社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2006－CSEC－34 (28)
2006／7／20

# 機器設定統合分析システムにおける対処案生成方式

岡城 純孝† 松田 勝志†

†NECインターネットシステム研究所 〒630-0101 奈良県生駒市高山町8916-47
E-mail: †{s-okajo@cd, mat@bp}.jp.nec.com

**あらまし** インターネットに対する様々な脅威からネットワークを保護するために，ネットワーク全体のセキュリティの一貫性を保ちつつ統一的にセキュリティ施策を実現する方法が求められている．しかし，様々な機能を持ったセキュリティ機器が混在しているため，それらの設定・管理は非常に複雑である．そこで筆者らは，相互に関係する複数のセキュリティ機器から設定情報を抽出し，機器に依存しないポリシー言語に変換して分析を行うことにより機器間の設定の矛盾検出を行う機器設定統合分析システムを開発している．

本稿では，検出された矛盾を解消するための対処案生成方式について述べる．また，ファイアウォールと侵入検知システムを対象とした試作システムについても述べる．本システムは機器間の設定の矛盾の検出とそれを解消する対処案生成を自動的に行うので，管理者は運用管理の手間を大幅に削減することができる．

**キーワード** セキュリティ，ポリシー，分析，ファイアウォール，侵入検知システム

# A Method of Correction Plan Generation in the Security Configuration Analyzing System

Sumitaka OKAJO† Katsushi MATSUDA†

† Internet Systems Research Laboratories. NEC Corp. 8916-47 Takayama-cho, Ikoma-shi, Nara, 630-0101 Japan
E-mail: †{s-okajo@cd, mat@bp}.jp.nec.com

**Abstract** In order to protect networks against network security threats, many security components with various security functions have been deployed. The configuration and management of those components are highly complex. Therefore, we have developed a security configuration analyzing system which can find security policy conflicts among the configurations of cooperated devices.

This paper especially describes a method of correction plan generation for configuration mismatch resolution. The paper also presents a prototype system which can resolve mismatches between firewall and IDS policies. This system automatically finds the mismatches and generates the plans. The system can reduce administrator's load and cost.

**Keyword** Security, Policy, Analysis, Firewall, IDS

## 1. はじめに

社外公開サーバの設置によって，ユーザは企業や組織へ24時間365日いつでもどこからでもアクセスしサービスを受けることができるようになった．しかし同時に攻撃者はいつでも攻撃を行うことが可能となり，企業や組織はネットワークに万全なセキュリティ対策を施す必要がある．

このような背景から近年，攻撃からネットワークやサーバを守るための様々な機能を持ったセキュリティ製品がベンダから提供されており，各組織は複数種類のセキュリティ機器を自身のネットワーク上で運用することでセキュリティ対策を行っている．しかし，セキュリティ機器の設定項目は膨大な量にのぼり，その設定が正しいかどうかを効果的に分析できるツールが欠けているために，セキュリティ管理者がネットワーク全域にわたってセキュリティを正確に管理することは非常に困難になってきている．

このため，管理者の負担を軽減し設定ミスを防止するセキュリティ運用管理技術が必要とされており，我々はこれを実現するセキュリティポリシー管理基盤の研究を行っている．特に，セキュリティ機器の中で最も利用頻度が高いと思われるファイアウォールとIDS（Intrusion Detection System；侵入検知システム）を対象とし，それぞれの機器の設定（ポリシー）を比較分析することによりセキュリティ状況の把握や設定の矛盾検出を行う機器設定統合分析システムの試作を行ってきた．

本稿では，分析により検出した設定間の矛盾について，他の部分の設定内容に影響を与えることなく矛盾を解消するための修正案を自動生成する機能について

報告する．

## 2. 機器設定比較分析システム

本節では，これまで筆者らが研究開発を行ってきた機器設定統合分析システムについて述べる．なお，以下の説明において，個々のファイアウォールルールや個々の IDS シグネチャ設定など，ある一連の意味を持つ設定情報を“ルール”，それらルールの集合による機器設定情報全体を“ポリシー”と呼ぶ．

### 2.1. セキュリティ運用管理の課題

近年，インターネットに対する脅威が高まっており，これに対して各組織は，ファイアウォール，ウイルス対策ソフト，IDS などのセキュリティ関連のハードウェアやソフトウェア（以下，セキュリティ機器）を積極的に導入している．その結果，次のような問題が顕在化している．

- **機器の個別管理による一貫性の欠如**

  セキュリティ機器の設定はそれぞれ個別の管理ツールなどにより管理されているため，同じ機能を持つセキュリティ機器であってもベンダやバージョンの違いにより別々に管理する必要があり，一貫性を保った運用管理が困難である．さらに，異なる機能を持つ別々のセキュリティ機器については，それらがうまく連携できているかを確認することはできない．

- **運用管理における管理者負担の増大**

  管理者は個別のツールを使い分けて作業しなければならないため負担が大きい．また，異なる機能を持つセキュリティ機器の設定の整合性については，管理者の知識や経験に基づいて管理されているため，設定ミスが発生する可能性も高い．

### 2.2. 機器設定統合分析

上記の問題を解決する目的で，ネットワーク全体の統合管理を行うツール[1][2]が登場しているが，1つのコンソールから複数機器の管理 GUI を呼び出したり，複数機器の設定情報を一元管理するにとどまっている．そこで筆者らは，設定情報から機種に依存しないポリシーを生成し分析することでセキュリティ機器を統合管理し，相互に関係する複数のセキュリティ機器の設定間の矛盾検出を行うことにより，セキュリティの一貫性を保ち管理者の負担を軽減する機器設定統合分析システムを提案してきた．機器設定統合分析システムの全体概要を図 1 に示す．提案システムでは，以下のようなステップでセキュリティ機器設定の分析を行う．

(1) 設定情報の抽出・ポリシー化
(2) ポリシー分析
(3) ポリシー矛盾検出

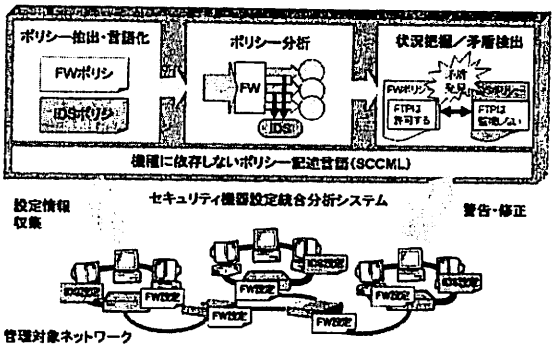


図 1 機器設定統合分析システムの全体概要

まず，SNMP や各機器の情報収集用の API，あるいは機器専用のエージェントプログラムなどを使って，管理対象ネットワーク上のセキュリティ機器から現在設定されている設定情報の収集を行う．次に，収集した設定情報を筆者らが設計したポリシー記述言語 SCCML[3]に変換する．SCCML は，ファイアウォールに代表されるアクセス制御系機器と，IDS に代表される監視系機器の設定を機種に依存せずにポリシーとして統一的に表現可能であり，セキュリティ機器の動作と機器によって制御されるオブジェクトからなるセキュリティ機器動作モデルに基づいた記述言語である．次に，生成したポリシーを分析し，設定間の矛盾を検出する．分析手法については，SCCML で表現されたポリシーに存在するオブジェクト属性の関連性に基づいてルールを対応付けることにより，異なるセキュリティ機器の設定間の統合分析を実現している[4]．これをポリシーコリレーション分析と呼ぶ．図 2 にポリシーコリレーション分析の概略を示す．

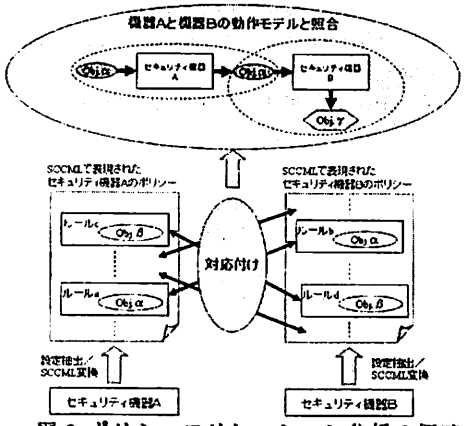


図 2 ポリシーコリレーション分析の概略

例えば，図 2 では異なる機能を持つセキュリティ機器 A と B のポリシーについて，Obj.α という同一オブジェクトの存在によってルール a とルール b を対応付けることができる．その後，SCCML における機器 A と機器 B の連携動作モデルと Obj.α を照合し，モデル上

で Obj.αの流れをシミュレートすることによりルール a とルール b の間の矛盾検出を行う．つまり，相互に関連しているにもかかわらず，機能・機種が異なるためにそれぞれ別個に管理されていた機器設定を統合して分析することができる．

## 2.3. 機器設定統合分析システム

本節では，これまで開発を進めてきたファイアウォール(FW)とネットワーク型 IDS(NIDS)を対象とした機器設定統合分析の試作システムについて述べる．

### 2.3.1. 設定抽出エージェント

設定抽出エージェントは，FW 設定（フィルタリングルール）と IDS 設定（監視ルール）を各機器から抽出する．これまでルータ製品の Cisco IOS ルータ，ファイアウォールソフトの FireWall-1, NetScreen, iptables および ipchains，IDS 製品の RealSecure NetworkSensor から設定情報を抽出するエージェントを実装した．FireWall-1 用エージェントは，OPSEC(Open Platform SECurity) [5]に準拠して実装を行った．その他の機器のエージェントについては，独自に実装を行った．

### 2.3.2. SCCML 変換モジュール

SCCML 変換モジュールは，抽出された設定情報を入力とし SCCML 形式のポリシーを出力する．これにより，ユーザはファイアウォールや IDS の設定を機器の違いを意識することなく統一的に扱えるようになる．SCCML 変換モジュールは，機種ごとに設定情報から SCCML へ変換する規則を保持している．

### 2.3.3. FW ポリシー分析モジュール

FW 設定は，一般的にパケットの属性である送信元 IP アドレス，送信元ポート，宛先 IP アドレス，宛先ポート，プロトコルの属性の組み合わせを条件として，通過許可あるいは通過禁止のアクション判別を記述したパケットフィルタリングルールのリストである．FW は，自身を通過するパケットの属性とリスト上の各ルールの属性条件とを順に照合し，そのパケットの通過の許可／禁止を判定する．また，各ルールにパケットの属性として記述される IP アドレスやポート番号には範囲指定が可能であり，それら範囲が一部重複することがあるため，ポリシー全体として結局どのようなパケットが通過を許可／禁止されるのかルールリストから把握することは困難である．

そこで FW ポリシー分析モジュールでは，各ルールに記述された属性条件をルールの順序関係を保持しながら｛送信元 IP アドレス，送信元ポート，宛先 IP アドレス，宛先ポート，プロトコル｝からなる 5 次元空間に射影し，5 次元空間を通過許可あるいは通過禁止の領域に分割することにより，ポリシー全体として通過するパケットと通過しないパケットを把握可能にしている[6]．

### 2.3.4. IDS ポリシー分析モジュール

IDS の設定は一般に，どのようなシグネチャを選択して監視を行うか，というルールの集合である．しかし，シグネチャは攻撃方法や OS の種類など，各 IDS に固有の方法で分類されており，製品に依存せずに監視状況を把握することが難しい．

そこで IDS ポリシー分析モジュールでは，各シグネチャが実際に監視するサービス（プロトコルとポート番号の組）に従ってシグネチャを分類する．これをプロトコルレイヤ分析と呼ぶ．これにより IDS ポリシーの監視状況が，機種に依存せずにサービスごとに把握できるようになる．

### 2.3.5. ポリシーコリレーション分析モジュール

プロトコルレイヤ分類によって，各 IDS ルールがどのサービスに関連するパケット，つまり，プロトコル属性とポート番号属性にどのような値を持つパケットに関連するルールであるかが判明しているので，プロトコル属性とポート番号属性に基づいて FW ルールと IDS ルールとを関連付けることができる．試作システムのポリシーコリレーション分析モジュールでは，FW ルールと IDS ルールの対応関係が，“『ファイアウォールで通過が許可されてネットワークを流れるパケットについてのみ IDS で監視を行う』という関係であれば矛盾のない設定である”，というセキュリティノウハウに基づき FW ポリシーと IDS ポリシーの矛盾検出を行っている．つまり，「ファイアウォールで通過が許可されてネットワークを流れるパケットを IDS で監視していない」という関係を“監視漏れ”として，また「ファイアウォールで通過が禁止されてネットワークを流れないパケットを IDS で監視している」という関係を“監視過剰”として検出する．図 3 に FW−IDS ポリシーコリレーション分析の概略を示す．

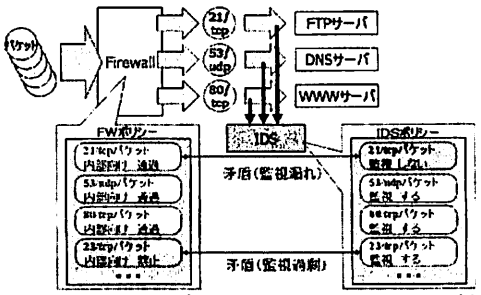


図 3　FW-IDS ポリシーコリレーション分析

## 3. 対処案生成機能

前節の分析によって矛盾が検出されたならば，それを速やかに解消する必要がある．本節では，検出された矛盾を解消する対処案の生成について述べる．

### 3.1. 対処案生成方式

ポリシーコリレーション分析により2つのポリシーの矛盾箇所はそれぞれ特定できているで，その矛盾を解消する修正ルールを生成する．試作システムでは，

- FWで通過を許可しているのにIDSで監視していない（監視漏れ）
- FWで通過を禁止しているのにIDSで監視している（監視過剰）

というルールの組み合わせを矛盾として検出する．そこで，矛盾として検出されたルールが上述の仮定を満たすようにポリシーを修正する．

しかし，ポリシーコリレーション分析では，入力された2種類のポリシーのみを使って矛盾検出を行っているため，どちらのルールが正しく，どちらのルールが誤っているのかまで判断することはできない．本システムでは，それぞれの場合についての修正ルールを生成する方式を採用した．管理者は，どちらのルールが正しいのかを判断した後，その対処案に従って修正ルールを適用することにより矛盾を解消することができる．

このとき，生成されたルールは，検出された矛盾のみを解消し他に影響を与えるものであってはならない．例えば，あるサービスについて矛盾を生じているFWルールを修正したために，別のサービスに関するパケットの通過が禁止されてしまったり，逆にこれまで通過を禁止されていたパケットの通過が許可されてしまったりしてはならない．

以下に，FWポリシーとIDSポリシーについてポリシーコリレーション分析の矛盾検出結果に対する対処案生成方式について述べる．

**「監視漏れ」矛盾に対する修正案生成**

まず，FWルールが正しくIDSルールが誤っている場合には，矛盾しているすべてのIDSルールについて「監視しない」から「監視する」ように修正すればよい．このとき，この修正によって新たな矛盾が生じることは無い．

次に，IDSルールが正しくFWルールが誤っている場合には，FWルールを「IDSで監視していないサービスに関するパケットについて，FWで通過を禁止する」状態にすればよい．しかしこのとき，矛盾しているFWルールについて単純に「通過を許可する」から「通過を禁止する」に修正してしまうと新たな矛盾が生じてしまう可能性がある．この問題とその解決策については3.2節で詳細に述べる．

**「監視過剰」矛盾に対する修正案生成**

FWルールが正しくIDSルールが誤っている場合には，矛盾しているすべてのIDSルールについて「監視する」から「監視しない」ように修正すればよい．このとき，この修正によって新たな矛盾が生じることは無い．

IDSルールが正しくFWルールが誤っている場合には，FWルールを「IDSで監視しているサービスに関するパケットについて，FWで通過を許可する」状態にすればよい．しかしこのとき，矛盾しているFWルールについて単純に「通過を禁止する」から「通過を許可する」に修正してしまうと，新たな矛盾が生じてしまう可能性がある．この問題とその解決策についても3.2節で述べる．

以上のように，基本的にIDSルールを修正する場合は矛盾を生じているIDSルールのアクション（監視する，あるいは監視しない）を変更すればよいが，FWルールを修正する場合は工夫が必要であることがわかった．

### 3.2. FWポリシー修正における問題とその解決

FWポリシーを修正するにあたっては，矛盾検出された各FWルールについて，

- IDSルールとのサービス（プロトコル，ポート番号）の整合
- ネットワークトポロジー（サービス稼動サーバのIPアドレス）との整合

の2つの問題を考慮する必要がある．

まず1つめの問題は，FWルールにおいてプロトコル属性の指定で“tcpおよびudp”のように複数のプロトコルが指定されたり，ポート番号の指定で“1番から10番”のように範囲指定されている場合に生じる．例えば宛先ポート番号について考えると，今，図4に示すように1番から10番までのポート番号について，FWポリシーで通過を許可するルールが1個設定されており，IDSポリシーで3番ポートのみ監視をせず，それ以外のポートについては監視するようにそれぞれ1個ずつ計10個のルールが設定されていたとする．このとき，ポリシーコリレーション分析により1番から10番までのポートについて設定している1個のFWルールと，3番ポートについて設定している1個のIDSルールが監視漏れ矛盾として検出される．ここで，検出されたFWルールのアクションを単純に“通過許可”から“通過禁止”に修正してしまうと，今度は3番ポートを除く9個のポートについて監視過剰矛盾となっ

てしまう．

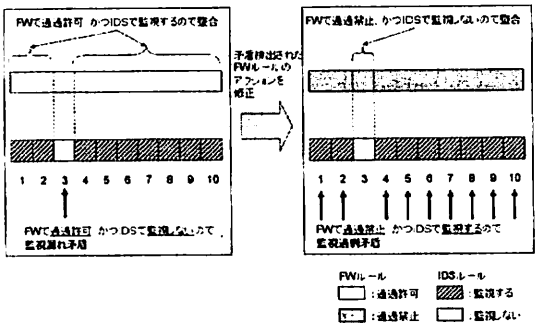


図 4 単純な修正によるポート番号の矛盾解消

そこで，矛盾として検出された FW ルール自体は修正せず，矛盾を生じている部分のポート番号のみを通過禁止とする FW ルールを新たに生成し，元の FW ルールの直前に挿入する（図 5 参照）．これによって新たな矛盾を生じさせることなく，検出された部分の矛盾のみを解消することができる．

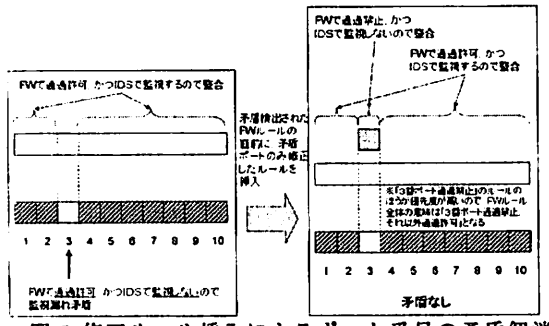


図 5 修正ルール挿入によるポート番号の矛盾解消

以上の処理を，矛盾している FW ルールのプロトコル属性，送信元ポート属性，宛先ポート属性について行うことにより，これら 3 つの属性の修正を行うことができる．

2 つめの問題は，FW ルールにおいて IP アドレス属性の指定でネットワークアドレス“192.168.1.0/24”のように範囲指定されている場合に生じる．今，図 6 に示すように DMZ 内のサーバ WWW（IP アドレス 10.59.24.5）上で稼動している HTTP サービスについて，IDS ルールで監視し，FW ルールでこの IP アドレスを含む範囲 (10.59.24.0/25，つまり 10.59.24.0 ～ 10.59.24.127 の範囲)を通過禁止にしている場合，これらのルールは監視過剰矛盾として検出される．このとき，この FW ルールのアクションを通過禁止から通過許可に修正してしまうと，サーバ WWW 以外の IP アドレスも修正されてしまう（図 7 参照）．これは結果的に，修正の必要のあるサーバ WWW の IP アドレス以外の範囲について，通過させる必要のないパケットを通過させてしまうことになる．これを防ぐために，管理対象ネットワークの構成情報をシステムに与える．

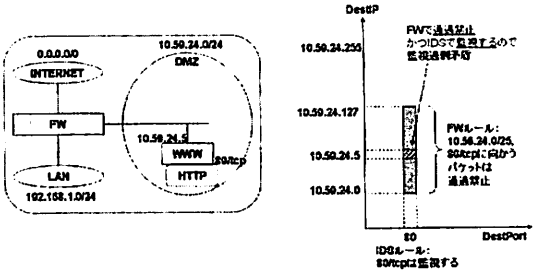


図 6 監視過剰矛盾の例

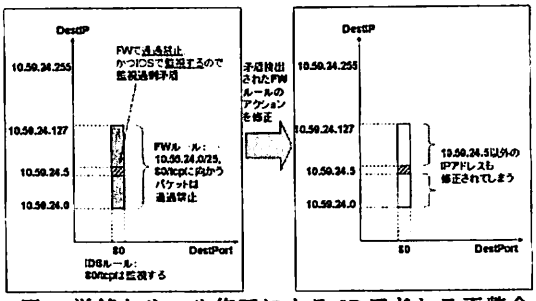


図 7 単純なルール修正による IP アドレス不整合

ネットワーク構成情報とはサーバごとの

- サーバ IP アドレス
- サーバ上の稼動サービス（複数種類可）

である．上の例ではサーバ WWW の，

- サーバＩＰアドレス：10.59.24.5
- 稼動サービス：TCP-HTTP(80/tcp)

がネットワーク構成情報となる．修正ルール作成時にこのネットワーク構成情報を参照し，矛盾を生じているサービスが稼動している IP アドレスのみを修正する（図 8 参照）．

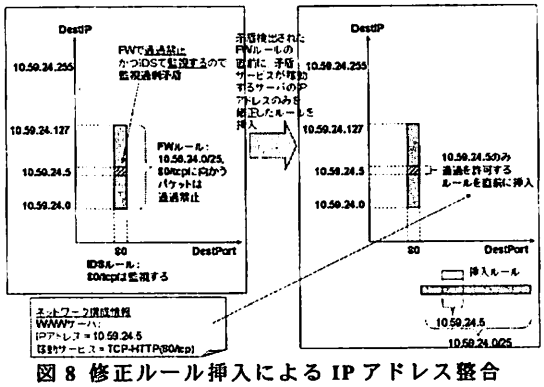


図 8 修正ルール挿入による IP アドレス整合

以上の FW ルール生成手順をまとめると以下のようになる．

1. 矛盾検出された FW ルールを取得（デフォルトルールを除く）
2. FW ルールのアクション属性について，監視漏

れ矛盾のときは“通過許可”から“通過禁止”に，監視過剰矛盾のときは“通過禁止”から“通過許可”に変更

3. FW ルールの属性のうちサービスに関係する属性（プロトコル属性，送信元ポート属性，宛先ポート属性）の値を，この FW ルールを対として検出された IDS ルールのそれぞれの値と重複する範囲の値に変更
4. ネットワーク構成情報を検索し，矛盾検出されたサービスが存在する場合には，FW ルールの宛先 IP アドレス属性の値を，稼動するサーバの IP アドレスの値に変更（このとき複数のサーバ上でサービスが稼動している場合には，稼動サーバ数と同数の修正ルールを生成）

上記手順により，生成した FW ルールを元のルールの直前に挿入することで，新たな矛盾を生じることなく検出された矛盾を解消することができる．

### 3.3. リスク値計算

対処案の生成と並行して，検出された矛盾がどれほど緊急な対処を必要とするかについてスコア付けを行う．このスコアをリスク値と呼ぶ．リスク値は，複数の矛盾が検出された場合に管理者がどの矛盾から優先的に対応すればよいかを判断するための指針となる値である．管理者はリスク値の大きい矛盾から対処を行うことで，より緊急の対処を要する矛盾の解消を順に行うことができる．

本システムでは，検出された矛盾について，

- 矛盾の種類による危険度
- 検出されたサービスの管理対象ネットワークにおける重要度
- 検出されたサービスの脆弱度

に基づいてリスク値を計算する．具体的には，以下の式によりリスク値を計算する．

$$R(c,s) = w(c) + f_w(s) \cdot f_v(s)$$
$$= w(c) + \{1 + \log_2(f(s,N)+1)\} \cdot \frac{n(s)}{S}$$

図 9 リスク値計算式

R(c,s)は，矛盾 c として検出されたサービス s のリスク値である．w(c)は矛盾 c の危険度であり，矛盾 c の種類と検出されたサービス s が管理対象ネットワークの中で稼動しているかどうかによって決まるリスクである．$f_w(s)$は管理対象ネットワークに固有のサービス s の重要度である．$f_v(s)$はサービス s の脆弱度である．

w(c)のリスクの大小を図 10 に示す．監視漏れのほうが監視過剰よりもリスクが大きい．これは，ファイアウォールが開いているにもかかわらず監視していない状態は，ファイアウォールが閉じているにもかかわらず無用に監視している状態よりも危険であるためである．さらに，監視漏れとして検出されたサービスの管理対象ネットワークでの稼動状況について，稼動していない場合は稼動している場合よりもリスクが大きい．これは，稼動していないサービスについてファイアウォールを開ける必要はなく，この部分がセキュリティホールとなる可能性があるためである．また，監視過剰として検出されたサービスについては，稼動している場合が稼動していない場合よりもリスクが大きい．これは，サービスが稼動しているにもかかわらずファイアウォールが閉じているため，提供するはずのサービスが提供できない可能性があるためである．w(c)は，リスクの大小に応じた値を重みとして与える．

| 不整合の種類 | サービスの有無 | リスク | 例 | リスク値の例 |
|---|---|---|---|---|
| 監視漏れ | 無し | 小さい | FWルール：HTTPは通過許可<br>IDSルール：HTTPは監視しない<br>サービス稼動状況：HTTPサービス無し | w(c)=0 |
| 監視漏れ | 有り | 大きい | FWルール：HTTPは通過許可<br>IDSルール：HTTPは監視しない<br>サービス稼動状況：HTTPサービス有り | w(c)=5 |
| 監視過剰 | 有り | やや小さい | FWルール：HTTPは通過禁止<br>IDSルール：HTTPは監視する<br>サービス稼動状況：HTTPサービス有り | w(c)=1 |
| 監視過剰 | 無し | 最も小さい | FWルール：HTTPは通過禁止<br>IDSルール：HTTPは監視する<br>サービス稼動状況：HTTPサービス無し | w(c)=0 |

図 10 不整合の種類によるリスク値

$f_w(s)$は，管理対象ネットワーク N における対象サービス s の稼動数 f(s,N)によって求められる．稼動数の多いサービスほど，そのネットワークにおける重要度が大きいことを表している．

$f_v(s)$は，入力 IDS ポリシーに含まれる全ルール（シグネチャ）数 S に占めるサービス s に関係するルール（シグネチャ）数 n(s)の割合である．関係するシグネチャの多いサービスほど，一般に攻撃対象として狙われやすく脆弱であることを表している．

$f_w(s)$と$f_v(s)$によるサービスの重要度計算により，図 10 に示した同じ種類の矛盾が複数検出された場合にも，管理対象ネットワークにおいて重要と考えられるサービス，一般に脆弱度が大きいと考えられるサービスほど，より大きな値としてリスク値が算出される．

## 4. 試作システム

3 節で述べた対処案生成機能を実装し，2 節で述べた機器設定統合分析システムに追加した試作システムについて以下に述べる．

図 11 に対処案生成モジュールの概略を示す．対処案生成モジュールは，設定抽出エージェントおよび SCCML 変換モジュールを介して得られた SCCML 形式の FW ポリシーと IDS ポリシー，ポリシーコリレーション分析モジュールによる分析結果である矛盾検出結果，ネットワーク構成情報を入力とし，対処案を出力する．ネットワーク構成情報は，セグメントトポロジ

一情報とサーバトポロジー情報からなる．セグメントトポロジー情報は，管理対象のネットワークセグメントの構成，および分析対象となるファイアウォールとIDSの配置場所に関する情報である．サーバトポロジー情報は，管理対象ネットワーク内のサーバの配置セグメント，各サーバ上で稼動するサービスに関する情報である．今回，ネットワーク構成情報は管理者がファイルとして与えている．図12にそれぞれセグメントトポロジー情報記述ファイルとサーバトポロジー情報記述ファイルの例を示す．

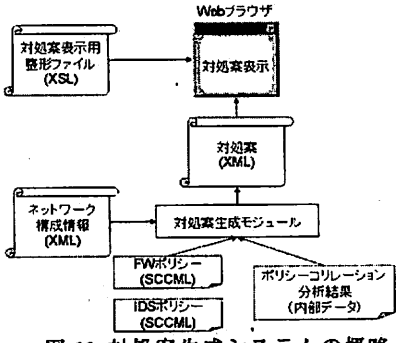


図 11 対処案生成システムの概略

セグメントトポロジー情報ファイル

```
<segment_topology>
 <segment name="INTERNET">
  <address>0.0.0.0/0</address>
 </segment>
 <segment name="DMZ">
  <address>10.56.123.0/24</address>
 </segment>
 <segment name="LAN">
  <address>192.168.1.0/24</address>
 </segment>

 <firewall name="fw0">
  <connect name="INTERNET">
   <nic name="eth0">207.100.5.233</nic>
  </connect>
  <connect name="DMZ">
   <nic name="eth1">10.56.123.253</nic>
  </connect>
  <connect name="LAN">
   <nic name="eth0">192.168.1.253</nic>
  </connect>
 </firewall>

 <nids name="nids0">
  <place name="DMZ"/>
 </nids>
</segment_topology>
```

サーバトポロジー情報ファイル

```
<server_topology>
 <server name="WWW">
  <in>DMZ</in>
  <ip>10.56.123.1</ip>
  <service>TCP-HTTP</service>
 </server>

 <server name="HTTPD">
  <in>DMZ</in>
  <ip>10.56.123.128</ip>
  <service>TCP-HTTP</service>
 </server>

 <server name="FTP">
  <in>DMZ</in>
  <ip>10.56.123.144</ip>
  <service>TCP-FTP</service>
 </server>
</server_topology>
```

図 12 ネットワーク構成情報ファイルの例

今，ポリシーコリレーション分析によって図 13 に示すような矛盾検出結果が得られたとする．1 つめの矛盾として，HTTP サービス(80/tcp)に関して 2 つの IDS ルールと 1 つの FW ルールの間で監視漏れ矛盾が検出されている．このときの状態を図 14 に示す．矛盾を生じている FW ルール (ContentsSecurity002) は，INTERNET セグメントから DMZ セグメントへの宛先ポート番号 1023 以下の tcp パケットの通過を許可するものである．

この矛盾に対してシステムが生成した対処案の表示画面を図 15 に示す．図 15 では，IDS ルールの修正方法として，2 つのルールを“監視する”に変更することが表示されている．また，FW ルールの修正方法として，矛盾として検出されたルールの直前に生成したルールを挿入することが表示されている．矛盾として検出されている HTTP サービスを稼動させているサーバは"WWW"と"HTTPD"の 2 つある（図 12 参照）ので，アクションを" 通過禁止"，宛先ポート番号を"80"，宛先 IP アドレスを"10.56.123.1"と"10.56.123.128"とする 2 つのルールが生成されている．これらのルールを元のルールの直前に挿入することにより，矛盾を生じている IP アドレスおよびポート番号部分のみを修正することができる．

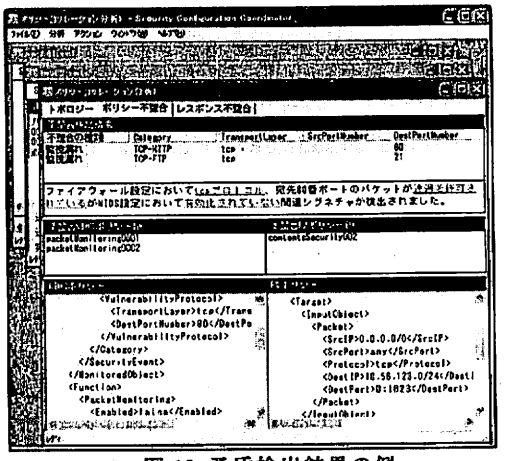

図 13 矛盾検出結果の例

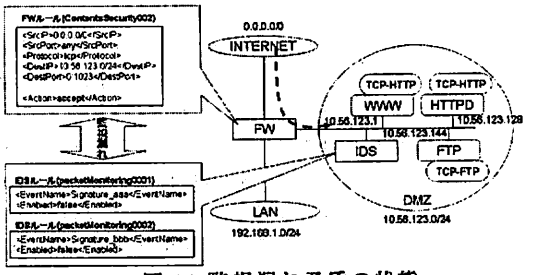


図 14 監視漏れ矛盾の状態

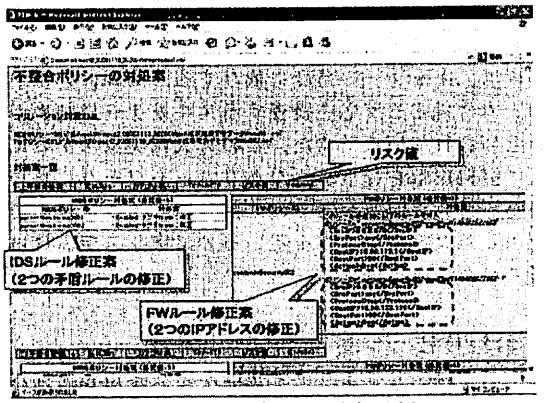


図 15 対処案の画面表示例

さらに図 15 では，図 13 で検出された 2 つの矛盾について，HTTP サービスに対して"6.292481"，FTP サービスに対して"5.106543"というリスク値がそれぞれ出力されている．図 14 に示すように，稼動サーバ数は HTTP サービスが 2 個，FTP サービスが 1 個であり，このときの IDS の全シグネチャ数 1727，そのうち HTTP サービスに関するシグネチャ数は 447，FTP サービスに関するシグネチャ数 92 であった．すなわち，提案するリスク値計算式は，同じ種類の矛盾であってもサービスの重要度に応じた値を算出することが確かめられた．

## 5. 従来研究

- **ダイナミックディフェンス（動的防御）**

ダイナミックディフェンスは，不正アクセスを検知した際の対応として不正アクセスに対する直接的な対応を行う手法の総称である[7]．具体的には，IDS により検知された不正アクセスについてファイアウォールの設定を変更してアクセスをブロックすることである．

ダイナミックディフェンスが対処療法的な技術であるのに対して，本システムの対処案生成機能は事前に FW ポリシーと IDS ポリシーを比較して設定のチューニングを行うものであり予防的な技術である．これらは互いに補完関係にある技術だと言える．

- 脆弱性スキャナー連携

IDS にネットワーク型の脆弱性スキャナーを統合し，IDS 設定のチューニングを自動的に行う製品が提供されている[8]．この自動チューニングは 3 つのステップで行われる．まず脆弱性スキャナーで監視対象の各機器の脆弱性を検査する．次に脆弱性検査の結果が IDS に送られる．最後に脆弱性検査の結果に基づいて，各機器の脆弱性と関連のある攻撃を検出するように IDS がチューニングされる．

脆弱性スキャナー連携は，予防的な技術であるという点で本システムに近い技術であるといえるが，検査精度はスキャナーの精度に依存する．実際，外部からの入力に対する反応を検査するスキャナーは精度が悪いといわれている．また，この方式は実システムに対して擬似攻撃を行う必要がありネットワークに負荷が掛かるという問題があるのに対して，本システムでは抽出した設定情報のみを用いるので実システムには影響を与えることはない．

## 6. おわりに

本稿では，セキュリティ機器から設定情報を抽出し，それらを分析することにより機器の設定間の矛盾検出と，それを解消する対処案を生成する機器設定統合分析システムについて述べた．特に，対処案生成機能として，修正ポリシー生成方式とリスク値計算方式について述べた．また，ファイアウォールと IDS を対象とする試作システムについて述べた．試作したシステムの対処案生成機能では，検出された矛盾箇所以外のポリシーには影響を与えないルール生成を実現している．管理者は，生成された対処案に基づきポリシーの修正を行うことで，新たな矛盾を生じさせることなく検出された矛盾のみを解消することができる．また，矛盾が複数検出された場合には，それぞれの矛盾についてどれほど緊急な対処を必要とするかを表すリスク値を計算する機能を設けた．管理者はこれを参考にすることでどの矛盾から優先的に対応すればよいかがわかる．

今後は，管理者が選択した対処案を自動的に機器に反映させる再設定機能の実装を行う．これは，選択された SCCML 形式の修正ポリシーを機器設定に逆変換した後，再設定を行えるように機能拡張した設定抽出エージェントを介して各機器にエンフォースすることで実現する予定である．また，対象機器の拡充や新しい分析方式についても検討を進めていく予定である．

## 文　献


[1] http://www.hitachi.co.jp/Prod/comp/soft1/jp1/

[2] http://www.symantec.com/Products/enterprise?c=prodinfo&refId=855&ln=ja_JP

[3] 岡城純孝，松田勝志，小川隆一，"セキュリティ運用管理のためのポリシー言語 SCCML，"情報処理学会研究報告，2004-CSEC-27，vol2004，NO.129，pp.89-94，Dec.2004.

[4] 岡城純孝，松田勝志，小川隆一，"セキュリティ運用管理における機器設定統合分析システム，"情報処理学会研究報告，2004-CSEC-28，vol2005，NO.33，pp303-308，Mar.2005.

[5] http://www.opsec.com

[6] 松田勝志，"マトリックス分解によるパケットフィルタリングルールの分析ー不要ルールと冗長条件ルールの検出ー，"情報処理学会研究報告，2005-CSEC-31，vol2005，NO.122，pp1-6，Dec.2005.

[7] JNSA ダイナミックディフェンス WG，"ダイナミックディフェンスの概要と適用について，" Dec.2000.

[8] 鳥居肖史，"プロアクティブセキュリティ，"Computerworld，2004 年 4 月号，pp.144-149，Apr.2004.