Vol.2015-DPS-162 No.6
Vol.2015-CSEC-68 No.6
2015/3/5

ベイジアンネットワークを利用した動的モデリングによるセキュリティリスク評価システムの開発

磯部義明[†1]　　杉本暁彦[†2]

年間約 5,000 件の脆弱性情報が米国 NIST から公開されており，情報システムの管理者は膨大な脆弱性情報からシステム固有の脆弱性に対するセキュリティリスクを評価し，優先度をつけて効率よく必要な対策を行う必要がある．NIST などが公開する脆弱性情報には，その脆弱性の技術的な特性のみに基づいて脅威を評価した CVSS 基本評価値が提示されている．しかし，管理対象のシステム構成において，その脆弱性に対するリスクや対策の要否，対策の優先度等の判断はシステム管理者に委ねられており，課題があった．そこで，本研究では収集したシステム情報から，ベイジアンネットワークによるリスク評価モデルを自動生成し，ネットワークトポロジーを含むシステム構成に基づいたセキュリティリスク評価システムを開発した．本報告では，開発したセキュリティリスク評価システムと，その評価結果について報告する．

Development of Security Risk Analyzing System based on Dynamically Modeling using the Bayesian Network

Yoshiaki ISOBE[†1]　Akihiro SUGIMOTO[†2]

About 5,000 vulnerabilities were disclosed in 2013 by the National Institute of Standards and Technology (NIST) of USA. As soon as vulnerabilities are disclosed, cyber-attacks that exploit the vulnerabilities increase suddenly. So system engineers must prioritize the vulnerabilities to deal with efficiently. Common Vulnerability Scoring System (CVSS) was standardized for risk assessment. But risk assessment for individual systems is entrusted to system engineers. Therefore we developed a risk assessment system that automatically makes modeling for risk assessment based on system configuration. This article introduces the risk assessment system and the assessment result.

1. はじめに

近年では，年間約 5,000 件のソフトウェア脆弱性情報が NIST(National Institute of Standards and Technology)[1]から公開されている[2]．一般的に，これら脆弱性情報が公開されると，公開直後から同脆弱性情報を利用した攻撃が急増する傾向にある．そのため，情報システムを運用するシステム管理者は膨大な脆弱性に対してリスクを評価し，優先度をつけて効率よく対策していく必要がある．

ソフトウェア脆弱性を評価する指標としては，CVSS(Common Vulnerability Scoring System)[3]と呼ばれる評価指標が存在する．CVSS は，脆弱性の技術的な特性に基づく基本評価(Base Metrics)，脆弱性を取り巻く状況に基づく現状評価(Temporal Metrics)，個々のシステム構成に基づく環境評価(Environmental Metrics)から構成される．一般的に，NIST などが公開した脆弱性情報には，脆弱性の技術的な特性に基づいてセキュリティ専門家が評価した CVSS の基本評価値のみが付されている．

一方で，CVSS の環境評価は，個々のシステム構成に基づく洞察が必要なため，システム管理者に委ねられていた．しかし，個々のシステムのリスク評価には，同システムに対する知識とセキュリティ知識の両方が必要となり，必ずしもシステム管理者が両知識を有しているとは限らない．そのため，システム管理者が脆弱性対策を行う上で，脆弱性がもたらすリスクを正しく評価できず，効率よく対策できない現状がある．

そこで，我々は，収集したシステム情報からリスク評価モデルを自動生成し，システム構成に基づいたリスク評価を行うシステムについて提案した[4]．本稿では，リスク評価システムの実装とその評価結果について説明する．

2. 脆弱性のリスク評価指標

2.1 CVSS

CVSS は，脆弱性の技術的な特性に基づく基本評価，脆弱性を取り巻く状況に基づく現状評価，個々のシステム構成に基づく環境評価から構成される．例えば，基本評価では，攻撃元区分（Access Vector）として，脆弱性攻撃の際に攻撃者（攻撃ノード）と攻撃対象（対象ノード）の間で必要となるリモート接続のタイプに応じて，３段階で評価されている．

CVSS の基本評価は，NIST や IPA が公開する脆弱性情報に付されているため，システム管理者にとって重要な評価指標であるが，基本評価のみに従って，脆弱性対策の優先度付けをすることは望ましくない．具体的には，図 1 のような例において，適切にリスクを評価できない．

†1 (株)日立製作所 横浜研究所
Hitachi Ltd., Yokohama Research Laboratory
†2 (株)日立製作所 横浜研究所
Hitachi Ltd., Yokohama Research Laboratory

Vol.2015-DPS-162 No.6
Vol.2015-CSEC-68 No.6
2015/3/5

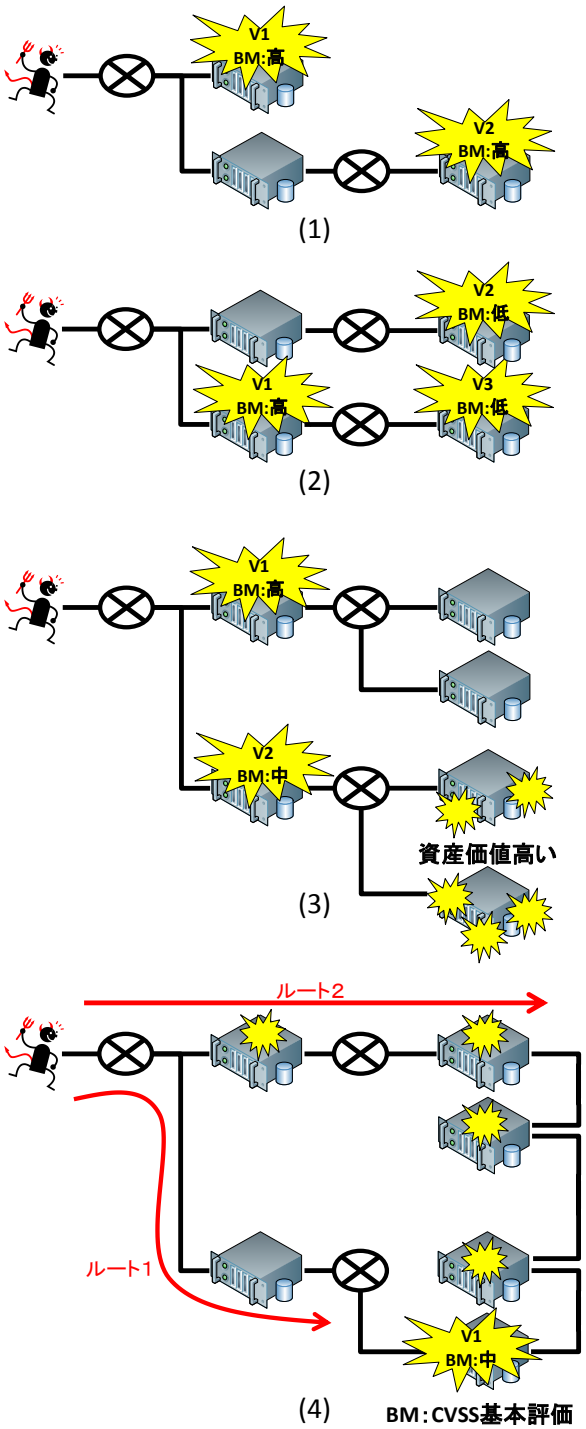


図 1　CVSS の基本評価のみでは評価しきれないリスク例

1. 技術的な観点からは脆弱性 V1，脆弱性 V2 が同程度のリスクとして評価される場合がある．しかし，攻撃者にとって，ネットワーク経由で直接アクセス可能な脆弱性 V1 を有する機器への攻撃の方が遥かに容易であり，システム構成の観点からは脆弱性 V1 のリスクの方が高い．
2. (1)同様，脆弱性 V2，脆弱性 V3 が同程度のリスクとして評価される場合がある．しかし，攻撃者にとって，脆弱性 V1 を攻略することで踏み台とできる機器が存在する分，脆弱性 V3 へ攻撃する方が容易である．このように脆弱性の分布状況観点からは脆弱性 V3 のリスクの方が高い．
3. 技術的な観点からは脆弱性 V1 のリスクの方が，脆弱性 V2 のリスクより高いと評価される場合がある．実際に，同脆弱性を攻略するだけであれば，脆弱性 V1 に対する攻撃の方が容易と考えられる．しかし，資産や他脆弱性の分布状況により，同脆弱性が攻略されることで，受ける影響が大きい場合，脆弱性 V2 のリスクを高く見積もる方が良い場合がある．
4. CVSS の基本評価のような技術的な観点にネットワーク階層構造の観点を加えて，リスク評価する方法も考えられる．しかし，攻撃者が脆弱性 V1 を攻撃するまでのパスが複数経路あるようなケースでは，適切にリスクを評価できない．

2.2 アタックグラフ

図 1 のようなリスクを分析する手法として，アタックグラフ(Attack Graphs)[5]と呼ばれる手法が存在する．アタックグラフとは，各機器における脆弱性や情報資産の保有状況，機器間のネットワーク接続性，アプリケーションサービスの稼働状況などから，脅威と脆弱性の関係をグラフモデル化する手法であり，アタックグラフによって脅威や脆弱性の関係を可視化されることで，適切なリスクの評価が可能になる．

アタックグラフは，当初セキュリティ専門家が机上においてセキュリティ分析する手段として，様々なモデルが検討され，近年では，要素技術として機械的にアタックグラフを生成する方式も検討されてきた[6][7]．しかし，文献[6][7]では，システム構成情報の取得から，公開されたセキュリティナレッジの収集，動的なモデルの構築，リスクの評価までを自動化することはできておらず，情報システムを管理するシステム管理者に対するセキュリティ運用支援システムとして，実用には至っていなかった．

以上より，本研究では，システム構成情報の取得から，公開されたセキュリティナレッジの収集，動的なモデルの構築，リスクの評価までを自動化することを目的とする．

3. 提案システムにおけるリスク評価モデル

本研究では，ペトリネット(Petri Net)[8]と呼ばれるグラフモデルを用いて，脅威と脆弱性の関係を構造化し，リスク評価するシステムを提案した[4]．本章では，提案するリスク評価モデルについて説明する．

3.1 提案した動的モデリング方式

ペトリネットとは，図 2 に示すように，物事の状態を“プレース”，発生する事象を“トランジション”，状態と事象の接続関係を“アーク”，事象発生した場合にアークで接続された状態に遷移する確率を“発火確率”と定義し，システムをモデル化したグラフモデルである．

Vol.2015-DPS-162 No.6
Vol.2015-CSEC-68 No.6
2015/3/5

ペトリネットは，一般的な有向グラフと比べ，複数の事象が並列的に生起した場合を前提条件として，状態遷移が発生するようなシステムのモデル化を可能とする．

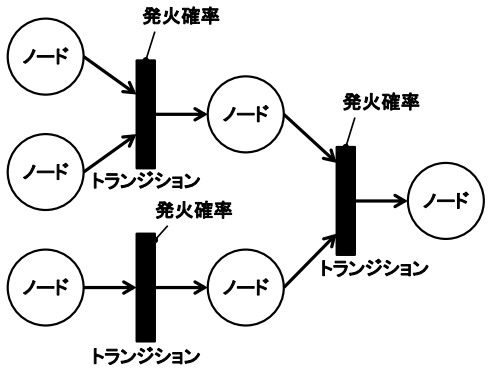


図 2　ペトリネットの例

提案方式では，上記ペトリネットを用いて，モデル化する上で，以下を定義する．

- 各機器において Exploit コードが実行できる状態，各機器が保有する認証情報が盗難された状態など，各機器が悪意のある攻撃者に侵害された状態をペトリネットにおける"プレース"として定義する．
- 特定の機器から特定の機器への攻撃手段をペトリネットにおける"トランジション"として定義する．攻撃手段としては，脆弱性をついた攻撃や，別機器から盗難した認証情報によるリモート操作，ハードニング不足によるセキュリティ設定の穴をついた攻撃などが考えられる．
- ネットワークトポロジに基づき，機器間においてメッセージの到達性があり，上記攻撃手段の前提条件に合致する"プレース"から"トランジション"へ"アーク"を接続するとする．例えば，上記攻撃手段が Exploit コードの実行を必要とする場合，Exploit コードの実行を表す"プレース"から同攻撃手段を表す"トランジション"に"アーク"が接続される．
- 上記攻撃手段により，発生する状態を表す"プレース"へ同攻撃手段を表す"トランジション"から"アーク"を接続するとする．例えば，脆弱性攻撃により Exploit コードの実行が可能となる場合，同脆弱性攻撃を表す"トランジション"から Exploit コードの実行を表す"プレース"へ"アーク"を接続する．

提案方式では，上記の定義により，機器間の関係性を図 3 のようなグラフモデルと構築する．

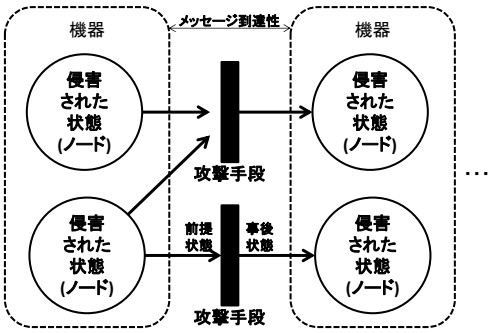


図 3　ペトリネットによるグラフモデルの例

3.2 提案方式の機能要件

上記グラフモデルを動的にモデリングするため，提案方式は，以下の 5 つの機能が必要と考える．

既報[4]では，F1～F3 まで，機能実装と検証について報告した．本報告では，F1～F3 の実装について概説し，続いて，F4，F5 の実装について詳述する．

F1.　脆弱性情報の収集・管理機能：インターネット経由でセキュリティナレッジ公開機関からソフトウェア脆弱性情報を収集する機能．

F2.　システム情報の収集・管理機能：管理システムから機器情報やソフトウェアスタック情報などのシステム構成情報を収集する機能．

F3.　機器と脆弱性の対応付け機能：機器情報と脆弱性情報をセマンティックに対応付ける機能．

F4.　システムリスクのモデル化機能：ペトリネットにより脅威と脆弱性の関係性をグラフモデル化する機能．

F5.　システムリスク値の計算機能：上記，グラフモデルからリスク値を計算する機能．

4. 提案方式を用いたシステムの実装方法

本章では，3.2 の各機能の実装について説明する．提案システムの全体構成については，図 4 に示す．

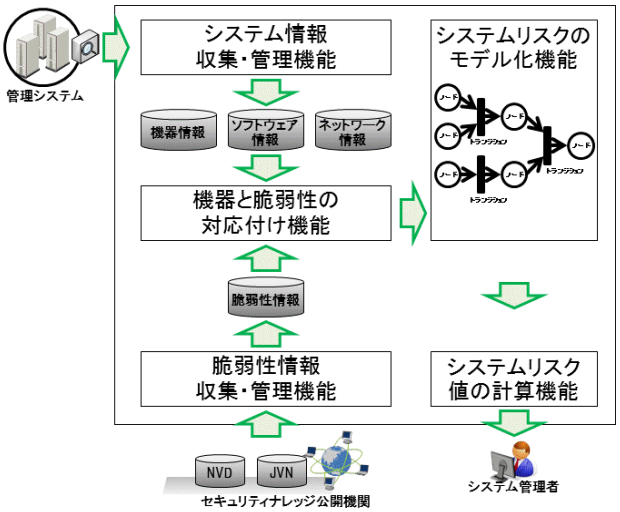


図 4　提案システムの全体構成

Vol.2015-DPS-162 No.6
Vol.2015-CSEC-68 No.6
2015/3/5

4.1 脆弱性情報の収集・管理機能

本機能は，インターネット経由で公的な脆弱性情報リポジトリである JVN（Japan Vulnerability Note）[9]や NVD（National Vulnerability Database）[2]から脆弱性情報を取得する機能である．これらサイトから，XML 表記で構造化された脆弱性情報を取得することが可能である．

4.2 システム情報の収集・管理機能

本機能は，脅威と脆弱性の関係をモデリングするために，システム情報として，機器情報，ソフトウェアスタック情報，ネットワーク情報を収集する機能である．各情報は以下の方法によって収集する．

1. 機器情報

   機器情報に関しては，機器にエージェントを導入することにより，OS の標準機能を利用することで様々な情報が取得可能である．例えば，弊社では，一般的なシステム管理に必要な機器情報を OS の標準機能を利用して，取得する方法と取得するツール(IT Report Utility)[10]を一般公開している．本システムでは，IT Report Utility に基づいて機器情報を取得する．

2. ソフトウェア情報

   提案システムでは，上記の IT Report Utility にて収集可能なソフトウェア情報に加え，システムの管理状況に応じて，Windows のレジストリ情報や Linux のパッケージ管理情報を取得するコマンド出力により，収集する．

3. ネットワーク情報

   ネットワークトポロジに基づき，機器間においてメッセージの到達性を判定するため，ネットワーク機器の情報も必要となる．ネットワーク機器情報の取得方法としては，IETF により標準的なプロトコル SNMP(Simple Network Management Protocol)[11]を利用した収集を行う．SNMP では，MIB と呼ばれるオブジェクトの階層構造を取るように，個々に情報に対して，Identifier が割付けられており，Identifier を指定することで情報の参照が可能になる．これにより，ポート情報，ネットワークインターフェース情報，IP アドレス情報，arp キャッシュ情報，MAC テーブル情報を取得する．

4.3 機器と脆弱性情報の対応付け機能

本機能は，インターネット経由で取得した公的な脆弱性情報と，機器から取得したソフトウェア情報とを対応付けし，機器ごとの脆弱性を識別する機能である．

本機能の実装は，機器から取得したソフトウェア情報は CPE 辞書[12]に基づいて置換して管理しておき，NIST IR 7696 で規格化された CPE ベースの照合アルゴリズム[13]に基づいて機器ごとの脆弱性を識別する機能の実装を行った．

4.4 システムリスクのモデル化機能，および，システムリスク値の計算機能

本機能は，脅威と脆弱性の関係をグラフモデル化する機能である．本稿ではベイジアンネットワークの利用したグラフモデル化により，システムリスク値を各ノードに対する攻撃到達の確率値を算出する実装を行った．

4.4.1 ベイジアンネットワークの適用

ベイジアンネットワークは，生起事象についての因果関係を条件付確率により解析し，不確実な環境下でユーザーの意思決定を支援する人工知能ツールとして注目されている．ベイジアンネットワークでは，原因と結果をノードとし，ノード間の関係を条件付き確率で結んだグラフモデルにより，複雑に絡み合った事象とその原因の関係を解析する．

ベイジアンネットワークでは各ノードへの隣接ノードからの遷移確率（条件付き確率）を入力することで，最上位ノードから任意のノード n までの遷移確率を計算することができる（図 5）．

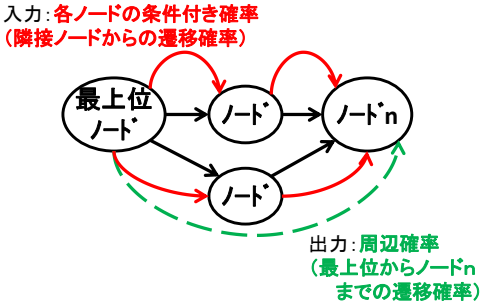


図 5　ベイジアンネットワークによる周辺確率計算例

各機器をノードとし，さらに機器の脆弱性を隣接ノードのトランジションとして遷移確率を設定する確率ペトリネットを構築し，各ノードの攻撃到達可能性を求める提案方法をベイジアンネットワークにて実装した．具体的には確率ペトリネットの各ノードと各トランジションをベイジアンネットワークのノードとして定義し，最上位ノードから各機器への攻撃到達可能性を周辺確率計算によりスコアリングすることで，確率ペトリネットを簡易モデル化した実装を行った．

4.4.2 処理の概要

処理の概要ステップは以下の通りである．

1. ネットワークトポロジの構築

   4.2 節にて収集したネットワーク情報を相互に突き合わせて，機器間の可達性をモデル化したネットワークトポロジを構築する

2. 非循環有向グラフの構築（図 6）

   各機器と 4.3 節で識別された各機器の脆弱性をそれぞれノードとして接続し，ネットワークトポロジ上に攻撃の起点を設定し，この起点からネットワークトポロ

ジの可達性モデルに基づき，可達する機器の脆弱性の有無により，アークを接続し，有向グラフを構築する．ここで，ベイジアンネットワークの制約から，「ノードの Hop 数が少ない機器から多い Hop 数にのみアークを接続する」ことを制約条件として，非循環有向グラフとして構築する

3. 条件付き確率表の設定
各ノードにベイジアンネットワークで計算するための条件付き確率表を設定する．ここで，機器ノードに関しては，各脆弱性のどれかにより侵害された状態か否かの 2 値となるように定義する（表 1）．脆弱性ノードに関しては，CVSS 基本評価値のベクトル要素の一つである攻撃容易性の値を利用し，例えば，表 2 のように設定する．

4. ベイジアンネットワークによるリスク値の算出
ベイジアンネットワークライブラリ[14]を利用し，任意の攻撃起点ノードから各ノードの周辺確率（遷移確率）を算出し，各ノードまでの攻撃到達可能性を数値化する．

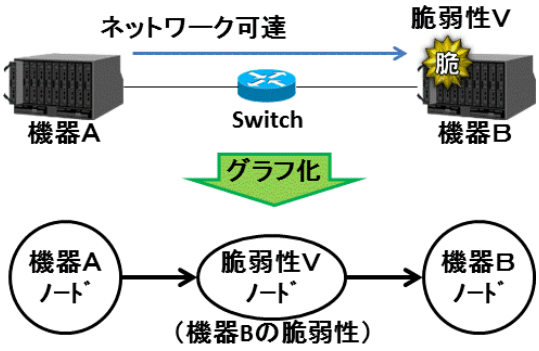


図 6　非循環有向グラフの構築

表 1　機器の持つ脆弱性Ａ，Ｂの侵害状況による機器ノードの条件付き確率表の例

	脆弱性A	True		False	
	脆弱性B	True	False	True	False
機器ノード	True	1.0	1.0	1.0	0
	False	0	0	0	1.0

表 2　隣接機器Ａ，Ｂの侵害を条件とした脆弱性ノードの条件付き確率表の例

	機器A	True		False	
	機器B	True	False	True	False
脆弱性ノード	True	0.7	0.7	0.7	0
	False	0.3	0.3	0.3	1

ベイジアンネットワークを利用することにより，複数経路があるノードに対しても，確率で正規化されたリスク値として，算出することが可能である．

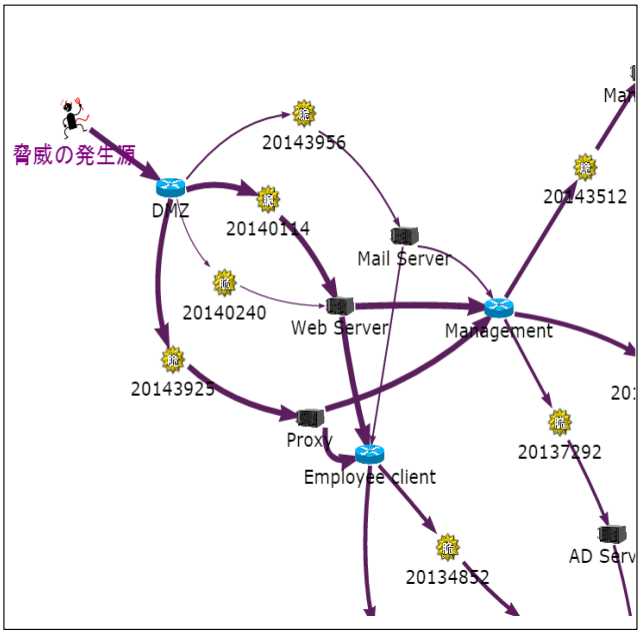
5. 評価実験

5.1 評価対象の管理システム

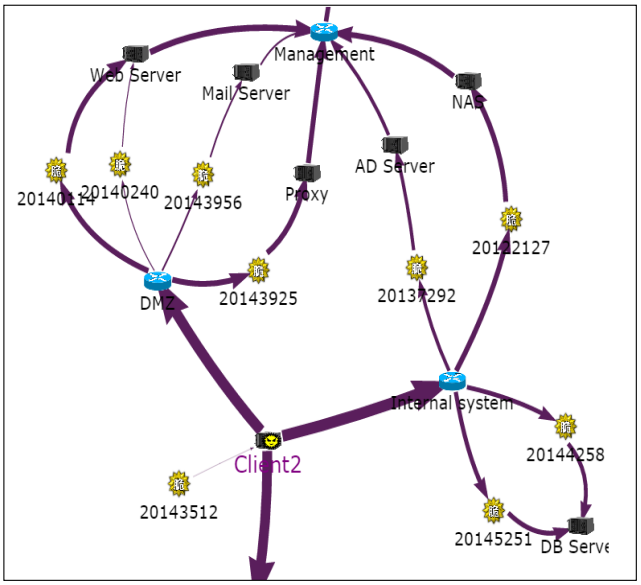
近年，多くの業務システムに採用されているＷｅｂ三階層システムで構築されたシステムを対象とする．ネットワークは，ＤＭＺ，管理されたネットワークゾーン，ユーザー（従業員）クライアントのネットワークゾーンで構成されているものとする．

5.2 評価結果および考察

図 7 に開発したプロトシステムのリスク評価結果のふたつの例を示す．ここで，脅威発生源からの各機器への攻撃到達可能性を遷移確率で計算・スコアリングした数値に基づき，矢印の太さで相対表示した表示例である．



（1）外部からの直接攻撃シナリオ



（2）マルウェアに侵害された機器からの攻撃シナリオ

図 7　リスク評価結果の例

（1）対策優先度の判定

評価システムにより，各機器の脆弱性ごとに脅威発生源からの攻撃到達可能性の寄与の大小がリスク値により明らかになる．図 7(1)は，インターネット介した外部ネットワークから各機器への攻撃到達可能性を評価した例である．これにより，脆弱性対策の優先度の判定が支援できるものと考える．

（2）様々な脅威シナリオへの対応

脅威シナリオにより脅威発生源を任意の機器ノードに変更することにより，動的に各脆弱性のリスク値が更新される．図 7(2)は，マルウェアに侵害された機器（ここでは，Client 2）からの攻撃到達可能性を評価した例である．このように，マルウェアインシデント発生時の影響範囲の特定業務を支援できるものと考える．

一方で、情報資産を考慮したリスク評価については、本評価システムでは扱っていない．この課題については，機器ノードに情報資産を割り付け，情報資産に対する機密性，完全性，可用性に対する要件を明確にすることで，リスク評価モデルを拡張可能と考える．

また，ベイジアンネットワークの制約から Hop 数により循環性を排除したグラフモデルによりスコア算出を行ったため，Hop 数が多い経路の方がより脆弱な場合に適切なリスク評価ができないことが想定される．この課題については，脅威発生源の機器ノードをこの脆弱な経路に沿って変更し，リスク評価を行うことで対応できる．

6. おわりに

本稿では，情報システムの管理者が公開された膨大なソフトウェア脆弱性の対策を支援するため，システム情報や脆弱性情報を収集し，自動的に脆弱性のリスク評価モデルを構築して，リスクを評価することで，脆弱性に優先度付けし，セキュリティ運用を支援するシステムの開発について報告した．

本稿では，特に，公開されている脆弱性情報に付加されている CVSS の基本評価のみに基づいたリスク評価の問題点について，システム構成情報に基づいた，脅威と脆弱性の関係性を表すモデルを構築して評価する方法を提案し，ベイジアンネットワークを利用したプロトタイプ開発について報告した．

本研究では，さらに，情報の取得からリスク評価までをオートメーション化するリスク評価システムのプロトタイプ開発に取り組んでいる．今後も継続して上記開発に取り組んでいくつもりである．

謝辞 本研究の評価にご協力頂いた皆様に，謹んで感謝の意を表する．

参考文献

1) NIST(National Institute of Standards and Technology). http://www.nist.gov/
2) NIST, "National Vulnerability Database (NVD) CVE Statistics". http://web.nvd.nist.gov/view/vuln/statistics-results?cves=on&pub_date_start_month=0&pub_date_start_year=2000&pub_date_end_month=11&pub_date_end_year=2014
3) Mike Schiffman, Gerhard Eschelbeck, David Ahmad, Andrew Wright, Sasha Romanosky, "CVSS: A Common Vulnerability Scoring System", National Infrastructure Advisory Council (NIAC), 2004.
4) 杉本暁彦，磯部義明：動的モデリングに基づいたリスク評価システム，CSS2014, pp.100-107 (2014)
5) Ou, Xinming, Singhal, Anoop, "Quantitative Security Risk Assessment of Enterprise Networks", Springer, 2011.
6) Xinming Ou, Sudhakar Govindavajhala, and Andrew W. Appel, "MulVAL: A logic-based network security analyzer", 14th USENIX Security Symposium, Baltimore, Maryland, U.S.A., August 2005.
7) Oleg Sheyner, Jeannette Wing, "Tools for Generating and Analyzing Attack Graphs", Lecture Notes in Computer Science Volume 3188, 2004, pp 344-371.
8) Meseguer, J. Montanari, el al. “information and computation 88”, 105-155, 1990.
9) JVN(Japan Vulnerability Notes)., https://jvn.jp/
10) （株）日立製作所，IT Report Utility. http://www.hitachi.co.jp/Prod/comp
11) IETF, ” A Simple Network Management Protocol (SNMP)”, RFC1157, https://www.ietf.org/rfc/rfc1157.txt
12) NIST, “Official Common Platform Enumeration (CPE) Dictionary ", https://nvd.nist.gov/cpe.cfm
13) M. C. Parmelee, H. Booth, D. Waltermire, K. Scarfone, “Common Platform Enumeration: Name Matching Specification Version 2.3”, NIST Interagency Report 7696(2011)
14) I. Witten, E. Frank, M Hall, “Data Mining: Practical Machine Learning Tools and Techniques”, 3rd Edition, Morgan Kaufmann Publishers(2011)