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

2008－CSEC－42（22）
2008 / 7 / 25

第3者マシンとの連携による不正侵入検知モデルの提案

藤井 雅和† 高橋 健一† 堀 良彰‡ 櫻井 幸一†‡

†（財）九州先端科学技術研究所 〒814-0001 福岡市早良区百道浜2-1-22

‡九州大学大学院システム情報科学研究院 〒819-0395 福岡市西区本岡744番地

E-mail: †{mafujii, takahashi, sakurai}@isit.or.jp, ‡hori@csce.kyushu-u.ac.jp

あらまし 近年，インターネットの広がりと共に不正侵入による被害が深刻になってきている．このような不正侵入への対策としてIDSやIPSなどの技術が研究開発されているが，基本的にこれらの技術では既知の攻撃にしか対応することができず，攻撃者に対して防御者が後手に回っているのが現状である．また，ゼロデイ攻撃やスピア攻撃の活発化，プログラムコードの肥大化などの要因も重なり，完全に不正侵入を防ぐことは難しいと考える．そこで，攻撃者が不正侵入したマシンを踏み台として利用することを難しくすることで，間接的に不正侵入を行う動機を軽減させて不正侵入を減らすための仕組みを提案する．本稿では，自サイトの通信ポリシを定義し，その通信ポリシに違反する通信を受信した第3者マシンが異常を通知することで不正侵入を検知するモデルを提案する． また，侵入者による通信ポリシ改ざんの対策手法についても述べる．

キーワード 不正侵入検知，ポリシ，踏み台攻撃，P2P

A Proposal of Intrusion Detection using Third-parties Support

Masakazu Fujii† Kenichi Takahashi† Yoshiaki Hori‡ Kouichi Sakurai†‡

†Institute of Systems, Information Technologies & Nanotechnologies

2-1-22 Momochihama, Sawara-ku, Fukuoka, 814-0001, Japan.

‡Department of Computer Science and Communication Engineering, Kyushu University, 744 Motooka Nishi-ku Fukuoka 819-0395, Japan.

Abstract Intrusions are one of the most important issues in the current Internet environment. A lot of researchers and companies elaborated countermeasure techniques such as Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). They usually rely on pattern matching. However, considering zero-day attacks and targeted attacks, we should assume that our machines may be corrupted anytime. Therefore we should consider what we can do under this assumption for a next generation security framework. In this paper, we focus on mitigating the spread of corrupted machines in the Internet world, and propose a new intrusion detection methodology using the support of third-parties' machines. In our proposal, when an attacker tries to attack other machine from a corrupted machine that the attacker already intrudes, the other machine notifies it to the corrupted machine's administrator. Since the attack can be noticed by the other machine, the attacker loses the motivation of attacking other machines from corrupted machines.

Keyword Intrusion Detection, Policy, Stepping Stone Attack, P2P

1. はじめに

近年，インターネットは普及し，多くの企業がインターネットを利用して活動をするようになった．しかし，インターネットは，その特徴である開放性がゆえに安全なネットワークではない．IPA/ISECの「コンピュータ不正アクセスの届出状況」によれば，サーバへの不正侵入事例は毎月報告されている［1］．自サーバに侵入された場合，その企業・団体は信用を失うと共に，侵入者が他サイトを攻撃することで加害者となり損害賠償を求められる可能性がある．そのため，システムへの不正侵入に対する十分なセキュリティ対策を施す必要がある．

一般的なセキュリティ対策として，迅速なセキュリティパッチ適用や不正侵入検知システム（IDS: Intrusion Detection System）［2］の導入がある．しか

し，攻撃者の技術は高度化しており，これらの対策だけで侵入を検知・防御することは難しい．例えば，パターンマッチングを基礎とする従来の不正侵入検知手法では，パッチ公開前にその脆弱性を突いた攻撃（ゼロデイ攻撃）やターゲットを絞った攻撃（スピア攻撃）を検知することができない．そこで，完全に侵入を防止することは無理であるという前提の下で新たな対策手法を考える必要がある．

悪意を持ったユーザは，自身の端末で直接攻撃をすると身元を特定されるため，一般に踏み台サーバを利用して間接的にターゲットを攻撃する．踏み台サーバ経由の攻撃は，追跡を困難にし，攻撃者の特定を難しくする．このため，踏み台サーバは攻撃者にとって必要なものとなっている．もし踏み台サーバが利用し難くなると，攻撃者は自身の端末から直接ターゲットを攻撃しなければならない．その場合，攻撃者は直接攻撃により身元が特定されやすくなる．このため，攻撃者がその活動を控えるようになると考える．

そこで，我々は攻撃者が不正侵入したマシンを踏み台として利用することを難しくすることで，間接的に不正侵入を行う動機を軽減するための仕組みを提案する．本提案では，侵入者が侵入に成功したマシン（マシン A）を使って他のマシン（第 3 者マシン）と通信しようとすると，その通信を第 3 者マシンが検査する．第 3 者マシンは，その通信が許可されたものでなければマシン A の管理者にメール等でその内容を通知する．侵入者は，未許可の通信を試みると検知されるため，許可された通信以外のアクセスを試みることができなくなる．この仕組みによって侵入者の行動を抑制し，不正侵入被害の拡散を防止することができる．

2. 関連研究

不正侵入を検知するシステムとして IDS(Intrusion Detection System) がある[2]．IDS は監視形態によって，ホスト型（HIDS: Host based IDS）とネットワーク型(NIDS: Network based IDS) に分類できる．また，侵入を検知する方法では，異常検知（Anomaly Detection）と不正検知（Misuse Detection)の 2 種類に分類できる．

異常検知方式は通常のアクセスパターン(プロファイル)を保持し，それを実際のアクセスと比較して大きく異なる場合に不正な通信として検知する．したがって，未知の手法も検知可能であるが，誤検知(False Positive)が多いという問題がある．一方，不正検知方式はシグネチャと呼ばれる不正アクセスの特徴情報をあらかじめ保持し，それを実際の通信データと比較して一致した場合に不正アクセスとして検知する．このため未知の不正な通信を検知できない（False Negative）という問題がある．そして，新しい手法の不正アクセスが発見される度にシグネチャを更新しなければならないという運用上の問題もある．

踏み台攻撃検出方式には，コンテンツベース検知方式 [3]とタイミングベース方式 [4]，コネクションベース検知方式 [5]がある．コンテンツベース検知方式は，受信および送信で流れるパケットの内容を手がかりに検知する．しかし，コンテンツベース方式はリモートログインパケットの内容を確認する必要があるため，パケットが暗号化されると適用できない．このことから近年ではタイミングベース方式による検出が主流となっている．タイミングベース方式は，リモートログイン時のユーザのキー入力ストロークに時間的な相関関係があることを利用して検出する．しかし，攻撃者がディレイや余計なパケットを挿入することで相関関係の検出を困難にすることができるとの指摘もある．コネクションベース方式は，踏み台攻撃時に発生するターゲットへの TCP コネクションの挙動を検知する．しかし，コネクションベース方式は，攻撃に時間差をつけられると検出できないという問題がある．

Security-Enhanced Linux（SELinux）[6]では，不正侵入後の被害を最小限に抑えるために，システムの根幹をなすオペレーティングシステム（OS）レベルでのセキュリティの向上がされている．SELinux は，任意アクセス制御（Discretionary Access Control, DAC）に加えて，強制アクセス制御（Mandatory Access Control, MAC）を採用することでセキュリティの向上を図っている．しかし，複雑なアクセス制御設定が必要であり，誤設定の可能性が高い．

インターネットファイアウォール[7]は，外部からの不正な通信を遮断し内部ネットワークを保護する．しかし，攻撃者はファイアウォールで許可された通信を使い侵入することが可能であり，また，ファイアウォール自身が脆弱性を持ち，侵入される危険性も考えられる．

SPF(Sender Policy Framework)[8]は，SMTP によるインターネットのメール配送を拡張する送信ドメイン認証技術のひとつであり，迷惑メール対策として実験的 RFC4408 で定められている．SPF では，電子メールの送信が認められているマシンを DNS の TXT レコードに定義する（送信者ポリシ）．電子メールを受信したメールサーバは，送信者ポリシを確認し，メールが本物か偽物か判断する．SPF は迷惑メール対策の技術だが，ポリシを使用する点で我々の提案モデルと類似している．

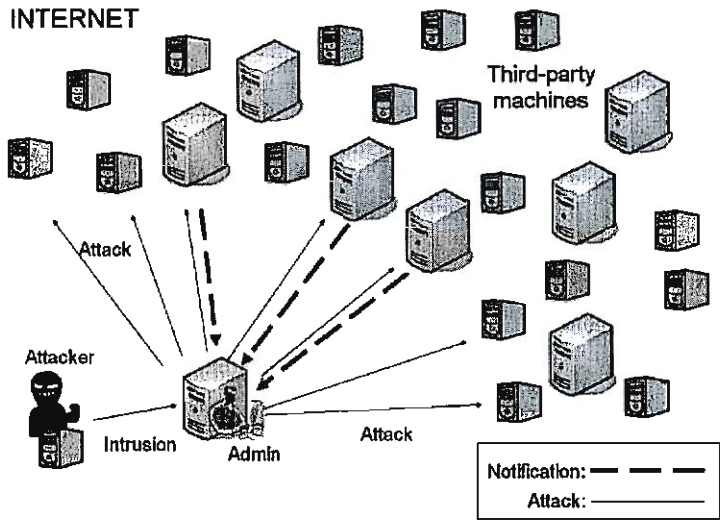

図 1　第 3 者からの通知による不正侵入検知

3. 不正侵入検知モデルの提案

IPA/ISEC の被害届けの報告には次のような被害事例がある.

【被害事例その 1】
事象　侵入

認知の発端
外部より,「貴方の管理しているサーバから攻撃を受けている」と連絡があった.
経過
アクセスログを調査したところ, 不特定多数の IP アドレスより SSH で使用するポートにパスワードクラッキング攻撃を受けていたことが判明. 推測が容易なパスワードが設定されていたアカウントで, 不正にログインされていた. さらに, そのアカウントが管理者権限に昇格されていた. 操作ログには, 外部サーバへの攻撃コマンドを実行していた形跡があった.

【被害事例その 2】
事象　侵入
認知の発端
組織外部から「貴方の組織内のパソコンから, SSH で使用するポートへパスワードクラッキング攻撃を受けている」との苦情メールが入った.
経過
調査したところ, 2 台の Linux マシンに侵入された形跡を発見. 内 1 台にパスワードクラッキングツールを埋め込まれ, 他サイト攻撃の踏み台として使われていたことが判明. どちらのマシンも, パスワードクラッキングを受けた形跡は無く, ID/パスワード共に一度の入力でログインに成功していた. Linux マシンに接続し端末として使用していた Windows マシンを調査したところ, ウイルス対策ソフトの機能が無効にされた上, キーロガーを含むルートキットを埋め込まれていた.

いずれの事例も不正侵入の認知の発端は外部からの通知である. システム管理者は, 外部からの通知を契機に不正侵入を検知（認知）している. 正規の ID/パスワードを利用してログインしコンピュータを操作された場合や高度な機能をもつルートキットなどでログを改竄された場合, 内部から不正侵入を検知することは難しい.

3.1. 提案モデル

通常, システム管理者は, 自システムへ不正侵入されることを望まない. しかしながら, どんなにセキュリティ対策を施していても, 不正侵入を完全に検知・防御することは難しい[9]. 自マシンは不正侵入をされ他のマシンを攻撃しているかもしれないし, SPAM を配信しているかもしれない. 優れた攻撃者は, その高い技術によって IDS による監視の無効化やシステムログを改竄するなど, セキュリティ対策を形骸化させ, マシンを使用し続ける. しかしながら, 攻撃者はすべてのマシンに侵入できるわけではない. ネットワーク上

には不正侵入されていない正常なマシンも存在している．もし，その正常なマシンが通信内容をチェックすれば，不正な通信であるかどうか気付くことができるはずである．不正な通信であることが判明した場合は，それに気付いたマシンが，不正な通信を発信しているマシンの管理者に警告をし，調査を促す（図 1）．普通のシステム管理者ならば，警告を受ければマシンの調査を開始し，マシンの復旧およびセキュリティ対策を施すだろう．我々の提案は，正常な第 3 者マシンが通信をチェックし，不正な通信が行われた場合に通知するフレームワークである．

我々の提案モデルでは，侵入者が侵入したマシン（SV1）から第 3 者マシンにアタックした場合，その通信をチェックする．もし，その通信が SV1 のシステム管理者の許可していない通信であれば，第 3 者マシンはそのことを SV1 のシステム管理者に通知する．通信がシステム管理者の許可したものであるかどうかは Policy（システム管理者がマシン毎に作成する通信ルール）を使うことで判断する．この Policy には発信ルールのみを定義する．第 3 者マシンは，ポリシで許可されていない通信を受信した場合，侵入者がいる可能性があるものとしてシステム管理者に通知する．もし許可されていない通信を行えば，その怪しい活動を検知されるため，侵入者は許可された通信以外のアクセスをすることができなくなる．

図 2 は提案するモデルの概要である．このモデルでの典型的な例を考える．例えば，システム管理者（*Admin*）は，企業のネットワークシステム構築において商用利用のための公開サーバを立ち上げる．DNS Server（SV1）もその 1 つである．DNS Server（SV1）は，他の DNS Server と DNS 通信（53/tcp, udp）することでドメイン名を IP アドレスに変換するため，DNS Server 同士が DNS 通信することは正常な動作であり問題はない．しかし，攻撃者が DNS Server（SV1）に侵入した場合，DNS Server（SV1）は DNS 通信以外のアクセスを第 3 者マシン（SV2）に行うかもしれない．攻撃者の目的は，リモートログインの試みやサービス拒否攻撃などを他サーバにすることだからである．提案モデルは，そのような通信を受信者側である第 3 者マシン（SV2）がチェックする．不正と判定した場合は，DNS Server（SV1）のシステム管理者（*Admin*）にその内容を通知する．受信した通信が不正かどうかを判定するために，第 3 者マシンはポリシ（Policy）を利用する．この Policy には，SV1 からの許可される外向きの通信ルールが定義されており，Policy に定義されない通信はすべて不正と判定する．不正な通信を受信した第 3 者マシンは，不正な通信の発信元のシステム管理者（*Admin*）にその内容を通知する．

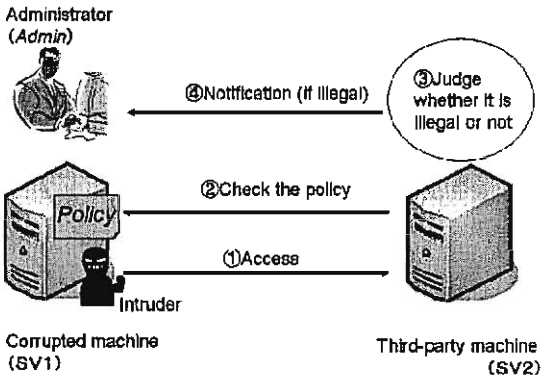

図 2　提案モデル概要

3.2. 通信ルール（Communication Rules）

我々の提案するモデルでは，通信を受信した第 3 者マシンがその通信内容をチェックする．第 3 者マシンは，不正な通信を検知した場合，その内容を発信元のシステム管理者に通知する．このため，ポリシには通信が不正かどうか判断できるだけの情報が必要である．ポリシは，複雑になると人的な設定ミスが発生するため，可能な限りシンプルであることが望ましい．そこで，ポリシはデフォルトですべての通信を拒否するものとし，システム管理者は許可する外向きの通信ルールを policy に設定する．そのパラメータとして，プロトコル名と宛先アドレスもしくは宛先ネットワークを使用する（図 3）．

```
<communication>
   <rule protocol = protocol name
                [ dest = IPaddress or network-address ] />
</communication>
```

図 3　通信ルール

通信ルールの設定は，XML のようなタグを使用する．<communication>と</communication> タグの間に設定し，各通信ルールの詳細は<rule>タグで設定する．protocol や dest は通信ルールのパラメータであり，protocol にはプロトコル名またはポート番号，dest には宛先アドレスを指定する．dest はオプションであり，明記されていない場合は，全アドレスへの通信を許可するものとする．図 4 の例の場合，2～4 行目が通信ルールで，2 行目は DNS 通信の発信を全アドレスに

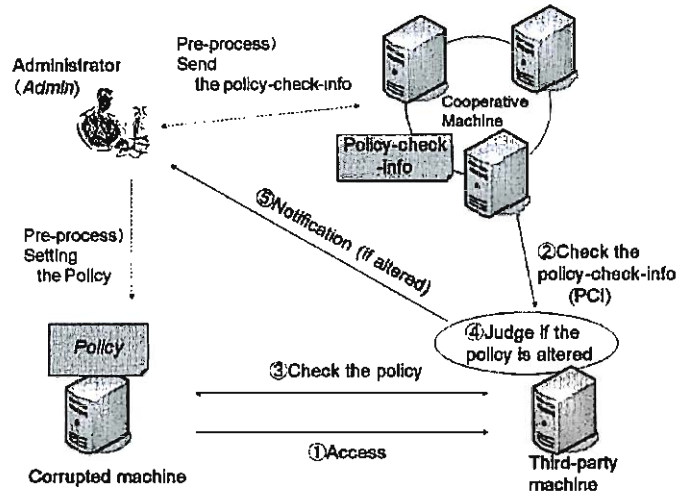

図 6　ポリシ改竄対策

対して許可するということを示し，3 行目は 192.168.123.1 との FTP 通信を許可することを示し，4 行目は 192.168.123.0/24 との SMTP 通信を許可することを示す．

```
1: <communication>
2:     <rule protocol = dns />
3:     <rule protocol = ftp dest = 192.168.123.1 />
4:     <rule protocol = smtp dest = 192.168.123.0/24 />
5: </communication>
```

図 4　通信ルール設定例

3.3. 通知ルール(Notification Rules)

提案モデルでは，システム管理者が許可していない通信が自マシンより発信された場合，その通信を受けた第 3 者マシンが不正侵入の可能性を発信元のシステム管理者に通知する．この時，通信を受けた第 3 者マシンは発信元のシステム管理者への連絡方法を知らなければならない．このため，通知方法をポリシに定義する必要がある．

```
<notification>
        <mail address = admin@lsit.or.jp />
</notification>
```

図 5　通知ルール設定例

図 5 は通知ルールの設定例である．第 3 者マシンは不正な通信を受信した場合，ポリシに指定された E メールアドレスに通知する．別の通知方法として電子掲示版（BBSs）などを利用することも可能と考える．その場合は，<mail>タグを変更することで対応する．E メールで通知を受ける場合，侵入者にその通知を削除されないように，信頼できるメールサーバから受信するようにすべきである．

4. ポリシ改竄対策

侵入した攻撃者がマシン内のポリシを改竄する可能性がある．このため，改竄を検知する仕組みが必要となる．改竄を検知するための方法の一つに，認証局(CA:Certification Authority)が発行する公開鍵証明書による電子署名を利用する方法がある．しかし，CA を利用する場合，公開鍵証明書を入手する手続きや証明書発行手数料などの時間と費用が掛かるといった問題や CA の安全性を十分に保証しなければならないといった問題がある．そこで，適当な第 3 者マシンが改竄を検知するための情報を持ち，第 3 者の協力によりポリシの改竄を検知する方法を導入する．

図 6 にポリシの改竄を検知する仕組みを示す．システム管理者(*Admin*)はポリシの改竄を検知するための情報を作成する．これをポリシチェック情報(PCI:Policy Check Information)と呼ぶ．システム管理者は，その PCI を第 3 者マシン(Cooperative Machine)に送信する．通信を受けた第 3 者マシン（Third-party machine）は，その通信の発信元の PCI を Cooperative Machine から取得し，通信元のポリシをその PCI を用いて評価する．ポリシが改竄されていることが判明した場合は，通信元の *Admin* にその事を通知する．

4.1. ポリシチェック情報(Policy Check Information)

ポリシチェック情報(PCI: Policy Check Information)は，ポリシの改竄をチェックために使用する．そのため，PCI として必要となる情報を次のように構成した（図 7）．

- ポリシのハッシュ値(MD:メッセージダイジェスト)(<check md>)
- ポリシが適用されるマシンの IP アドレス情報(<check address>)
- 通知先情報(<notification>)
- アップデート用情報(<hash-chain>)

```
<policy-check-info>
  <check md = Message Digest Value
        address = IP address />
  <notification><mail address = Mail-address /></notification>
  <hash-chain> Hash Chain Value </hash-chain>
</policy-check-info>
```

図 7 ポリシチェック情報（PCI）

これらの情報は，PCI ファイルの<policy-check-info>と</policy-check-info>タグの間に設定する．<check>タグには md と address のパラメータがあり，md にはポリシのハッシュ値を，address にはポリシが適用されるマシンの IP アドレスを設定する．ポリシの改竄検知は，発信したマシンのポリシのハッシュ値とそのマシンの PCI に記載された md 値を比較することで行う．<notification>タグには，3.3 節で説明した通知ルールを設定する．図 7 では，ポリシの改竄を検知した場合，指定した E-mail アドレスに連絡することを示している．<hash-chain>タグには，PCI のアップデートが正当なものかをチェックするハッシュチェイン値を設定する．ハッシュチェインについては 4.3 節で述べる．

4.2. 協力マシン(Cooperative Machine)

協力マシン(Cooperative Machine)は，ポリシの改竄検知に協力する第 3 者マシンである．Cooperative Machine の役割は，PCI を管理すること，他マシンから PCI を要求されたときに該当する PCI を提供することである．Cooperative Machine の選定には情報の分散管理手法である分散ハッシュテーブル(DHT: Distributed Hash Table，代表例 Chord[10,11]) を用いる．DHT を用いた P2P 環境では，ハッシュ値をキーとする情報を管理するマシンが選定される．そのマシンが Cooperative Machine となる．次式のハッシュ関数によって選ばれるハッシュ領域(hashspace)を担当するマシンが，*IP address* のマシンの PCI を管理する．

$$hashspace = hash(N,\ IP\ address)$$

N は管理させたい Cooperative Machine の数を表し，*IP address* はポリシを持つマシンの IP アドレスである．N が 1 のときは Cooperative Machine は 1 台となり，N を 3 とした場合は Cooperative Machine は 3 台となる．つまり，N の値を変えることで PCI を管理する Cooperative Machine の数を増やすことができる．仮に N を 1 とするならば，ポリシの改竄検知はその 1 台に頼ることになる．このため，N の値を大きくし，PCI を管理する Cooperative Machine を複数台にすることで，PCI 自身の改竄に対する強度を高くするのが望ましい．通信を受けた第 3 者マシンは，複数台の Cooperative Machine から PCI を入手し，PCI 自体の改竄をチェックすることができる．

4.3. ポリシチェック情報の更新

システム管理者が何らかの理由でポリシを修正した場合，PCI を更新する必要がある．その時，正当な管理者のみが Coorerative Machine 上の PCI を更新できるように，ハッシュチェインの値を利用する．図 7 で示した<hash-chain>タグには，PCI のアップデートが正当なものかをチェックするハッシュチェイン値を設定する．ハッシュチェインはシード S にハッシュ関数 H を 1～n 回適用することで得られたデータ $H^1(S), H^2(S), \ldots, H^n(S)$ の列である．$H^i(S)\ (i<n)$ を得たとしても，$H^j(S)\ (j<i)$ を求めることはできない．PCI のアップデート時にハッシュチェイン値を逆順に設定することで，侵入者が不正に PCI をアップデートすることを防ぐことができる．例えば，現存の PCI<hash-chain>値が $H^i(S)$ であれば，アップデート時には <hash-chain> 値を $H^{i-1}(S)$ とした PCI を Cooperative Machine へ送る．Cooperative Machine は，受け取った PCI のハッシュチェイン値 $H^{i-1}(S)$ から求められるハッシュ値と現在 Cooperative Machine が管理している PCI のハッシュチェイン値が一致するか確認した上で PCI のアップデートを受け入れる．このように，アップデート時に Cooperative Machine が PCI のハッシュチェイン値をチェックすることで，不正なアップデートを防ぐことができる．

5. 提案モデルにおける考察

5.1. 提案モデル導入の効果

提案モデルを導入した場合，システム管理者はこれまで気付き得なかった不正侵入の検知ができるようになる．第3者側からすれば，一見リソースを他人に貸しているだけに見えるが，第3者も他者からの通知で不正侵入を検知することができる．すなわち，提案モデルを導入することで，お互いの異常を監視し合う相互扶助の関係になる．本モデルを導入するマシンが増えれば増えるほど，検知は早くなり被害を最小限に抑えることができる．例えば，全マシンの4分の1が提案モデルを導入した場合，攻撃者はランダムに攻撃対象を選定すると，4回の不正通信の内1回が検知されることになる．このように，攻撃者からすれば，侵入を果たしてもインターネット上に不正侵入を通知するマシンが多数存在することになるため，迂闊な攻撃活動ができなくなる．我々は，提案システムが攻撃者に対しての抑止力としての働き，最終的にインターネット上から踏み台サーバが減ることにつながると考えている．

5.2. 誤検知について

False Positive としての誤検知には，自システムに侵入されていないにも関わらず通知が来る場合と攻撃者に偽通知メールを送りつけられる場合の2通りが考えられる．前者の場合は，通知は管理者にとって有用なメッセージとなる．侵入されていないにも関わらず通知を受け取った場合，意図しない通信が外部に発信されていることになる．この場合，システム管理者は，自分が気付いていない何らかの設定ミスやシステムの異常状態に気付くことができる．つまり，このような通知は不正侵入検知以外の異常を知るキッカケになり，システム管理者にとってメリットとなる．後者の場合は一種の DoS 攻撃である．この場合，自システムが DoS 攻撃の対象となっていることが分かり，インターネット接続に対する警戒心を高めることができる．しかし，多数の偽通知メールを長期間に渡り受け取ることは，管理者の通知に対する意識を低くする可能性があるため，ポリシと通知内容の比較や Filtering により偽通知メールを自動選別するための仕組みが必要となる．

攻撃者が侵入しているにもかかわらず，他サーバへの攻撃が発生していない場合，False Negative が発生する．本提案における検知方式は，他マシンへのアクセスを契機として侵入を検知するため，このようなケースでは，侵入者を検知することができない．しかし，他マシンに攻撃していないため，不正侵入被害の拡散を防止しているといえる．

5.3. 既存技術との関係について

本提案モデルは，従来の IDS とは目的が異なる．従来の IDS は，監視カメラのように機能し，攻撃者の侵入行為を検知することを目的にしている．IDS の守るべき対象は，IDS を導入しているシステムのみに限られている．一方，提案モデルは，不正侵入は完全には防げないという前提のもと，悪意を持った攻撃者が不正活動をし難いネットワークを作るのが目的である．従来の IDS は，自システム環境内に導入・設置して不正侵入を検知(内部検知)するのに対して，提案モデルではマシン間の協力によりお互いに不正侵入の可能性を通知することで不正侵入を検知(外部検知)する．内部検知の場合，攻撃者は自身の存在を秘匿するために Rootkit のインストールや監視ツールの機能を停止し，自システム内で攻撃者を検知できなくする可能性がある．しかし，外部検知の場合，攻撃者が侵入していないクリーンな第3者マシンは多数存在するため，攻撃者が第3者マシンすべてに侵入し，検知不能とすることはできない．両者の仕組みを併用することで，より信頼できる不正侵入の検知ができるものと考えている．

6. まとめ

本研究では，不正侵入を検知することを目的として，第 3 者の助けを借りた不正侵入検知モデルを提案した．提案モデルは，システム管理者の許可する通信をポリシに定義し，そのポリシに違反する通信を受信した第3者マシンに通知してもらうことで不正な活動を検知する．我々は提案モデルに対するポリシの改竄対策として，P2P 環境を利用した改竄検知方法について検討するとともに本モデル導入によるメリットについて考察を行った．提案モデルを従来の不正侵入検知システムと併用することでより信頼できる侵入検知ができる可能性があること，また，本モデルの普及が侵入の検知だけにとどまらず，攻撃者の行動を抑止できるネットワーク作りに役立つことを述べた．

今後は本提案モデルの有効性についての検証を行う予定である．

文　献

[1] 独立行政法人　情報処理推進機構（IPA）セキュリティセンタ，
http://www.ipa.go.jp/security/index.html

[2] Herve Debar, Marc Dacier, AndreasWespi. Towards a taxonomy of intrusion-detection systems. Computer Networks, vol.31, pp.805-822, 1999.

[3] Stuart Staniford-Chen, L. Todd Heberlein. Holding Intruders Accountable on the Internet. Proc. 1995 IEEE Symposium on Security and Privacy (SP' 05), pp.39-49, 1995.

[4] Yin Zhang, Vern Paxson. Detecting Stepping Stones. Proc.9th USENIX Security Symposium, Vol.9, pp.171-184, 2000.

[5] 竹尾大輔, 伊藤将志, 鈴木秀和, 岡崎直宣, 渡邊晃, " コネクションベース方式による踏み台攻撃検知手法の提案," 情報処理学会論文誌, Vol.48, No.2, pp.644-655, 2007.

[6] NSA, Security-Enhanced Linux, http://www.nsa.gov/selinux/.

[7] D. Brent Chapman, Elizabeth D. Zwicky. Building internet Firewalls, O' REILLY, 1995.

[8] Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, http://www.ietf.org/rfc/rfc4408.txt

[9] R. A. Kemmerer, G. Vigna, Intrusion detection: a brief history and overview, Computer Vol.35 Issue4, pp.27-30, April 2002.

[10] Chord, http://pdos.lcs.mit.edu/chord/

[11] Ion Stoica, Robert Morris, David Liben-Nowell, David R. Karger, M. Frans Kaashoek, Frank Dabek, HariBalakrishnan. Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications, IEEE/ACM Trans. Networking, Vol.11, No.1, pp.17-32, 2003.