安全なネットワークサーバ構築のための階層化した要件

吉田真澄*　　岡田　正**

Hierarchical Requirements for Developing a Secure Network Server

Masumi YOSHIDA and Tadashi OKADA

We are trying to resolve the problem of security when developing information systems on local area for informationalization aiding. In this report, we discuss the specification that adapted knowledge of the security requirement engineering. We propose systematic measures for networking server's safety and hierarchical requirements for a secure network server. This study is going to help informationalization at local society by our laboratory.

Key Words : Secure Network Server, Security Requirement Engineering, Informationalization Aiding

１．は　じ　め　に

近年，さまざまな業種の企業が経営システムの情報化を盛んに行っており，多様な情報をコンピュータで一元管理するのが当たり前な時代となっている．そこで我々の研究室では研究目的の一つとして，早くから地域社会の情報化の支援を行ってきた（参考文献1とその引用文献）．この延長上として，地元企業の各種業務の情報化を支援することで地域社会に貢献するとともに，より具体的で効率のよいシステム開発の手段について，実地での研究を行うことを大きな目的としている．

システム開発を行う際には，企業の運営環境などを考えて，企業にとって適切なシステムを提案しなければならない．しかし，多様な条件が複雑に絡み合う中でのシステム設計は非常に難しく，はっきりとした設計手段は確立されていないといえる．また，近年では情報セキュリティの重要さがさまざまなところで指摘されており，セキュリティを念頭においたシステム設計の需要はますます広がっている．

本報告では，システム設計から運営管理までの体系的なセキュリティ向上が可能な標準仕様の作成に向けて，調査・考察を行った内容を述べる．2章では，セキュリティ要求工学に基づくセキュリティ概念を整理し，3章で階層化されたセキュリティ仕様を提案する．これらの仕様を，実際の研究成果にフィードバックする事例を，4章で議論している．

原稿受付　平成21年8月31日

＊ 専攻科電子・情報システム工学専攻科生

＊＊情報工学科

２．セキュリティの考え方と困難さ

情報通信技術の発達により，インターネットを使った多様な活動が社会に浸透してきた．それに伴い情報漏洩やデータ改ざんなどの攻撃も悪質化・複雑化しており，セキュリティの重要性はますます広がっている．これまでの多くは，事故が発生してからパッチを充てるなどといった，その場しのぎの対応をすることに留まっていた．しかし，それでは根本的な解決にはなっておらず，また，そのパッチによるシステムの脆弱性が新たに生まれるかもしれない．こうした流れから，近年では特に根本的にセキュリティ高めることへの関心が高まっている．

2. 1 セキュリティ要求工学

前述したようなIT社会の状況により，近年ではシステム開発の初期段階からセキュリティにより強い関心を持ち，新しい攻撃方法にも柔軟に対応できる安全なシステムを開発することが奨励されている．そのための手段としてセキュリティ要求工学[2)]が注目を集めており，本研究でもセキュリティ要求工学を軸に，体系的なセキュリティ対策を考えていく．

セキュリティ要求工学とは，システム開発の際に「どのようなシステムが必要なのか」，「このときに誰がシステムに携わるのか」などといった要求を分析し獲得する要求工学を，セキュリティの分野について拡張したものである．要求を洗い出し，その要求が必要な理由を分析し，それらを整理したのち，要求を満たす対象を明らかにし，システムの要求を明らかにして，最終的な要求仕様とする．セキュリティ要求を分析する際には，図式化して効率よく整理することが一般的であり，先に述べた要求自体を「What」や「要求」，要求が必要な理由を「Why」や「ゴール」，要求を満たす対象を「Who」や「対象」などとおき，それぞれの関連性を問いながら要

求を分析していく．

セキュリティゴールを定めるためには，システムに対する脅威の種類や，その脅威による被害の程度といったリスク分析が必要になる．攻撃の確率や頻度を予測し，それによって生じる実際的な被害を予測算出する．被害の例としては，顧客情報流出や破壊などのデータ損失に対する金額や，システム復旧およびセキュリティ追加開発などに必要な金額といった実際的な損失はもとより，顧客に対する謝罪にかかった副次的な労働時間や，詫び状に添えるお詫びのクーポンなどにかかる費用，風評被害による製品の希求力低下が招く売り上げの減少など，さまざまな角度から被害の程度を予測していく必要がある．

2.2 セキュリティ要求獲得の難しさ

セキュリティ要求の獲得は，セキュリティという分野ならではの難しさがいくつか存在し，以下の３つに大別することができる．

(1) 取り扱う情報の幅が広く関連性が複雑であること
(2) 技術の発展や経済情勢などによりシステムが置かれる状況が刻々と変化すること
(3) セキュリティと利便性・セキュリティと開発費用などのトレードオフ

まず，システムのセキュリティについて考える場合，ソフトウェア開発・プログラミング・アルゴリズムの知識はもちろん，経済的な知識や法律に関する知識など，(1)の幅広い情報が必要になる．セキュリティの分野においては，このような複数の情報が複雑に起因し合い，特定の要求の獲得を難しくしているといえる．さらに，攻撃者により行われるセキュリティに対する攻撃を予測する場合は，スクリプトやセッションを利用する攻撃から，オフィスや管理室への侵入や，クレジットカード盗むなどの物理的な攻撃まで，例を挙げるときりがないほど考慮できる可能性が膨大になるという特徴もある．また，資産やセキュリティ原則などを決定する際には，経営に関わらなければならず，経営陣などのステークスホルダ（利害関係者）と関わりを持つ必要もある．

つぎに，MS-Windows[3]など大手ベンダが日々アップデートを繰り返していることから分かるように，(2)のようにセキュリティに対する脅威は日々新たに発見されている．そのため，すべての予想される脅威からシステムを完璧に守ることはほぼ不可能である．しかし，セキュリティ要求に対する分析を適切に行うことによって，新しい攻撃にも柔軟に対応し，攻撃による被害を減らすことができるシステムの開発を目標とすることは大切である．現状では，このような意味での安全なシステムを開発すべきである．

(3)に関しては，セキュリティの費用対効果は目に見える形ではっきりとは確認しづらいため，現在の，納期と開発費用がともに減少しているIT業界の状況とも相まって，システム開発におけるセキュリティへの優先度は低いものとなっている．セキュリティ要求分析および要求の獲得を行うためには，経営者・株主・技術者などさまざまな人間が関与する必要があり，また昨今の世界的不況から，ステークスホルダの関心をセキュリティ強化のための追加投資などに向けさせるのはより難しくなっていると考えられる．

以上のような原因が相互に関連し合い，適切なセキュリティ要求の分析・獲得を行うことは非常に難しいとされている．

2.3 セキュリティ要求獲得の手法

最近では，セキュリティ要求の定義・分析をより効果的に行うために，さまざまな研究機関からセキュリティ要求の方法論が提案されている．セキュリティ要求工学では，ステークスホルダとの対話により得られる断片的な情報を分析・整理し，最終的にシステム設計のために最適化された要求仕様書を作成するのが一連の流れである．さらに，得られた情報やニーズの背後には，より抽象度の高い要求や，より上位の目的が隠れていることがある．これを見つけ出しゴールとして明確にする「ゴール指向要求工学」が注目されている．主なゴール指向要求要求工学手法論には，Secure Tropos[4]やKAOS (Knowledge Acquisition in automated Specification)[5]などがある．

セキュリティにおいても，システム開発においても，要求定義・設計など開発初期の段階での不備によるソフトウェアの欠陥が大きな原因となっている．なので，セキュリティ要求工学では，いずれの手法も，パッチをあてる，アプリケーションの更新を行う，といったその場しのぎのセキュリティ対策では対応しきれないセキュリティ攻撃の現状を鑑みて，システムの要求定義・分析の段階からセキュリティを意識し，見落としを減らし欠陥が少なく攻撃されにくいシステムの開発を支援するような仕組みになっている．

３．階層化されたセキュリティ仕様書

上述したセキュリティ要求工学を元に，体系的な安全対策を考えるために機能階層別のセキュリティに関する仕様書を考えることから始めた．我々は，

(1) サーバマシン自体の安全性を考える「ハードウェア」
(2) マシンの稼動効率と安全性を視野に入れた「OS・システムコア」および「システム設定」
(3) 運用するコンテンツのコアに関する安全性を考

える「アプリケーション」
(4) コンテンツ利便性と安全性について考える「アプリケーション上で動くプログラム（サービス）」
(5) システムを運営する上での安全性およびシステムを利用するユーザの視点での安全性について述べる「システム運営・利用者」

の５つの階層で構成することにした．

3.1 ハードウェア

ハードウェアの面では，主にサーバマシンの発熱や埃，クラッシュによるデータ破損への対策などについて考えていく．例えば，発熱量を抑えたマルチコアCPUを利用したり，大型高速ファンを用いたりするなど，サーバ内の部品を換装するのが基本的な対応である．サーバの形状によっては複数のファンを設置することも可能なので，導入の際の機種選定においてそのようなことについても考えるとよい．ハードディスクの発熱に対応するために，SAN (Storage Area Network) を利用してサーバ本体と記憶部分を分離して熱源の分散を行うことも有効である．

CPUを高性能化した場合，サーバ単体の発熱量が増えるため，複数のサーバを保管するサーバラックにも必然的に発熱に対する対策が求められる．セキュリティの面を考慮すると，簡単には触れられないようにラックを防御したほうがいい．しかし，熱がこもってしまい熱によるサーバダウンなどを引き起こしてしまう可能性がある．排熱を効率よく行うには，例えばラックの前面をガードするなら，背面や側面に冷却ファンを取り付けることが可能な種類のラックを選定する．最近はサーバラック向け空調機器も充実しているので，そのような製品を利用するのもよい．

ただ，サーバラックを保管しているデータセンタ全体の空気の流れに背くようなファンの設置を行うと，効率的に排熱が行えなくなるため，設置位置は慎重に考慮する必要がある．さらには，データセンターの冷却装置や空調機器は，設置場所を考慮しないと空気を全体的に循環させることができず，一部のマシンが冷却の恩恵を得られないなどといった問題が起こる．そのため，データセンター稼働前からの室内設計にも気を配る必要がある．

3.2 システムおよびアプリケーション

OS・システムコアでは，本研究室で使用しているFreeBSD[6]のセキュリティ勧告を紹介し，そこから各種OSに共通するようなセキュリティホールへの考察を行う．例えば，2008年のセキュリティ勧告では，特定の入力がおこなわれた場合に，メモリ汚染を引き起こすために，特定エリアのメモリ領域の上書きを実行される危険性があることや，デフォルトでは有効になっていないものの，ftpd (8) に特定のコマンドを使うことでクロスサイトリクエスト偽装攻撃に使われる可能性があることが明らかになった．これらは対象の機能を使用不可に設定したり，最新バージョンへのアップデートを行うことにより解決できる．

システム設定では，アカウント・ログ・メール・ネットワーク設定などといった，基本となる設定変更について述べ，セキュリティの向上を図るとともに，運営の効率化に繋がる設定方法を考える．例えばアカウントの作成は，利用するユーザをきちんと把握し，必要最小限の数だけ作成する．適切にグループ分けを行い，利用権限を明確にし，管理者権限を管理責任者以外は利用できないようにする．パスワードは定期的に変更する．ログファイルは1ヶ月や半年など，決められた期間でローテートし，バックアップファイルとして別のストレージに保存するのがよい．

メールサーバでは，迷惑メールを振り分ける機能を備えておくことや，迷惑メールが送られてきた場合，メールのヘッダ情報を確認することにより，迷惑メールを送る相手を突き止めることができる．ヘッダ情報にはメールを送信する上での重要な情報が多く記載されており，この情報を理解することにより，迷惑メールの被害を少なくすることができる．特に「Received」の項目にはメールの配送方法が記載されており，この情報からこれが意図して送られた迷惑メールであるかを突き止めることが可能である．

アプリケーションおよびアプリケーション上で動くプログラム（サービス）では，実際に運営するシステムのコードの面でのセキュリティ強化や，プログラムの死角を狙った不正アクセス，スクリプト埋め込みなどの攻撃に対する対策について考えるようにしている．例えば，検索エンジンなどのWebサイトのキーワード入力欄にスクリプトを含んだタグを打ち込むと，そのサーバの脆弱性の度合いによって，cookieを吐き出したり，読み出されたcookieデータが第三者のサーバに転送されるなどの可能性があるクロスサイトスクリプティングは，近年の新しい攻撃方法の一つである．この脆弱性は，ユーザが入力した値をそのまま利用するアプリケーションなどで起こりうるため，プログラムレベルでのデータ入力形式のチェックなどが必要となる．安全性が高いと認められているモジュールなどをがあれば，積極的に用いるとよい．

3.3 運用管理

システム管理・利用者では，人的要因によるシステム障害やデータ破損，また近年多発している情報流出などについて，意識の向上と具体的な対策について考えていく．例えば，ログインに認証回数制限を設けることは，セキュリティを高めることが可能

な管理方法である．通常は，回数を越えた対応として，ID凍結などを行う．これにより，ブルートフォースアタックなどによる不正ログインに対抗することができる．しかし，パスワードを間違えて覚えていた管理者のIDが凍結されることも予測されるので，IDとパスの照会や，ID凍結解除申請などの機能を備えておく．

近年USBフラッシュメモリが普及し，誰でも手軽により大容量のデータを持ち運ぶことが可能になった．このことも影響して，顧客情報や企業機密などの流出が頻繁に起こり問題となっている．オフィスで仕事を行う際は，仕事用データと私用データを一つのフラッシュメモリに記憶せず，仕事用フラッシュメモリを準備するのが望ましい．仕事は自宅に持ち込まずに，すべて社内で消化することもセキュリティ向上につながる．

さらに，自宅で仕事を行う際は，ファイル交換ソフトを利用しているコンピュータに仕事用データを保存している記憶媒体をを接続しない．あるいは，それらを接続の際にはインターネット回線を切断するなどの対応を行う．

また，社員へのセキュリティに対する教育も重要である．セキュリティに対する知識を深めることにより，ヒューマンエラーや人間の心理を利用したセキュリティ侵害を減らすことができる．例えば，電話でシステム管理者や経営関係者などになりすまして情報を聞き出したり，サーバルームへ不正に侵入したりなどの攻撃が考えられる．また，ゴミ箱から捨てられたテキストを採取し社内の情報を読み取られないように，縦横双方向の比較的新しいシュレッダーで使用済み文書を処理したりと細部にまで注意を寄せることが必要である．

4．応用システムと適用例

4.1 OSSによるシステム開発

近年のオープンソースの急速な発展に伴い，本研究でも安価もしくは無償でありながら，機能的に商用のもと遜色のないオープンソースソフトウェアOSSを活用してシステムを構築していく．OSにはFreeBSD[6]，Webサーバには Apache[7]，DBMSにはPostgreSQL[8]などを利用する．

4.2 CMSを利用したシステム

一世代前と比べ，現在ではインターネットをただ利用するだけでなく，多くの人が自分の持つ情報を発信し，同じ趣味や見解を持つ人々との交流を図り，情報交換を行っている．表現の場もWebサイトや掲示板のみだったものが，ブログやSNSなどの新たな技術の登場により，より簡単に，そして多様な情報の発信を可能としている．

そこで本研究では，コンピュータに関する詳しい知識がなくても自分の情報を発信するシステムであるCMS (content management system) を利用したシステムの開発を行う．

CMSはすでにオープンソースとしてかなりの数のものが存在するので，まずはその選定から行った．初めは海外のものを利用してみた．その後，システム完成後の管理の容易さを考え，島根県が開発した「島根県CMS」[9]を利用することに決定した．日本語環境で開発されたCMSであるため，ドキュメント翻訳の手間が省け，システムに組み込んだあとのマニュアルの作成も容易である．しかし，未習熟の言語であるRubyにより開発が行われているため，新に言語学習を行う必要がある．

4.3 セキュリティ仕様書の適用

これまで学習したセキュリティ要求工学の中からSecure Troposでの要素モデルによるゴールの分析を行い，図１のようなモデルを作成した．丸はアクター，長丸はソフトゴール，四角はハードゴール，八角形はセキュリティ制約をそれぞれ示している．Dは依存を表すDependenceの頭文字であり，Dの向きにアクターが依存していることを示す．

研究室で開発しようとしているシステムは，サーバにCMSをインストールし，それをシステムユーザが利用してウェブサイトを開設するという実験的な物なので，ここでのアクターは開発者・システムユーザ・閲覧者の三者となった．閲覧者はシステムユーザに対して，システムユーザは開発者に対して，個人情報を漏らしてはならいという制約を設ける．

閲覧者が設ける制約とシステムユーザが設ける制約は，同じ制約だが性質が異なる．システムユーザ

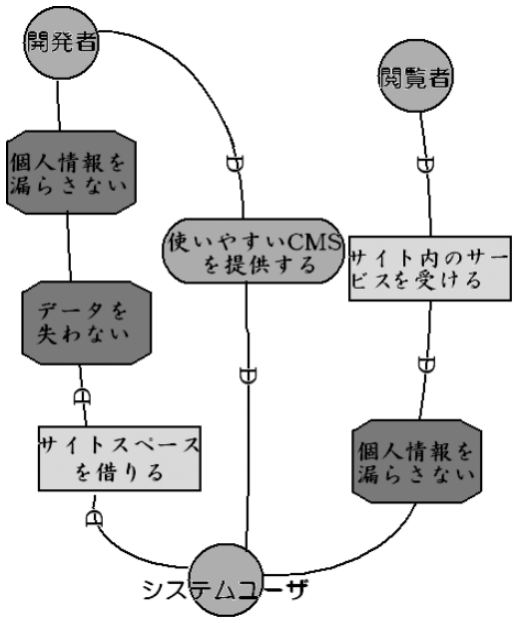

図１　システムのゴール分析

は，閲覧者から得た個人情報をウェブサイト運営に関してのみ利用することで制約を満たすことができる．一方，開発者は，システムユーザの個人情報を外部からの攻撃から守る必要がある．また，閲覧者の個人情報は，開発者が開発したシステム上に記録されるため，開発者の制約には閲覧者の個人情報の保護も含まれることになる．

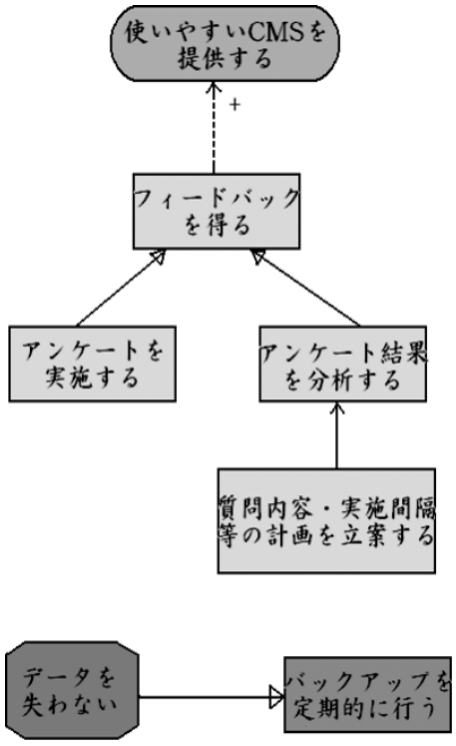

図２　ゴールの詳細分析

図２はゴールを詳細に分解したものである．使いやすいCMSを提供するには，システムユーザからのフィードバックが有効であると考えた．そのためにはアンケート内容を立案し，実施し，結果を分析して追加開発などに反映させるなどの対策が考えられる．

また，データを失わないというセキュリティ制約は，バックアップを定期的に行うという対策によって満足させることができるので，図２のように表した．このときの四角をセキュアゴールという．

５．あ と が き

本報告では，階層化されたセキュリティ仕様書の作成に，セキュリティ要求工学の考え方を取り入れた検討内容を述べた．セキュリティ仕様書はまだ完成していないので，まずはそれを完成させることが必要である．物理的な問題やサーバ内部の問題など取り扱うことは非常に多いものの，できるだけ多くの範囲を網羅した体系的な文書を作成し，今後のシステム開発に役立てたいと考えている．

CMSのシステムはOSやデータベースなどすべてオープンソースにより構築されており，カーネルプログラムレベルでの修正も可能である．ただ，高い技術力が必要であり現状で完全に処理するのは難しい．FreeBSDの高い安全性と柔軟性を活かし，セキュリティ仕様書に記述した項目と照らし合わせながら，外部からの攻撃などを想定した安全対策を施したサーバチューニングを行うためのモデルとしてふさわしいので，今後とも検討を進めていきたい．

参 考 文 献

1) 松本哲始・岡田正：セキュリティに配慮したインターネット応用システムの構築，津山工業高等専門学校紀要 Vol.49 (2007) 47-52
2) 吉岡信和・田口研治（編）：特集・セキュリティ要求工学の実効性，情報処理 Vol.50 No.3 (2009.3)
3) マイクロソフトセキュリティホーム：http://www.microsoft.com/japan/security/default.mspx
4) Haralambos Mouratidis・田口研治：セキュアトロポス (Secure Tropos) 概論，情報処理 Vol.50 No.3 (2009.3) 198-202
5) 田原康之・Axel van Lamsweerde・Emmanuel Letier：KAOSによるセキュリティ要件の獲得・分析，情報処理 Vol.50 No.3 (2009.3) 203-208
6) the FreeBSD Project: http://www.freebsd.org/
7) The Apache HTTP Server Project: http: http://httpd.apache.org/
8) PostgreSQL: http://www.postgresql.org/
9) PrefShimaneCMS- 島根県 CMS 公式サイト : http://projects.netlab.jp/PrefShimaneCMS/