セキュリティ標準に基づいたITシステム設計支援ツールの開発

芦野 佑樹† 髙橋 雄志‡ 森田 陽一郎† 島 成佳† 岡村 利彦†

勅使河原 可海†† 佐々木 良一††

†NEC クラウドシステム研究所
211-8666 神奈川県川崎市中原区下沼部 1753
y-ashino@cw.nec.jp.com, y-morita@dc.jp.nec.com,
shima@ap.jp.nec.com, t-okamura@da.jp.nec.com

‡創価大学大学院工学研究科
192-8577
東京都八王子市丹木町 1-236
e08d5203@soka.ac.jp

††東京電機大学未来科学部
120-8551 東京都足立区千住旭町 5
teshiga@isl.im.dendai.ac.jp, sasaki@im.dendai.ac.jp

あらまし 政府機関や業界団体は，ITシステムに対してセキュリティ標準を満たすよう要求している．しかし，セキュリティ標準は，汎用的な記述であるため，個々のITシステムに対する具体的な対策方法まで落とし込まれていない．したがって，ITシステムを設計するシステムエンジニアは，技術的な知識の他に，セキュリティ標準を読解するスキルが要求される．本論文では，ITシステムの構成がセキュリティ標準に基づいているかを評価するセキュリティモデリング技術を提案し，それに基づき，システムエンジニアの負担軽減を目的として，セキュリティ標準の一つであるPCI DSSに対応したITシステムの設計支援ツールを開発したので報告を行う．併せて，ツールの実装および運用に関する課題についても報告を行う．

Development of IT System Design Support Tool Based on Security Standards

Yuki Ashino† Yuji Takahashi‡ Yoichiro Morita† Shigeyoshi Shima†
Toshihiko Okamura† Yoshimi Teshigawara†† Ryoichi Sasaki††

†Cloud System Laboratories,
NEC Corpration
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8666, JAPAN
y-ashino@cw.nec.jp.com, y-morita@dc.jp.nec.com,
shima@ap.jp.nec.com, t-okamura@da.jp.nec.com

‡Graduate School of
Engineering, Soka University
1-236, Tangi-machi,
Hachioji-shi, Tokyo, 192-8577,
JAPAN
e08d5203@soka.ac.jp

††School of Science and Technology for Future Life, Tokyo Denki University
5 Senju-asahi-cho, Adachi-ku, Tokyo, 120-8551, JAPAN
teshiga@isl.im.dendai.ac.jp, sasaki@im.dendai.ac.jp

Abstract
Government organizations and industry organizations demand that IT systems are

based on security standards. However, because they are written in very general descriptions, a system engineer requires reading ability of security standards in addition to a broad knowledge of security. In this paper, to reduce a system engineer burden of system design based on security standards, we propose the method named "four-layer knowledge based model" of security standards. Also we show an IT system design support tool based on PCI DSS and identify issues in implementation and maintenance the tool.

1. はじめに

政府機関や業界団体は，IT システムに対してセキュリティ標準を満たすよう要求している．セキュリティ標準とは，IT システムの運用に関する組織体制や技術的対策などの監査基準が記載されているドキュメント群である．代表的なセキュリティ標準としては，政府機関によって策定された政府機関統一基準[1][2]やクレジットカード業界団体によって策定された Payment Card Industry Data Security Standard(PCI DSS)[3]が存在する．セキュリティ標準は，定期的なシステム監査や審査で用いられることを前提として記載されているため，あらゆる組織に対応できるように，具体的な対策方法や設定の粒度まで落とされていない．

したがって，システムエンジニアには，IT システムを設計する際に，セキュリティの知識に基づいてセキュリティ標準に記述されている内容を理解して具体的な対策技術や設定に解釈する能力が求められる．しかしながら，経済産業省や情報処理推進機構がセキュリティ人材育成について触れているように，セキュリティ標準を活用できるシステムエンジニアの確保は難しい[4][5]．このことから，セキュリティ標準の知識が十分でない IT システムの設計経験のあるシステムエンジニアがセキュリティ標準に基づいた設計するための支援ツールが必要である．

筆者らは，システムエンジニアを支援する目的として，セキュリティ標準に基づいた IT システム構成であるかを判定するために，(1)セキュリティ標準の 4 階層モデル，(2)IT システム構成を表現するモデル化のアプローチに基づくセキュリティモデリング技術を提案してきた[6]．本論文では，セキュリティモデリング技術に基づき，システムエンジニアやコンサルタントのヒアリングを通じて機能要件をまとめた上で，PCI DSS 対応の IT システム設計支援ツールを開発したので報告する．本論文の構成は，第 2 章ではセキュリティ標準を活用する課題と解決するための提案手法について述べ，第 3 章で提案手法に基づいて実装した設計支援ツールについて述べる．第 4 章では，ツールの実装および運用に関する課題について言及した後，5 章でまとめを行う．

2. 提案手法

2.1 セキュリティ標準を活用する際の課題

これまでの研究から，システムエンジニアが IT システムの設計時にセキュリティ標準を活用するためには，次に述べる要件を満たす必要がある．

(要件 1) セキュリティ要件から実現方式の導出

セキュリティ機能の設計は，IT システムで保護するべき情報資産と脅威を整理し，セキュリティの基本的な方針としてセキュリティ要件を決定した上で，セキュリティ要件に対応した実現方式の検討する形で進められる[7][8]．しかし，セキュリティ標準には，セキュリティ要件に対応する実現方式の記載はない．そのため，システムエンジニアには，セキュリティ要件と実現方式に関する知識をあらかじめ有する必要がある．

(要件 2) 実現方式に必要な機能と設定項目

セキュリティ標準に記載されている内容は，実現方式によって解釈が異なる場合がある．その

ため，システムエンジニアは，セキュリティ標準から実現方式に適した設定項目（以下，付随機能要素）を抽出する必要がある．

例えば，政府統一基準には，セキュリティ要件として，主体認証機能に関する記述がある．主体認証機能とは，あるコンピュータリソースに対して，あるシステムの利用者が正当なアクセス権を持つことを確認する機能である．ここには，主体認証機能で用いる主体認証情報は，利用者によって定期的に変更することを推奨している．主体認証機能の実現方式が，ID とパスワードを用いた認証（以下，ID/PW 認証）ならば，主体認証情報はパスワードであるため，利用者に対してパスワード更新機能という形で主体認証情報の更新機能を提供できる．その一方で，指紋などを用いて認証する実現方式（以下，生体認証）の場合は，主体認証情報が生体情報であるため，利用者に対して生体情報の変更を求めることは難しい．したがって，生体認証を実現方式に選んだ場合，主体認証情報の更新機能を利用者に提供しないといった判断が必要である．

（要件 3）新たなセキュリティ要件への対応

選択した実現方式によって，新たなセキュリティ要件が発生することがある．したがって，システムエンジニアは，実現方式を選択することによる新たなセキュリティ要件の抽出と，的確な実現方式の選択を行う必要がある．

例えば，政府統一基準には，通信路上をデータが流れる場合は暗号化するよう求めている．例えば，ID/PW 認証を選択した場合，通信路上を ID とパスワードが流れるため，これらを保護するために暗号化通信というセキュリティ要件が新たに発生し，対応する実現方式として VPN や SSL などを選択する必要がある．

（要件 4）構成変更に伴う迅速な再評価

IT システムの仮想化[9]により，今まで以上に IT システムの構成変更が柔軟にできるようになってきた[10]．システムエンジニアが，上記の(要件 1)～(要件 3)を実施できたとしても，IT システムの構成が一部または全体が変更された際は，改めてセキュリティレベル評価を行う必要がある．セキュリティレベル評価は時間がかかるため，仮想化のメリットを活かせない可能性がある．

したがって，システムエンジニアには，IT システムの構成が変更されたとしても，迅速にセキュリティレベル評価を実施できる必要がある．

2.2 セキュリティモデリング技術

システムエンジニアの負担を軽減させるには，2.1 節で述べた(要件 1)～(要件 4)をコンピュータにより処理する必要がある．そのためには，セキュリティ標準と IT システムの構成とマッチングできるようにデータ化するアプローチが必要である．

2.2.1 項では，セキュリティ標準のモデル化について述べ，2.2.2 項で IT システム構成を表現するモデル化について述べる．

2.2.1 セキュリティ標準のモデル化

セキュリティ標準のモデル化の目的は，システムエンジニアに対する入力支援と(要件 1)(要件 2)(要件 3)，IT システム構成とセキュリティ標準とのパターンマッチング支援(要件 4)である．両目的を果たすための構成としてセキュリティ標準は，次に述べるように，セキュリティ標準の 4 階層（以下，4 階層モデル）で表現する．

図 1. セキュリティ標準の 4 階層モデル

第 1 層: セキュリティ要件

第 1 層は，セキュリティ要件を格納する．例えば，政府統一基準の場合は 2.2.1 項以下にある，主体認証機能，アクセス制御機能，権限管理機能などと言ったものが該当する(図 1: 第 1 層)．

第 2 層: 実現方式

第2層は，第1層のセキュリティ要件に対応した実現方式を格納する．例えば，セキュリティ要件である主体認証機能に対応できる実現方式として，ID/PW 認証，生体認証，ハードウェア認証などを格納する．このデータ構造を持つことにより，セキュリティ要件と実現方式の対応付けができるようになり，(要件 1)を満たすことができる(図 1: 第 2 層)．

第 3 層: 付随機能要素

第 3 層は，実現方式に対応する付随機能要素を格納する．例えば，ID/PW 認証に対応する付随機能要素は，パスワードを保存する際に用いる暗号アルゴリズムや，通信路暗号化方式などを格納する．このデータ構造により，実現方式の付随機能要素を表現できるようになり，(要件 2)に対応できる．また，新たに発生するセキュリティ要件と実現方式との対応についても表現できるようになり，(要件 3)に対応できる(図 1: 第 3 層)．

第 4 層: 判定ルール

第 4 層は，システムモデル上に存在する付随機能要素の内容が，セキュリティ標準に基づいているかを判定するプログラムを格納する．プログラムは，付随機能要素がセキュリティ標準に基づいていると判定した場合は合格とし，そうでない場合は不合格とする．この判定ルールにより，システム構成の変更が行われても，コンピュータにより迅速にセキュリティレベル評価を行うことができるので，(要件 4)に対応できる(図 1: 第 4 層)．

2.2.2 IT システム構成のモデル化

IT システムの構成を表現するデータ形式は，判定ルールが処理しやすい構造である必要がある．2.2.1 項で述べた第 4 層に格納されるプログラムは，(1)コンピュータ同士の経路探索，(2)コンピュータリソースに割り当てられた実現方式および付随機能要素に対してセキュリティレベル評価を行う．したがって，システムモデルは，コンピュータの到達性が検証できるモデル（以下，コンピュータリソースモデル）と，コンピュータリソースと実現方式および付随機能要素の関係を表現するモデル（以下，セキュリティ機能モデル）の 2 種類が必要であると考えた．

コンピュータリソースモデルは，コンピュータ同士のネットワーク構成を表現するために，コンピュータとネットワークインタフェースカードを構成要素とし，ネットワークインタフェースカード同士の結線を表現できる必要がある．セキュリティ機能モデルは，コンピュータにインストールされた実現方式と付随機能要素を表現できる必要がある．

上記二つの表現ができるモデルをシステムモデルと名付けた．

3. 設計支援ツールの実装

今回実装するツールの機能要件は，システムエンジニアとシステムコンサルタントのヒアリングを基に定義した．

3.1 節では，ヒアリングを通じて定義した機能要件と実装範囲について述べ，3.2 節以降では機能要件に基づいた実装について述べる．

3.1 機能要件と実装範囲

2.1.節で述べた要件および 2.2 節で述べたアプローチについて，複数のシステムエンジニアおよびセキュリティコンサルタントにヒアリングを実施し，次に述べる機能要件に基づいて実装を行う．

(機能要件 1) 4 階層モデルの実装

ヒアリング結果から，セキュリティコンサルタントは，セキュリティレベル評価を行う際，技術的対策可能な項目と非技術的対策で対応する項目に分けて評価する．そこで，ツールに用いる 4 階層モデルを実装する際は，技術的に対策可能である部分を明確化した上で実装することとした．

また，今回のヒアリングを実施したコンサルタントが PCI DSS に精通していた点と，ISO27000 シリーズ[12]や FISC[13]に比べ技

術的な記述が多く記載されている点から，今回の実装はPCI DSSを選択した．また，今回の実装ではPCI DSSの第1章から第2章とし，評価内容は技術的に対策可能な項目とした．

(機能要件 2) データ入力の簡素化

システムエンジニアからのヒアリングの結果，入力の手間が少なく，かつ，入力したデータがそのままシステム設計関連のドキュメントとして活用できる形が望ましいとの見解を得た．

入力するデータ量は可能な限り少なくさせ，かつ，入力データを後からシステム構成図として活用できるように，広く使われているモデリングツールのインタフェースを応用する．

(機能要件 3) 評価結果の明確化

セキュリティコンサルタントは，技術的に解決できない場合は運用で対応を行うといった代替案を検討する必要がある．そのため，判定結果はセキュリティ標準と対応した項目と併せて出力するべきとの見解を述べた．システムエンジニアは，入力したデータの内，どのデータがどのような判定結果であるのか，また，次にどうするべきなのかアドバイスが欲しいとの見解を述べた．

以上のことから，(機能要件 3-1)セキュリティ標準の項目と対応した評価結果一覧機能，(機能要件 3-2)入力データ判定結果の反映機能，(機能要件 3-3)アドバイス機能を実装することにした．

3.2 PCI DSS の 4 階層モデル実装

3.2.1 セキュリティ要件と実現方式の確定

PCI DSSは各章がそれぞれセキュリティ要件として位置付けられているため，第1章のセキュリティ要件はファイアウォール，第2章のセキュリティ要件はデフォルト値の不使用とした．第1章の実現方式は，パケットフィルタリングとパケットフォワーディングとした．第2章のセキュリティ要件は，ITシステム全般に対する指示であるため，具体的な実現方式は存在しない．そのため，第2章に対応する実現方式は設けず，記載内容は全実現方式に対する付随機能要素として位置付けた．

3.2.2 付随機能要素の整理

(機能要件 1)の観点で，実装範囲を限定するため付随機能要素の内，システムモデルとして入力が不可なものを分類した．また，(機能要件 2)の観点で，システムモデルで表現できる付随機能要素の内，ツールの利用者が入力を省略できる項目とそうでない項目に分類した．(表 1)(表 2)は，第1章から第2章の付随機能要素の内，非技術的項目，強制付帯項目，調整必須項目に分類した表である．詳細は以下に述べる．

(1) 非技術的項目

技術的な対策ではない項目は，非技術的項目として実装の対象外とした．例えば，PCI DSS 1.1.2.a は，ネットワーク図を作成することを指示している．このように，ドキュメント類の整備はコンピュータリソースモデルやセキュリティ機能モデルでは表現できないため，非技術項目として扱う．

(2) 強制付帯項目

システムモデルで表現できる項目の内，実現方式に無条件で付されるべき機能や設定については，セキュリティレベル評価の対象外とした．例えば，PCI DSS 1.3.6 は，ステートフルインスペクション機能は必須であるとしている．搭載が必須である以上，搭載しない選択肢は存在しないため，セキュリティレベル評価の対象外とした．

(3) 調整必須項目

システムモデルで表現できる項目の内，システムモデルの内容によってセキュリティレベル評価の合格条件が変化するものを調整必須項目と定義して，実装の対象とした．例えば，PCI DSS 1.2.1 はフィルタリングルールに関する記述であり，IT システムの構成によってフィルタリングルールは変化しうる．したがって，セキュリティレベル評価の対象となり，判定ルールを実

表 1. 第 1 章の分類

	1.1.1.	1.1.2. a)	1.1.2. b)	1.1.3. a)	1.1.3. b)	1.1.4.	1.1.5. a)	1.1.5. b)	1.1.6. a)	1.1.6. b)	1.2.1. a)	1.2.1. b)	1.2.2.	1.2.3.	1.3.1.	1.3.2.	1.3.3.	1.3.4.	1.3.5.	1.3.6.	1.3.7.	1.3.8. a)	1.3.8. b)	1.4.a	1.4.b
非技術項目		○	○	○	○	○	○	○	○	○															
強制付帯	○												○					○		○		○	○		○
調整必須項目											○	○		○	○	○	○		○		○			○	

表 2. 第 2 章の分類

	2.1.1. a)	2.1.1. b)	2.1.1. c)	2.1.1. d)	2.1.1. e)	2.2.1. a)	2.2.1. b)	2.2.2. a)	2.2.2. b)	2.2.3. a)	2.2.3. b)	2.2.3. c)	2.2.4. a)	2.2.4. b)	2.2.4. c)	2.3.a.	2.3.b.	2.3.c.	2.4
非技術項目									○	○				○	○				
強制付帯	○	○	○	○	○			○			○	○	○			○	○		○
調整必須項目						○	○											○	

装する.

3.2.3 オブジェクトの定義

調整必須項目の判定ルールを実装するにあたり, システムモデル中に存在するべきコンピュータリソースやセキュリティ機能(以下, オブジェクト)が定義される場合がある. 例えば, PCI DSS 1.3.1 のセキュリティレベル評価を行うためには, システムモデル上に, DMZ, ファイアウォール, ファイアウォールのフィルタリングルールの他に, ウェブサービスが配置されている必要がある. このようにシステムモデルに存在すべき要素は, 判定ルールを実装することによって定義される. 以下は, システムモデル中に含まれる必要のあるオブジェクトを表す(表 3).

表 3. オブジェクト

物理マシン	ウェブクライアント
仮想マシン	SSHクライアント
有線ネットワークインタフェース	無線LANカード
仮想マシンモニタ	無線LAN基地局機能
ブリッジサービス	無線LANクライアント
NATサービス	VPNクライアント
VLANサービス	DNSサーバ
カード会員DB	一般従業員作業
公開ウェブサーバ	サーバ管理者作業

3.2.4 判定ルールの実装

3.2.2 および 3.2.3 で述べた内容を基に判定ルールを実装した. 第 1 章の判定ルールは, サービスへの到達性を検証するものである. システムモデルからサービスに対する経路を算出し, その経路上に存在するファイアウォールのフィルタリングルールを基に到達性を調べる. 第 2 章のほとんどが強制付帯に分類される項目であったため, 要調整機能に関しては関連する実装方式の付随機能要素として実装した.

判定ルールの出力結果には, システムエンジニアに対するアドバイスする目的で, セキュリティレベル評価結果の理由を付した.

3.3 システムモデルと GUI モデル

2.2.2 で述べたシステムモデルの実装には, 業界標準化団体 DMTF(Distributed Management Task Force)[14]が策定する CIM(Common Information Model)[15]を参考にした. CIM は, コンピュータやネットワークおよびコンピュータにインストールされているサーバプログラムやその設定を表現するモデリング言語である. CIM の特徴は, オブジェクト同士の関連性を関連クラスで表現する点にある. そのことにより, システムモデル全体から特定のオブジェクトを検索が容易になる.

しかし, そのまま CIM の概念をツールの入力データとして用いるとオブジェクトと関連クラスの両方を入力する必要がある. (機能要件 2)で定義したように入力の負担を減らすために, 両端のクラスによって関連クラスは一意に決まるという特性に着目した. この特性により, 二つのオブジェクト間に必要な関連クラスを自動的に導出できるため, ツールの入力データはオブジェクト間の関連を意味する結線のみ表現できる. この

ようにツールから入力するモデルをGUIモデルとした(図 2).

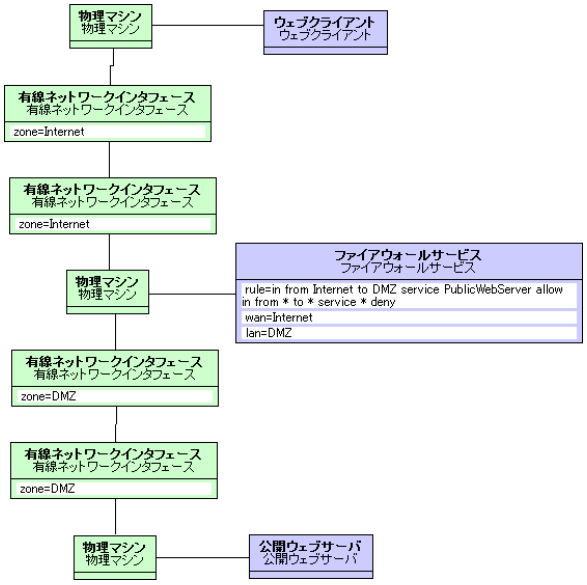

図 2. GUI モデル

GUI モデルに対してセキュリティレベル評価を行う際は, システムモデルに変換してから行うこととした. ツールのインタフェースは, 任意のモデルを編集する機能を提供する開発フレームワーク EMF[16]を拡張する形で実装する.

4. 考察

4.1 実装結果

3.1 節で定義した(機能要件 1)～(機能要件 3)の結果について述べる.

4 階層モデルの実装については, 3.2 節で述べたとおり, 実装の範囲をシステムモデルで表現できる技術的対策かつシステムモデルの内容によって合格条件が異なる項目に限定し, 判定ルールの実装を行った. 使用した言語は, C#で全ソースコードの行数は 7,450 行であった.

判定ルールの実装を通じて, システムモデルとして入力するオブジェクトを定義した. システムモデルは, CIM の概念を取り入れたため, オブジェクトの関係を表現する関連クラスが必要である. 入力するオブジェクト数をできるだけ減らすために, GUIモデルを定義し, インタフェースに実装した.

セキュリティレベル評価は, GUI モデルをシステムモデルに変換し, 判定ルールによって行う.

出力結果は, 評価結果は, スプレッドシートとして, PCI DSS の項番ごとにセキュリティレベル評価の結果を合格または不合格で表示する(図 3). このことで, 入力したITシステム構成全体の評価を一覧で把握することができる.

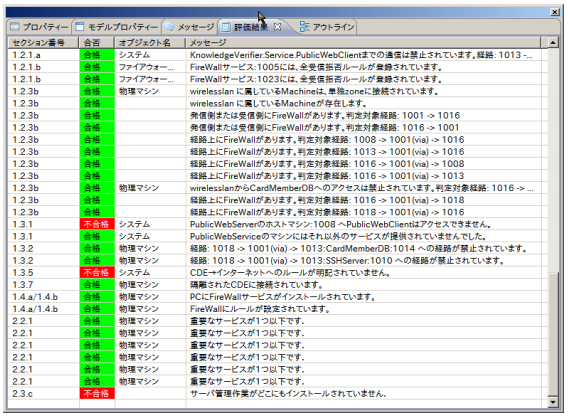

図 3. スプレッドシートへの評価結果出力

GUI モデルにも評価結果を反映させるようにした. オブジェクトと関連する付随機能要素が不合格の場合, 枠や文字を赤で表示した. また, このためにエラーメッセージを表示させることで, どのように修正するべきかのガイドを表示する(図 4).

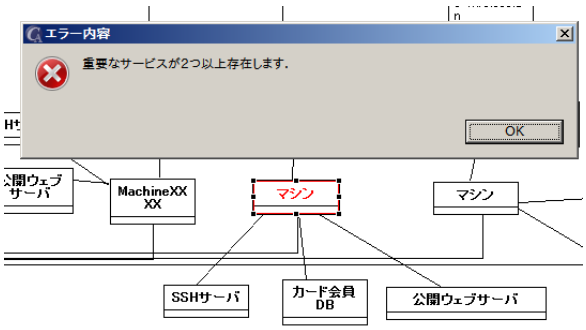

図 4. GUI モデルへの結果反映

以上のとおり, ヒアリングを通じて定義した(機能要件 1)～(機能要件 3)を実装した. しかしながら, 現在の所, 本ツールについて, システムエンジニアやセキュリティコンサルタントの評価を受けていない.

そのため, 今後は, 実装した機能要件について評価する方法について検討を行う.

4.2 システムモデル未確定問題

3.2 節で述べたとおり，セキュリティレベル評価する上で必要なシステムモデルは，セキュリティ標準の 4 階層モデルに基づく実装を行うまで確定しない．そのため，セキュリティ標準が異なるごとに，システムモデルの再定義が必要となる可能性がある．このままでは，ツールが使用するセキュリティ標準が異なるごとに，GUI モデルを入力し直す手間が発生する．

この問題に対するアプローチの一つとして，異なるセキュリティ標準間の相関関係を利用する手法の活用が考えられる．髙橋らの研究では，複数のセキュリティ標準間の相関関係を数値化することを試みている．この技術を使うことで，あるセキュリティ標準に対するセキュリティレベル評価合格の結果を，別の標準に対応するといった応用が考えられる[1]．

4.3 メンテナンス問題

3.2 節で実装した 4 階層モデルに基づくデータは，セキュリティ標準の改定や，新たなセキュリティ標準へ対応するためにメンテナンスが必要である．このメンテナンスを担当する者は，セキュリティ標準の記載内容から 4 階層モデルに分解し，判定ルールの実装を通じてシステムモデルを定義できる能力を有さなければならない．セキュリティ標準は複数存在し改定され続けることから，4 階層モデルに基づくデータをメンテナンスする担当者の負担は大きい．

4 階層モデルデータのメンテナンス方法を含めた検討が必要である．

5. まとめ

筆者らは，セキュリティ標準を IT システムの設計に活用するためには四つの要件を定めた．その要件を満たすためのアプローチとして，セキュリティ標準の 4 階層モデル化と IT システムの構成を表現するシステムモデル化について提案した．複数のコンサルタントやシステムエンジニアからのヒアリングに基づき，ツールの機能要件を定義し，PCI DSS を対象とした IT システム設計支援ツールを実装した．その結果，ツールの運用に関する新たな課題が判明した．

将来的に，このツールは，システムエンジニア以外にもセキュリティコンサルタントの利用を想定して検討を進めたい．そのためには，第 4 章で述べた課題に加え，セキュリティコンサルタントが本ツールを実際の現場で利用することを想定し，本ツールの有効性を評価する方法を今後の課題とする．

参考文献

[1] 内閣官房情報セキュリティセンター: 政府機関の情報セキュリティ対策のための統一規範, http://www.nisc.go.jp/active/general/pdf/kihan24.pdf

[2] 内閣官房情報セキュリティセンター: 政府機関の情報セキュリティ対策のための統一技術基準(平成 24 年度版), http://www.nisc.go.jp/active/general/pdf/k305-111.pdf

[3] Payment Card Industry Security Standards Council: PCI SSC Data Security Standards, https://www.pcisecuritystandards.org/security_standards/

[4] 情報処理推進機構:「情報セキュリティ人材の育成に関する基礎調査」報告書について http://www.ipa.go.jp/security/fy23/reports/jinzai/

[5] 経済産業省, IT 人材の育成: http://www.meti.go.jp/policy/it_policy/jinzai/

[6] 芦野佑樹, 森田陽一郎, 小泉純, 岡村利彦: セキュリティ標準に基づいたセキュリティレベル評価技術の検討, CSEC, 2013-CSEC-60(35), 1-8, 2013-03-07

[7] 大西 克美: できるエンジニアのセキュリティチェックポイント, プロジェクトを成功に導くシステム開発のスキルアップ教本, 日経 BP ムック, pp.100-113 (2007).

[8] 中塩 慎一: ゼロから学ぶアーキテクチャ設計, IT アーキテクトのためのシステム設計完全ガイド 2008, 日経 BP, pp.54-59 (2007).

[9] 日本 BP 社: すべてわかる仮想化大全 2013, 2012/12

[10] さくらインターネット: さくらインターネット、サーバやネットワークを自在に構築できるパブリッククラウド「さくらのクラウド」を 11 月 15 日より提供開始

[11] ～2011 年 11 月開所の石狩データセンターでの第一弾サービスとして登場～: http://www.sakura.ad.jp/press/2011/1108_cloud.html

[12] The ISO 27000 Directory, http://www.27000.org/

[13] 金融情報システムセンター: 金融機関等のシステム監査指針, 2007/03

[14] http://www.dmtf.org/

[15] http://www.dmtf.org/standards/cim

[16] Dave Budinsky, Frank Paternostro, Marcelo Merks, Ed Steinber: EMF: Eclipse Modeling Framework (Eclipse Series), 2008/12/16

[17] 髙橋雄志, 篠宮紀彦, 勅使河原可海: セキュリティ標準間の関連情報作成手法の検討とその適応, CDS, 2013-CDS-7(1), 1-8, 2013/05