　　論

本論文では，アクタ関係行列（ARM）を拡張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）を提案した．SARM は i* に対してセキュリティ対応をした i*-Liu 法で作成するモデルと等価性を持って表現可能である．SARM と i*-Liu 法を比較・評価した結果，SARM の網羅性の高さと i*-Liu 法の一覧性の良さを確認した．さらに，両手法の特性を生かした効果的な使用方法として，SARM と i*-Liu 法の組合せレビュー手法を提案した．また，SARM と i*-Liu 法については定式化し，一般的に相互変換できることを論理的に証明し両手法の等価性を明らかにした．また具体例による確認も実施した．アクタ間の詳

© 2011 Information Processing Society of Japan

細レビューを実施可能なことも SARM の効果である．これらの手法の本格的な評価ならびに変換ツールの開発は今後の課題である．

謝辞　数々の有益なご指摘を賜った査読者の方々をはじめ，本論文の作成にご協力いただいた田中研究室や，この研究をするにあたりサポートしていただいた（株）NTT データの皆様，励ましをいただいた恩師，家族に謹んで感謝の意を表する．

参 考 文 献

1) Kotonya, G. and Sonnmerville, I.: *Requirements Engineering*, John Wiley & Sons (2002).
2) Dubois, E. and Mouratidis, H.: security requirements engineering: past, present and future, *Requirements Engineering*, Vol.15, No.1, pp.1–5 (online), DOI:010.01007/s00766-009-0094-8 (2009).
3) Liu, L., Yu, E. and Mylopolos, J.: Security and Privacy Requirements Analysis within a Social Setting, *Proc. IEEE International Conference on Requirements Engineering* (*RE 2003*), pp.151–161 (2003).
4) Yu, E.: i*homepage, i* (online), available from ⟨http://www.cs.toronto.edu/km/istar/⟩.
5) Letier, E.: Reasoning about Agents in Goal-Oriented Requirements Engineering, Université Catholique de Louvain (2001).
6) 井部己文，山本修一郎，佐藤友合子：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，情報処理学会研究報告，ソフトウェア工学研究会報告，Vol.2007, No.0107, pp.1–6 (2007).
7) 山本修一郎：連載第 39 回アクタ関係分析，ビジネスコミュニケーション,（オンライン），入手先⟨http://www.bcm.co.jp/site/youkyu/youkyu39.html⟩.
8) Yamamoto, S., Ibe, K., Verner, J., et al.: Actor Relationship Analysis for the i* Framework, *Proc. International Conference on Enterprise Information Systems* (*ICEIS 2009*), pp.491–500 (2009).
9) Sindre, G. and Opdahl, L.A.: Eliciting security requirements with misuse cases, *Requirements Engineering*, Vol.10, No.1, pp.34–44 (2005).
10) Chung, L., Nixon, B., Yu, E., et al.: *Non-Functional Requirements In Software Engineering*, Academic Publishers (1999).
11) Mouratidis, H.: Secure Tropos homepage (online), available from ⟨http://www.securetropos.org/⟩.
12) Li, T., Liu, L., Elahi, G., et al.: Service Security Analysis Based on i*: An Approach from the Attacker Viewpoint, *Proc. 34th Annual IEEE Computer Software and Applications Conference Workshops*, pp.127–133 (2010).
13) 玉井哲雄：問題フレームについて，情報処理，Vol.49, No.4, pp.364–370 (2008).
14) Lin, L., Nuseibeh, B., Ince, D., et al.: Introducing Abuse Frames for Analysing Security Requirements, *Proc. IEEE International Conference on Requirements Engineering* (*RE 2003*), pp.371–372 (2003).
15) 金子朋子，山本修一郎，田中英彦：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，情報処理学会 CSS（コンピュータ・セキュリティ・シンポジウム）富山，pp.721–736 (2009).
16) Web アプリケーションセキュリティ強化—脅威とその対策（オンライン），入手先⟨http://msdn.microsoft.com/ja-jp/library/ff648641.aspx⟩.
17) 大久保隆夫，田中英彦：セキュリティアスペクトの設計手法，電子情報通信学会，2008 年暗号と情報セキュリティシンポジウム（SCIS2008）(2008).
18) IPA：脆弱性対策情報データベース（オンライン），入手先⟨http://jvndb.jvn.jp/⟩.
19) Guttorm, S. and Andreas, L.O.: Eliciting security requirements with misuse cases, *Requir. Eng.*, Vol.10, pp.34–44 (2005).

(平成 22 年 12 月 11 日受付)
(平成 23 年 6 月 3 日採録)

金子　朋子（正会員）

1988 年慶應義塾大学文学部卒業．同年（株）NTT 入社．（株）NTT データにてシステム開発業務に従事．1993 年創価大学法学部卒業．2008 年情報セキュリティ大学院大学情報セキュリティ研究科博士前期課程修了．現在，（株）NTT データ品質保証部所属．情報セキュリティ大学院大学客員研究員．プロジェクトマネジメント学会会員．文部科学省認定プログラム・サーティフィケート取得者（情報セキュリティスペシャリスト，ソフトウェアスペシャリスト）．

© 2011 Information Processing Society of Japan

山本修一郎（正会員）

1977 年名古屋工業大学情報工学科卒業．1979 年名古屋大学大学院工学研究科情報工学専攻修了．同年日本電信電話公社入社．2002 年（株）NTT データ技術開発本部副本部長．2007 年同社初代フェロー，システム科学研究所所長．2009 年東京工業大学統合研究院医療情報プロジェクト特任教授．同年名古屋大学情報連携統括本部情報戦略室教授．ソフトウェア工学，要求工学，IC カードプラットフォーム，ユビキタスコンピューティング，知識創造デザインの研究に従事．情報処理学会業績賞，電子情報通信学会業績賞，逓信協会前島賞等受賞．博士（工学）．著書に，『要求定義・要求仕様書の作り方』（ソフト・リサーチ・センター，2006）『〜ゴール指向による〜システム要求管理』（ソフト・リサーチ・センター，2007），『すりあわせの技術』（ダイヤモンド社，2009），『CMC で変わる組織コミュニケーション』（NTT 出版，2010）等．電子情報通信学会知能ソフトウェア工学研究会研究専門委員長（2000〜2002 年），人工知能学会知識流通ネットワーク研究会主査（2007 年〜）．情報処理学会，電子情報通信学会，日本ソフトウェア科学会，人工知能学会，日本情報経営学会，ACM，IEEE 各会員．

田中　英彦（フェロー）

1970 年東京大学大学院博士課程修了，工学博士．東京大学工学部教授，同情報理工学系研究科教授・研究科長を経て，2004 年情報セキュリティ大学院大学教授，研究科長．計算機アーキテクチャ，分散処理，知識処理，デペンダブル情報システム等に興味を持つ．著書に『非ノイマンコンピュータ』『計算機アーキテクチャ』『Parallel Inference Engine』等がある．日本学術会議会員，情報処理学会，電子情報通信学会，人工知能学会各フェロー，IEEE ライフフェロー．

© 2011 Information Processing Society of Japan