情報セキュリティマネジメントのためのリスク分析ツールの改良

2005MT015 舟橋 篤史　　2006MI168 鈴木 幹也
指導教員　後藤 邦夫

1　はじめに

近年, 情報化・ネットワーク化に伴い, セキュリティ対策の不備に起因する機密情報や個人情報の外部への漏洩, コンピュータウイルス, 不正アクセス行為やシステムダウンによる事業の中断など様々なセキュリティ事故が多発しており, 多くの組織が財団法人日本情報処理開発協会 (JIPDEC) が制定している情報セキュリティマネジメントシステム (以下, ISMS) [1][5][6] の認証を取得している. しかし, ISMS を取得するために, 何百から何千の情報資産を分類し, 評価するには相当な手間がかかる. また, 評価基準を定めないと正しい評価はできない.

本研究では, これから ISMS を導入する企業などの少しでも助けとすることを目的とし, セキュリティマニュアル作成の基本となるリスクマネジメントの部分を自動化するシステムを試作する.

2008 年度丸山, 田口の卒業研究「情報セキュリティマネジメントのためのリスク分析ツールの試作」[4] で基本となるシステムの流れを構築した [3]. この研究では複数のユーザが利用すると想定して基本となるデータベースを作成し, それをもとにユーザ専用のデータベースを複製することで複数のユーザの利用を可能とした. 過去のシステムはユーザがシステムを利用する上でセキュリティ面に欠けるところがある. ユーザログイン時にユーザ名さえ入力すれば誰でも利用できてしまうことや, 過去の利用情報を誰でも閲覧できてしまう点を, データベースの改良と PHP スクリプトの修正と追加をすることから, セキュリティ強化を中心に, システムを構築した. なお, このシステムの考案, 構築は共同で行い, その中でも特に鈴木はリスクアセスメントの部分, 過去のシステムのエラー箇所の修正のプログラミング担当し, 舟橋はユーザ登録部分, リスク算出部分のプログラミングを担当した.

2　情報セキュリティマネジメントシステム

本節では, ISMS 及び, 本研究で提案するシステム (以下, 本システム) の自動化する部分が ISMS でどのように定義されているかを説明する.

ISMS とは, 情報セキュリティの個別の問題毎の技術対策の他に, 組織のマネジメントとして, 自らのリスクアセスメントにより必要なセキュリティレベルを決め, プランを持ち, 資源配分して, システムを運用することである. 2002 年 4 月から本格的に運様が始まり, 現在日本国内で ISMS を取得している組織数は 3342 あり (2009 年 12 月 3 日現在), 多くの組織が取得することにより組織のセキュリティーレベルの向上を計ろうとしていることが分かる [6].

組織が保護すべき情報資産について, 機密性, 完全性, 可用性をバランス良く維持し改善することが ISMS の基本コンセプトである. 機密性の定義は, 「認可されていない個人, エンティティ又はプロセスに対して, 情報を使用又は非公開にする特性」と規定され, 完全性の定義は, 「資産の正確さ及び完全さを保護する特性」と規定されている. また, 可用性の定義は,「認可されたエンティティが要求したときにアクセスが可能である特性」と規定されている. さらに, 情報の機密性の評価値とは,「情報資産が漏洩した場合の影響度」として捉えることができ, 情報の完全性とは, 情報資産が滅失・毀損した場合の影響度として捉えることができる. また, 情報の可用性の評価値とは, 「情報資産が利用できない場合の影響度」として捉えることができる. 情報資産の重要度に応じ, それに必要な対策をとり維持・監視することで, 重要な情報資産をリスクから守り機密情報の漏洩を未然に防止することができる.

ISMS で採用されているモデルは, 「Plan-Do-Check-Act(PDCA) モデル」である. 本システムで自動化するのは, その中でも Plan(ISMS の確立) のリスクマネジメントの部分である. その部分を以下に示す.

1. リスクを特定する
2. リスクを分析し, 評価する
3. リスク対応のための選択肢を特定し, 評価する
4. リスク対応のための管理目的及び管理策を選択する

この ISMS を認証取得する場合, 「外部のコンサルタントを利用して取得する方法」と, 「自社で取得活動をする方法」が一般的である. しかし, 多大なコストが掛かかることや, 独自に取得しようとした場合では時間がかかる. そこで, 本システムを利用し, 特に手間の掛かる資産選択やリスク・コストなどの算出の仕訳と計算をシステムに任せることで, より安く, より時間を掛けずに ISMS 取得へ取り掛かることができる.

3　システムの実現

本節では, システムの構築による効果と詳細について説明する.

3.1　過去のシステム問題・改良点

はじめに, システムの改良点を述べる.

1. 新規ユーザ登録, ユーザログイン時のパスワード認証についての改良.
2. ユーザ登録時のユーザ専用テーブルの作成.
3. ユーザ利用結果の表示.
4. 専門用語, 数値入力の説明案内.

5. 資産名・脅威名・脆弱性名の新規追加登録機能.
6. 文字化けや, システムエラーの修正.

次節以降, 改良点の詳細を示す.

本システムでは, データベースとして PostgreSQL を使用し, Web アプリケーションを実現するためのスクリプト言語として PHP を使用する. PHP を選択した理由としては, コンパイルを必要としないこと, HTML に埋め込むことができること, データベースの連携に優れているといえる.

過去のシステムでは, セキュリティ面における安全性に欠けることが一番の問題と考えられる. ユーザ登録時にパスワード認証がなく, 本人以外でもユーザ名さえ分かれば他人の情報を閲覧かつ, システム利用ができてしまう.

またユーザ登録時に各ユーザ専用のデータベースを作成していたため, データベース数がユーザの数に比例して増えることから, 無駄な処理が多く, また容量の圧迫になる.

本システムでは, ユーザログイン時に過去の利用結果一覧を表示可能にした. また他の利用者に自分の利用情報を見られないように, 個々の結果を格納するテーブルを新規ユーザ登録ごとに作成し, セキュリティ面を強化した. 知識の無い人にとっては専門用語など, 何を入力して良いか分かりにくくなっていたため, 簡単な用語の説明などもつけた.

また利用者間で新しい資産・脅威・脆弱性の情報の共有をするため, 本システムでは利用者が資産名・脅威名・脆弱性名の登録を利用者が追加することができるようにした.

これらも踏まえ, 本研究では, 過去のデータベースと PHP のエラーを見直し, 実際に稼働していなかったシステムを, 修正・改良することから, システムの実装を可能にした.

次節以降, 問題点の改良を示す.

3.2 パスワード認証

パスワード認証を追加することによって本人以外がシステムを利用できないようにする. パスワード認証の流れを図 1 に示す. 新規ユーザ登録の際にユーザ名並びにパスワードを入力する. 入力後, 次のページに入力した情報を送信し, パスワードを暗号化関数である crypt 関数 [2] を用いてハッシュ化する. 暗号化式は, salt 引数によってきまる. 暗号化した文字列と同時に入力されたユーザ名を 'usr' のテーブルに格納し, 登録作業をする. なお登録ページで未入力があった場合はもう一度作業を繰り返す. ここで正常に登録ができれば, 作成したテーブルに insert した作業となる.

メニューページからはユーザのログイン作業ができる. あらかじめ登録したユーザ名とパスワードを入力し, 次のページに送信する. 入力されたユーザ名, 並びにパスワードを受け取ったページでは select の作業をする. ユーザ名と暗号化されたパスワードを 'usr' のテーブルから検索し, 入力したパスワードとの比較をし, 格納されたものと同じであれば, 資産の選択ページに遷移し, システムの利用ができる. また一致しない場合はエラーページを表示し, ログイン画面に戻す.

3.3 ユーザ専用テーブル・データベース

本システムでは, 新規ユーザ登録時に, ユーザ名と暗号化されたパスワードを格納するための, ユーザ名とパスワードは 'usr' のテーブルに格納する. 本システムでは新規ユーザ登録時に基本となるテーブル 'my_select', 'select_manage' をコピーし, 新たなユーザ専用のテーブルを作り, 情報を格納する. これにより毎回データベースを作成する無駄がなくなる. また個人の利用情報を他人に見られないようにすることができた. 作成されるテーブルは, 入力したユーザ名が反映される. 例えば登録されたユーザ名が, 「 nanzan 」 であった場合, 'my_select' をコピーしたテーブルが'nanzan_select', 'select_manage' をコピーしたテーブルが 'nanzan_manage' となる. 'my_select' には, 資産名・脅威名・脆弱性名・機密性・完全性・可用性・脅威評価・脆弱性評価・許容範囲・評価の情報を格納し, 'select_manage' には資産名・管理策・機密性・完全性・可用性・脅威評価・脆弱性評価・手間・コストの情報を格納する. これらのテーブルについてのデータ定義を表 1 に示す. またデータベースの繋がりとして, ER 図を図 2 に示す. 'my_select' と 'select_manage' の 2 つのテーブルがあるのは, リスク値算出時に, 自ら決めたリスク値の許容範囲内であれば ' ユーザ名_select' に登録し, 許容範囲外であれば, 管理策, コスト, 手間と新たに見直した脅威と脆弱性の評価を登録するために ' ユーザ名_manage' に格納する. また, 本システムでは, セキュリティ面を重要視するため, ユーザ名とパスワードによるログインをし, たユーザのみが過去の利用状況, 利用結果を閲覧できるようにした.

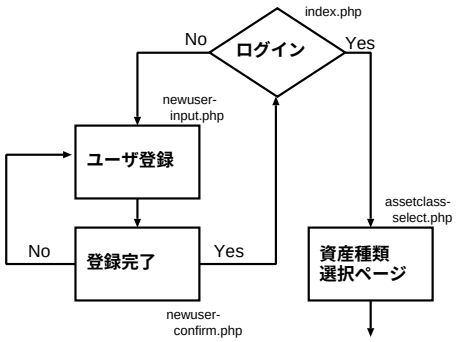

図 1 パスワード認証の流れ

1. 新規登録時にパスワードのハッシュをする.
2. パスワードとユーザ名を 'usr' のテーブルに格納.
3. 利用者情報格納テーブル, ' ユーザ名_select', ' ユーザ名_manage' の作成.
4. ログイン時にユーザ名とパスワードの入力.

5. ユーザを 'usr' のテーブルから検索.
6. パスワードを 'usr' のテーブルから検索.
7. 格納されているパスワードと入力したパスワードのハッシュしたものを比較.

先ほど述べたように, 表 1 にデータ定義の一覧を示し, ER 図を図 2 に示す.

表 1　データ定義一覧

テーブル名	項目名	
資産リスト	*資産名	資産分類
分類リスト	*資産分類	脆弱性分類
	脅威分類	
脆弱性リスト	*脆弱性分類	脆弱性名
脅威リスト	*脅威分類	脅威名
管理策リスト	*分類 no	*目的 no
	*項目 no	項目名
	管理策	脆弱性
管理策目的リスト	*目的 no	目的分類
	目的名	
管理策分類リスト	*分類 no	分類名
usr	*uname	passwd
my_select	*資産名	機密性
	完全性	可用性
	脅威名	脅威評価
	脆弱性名	脆弱性評価
	許容範囲	評価
select_manage	*資産名	管理策
	機密性	完全性
	可用性	脅威評価
	脆弱性評価	手間
	コスト	

（ *は主キー ）

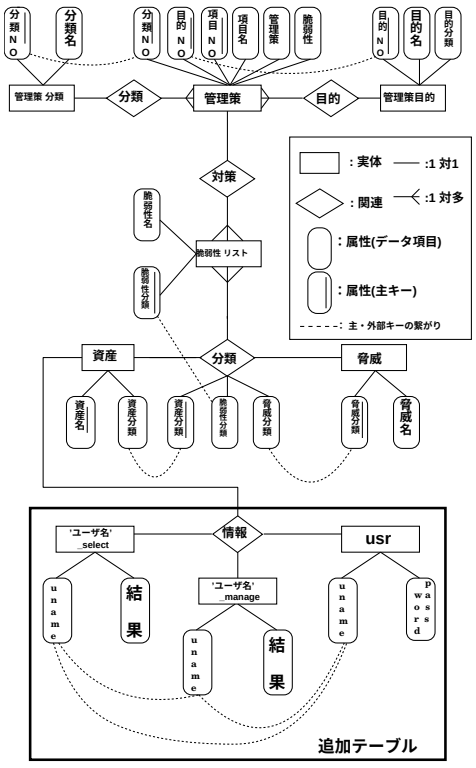

図 2　ER 図

3.4　システムエラー修正

過去のシステムでは 文字指定がされていなかったので, 文字化けが発生し, システム利用中に障害を与えていた. 本システムでは, 文字化けによる障害を無くすため, UTF-8 に文字指定をし, 全体を統一した. またリンク先の間違いにより, システム終了まで行かないパターンを回避するため, システムの流れを見直した. システムの流れを図 3 に示す.

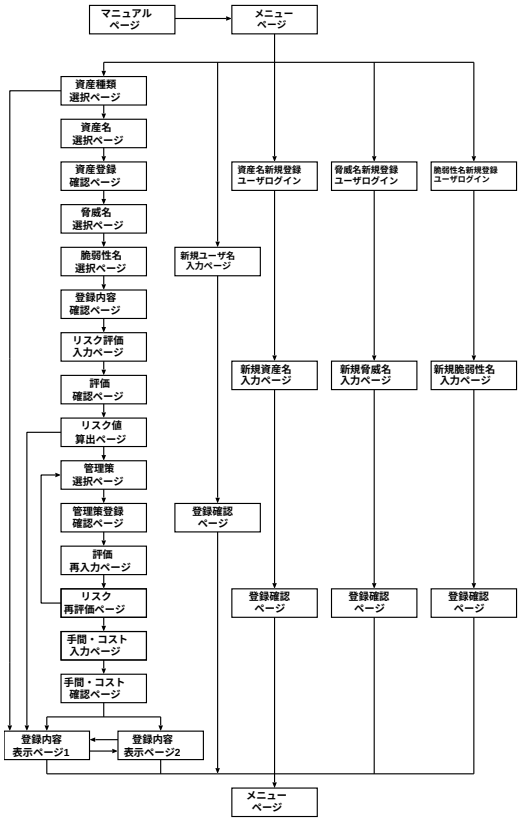

図 3　システムの遷移図

4　システムの評価

本節では, システムの評価について述べる.

4.1　作成者の評価

作成者の視点からの評価では, システムが必要な用件を満たしているか機能テストと例外処理のシステムテストをする. Web アプリケーションでは, ある入力やアクションに対して表示される結果や実行される処理が正しいかどうか確認するテストが機能テストである. 例外処理テストとは, エラー処理機能やエラーからの回復機能が正常に動作するかを確認するテストである. 今回の機能テストでは, 画面の遷移, 画面表示, データベースの更新確認をした. 画面の遷移を確認する過程で同時に画面表示の確認もできた. データベースの更新の確認は, 画面の遷移, 画面表示の確認終了後に, 確認作業中に登録された内容が正しく更新されているかを確認した.

4.2 利用者の評価

利用者の視点から本システムの評価する方法を説明する.

実際に利用者にシステムを使ってもらい, アンケートを実施する. アンケートには情報分野を専攻する学生 17 人, 並びに情報分野以外の専攻, つまり専門知識のない学生 31 人を対象に実施した. 次に評価のポイントを述べる.

- システムの使いやすさ
  - 操作 (進んでいく中で操作に困ることが無かったか等)
  - 入力 (入力内容に困ることが無かったか, 最低限の入力に抑えられているか)
  - 誘導 (困ること無く進められたか, 間違えた場合の誘導は適切だったか等)
- システムの性能
  - リスクの分析の役に立ったか
  - このツールを利用することでリスク分析の手間は軽減されたか

4.3 アンケート・評価結果

評価は, 1 (Bad) 〜 4 (Good) の 4 段階で評価する. 4 段階評価にした目的は, 偶数段階であれば, 必ず良いか悪いかを評価しなければならないからである. まず, 評価結果の全体の平均値, 平均 A(情報分野専門学生の全体平均), 平均 B (情報分野非専門学生の全体平均) を表 2 に示す.

表 2　評価の平均値

評価項目	平均値	平均 A	平均 B
操作	3.23	3.11	3.29
入力	3.10	3.05	3.13
誘導	3.25	3.29	3.23
役に立ったか	2.10	2.71	1.77
手間が軽減された	3.29	3.53	3.16
適切な管理策が作成された	2.90	2.88	2.90

次に利用者からのコメントを示す.
システムを利用したうえでの感想のまとめ.

- 実際にたくさんのチェック項目があるため, システムを使わないで資料などから分析するのは大変であると感じた.
- インターフェイスは見やすかったが, 知識の無い人だと何をしているか分かりにくいと思う.
- 何を入力に関していくかは分かったが, 実際に ISMS を取得する予定はないのでこのシステムの利用価値を感じにくい.

以上より, 専門用語などに説明を加え専門知識の無い人にとっても分かりやすくしたが, 実際入力には困ることは無くても, ISMS を身近に感じていないため, システムの利用価値を感じにくいと分かる. 特に今回のアンケートは専門知識のある人と無い人の割合を 1 対 2 で実施したため, 評価の結果も「役に立った」の項目が極端に低くなっていると考えられる. 実際, システム利用前に ISMS のコンセプトを説明する必要もあった.

しかし, 操作や入力に関しての性能評価は高かった. 実際に何を入力したら良いかなどは, 分かりにくい箇所にリンクを付け, 誘導を行ったからであると考えられる.

5 おわりに

本研究では, リスク分析とリスク算定と管理策選択の自動化を過去のシステムを基に改良することから, システムを作成し, 実現することができた.

Web アプリケーションとデータベースの設計を見直し, パスワード認証によるログインシステムのためのパスワード暗号化を使った新規ユーザ登録システムを作成した. また新規ユーザ登録ごとに利用データを格納するテーブルを作成し, 利用結果表示時も, ログインしたユーザのみしか閲覧できないようにした. よって目的であった, ユーザ管理におけるセキュリティ面の向上を計ることができた.

また, 登録されていない資産・脅威・脆弱性を利用者が新しく追加できるようにし, 利用者間での情報の共有を可能とした. 実際, 過去のシステムでは, プログラム自体に欠陥がありシステムが利用できないトラブルがあった. 最終的には, 修正をし, システムの利用を可能とした. 現状, 世間一般では ISMS について無知な人が多いことは否めない, 今後システム利用も踏まえ, より多くの人に ISMS を知ってもらう必要がある.

参考文献

[1] @ IT －アットマーク・アイティ：情報セキュリティマネジメントシステム基礎講座, http://www.atmarkit.co.jp/index.html (accessed Aug 2009).

[2] Cephid：crypt による暗号化の基礎, http://www.ss.iij4u.or.jp/˜somali/index.shtml (accessed Aug 2009).

[3] 富士通ソーシアルサイエンスラボラトリ：リスク分析・対策立案ツール, http://www.ssl.fijitsu.com/products/network/bs7799/racontis-web/ (accessed Aug 2009).

[4] 丸山美希, 田口佳美：情報セキュリティマネジメントのためのリスク分析ツールの試作, 卒業論文, 南山大学数理情報学部情報通信学科 (2008).

[5] 日本規格協会/編：対訳 ISO/IEC27001：2005(JIS Q 27001：2006) 情報セキュリティマネジメントシステム

[6] 財団法人日本情報処理開発協会 (JIPDEC)：情報マネジメントシステム推進センター, http://www.isms.jipdec.jp/ (accessed Aug 2009).