ネットワークシステムにおける
セキュリティ監視技術に関する研究

平成 15 年度

竹森　敬祐

あらまし

近年，世界的な規模で広がりをみせるサイバーテロの脅威が高まっている．2001年7月にはCodeRedと呼ばれるコンピュータウィルスが，2003年1月にはSlammerコンピュータウィルスが，同年8月にはBlasterコンピュータウィルスが現れ，多数のホストがこれらに感染していまい，インターネットが一時的にサービス停止状態に陥るなどの影響が出た．このようなサイバーテロの脅威を最小限に抑え，安心・安全なネットワークシステムを提供するためには，各地のネットワークシステムの状態を的確に監視し，セキュリティログ等による未知の攻撃情報の収集を行い，これらの攻撃情報から，いちはやく脅威を分析して対策を促すセキュリティ監視センタ (SOC: Security Operation Center) の構築が重要な課題となっている．

本論文では，ネットワーク上で発生するコンピュータへの侵入攻撃やサービス不能攻撃等のサイバーテロを即座に発見し，その挙動を把握するSOC運用に必須な要素技術として，上記のシステムリモート監視，攻撃情報収集，セキュリティログ分析についてそれぞれ新たな手法を提案している．提案方式のシステム実装および評価実験を通じて，これまで検知できなかった攻撃やその被害を迅速に把握できることを明らかにし，提案方式の有効性を提示する．攻撃を的確に把握してその情報を防御へと活用することで，ネットワーク運用者と利用者にとっての安全なネットワーク環境を実現できる．以下に具体的内容を示す．

第 1 章では, 本研究の背景となるサイバーテロの検知手法に関する従来研究を概

観し，本研究の目的と位置付けを明確にしている．

第 2 章では, Web サーバが管理するホームページ用のファイルをリモートから監視することで，改竄攻撃やネットワークサービス停止攻撃による異常を検知できる Web サーバリモート監視システムを提案している．本提案システムでは，リンクページを辿ることにより監視対象となるファイルを自動的に抽出する機能と，ファイルのヘッダ情報やハッシュ値の変化に注目して改竄を判定する機能を有しているため，運用者の負担がほとんど無い状態において高い確率で Web サーバに対する攻撃を検知できる．大規模監視の実現性検証のために，インターネット上での監視処理速度に関する実験を行い，ネットワークへ与える負荷を軽減しつつ，局所的な輻輳に影響を受けない安定したリモート監視システムであることを定量的に明らかにしている．

第 3 章では, 未知の攻撃情報を収集するためのシステムとして，侵入検知システム（IDS: Intrusion Detection System）と連携して，不審な通信コネクションを本来のシステムからおとりシステムへと誘導して行動ログを収集するトラップ型おとりシステムの誘導手法を提案している．本誘導手法は，本来のシステムとおとりシステムの通信状態の同期を図っておくことで，誘導処理の高速性と通信シナリオの継続性を確保でき，侵入者におとりシステムの存在を気付かれない方式となっている．そして FTP サービスならびに Web サービスへの実装を行い，実験用ネットワークを用いて評価を行った結果，侵入者に気付かれないレベルの高速な誘導を実現できていることを確認している．このシステムにより，本来のサービスを提供しつつも侵入者の挙動・攻撃手法を容易に収集することが可能となる．

第 4 章では, 広域ネットワークの各地に設置された IDS から出力される攻撃検知ログを統合管理して，異常なイベントを客観的に抽出する IDS ログ分析支援システムを提案している．本システムでは，ログに含まれる各種イベントの異常性につい

て順位付けし，長期間のイベント傾向を比較対象として短期間のイベントの発生状況から，異常性を客観的な数値で評価する分析手法を提案している．各地で運用されている IDS の攻撃検知ログを用いて評価を行った結果，多量のイベント情報の中から，従来では発見が困難であった異常なイベントを特定できること，検証不要なイベントを排除できることを確認している．これにより，広域ネットワークを監視する運用者の作業負担を大幅に軽減するとともに，ネットワーク上で発生している攻撃の挙動を容易に把握することが可能になった．

第 5 章は結論であり，本論文の内容を総括している．

目 次

第1章

緒論

1.1 研究の背景

近年，世界的規模で広がりをみせるサイバーテロの脅威が高まっている．2001年前半には，多くの省庁のホームページが改竄されてしまう事件や，同年7月にはCodeRedと呼ばれるコンピュータウィルス（狭義にはインターネットワームもしくは単にワームと呼ばれる）がWebサービスを停止させてしまう事件が発生した．また，2003年1月にはSlammerワーム，同年8月にはBlasterワームに，多数のホストが感染してしまい，インターネットが一時的にサービス停止状態に陥るなどの影響が出た．このようなサイバーテロの脅威を最小限に抑え，安心・安全なネットワークシステムを提供するためには，各地のネットワークシステムの状態を的確に監視し，セキュリティログや未知の攻撃情報の収集を行い，これらの攻撃情報からいちはやく脅威を分析して警戒を呼びかけるセキュリティ監視センタ (SOC: Security Operation Center) の構築が重要な課題となっている．

サイバーテロ監視の歴史は，システムログ監査にはじまり，侵入検知システム (IDS: Intrusion Detection System) の研究開発，おとりシステムによる未知の攻撃分析，そしてこれらの技術を統合したSOCの構築へと繋がる．

図1.1にサイバーテロ監視技術の発展の様子を示す．セキュリティ監視の研究は

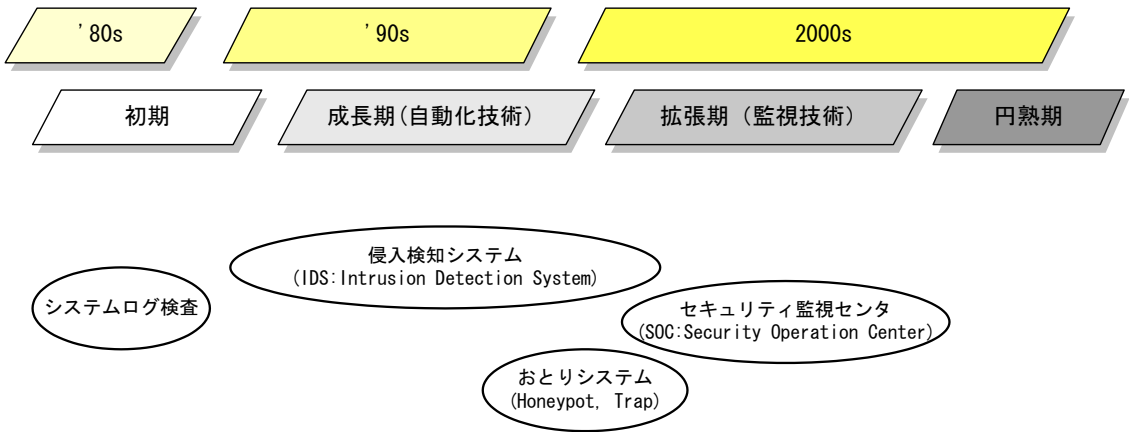

図 1.1: 侵入検知技術の発展

20 年近くの歴史を持っている．発端はシステムログのセキュリティに関わる事項を取り出した検査にあると言われている [1, 2].

1990 年代初頭に，監査の機能を自動化した様々な IDS が登場した．この中には，ネットワーク上を流れるパケットを監視して不正な型を持つパケットを検知する不正検知 (Misuse Detection) 方式の IDS や，ユーザの振る舞いを監視して統計的に普段と異なる行為を異常と位置付けて検知する異常検知 (Anomaly Detection) 方式の IDS などがある．

インターネットの普及が急速に進みはじめた 1990 年代後半には，日々多数のワームや攻撃ツールが出回るようになり，これまで経験したことのない未知の攻撃に対する脅威が高まった．そこで，侵入者を誘き寄せて侵入手法を収集するおとりシステムに関する研究が行われるようになった．ここで得られた知見は，防御対策の確立や IDS の検知アルゴリズムの高度化に寄与してきた．

しかしながら，攻撃による影響が瞬時にインターネット全体に波及するようになった現在，ネットワークシステムを個別に監視する手順では，攻撃の拡散速度や被害

状況を的確に把握することができない問題が明らかになってきた．そこで，広域ネットワークを統合監視する SOC の構築が開始された．SOC では，以下の項目が要求条件として挙げられている．

要件 1 各地の Web サービス等のネットワークシステムの状態をリモート監視すること

要件 2 侵入者の行動ログを収集して未知の攻撃に関する脅威を把握すること

要件 3 各地に設置した IDS ログを統合管理して攻撃の兆候や被害を把握すること

本論文は，このような背景と要件から，SOC から定期的に各地のネットワークシステムの状態を監視するリモート監視型 IDS の研究，積極的に侵入者の行動ログを収集して未知の侵入手法を把握するためのプラットホームとなるおとりシステムに関する研究，各地に設置された IDS のログを SOC において収集・管理して異常なイベントを検出する研究について進めたものである．本論文のサイバーテロ監視技術によって収集・分析された情報を，防御対策へと活用することで，ネットワーク運用者ならびに利用者にとって安心・安全なネットワーク社会の実現に向けた発展が期待される．

1.2　侵入検知システム

本節では，セキュリティ侵害の検知，通知を行う IDS に関する技術について概説する．

IDS は，その監視の対象として，ホスト上のログファイルやその他のファイルからデータを収集して検知するホストベース IDS と，ネットワークを流れるパケットからデータを収集して検知するネットワークベース IDS の二つのタイプに分類される．

またその検知アルゴリズムとして，既知の不正なパターン（一般的に，攻撃シグネチャと呼ばれる）をあらかじめファイルとして用意しておき，この攻撃シグネチャと収集したデータを照合して一致する項目があれば攻撃とみなす不正検知 (Misuse Detection) アルゴリズムと，正常な状態をあらかじめ特徴量 (一般に，プロファイルと呼ばれる) として用意しておき，このプロファイルと収集したデータを照合して一致しないかもしくは掛け離れた状態であれば攻撃とみなす異常検知 (Anomaly Detection) アルゴリズムの 2 つに分類することができる．

一般的な IDS の構成を図 1.2 に示す．イベント収集部は，システム環境の中から侵入検知に必要なイベント情報を入力として獲得する．イベント分析部は，侵入を検知するために不正検知もしくは異常検知を行うモジュールである．二つの検知アルゴリズムに関して様々な研究がなされており [3]-[16]，IDS の検知率や処理速度などを決定するコア技術を持ったモジュールである．イベント DB は，取得したイベント情報をプロファイルとして格納しておく DB である．知識 DB は，既知の攻撃情報を攻撃シグネチャとして格納しておく DB である．管理コンソールは，アラームとして出力されたログファイルを時系列で閲覧したり，ログに含まれるイベントの頻度グラフとして閲覧する機能を持つ．多量に出力されるログの中から効率的に異常なイベントを抽出する分析アルゴリズムに関する研究に注目が集まっている [12]．

1.2.1　ホストベース IDS の監視対象

ホストベース IDS で監視対象のファイルとして，オペレーティングシステム (OS) ログ，システムログ，アプリケーションログ，システムファイル，アプリケーションファイルなどがある．この他，OS から発行されるシステムコールを入力とするホストベース IDS もある．

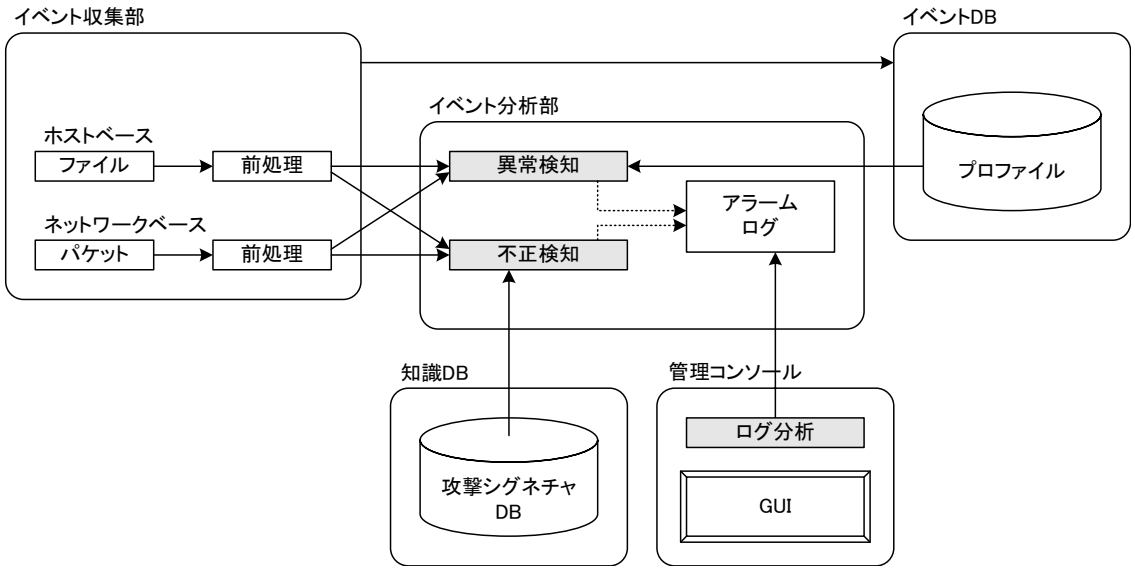

図 1.2: IDS の構成

OS ログは，カーネルレベルで処理されるため，後述するシステムログに比べてより信頼性があると言われているが，冗長な情報を含んでいるため解析が困難なことが多い．システムログは，システムの運用に関するログのうち，OS ログ以外のものを指す．UNIX での多くのシステムログは，syslog のライブラリを用いて生成される．たいていの場合，システムログはテキストファイルとして保存されるため，管理者にとっても操作しやすく，侵入検知の入力情報として扱いが容易である．アプリケーションログは，Web サーバや Mail サーバなど，アプリケーションが生成するログを生成する．アプリケーションの稼動状況や利用の記録に用いられるが，中に含まれるエラーログを調査することで，攻撃を検知できる．システムファイルは，システム起動や動作を既定したファイルであり，システム運用に重要なファイルである．これが改竄されると，システム運用に大きな支障となるため，ファイルの変更は厳しく管理されている．アプリケーションファイルは，前述のシステムファイル

以外のアプリケーションに関わるファイルであり，例えば，Web サーバや Mail サーバの動作を既定するファイルや，アプリケーションで取り交わされるファイルなどであり，ホームページやメールはアプリケーションファイルの一部である．

1.2.2　ネットワークベース IDS の監視対象

ネットワークベース IDS で監視対象の主なパケットとして，ICMP パケット，TCP パケット，UDP パケットがある．

ICMP パケットとは，OSI 参照モデルの第 3 層のネットワーク層に属するプロトコルパケットである．ネットワーク層の主な役割は，データ配信のためのアドレス管理や経路選択である．TCP パケットとは，OSI 参照モデルの第 4 層のトランスポート層に属するプロトコルパケットである．TCP では，パケットの流量コントロールや転送順序の制御など，信頼性の高い通信を実現している．UDP パケットとは，OSI 参照モデルの第 4 層のトランスポート層に属するプロトコルパケットである．UDP では，前述した TCP に比べて信頼性は劣るが，転送速度を重視したコネクションレス型のプロトコルである．

ここでネットワークベース IDS は，送信方向のみのパケットもしくは返信方向のみのパケットのどちらか片方向のパケットのみを監視して攻撃を検知する方式と，送信方向のパケットに対する返信方向のパケットの両方向を監視して攻撃を検知する方式がある．前者は，ステーションレス監視方式であり，簡易な設計で実装可能である．後者は，ステートフル監視方式であり，攻撃コマンドを送信したときの返信パケットの状態を比較分析することで，様々な攻撃を検知できることと，攻撃の成否をある程度判断できる．ただし，ステートを管理するためのメモリや CPU 処理負荷が増大するため，多量の攻撃を同時に実行されることで，検知不能に陥る可能性

が高い.

1.2.3　不正検知の研究

不正検知とは，ログファイルやパケットからの入力情報と，あらかじめ登録しておいた攻撃シグネチャとを比較して，不正を検知する手法である．よって，検知アルゴリズム自体は簡易であり，いかに多くの攻撃シグネチャを持たせることができるかがポイントになる [3]．日々新たな攻撃が発見され，かつネットワークの高速化が進む中，不正検知方式を用いたシステムの性能評価に関する研究が行われている．性能の指針として，検知可能な攻撃の種類数と，単位時間あたりに監視できるパケット数があげられる．前者は攻撃検知率と呼ばれ，後者はパケット処理率と呼ばれることもある．各種不正検知システムに関して，この二つの指標を適切な基準で評価する必要があり，その基盤技術に関する研究がなされている [4].

1.2.4　異常検知の研究

異常検知とは，観測したイベントを過去のイベントに対する偏差として評価する手法である．代表的な偏差の評価方法として，ニューラルネットワーク分析 (Neural Networks Analysis)，クラスタ分析 (Cluster Analysis)，オートマトン分析 (Automaton-Based Analysis) 等の研究がなされてきた.

ニューラルネットワーク分析は，学習，最適化，自己組織化などの機能を持った手法である．まず，ログファイルやパケットからの入力を適切な形式に変換する．次に，ニューラルネットワークに対して正常な状態をネットワーク構造の形として学習させる．ニューラルネットワークは，学習の前提を事前にデータ構造などの形で

与える必要がなく，自動的に状態の学習を進めることができる．ただし，異常を検知した場合に，異常と判断した理由がわからず，その有効性を証明できなかったり，学習に失敗することなどあり，活用のための研究が行われている [8]-[12].

クラスタ分析を用いた異常検知には二つの方法ある．一つは，通常時の振る舞いによって生じるデータをいくつかのクラスタに分類しておき，新たなデータを取得してそのデータがどのクラスタにも所属しないときに異常とする手法である．もう一つは，正常な状態と異常な状態のクラスタをそれぞれ定義しておき，異常クラスタに分類されるのを検知する方法である．クラスタ分析は，大量のネットワークシステムやパケットを監視する場合に有効である．一般に，同じ機能を持つネットワークシステムのサービスプロファイルは類似度が高いと考えられ，また，同じサービスから発生するパケットについてもそのプロファイルは類似度は高くなると考えられる．対象をグループ化してプロファイリングを行うことで，個々のオブジェクトの振る舞いを一般化することができる [13, 14]. クラスタ分析を用いた異常検知手法としては，k 最近隣法 (k-nearest-neighbor method), k 平均法 (k-means method), 自己組織化マップ, 統計クラスタリングなどのアルゴリズムが用いられている．

オートマトン分析は，普段のネットワークシステムの状態を全てモデル化しておき，分析を行いたい期間のシステム状態を入力として与えたときに，どのモデルにも所属しないときに異常とする手法である．ネットワークサービスプロトコル上で流れるパケットの種類からオートマトンを作成して異常を検知する研究や，システムコールに関するオートマトンモデルから異常を検知する研究が行われている [15, 16].

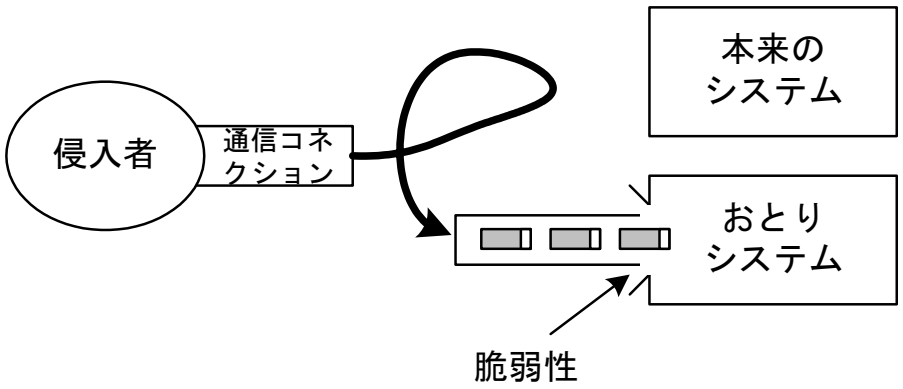

図 1.3: ハニーポット方式の構成

1.3　おとりシステム

本節では，様々な攻撃情報，特に未知の攻撃手法を収集する技術として，おとりシステムに関する技術について概説する．おとりシステムとは，侵入者に偽りの情報を与えて本来のシステムを保護したり，侵入者の行動を分析するための情報を収集する基盤システムのことである．おとりシステムには，待ち受け型のハニーポット (Honey Pot) 方式と，強制誘導型のトラップ (Trap) 方式の二つに分類される．

1.3.1　ハニーポット方式の研究

ハニーポット方式では，本来のネットワークシステムとは別に，侵入者の興味を惹くための脆弱性を持つおとりのネットワークシステムを設置して，これに誘き寄せられた侵入者を泳がせておき，その行動ログを収集する．図 1.3 にハニーポット方式の概念モデルを示す.

ハニーポット方式は，本来のシステムとは独立したおとり専用のシステムを設置するだけであり，実装は容易である．本技術に関する研究としては，その運用手順

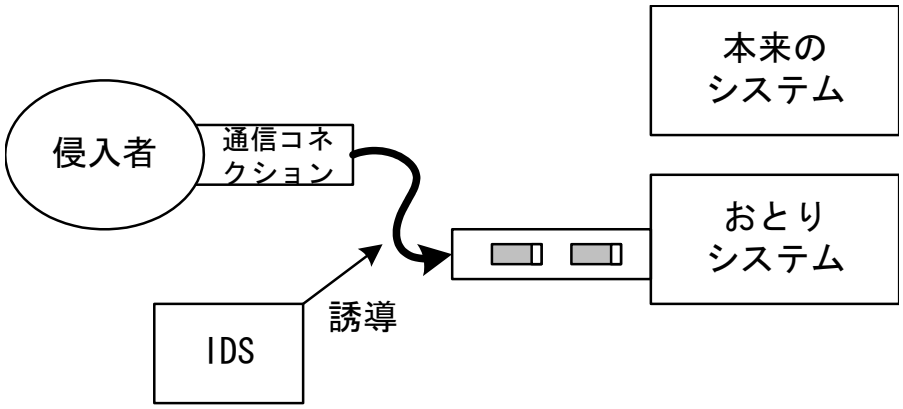

図 1.4: トラップ方式の構成

に関するものが多く，主に米国の企業や大学を中心に”The Honeynet Project”として攻撃情報を収集，分析が行われている [17]-[19]．日本でも慶應大学にて，効率的に攻撃情報を収集するための運用技術について研究がなされており，FTP サービスや HTTP サービスに関する未知の攻撃手法の発見に繋がっている [20].

1.3.2　トラップ方式の研究

トラップ方式では，本来のシステムと同じセキュリティ対策を施したおとりシステムを本来のシステムに併設しておき，本来のシステムにアクセスしてくる全ての通信を IDS で監視して，不審な行動が検知されるとその通信コネクションをおとりシステムへと誘導して，その行動ログを収集する．図 1.4 にトラップ方式の概念モデルを示す.

トラップ方式は，侵入者を確実におとりシステムへと誘導できること，本来のシステムが潜在的にもつ脆弱性情報を高い確率で収集できることなど，ハニーポット方式に比べて安全面と情報収集面における優位性から注目されている [1, 21].

1.4　セキュリティ監視センタ

近年，サイバーテロ対策の一環として，セキュリティ監視センタ (SOC) の設置が進められている．本節では，SOC に集められるセキュリティ侵害に関する事件の対応 (一般に，インシデントレスポンスと呼ばれる) と，そのインシデント対応で重要な役割を果たすセキュリティログ分析技術について概説する．

1.4.1　インシデントレスポンスチームの活動

米国では，コンピュータシステムのセキュリティログの監査を専門に行うセンタとして，1988 年に CERT/CC (Computer Emergency Response Team/Coordination Center) が，いわゆるインシデントレスポンスチーム（IRT：Incident Response Team) と呼ばれる組織として活動を開始した [22]．CERT/CC では，不正アクセスを受けた組織からの届け出があった場合，技術的なアドバイスを提供するほか，通報された情報を基に分析を行い，再発防止のための情報を公開している．開局当時の CERT/CC によるセキュリティログの監査結果は，個々のコンピュータシステムが独立して利用される場面が多く，各システムに特化したものであった．

近年，インターネット利用環境の整備が急速に進められる中，同時多発的に大規模・広範囲に行われる不正アクセスについては，一つのサイトから得られる情報だけでは全体像を解明し難い問題が明らかになってきた．同じ情報ではあっても，関連する複数のサイトからの情報を統合比較することで，影響範囲を特定し，必要な対策を検討するために役立つ情報を得られる．近年，CERT/CC ではネットワークシステム間にまたがるセキュリティログも監査の対象とするべく，不正アクセスによる被害情報を広く収集・分析し，対応を促すセンタの役割へと移行してきた．日本でも 1996 年

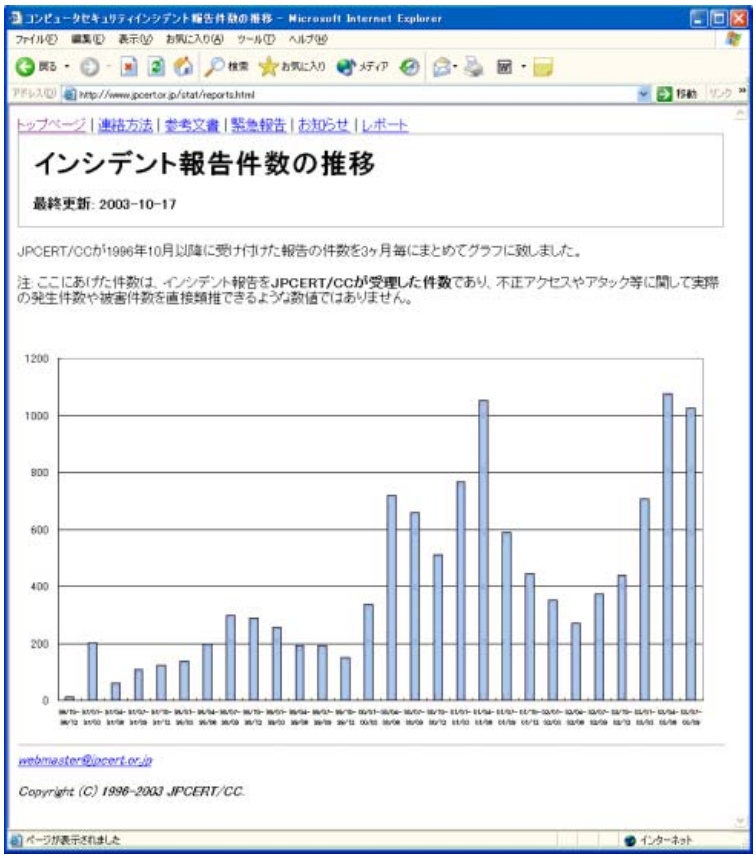

図 1.5: JPCERT/CC に寄せられたインシデントの件数

10 月に，JPCERT/CC (Japan Computer Emergency Response Team/Coordination Center：日本コンピュータ緊急対応センタ) が活動を開始した [23]-[25]．これらのセンタが，現在構築が検討されている SOC の原型となっている．

図 1.5 に，JPCERT/CC で公開されているインシデント件数の推移を示す．棒グラフは，四半期ごとのインシデント件数を表しており，2000 年 1 月頃から問い合わせ件数が増加しており，2002 年 4 月に減少したものの，最近では再び増加傾向にある．

既存の IRT では，インターネットユーザからの申請により情報収集しており，侵害後に残された情報だけでは，詳細を把握しきれない問題がある．また，短時間で

急激に広がるインターネットワームなどの攻撃に対して，分析の遅延が問題になっている．そこで，各地のネットワークに IDS やおとりシステムを仕掛けて，積極的にセキュリティログを収集して分析を行う SOC の構築がはじまった．日本でも京都大学などでは，学内のネットワークの各所に IDS を仕掛けて，イベントの傾向把握やインターネットワームの感染活動を監視する試みも行われている [26].

1.4.2　セキュリティログ分析の研究

各地のネットワークに仕掛けた IDS からのログを統合管理して，インシデント対応に役立てるセキュリティログ分析に関する研究が注目されている．ここで，IDS が捕捉しようとする特定の行為を EoI(Event of Interest) と呼ぶことがある [2]．この EoI に関する情報を的確に把握することが重要であるが，IDS の能力に依存することが多い．IDS の能力を測る指標として，フォルスポジティブ (False Positive) とフォルスネガティブ (False Negative) がある．フォルスポジティブとは，EoI でない行為を攻撃として検知してしまう誤検知のことを指し，フォルスネガティブとは EoI が発生しているにも関わらず，それを攻撃として検知できないことを指す．未知の攻撃に対する脅威が高まっている現在，殆どの IDS がフォルスポジティブに設計されている．この場合，疑わしいと思われる全ての行為に対してアラームを発するようになり，フォルスポジティブの割合が著しく増加してしまう．フォルスポジティブが度重なると管理者はアラームに対して鈍感になってしまい，本当の侵入に対する警報を見逃してしまう危険性がある．そこで，いかに多量に出力されるアラームの中から，異常な EoI を抽出するかが研究の中心となる．

一つの試みとして，ログの出力傾向を把握するために，ログに含まれるアラームの説明文を棒グラフで表示することで，他と異なるログを視覚的に検出し易くする

システムが提案されている [27]．他の試みとして，ニューラルネットワークを用いて誤検知ログを学習して，次々に出力される冗長なログを削除する研究も行われている [28]．

1.4.3　監視情報を活用する攻撃対策技術の研究

収集した情報を，今後の脅威への防御対策に活用する必要がある．例えば，攻撃を検出したサイト管理者への連絡や，改善が図られない場合のサイト情報の公開，攻撃状況を基にした防御策の提案，ネットワークからの自動排除など，監視情報の活用手段は様々な場面が考えられる．攻撃状況を基にした防御策の提案技術として，過去に発生した一連の攻撃シナリオから未来に発生する攻撃を予測して，ネットワーク管理者に対策を促すシナリオ追跡型の監視・防御技術が注目されている [29] [30]．これは，IDS から出力されるログからイベント遷移を表すオートマトンモデルを導出しておき，注目する攻撃がどの段階まで進行しているのかを把握して，未来に発生しうる攻撃シナリオをネットワーク運用者に提示する技術である．SOC による監視情報を防御へと活用することは重要であり，ネットワーク利用者にとって安全なネットワーク社会の実現に向けた対策へと繋がっていく．

1.5　既存技術の課題

1.2 節から 1.4 節では，セキュリティ監視センタ (SOC) におけるネットワークシステムに対するサイバーテロ監視技術について述べてきた．しかしながら，攻撃手法とその被害の多様化が進む中，広域かつ詳細なネットワーク監視を実現するためには，Web サービスに代表される各地のネットワークシステムの異常を外部からの視

表 1.1: サイバーテロ監視に関する既存技術の課題

リモート監視技術（要件1）	背景	・Webサービスに代表されるネットワークシステムへの攻撃が多発してきた. 　－ 改竄攻撃／DNS情報詐称によるなりすまし攻撃／ネットワーク資源を枯渇させるDDoS攻撃 ・既存のIDSでは検知しきれない攻撃があることが判明した.
	必要技術	・外部からの視点でWebサービスへの攻撃を検知するリモート監視システムが必要である.
	課題	・Webサービスに対する各種攻撃を検知できなければならない. ・ネットワーク輻輳による影響を受けない安定した監視を行わなければならない.
情報収集技術（要件2）	背景	・未知の攻撃に対する脅威が高まる中，侵入者の行動ログを収集するおとりシステムが注目されるようになってきた.
	必要技術	・サイト独自の脆弱性情報を収集できるトラップ方式のおとりシステムが必要である.
	課題	・誘導すべき不審な通信コネクションが検知されると，直ちにその通信コネクションをおとりシステムへと誘導して正規システムを守らなければならない. ・誘導時に，本来のシステムとおとりシステム間の通信シナリオを一致させなければならない.
セキュリティログ分析技術（要件3）	背景	・サイバーテロによる影響がネットワークの広域に及ぶようになる中，新たな攻撃の兆候や攻撃被害を迅速に把握するためのIDSログ分析に関する研究が注目されるようになってきた. ・IDSから出力されるログには，誤検知／多重検知／対策済み検知／繰り返し検知などの冗長な情報が含まれており，監視作業が煩雑である.
	必要技術	・各地のネットワークシステムに設置したIDSログを効率的に統合分析する技術が必要である.
	課題	・IDSごとにフォーマットや出力特性が異なるログを，統合管理しなければならない. ・多量に出力されるログの中から，異常なイベントを的確に抽出しなければならない.

点で監視する技術がないこと、未知の攻撃情報を安全かつ確実に収集する技術がないこと、セキュリティシステムから出力されるログを効率的に分析する技術がないこと等の課題がある。これらの課題を表 1.1 にまとめる．

リモートシステム監視技術の課題

近年，Web サービスを提供するネットワークシステムに対する攻撃が社会問題となっている．この攻撃を監視するために，IDS に関する研究が注目を集めている．Web サービスに対する攻撃として，ホームページファイルの改竄攻撃や DDoS(Distributed Denial of Service) 攻撃があるが，前者の攻撃を検知するためには，ホストベース異常検知方式の IDS が適切であり，後者の攻撃を検知するためには，ネットワークベース異常検知方式の IDS が適切である．ホストベース異常検知方式の IDS の場合，ホ

ストに常駐して監視対象ファイルの変更を定期的に検査することになる．具体的には，ホームページ用のファイル作成者がその正当性を証明するための署名データを一方向性ハッシュ関数で作成し，ホームページファイルならびに署名データの両方を Web サーバ上で管理することで，作成者が意図しないファイルの変更を確実に検知することができる [31]．ネットワークベース異常検知方式の IDS の場合，Web サーバに繋がるネットワーク上のトラヒックを監視しておき，多量な接続要求や，Web サービスには関連のないトラヒックを監視することで攻撃を検知できる [1].

しかしながら，外部の DNS(Domain Name Server) スプーフ攻撃で偽の Web サーバへと閲覧者が誘導されてしまうことを検知できない問題，DDoS 攻撃によるネットワーク資源の枯渇の状況を外部覧者の立場から評価できない問題などがある．そこで、外部の閲覧者の視点で各種攻撃をリモート監視する方式が考えられるが、ネットワーク輻輳などの影響を受けてしまうことが問題になる．

情報収集技術の課題

ここまでは、攻撃を受けた後の対応を迅速に図るためのリモート監視システムに関して述べてきたが、侵入を未然に防ぐための攻撃手法収集技術としてのおとりシステムもサイバーテロ監視に重要な役割を果たす。侵入者の行動ログを収集するおとりシステムとして，待ち受け型のハニーポット方式と強制誘導型のトラップ方式がある．前者の方式の場合，一般的な脆弱性を持たせたシステムに誘き寄せられた侵入者の行動ログを収集することを目的としており，収集される情報も一般的な攻撃情報になりがちである．また，本来のシステムを守るという目的では，ランダムに攻撃対象を選択する侵入者に対しては，無効な方式である．後者の方式の場合，本来のシステムと同じセキュリティレベルの設定をおとりシステムに施しておくこ

とで，本来のシステムに潜在する脆弱性に関する攻撃ログを収集できる．また，危険な通信コネクションを強制的に本来のシステムから隔離することで安全を確保できる．こうした優位性から，おとりシステムとしてトラップ方式が注目を集めている [1, 21].

しかしながら既存のトラップ方式における強制誘導の制御は，通信コネクションの開始時点で誘導しており，不正が検知された次の通信コネクションから対応している．もし，不正の検知された TCP コネクションが継続されれば，本来のシステムに接続されたままになり，攻撃を受けてしまう問題がある．また、誘導時に，本来のシステムとおとりシステム間の通信シナリオに不整合が生じるため、侵入者に気付かれてしまう問題がある。ここで不整合とは、TCP コネクションの再接続やアプリケーションレベルので再ログイン処理などである。

セキュリティログ分析技術の課題

サイバーテロによる影響が瞬時にインターネットの広域に波及するようになってきた現在、侵入後の対応の迅速化や未知の攻撃手法の把握に加えて、セキュリティ事象への対応（一般にインシデントレスポンスと呼ばれる）や，各地の IDS から出力されるログの分析が重要になってきた．インシデントレスポンスに関しては，既存の OS ベンダや CETRT/CC や JPCERT/CC などにより，セキュリティ侵害に関する情報収集と対策方法の広報がなされている [22, 23]．セキュリティログの分析に関しては，各地からのログを統合分析することで，新たな攻撃予兆の発見やネットワーク間の特徴比較による異常検知が可能であり，盛んに研究が行われている [26, 27].

しかしながら，複数の IDS を纏めてセキュリティログを分析するには，設置されている IDS の種別ごとにログフォーマットや運用手順が異なるため，簡単に統合運

用できない問題がある．これにより，広域や個々のネットワークを同じ基準で監視することができない．また，管理者に必要な情報は全て提供するという思想で，疑わしいイベントは全て報告する False Positive に設計されがちな監視ポリシによって多量のログが出力されてしまうことで、微かな痕跡を見逃してしまう問題がある．この冗長なログを削減する試みとして，運用パラメータの最適化手法があるが，予備作業に掛かるコストや最適化パラメータを他の IDS に転用できない等の問題がある．視覚的に異常を強調することで冗長なログを目立たなくする試みもあるが，運用者の経験や主観に依存する痕跡検出作業への信頼性に疑問が残る．さらに，出力されるログの特徴が，監視対象のネットワーク構成の変化や新たな攻撃の出現によって日々変動していることで，IDS に付随する既存の頻度分析機能だけでは，異常なイベントを的確に検出できない問題がある．

1.6　課題解決のための提案および既存技術に対する位置付け

1.6.1　課題解決のための提案

1.5 節で述べてきた課題を解決するための本研究における提案技術とその効果について，表 1.2 に示す．

本研究の一つとして 2 章では，Web サービスに対するリモート監視技術の検討であり，Web サーバごとに IDS を設置することなく，ファイルの整合性検査と DDoS 攻撃を検知するリモート監視システムを提案している．

また，本研究の一つとして 3 章では，おとりシステムの中でも有効な攻撃ログを収集できるトラップ方式に関する検討であり，本来のシステムとおとりシステムを

表 1.2: 提案技術とその効果

リモート監視技術（要件1）	提案方式	・各地のネットワークシステムを統合管理するSOCから，ホームページファイルを定期的にダウンロードして検査する方式を提案する． ・定期的な監視処理の独立起動と，待ち行列を用いた監視処理の並列化を提案する．
情報収集技術（要件2）	提案方式	・侵入者からの通信コネクションを誘導するアクセス制御装置を導入する． ・アクセス制御装置は，クライアントからの通信コネクションを本来のシステムとおとりシステムの両方に確立して，両者の通信コネクションの状態を同期させる方式を提案する．
セキュリティログ分析技術（要件3）	提案方式	・各種IDSログに共通に含まれるパラメータを統合管理する手法を提案する． ・長期間のイベントの検知状況を比較対象として，短期間のイベントの検知状況から，IDSログに含まれる各種イベントの異常性について客観的に順位付けする手法を提案する．

併設して侵入者の通信コネクションを動的におとりシステムへと誘導する制御手法を提案している．

また，本研究の一つとして 4 章では，各地のネットワークに設置された IDS から出力されるログを統合管理して分析する技術に関する検討であり，時間軸上でのイベント数の変化の異常性を客観的に評価する分析手法を提案している．

以上，本研究での提案技術を集めた SOC の概要を図 1.6 にまとめる．

Web サービスに関するリモートシステム監視技術

表 1.1 の課題で示した Web サービスへの各種攻撃を確実に検知するために，2 章において，外部の SOC からの視点で監視する Web サーバリモート監視システムを提案する．これは，SOC から定期的にホームページファイルをダウンロードして，あらかじめ登録されているファイルと比較することで改竄攻撃を検知する方式である．本提案システムは，ホストベース異常検知の要素と，SOC によるリモートシステム監視の要素を併せ持っている．外部の SOC から監視することで，閲覧者の視点

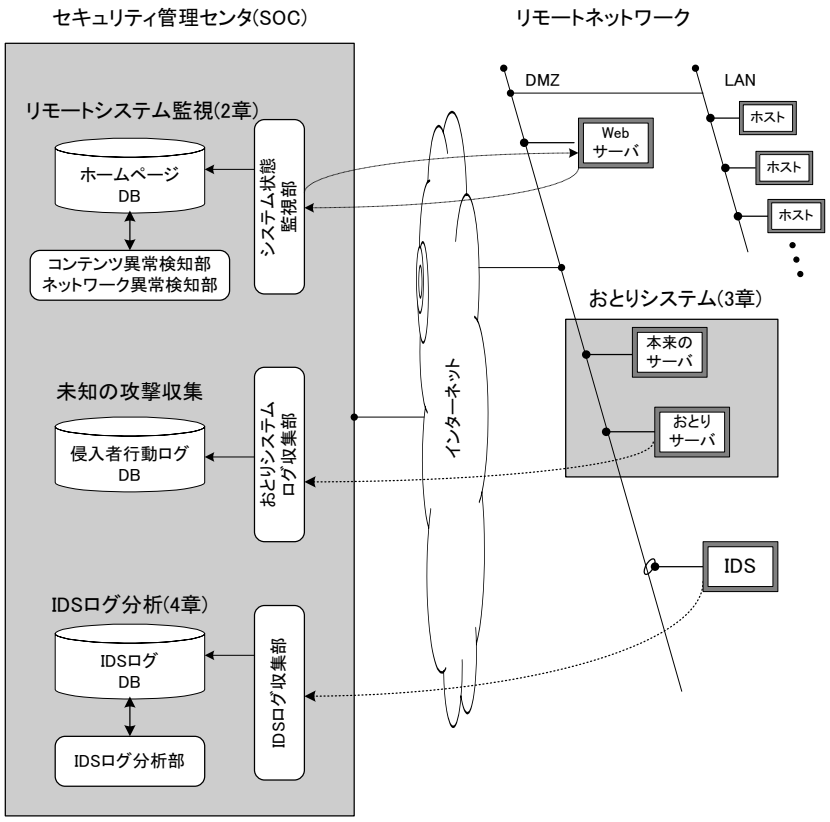

図 1.6: 本論文の提案技術を集めた SOC

に立って DNS スプーフ攻撃を検知できることや，ファイルをダウンロードするときの速度を測定することで DDoS 攻撃による Web サービスレスポンスの低下や停止を検知できるようになる．また，標準的な Web プロトコルによりファイルをダウンロードできるため，OS ごとに IDS を必要とせず，監視の統合化によるコストの低減を図ることができる．さらに，表 1.1 のネットワーク輻輳の課題を解決するために，定期的に独立した処理プロセスを起動して監視する手法と，処理プロセスをさらにスレッドに分割してファイルを並列にダウンロードして検査を行う手法を提案する．監視処理能力についてインターネット上での実証実験の結果，インターネッ

ト上の局所的な輻輳に影響を受けない安定したリモート監視を実現できていることを定量的に明らかにする．

おとりシステムによる未知の攻撃情報収集技術

表 1.1 の課題で示した正規システムを保護しながら情報収集を円滑に行うために，3 章において，IDS を利用して不審な挙動を知らせるアラームをきっかけに，通信コネクションの開始時点のみならず継続中の通信コネクションについても，正規のシステムからおとりシステムへと迅速に誘導する手法を提案する．さらに，表 1.1 の誘導時の通信シナリオの一貫性を確保するために，あらかじめ誘導前のクライアントからの通信を正規システムとおとりシステムの両方へ繋げて，正規システムとおとりシステムの通信状態を同期させておき，誘導後の通信シナリオを継続させる手法についても提案する．本誘導機能は，様々な通信アプリケーションサービスに適用できるが，本研究では FTP ならびに HTTP サービスへ適用したときの実装および評価を行い，侵入者に気付かれないレベルの高速な誘導を実現していることを明らかにする．

SOC におけるセキュリティログ分析技術

表 1.1 の課題で示した各地の IDS ログを統合管理するために，4 章において，各種 IDS に共通して含まれる Attack Signature，Source/Destination Port，Source/Destination IP の 5 つのパラメータを扱う統合 DB フォーマットを提案する．さらに，表 1.1 の異常なイベントを客観的に検出するために，各パラメータの過去の長期プロファイルを基準データにして，最近の短期プロファイルのデータの変化量を異常率として算出する手法を提案する．各地で運用されている IDS ログを用いて評価を行った結

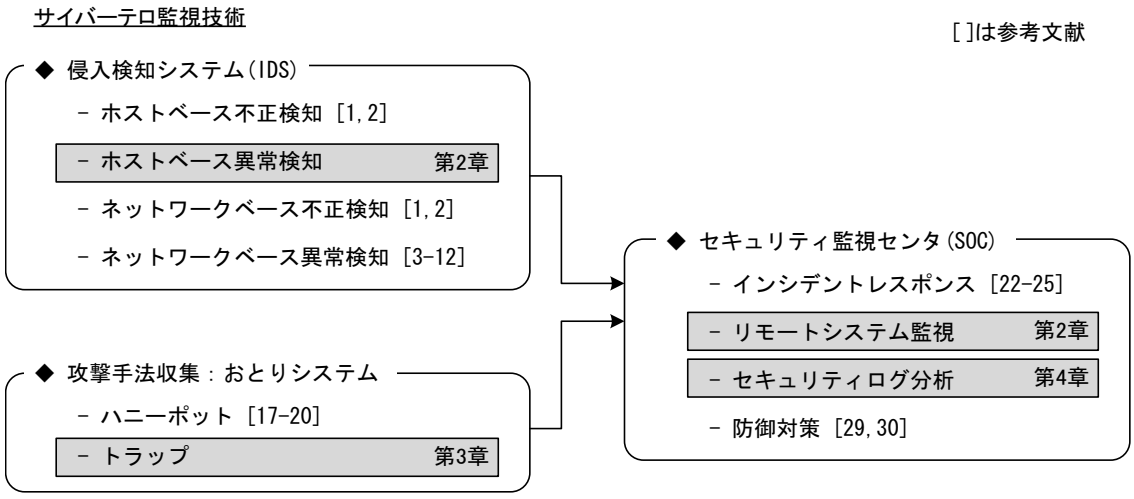

図 1.7: 既存技術に対する本研究の位置付け

果，従来からの頻度分析データの中から，冗長なイベントを特定できること，発見が困難であった微かな痕跡を検出できることを明らかにする．

1.6.2　既存技術に対する本研究の位置付け

図 1.7 にネットワークシステムのサイバーテロ監視技術における本研究の位置付けを示す．図には，既存技術の参考研究番号も示してある．

従来の IDS に関する研究として，ネットワーク上を流れるパケット，もしくは，ホスト OS やアプリケーションから出力されるログの中から，攻撃に関わる記録を検出する研究が行われてきた [3]-[12]．近年，Web サービスへの攻撃が問題になる中，従来からの IDS では攻撃の試みを検知することはできるものの，実際に被害が出たことを確実に検知するには至っていなかった．そこで本研究では，ホスト上のファイルを定期的に検査するホストベース異常検知技術を応用して，これらの被害を迅速に検知する手法を提案している．また，従来の IDS は監視対象のホストもしくは

ネットワーク上に設置されていたが，本研究では IDS の機能を外部の SOC に切り出しており，サイバーテロ監視技術の新たな手法を提案している．

従来のおとりシステムとして，脆弱性のあるホストをネットワーク上に設置して，ここへクセスしてくる侵入者のログを収集するハニーポット方式のおとりシステムが，その実装の容易性から研究・調査されてきた [17]-[20]．しかし，正規のシステムへの侵入を試みる通信を自動的におとりへ誘導して情報収集するトラップ方式のおとりシステムが，収集されるログの有用性と本来のシステムを守れる効果から，その概念モデルが注目されていた [1, 21]．本研究では，トラップ方式のリアルタイムな誘導制御を提案・実装しており，トラップ方式のおとりシステムの実現に寄与している．

SOC 構築における技術の一つとしてのインシデントレスポンスは，CERT/CC や JPCERT/CC の活動により達成されてきた [22]-[25]．また，各地のシステムを統合的にリモート監視する技術については，本研究の 2 章において達成する．ここで，IDS から出力されるログの分析については，単一種類の IDS ログを対象にして侵入を検知する研究がなされていた [14]．しかしながら，広域に及ぶ異常を発見するためには，各種 IDS のログを統合管理して，各地のネットワークの状況を考慮して分析する手法がなかった．そこで，本研究では，各種 IDS ログに共通して含まれる事象に注目して，その時間的な変動から異常なイベントを客観的に抽出する分析手法を提案している．防御対策技術については，上記分析手法で出力される情報を活用する研究は開始されたばかりである [29, 30]．今後いかに正確な分析結果を，いかに活用するかという技術が，ネットワークオペレータならびにユーザにとっての安心・安全なネットワーク社会の構築に寄与することになる．

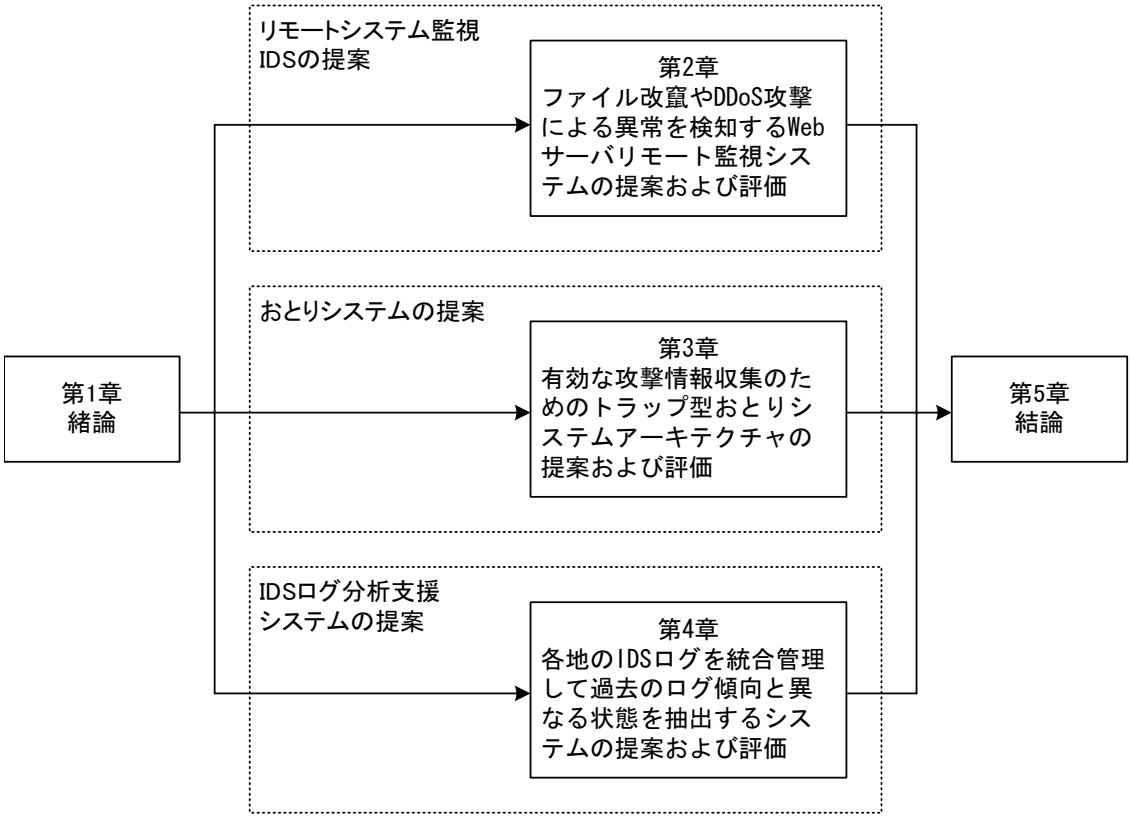

図 1.8: 本論文の構成

1.6.3　本論文の構成

本論文は，図 1.8 に示す構成から成る．第 1 章は，緒論である．第 2 章は，Web サーバに対するファイル改竄や DDoS 攻撃による異常を検知するリモート監視型 IDS に関する提案である．第 3 章は，システム固有の攻撃情報を収集できるトラップ型おとりシステムに関するアーキテクチャの提案である．第 4 章は，セキュリティ監視センタにおけるログ分析支援システムに関する提案である．第 5 章は，結論であり，本論文の内容を総括するとともに，今後のサイバーテロ監視技術と，ここで収集された情報を基に防御対策へと活用するための技術の展開について概観する．

第1章参考文献

[1] E. Amoroso, "Intrusion Detection", Intrusion.Net Books, New Jersey, 1999.

[2] 武田 圭史, 磯崎 宏, "ネットワーク侵入検知", ソフトバンクパブリッシング, Jun. 2000.

[3] S. Northcutt, M. Cooper, M. Fearmow and K. Frederick, "Intrusion Signatures and Analysis", New Riders Publishing, Indianapolis, Jan. 2001.

[4] 貝嶋 創, 磯崎 宏, 武田 圭史, 武藤 佳恭, "侵入検知システム評価方法に関する研究", 情報処理学会, 第 62 回全国大会, 7F-01, Mar. 2001.

[5] S. Northcutt, "Network Intrusion Detection an Analyst's Handbook", New Riders, 1999.

[6] P. E. Proctor, "The Practical Intrusion Detection Handbook", Prentice Hall, New Jersey, 2001.

[7] K. Ilugun, R. A. Kemmerer and P. A. Porras, "State Transition Analysis: A Rule-Based Intrusion Detection Approach", IEEE, Transactions on Software Engineering, Vol. 21, No. 3, Mar. 1995.

[8] H. Debar, M. Becker and D. Siboni, "A Neural Network Component for an Intrusion Detection System", IEEE, 1992 Computer Societyj Symposium on Research in Security and Privacy, pp. 240–250, 1992.

[9] J. M. Bonifacio Jr, A. M. Cansian, A. C. P. L. F. de Carvalho and E. S. Moreira, "Neural Network Applied in Intrusion Detection System", IEEE, International Joint Conference on Neural Networks, pp. 205–210, 1998.

[10] A. K. Ghosh and A. Schwartzbard, "A Study in Using Neural Networks for Anomaly and Misuse Detection", 8th USENIX Security Symposium, 1999.

[11] W. Lee, S. J. Stolfo, and K. W. Mok, "Adaptive Intrusion Detection: A Data Mining Approach", Artificial Intelligence Review, Vol.13, No.6, pp. 533–567, 2000.

[12] 鴨田 浩明, 馬場 達也, 小久保 勝敏, 松田 栄之, "ニューラルネットワークを利用した不正アクセス被害予測方式の検討", 情報処理学会, 第 62 回全国大会, 1S-1, pp. 283–284, Mar. 2001.

[13] D. Marchette, "A Statistical Method for Profiling Network Traffic", Workshop on Intrusion Detection and Network Monitoring (ID'99), 1999.

[14] K. Julisc, "Mining Alarm Clusters to Improve Alarm Handling Efficiency", 17th ACSAC, December 2001.

[15] C. C. Michael and A. Ghosh, "Two State-based Approaches to Program-based Anomaly Detection", DARPA, DAAH01-98-C-R145, 1998.

[16] R. Sekar, M. Bendre, D. Dhurjati, P. Bollineni, ”A Fast Automaton-Based Method for Detecting Anomalous Program Behaviors”, IEEE, 2001 Symposium on Security and Privacy, (S&P’2000), pp. 144–155, May 2000.

[17] The Honeynet Project, `http://project.honeynet.org/`

[18] The Honeynet Project Members, ”Know Your Enemy”, Intrusion.Net Books, New Jersey, Jul. 2000.

[19] L. Spitzner, ”Honeypots”, Addison-Wesley, Boston, USA, 2003.

[20] 渋谷 芳洋, 小池 英樹, 高田 哲司, 安村 通晃, 石井 威望, ”高対話型おとりシステムの運用経験と評価”, コンピュータセキュリティシンポジウム 2003 (CSS’2003), pp. 587–592, Oct. 2003.

[21] 宮川 明子, 稲田 徹, 後沢 忍, ”不正侵入者を外部ネットワークに設置したおとりサーバへ誘導するセキュリティシステムの検討”, 電子情報通信学会, 情報セキュリティ研究会, ISEC2001-49, pp. 225–230, Jul. 2001.

[22] Computer Emergency Response Team/Coordination Center (CERT/CC), `http://www.cert.org/`

[23] Japan Computer Emergency Response Team/Coordination Center (JPCERT/CC), `http://www.jpcert.or.jp/`

[24] 大林　正英, 石田　晴久, ”インターネットにおける不正アクセス対応と JPCERT/CC”, 電子情報通信学会, FACE98-24, pp. 23–27, Dec. 1998.

[25] 山口 英, 鈴木 裕信, ”情報セキュリティ”, 共立出版, Sep. 2000.

[26] 沢田 篤史, 高倉 弘喜, 岡部 寿男，”開放型大規模ネットワークのための IDS ログ監視支援システム”, 情報処理学会論文誌, Vol. 44, No.8, pp. 1861–1871, Aug. 2003.

[27] 高田哲司, 小池英樹：見えログ：情報視覚化とテキストマイニングを用いたログ情報ブラウザ，情処論文誌, Vol.41, No.12, pp.3265-3275, 2000 年 12 月.

[28] 宮地玲奈，小宅宏明，川口信隆，岡田謙一，重野寛：機械学習によるネットワーク型 IDS の fales positive 削減手法の提案，情処技報，CSEC，2003 年 5 月.

[29] 田村 研輔，松浦 幹太，今井 秀樹，”データマイニングを用いた IDS ログ分析結果の活用”，電子情報通信学会，2004 年 暗号と情報セキュリティシンポジウム (SCIS2004)，3B4-4，pp.1155–1160, Jan. 2004.

[30] 栗林 利光，白石 善明，森井 昌克，”イベント依存モデルによる不正アクセスの被害予測”，電子情報通信学会，2004 年 暗号と情報セキュリティシンポジウム (SCIS2004)，3B3-2，pp.1029–11034, Jan. 2004.

[31] 可部孝二, 曽我正和, 西垣正勝, 田窪昭夫, “ ホームページ改竄パトロール方式”, 情報処理学会, コンピュータセキュリティ研究会, CSEC-8-30, pp.173–178, Mar. 2000.

第2章

Webサーバリモート監視システムの実装および評価

1章では，SOCにおけるセキュリティ監視の要件として，

(i) 各地のネットワークシステムの状態をリモート監視すること

(ii) 未知の攻撃情報を収集する基盤技術を確立すること

(iii) 各地のセキュリティシステムから出力されるログを統合分析すること

について述べてきた．(i)は，ネットワークシステムへの侵入を完全に防ぐとこはできないことを前提としており，侵入後の被害をいち早く見つけ出し，迅速な対応を図るための事後対策の要件である．本章では，(i)の要件を達成するためのシステム提案ならびに実装・評価を行う．

2.1 概要

近年，インターネットの代表的なサービスであるWebサーバに対する不正アクセスが大きな社会問題となってきている．Webサーバを構成する機器には，日々セキュリティホールが報告されており，その全てに対して迅速かつ適切なセキュリティ

対策を図ることは難しく [1]，悪意のある侵入者による攻撃や，インターネットワーム [2] による攻撃を完全に防ぐことは困難である．このような中，トラヒックやログデータを参照することで攻撃を検知する侵入検知システム [3] や，侵入を試みる者の挙動を収集してシステムのセキュリティホールを塞ぐおとりシステム [4]，ホスト上でファイルを定期的に監視して変更を検知する改竄検知システム [5]–[8] などに関する研究，開発が盛んに行われている．

しかしながら侵入検知システムの場合，監視対象になっていない外部ネットワーク機器への攻撃，例えば最近問題となっている DNS スプーフ攻撃 [9],[10] による Web サーバのなりすまし等を検知することができない．また，DDoS 攻撃を受けたときの外部からみた影響について評価することができない．さらに，未知の手法により攻撃された場合，これを検知することはできず，改竄を未然に防ぐことができない．現在の対策技術として，Web サーバに常駐してファイルを監視し続けておき，いざ改竄された場合でもいち早くアラームをあげて，被害を最小限に抑えることを目的とした改竄検知システムがある．このシステムの場合，OS 毎に専用のシステムを用意する必要があることや，侵入された場合に改竄検知システム自体が攻撃されることで，検知サービスを止められる恐れもある．

そこで本章では，上述の検知システムの問題を解決するために，検知機能を検知対象サイトの外部に設置して，簡単な操作で抜けのない Web ページの検知を一元的に実行する Web サーバリモート監視システムを提案する．リモートから監視を行うことで，複数の Web サーバを一括して管理できるようになる反面，ネットワークへ与える負荷を軽減する機能が必要となる．また，ネットワーク遅延の影響を抑えるための監視処理の並列化機能を実装し，評価を通じて，提案システムの有効性，実現性について述べる．以下 2.2 節において，Web サーバに対する攻撃，既存の検知システムの原理，および，その問題点について説明し，2.3 節で問題点を解決すべ

く，監視システムを提案する．2.4 節でリモート監視を行う上での設計，実装手法について述べ，2.5 節でインターネット上での性能評価を実施する．さらに，2.6 節で全体の要約と共に，本システムの検討課題について触る．

2.2 攻撃例と既存の攻撃検知システム

ここでは，SOC によるリモート監視技術において，検知すべき Web サーバに対する攻撃を説明する．

2.2.1 Web サーバに対する攻撃例

DNS スプーフ攻撃

DNS スプーフ攻撃は，攻撃者が DNS における Domain Name と IP アドレスの対応を書き換えることで，正規サーバになりすましたサーバへと閲覧者を誘導する攻撃である．本攻撃は，Web サーバ自体のセキュリティを確保するだけで防御することはできない．

DDoS 攻撃

DDoS 攻撃は，例えばトロイの木馬プログラムに感染したホストから，特定のターゲットホストへトラヒックを集中させることで，ターゲットホストが属するネットワークサービスを不能にする攻撃である．

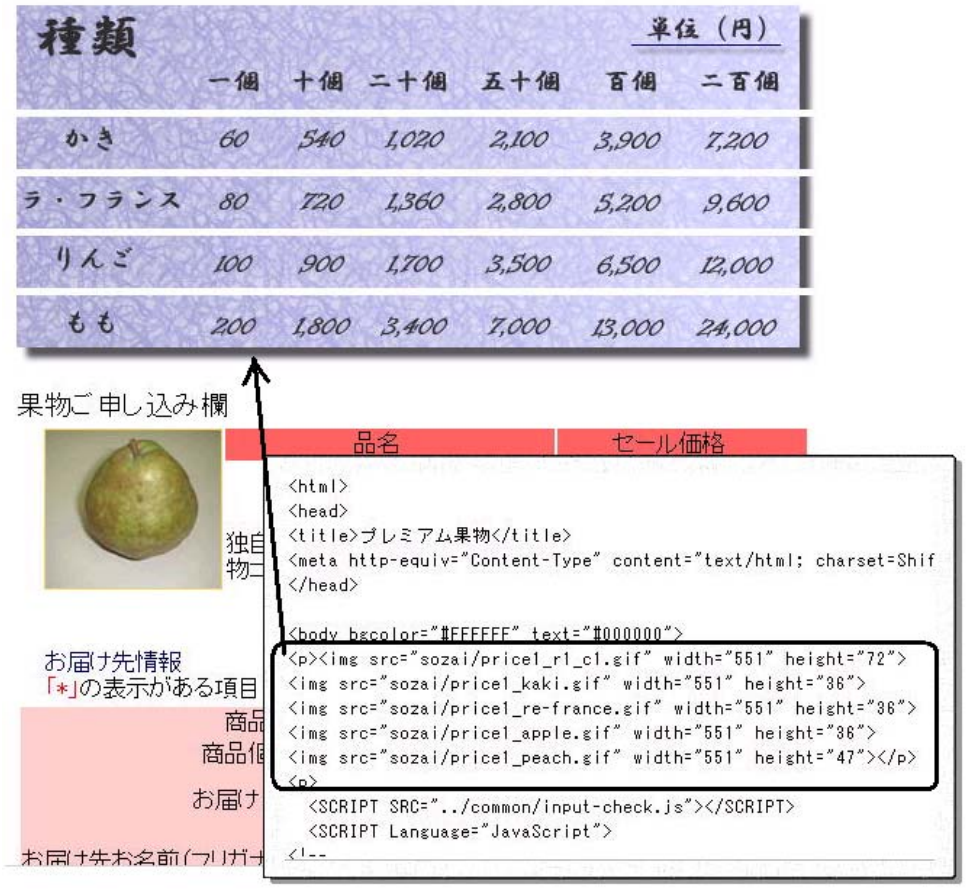

図 2.1: Web ページの構成例

改竄攻撃

多くの Web ページは，基本となる HTML ファイルと，これを構成するファイルの集合から構成されている．E-コマースサイトを例に，この様子を図 2.1 に示す．コンテンツ改竄攻撃者は，この基本となる HTML ファイル自体を改竄することで，ページの全体的な見た目を変更したり，一部の構成ファイル，例えば価格ファイルを変更したりすることで，サイト運営に支障を与える．

2.2.2 既存の攻撃検知システムの原理

侵入検知システム

既存の侵入検知システムは，あらかじめ登録された攻撃パターンとのマッチングにより検知をしており，監視するシステム環境，例えばネットワーク構成等を考慮したものではない．

改竄検知システム

改竄攻撃を検知する手法として，一方向性ハッシュ関数 [11] を用いたファイル監視システムがある [5],[6]．これは，監視対象のオリジナルファイルから求めたハッシュ値をその特徴として保持しておき，定期的に対象ファイルを取得してそのハッシュ値と比較することで，改竄を検知する手法である．本手法の信頼性は，オリジナルファイルのハッシュ値と，改竄後のハッシュ値を一致させることが極めて困難であることに依っている．改竄検知システムは，ホストに常駐して監視対象ファイルの変更を定期的に検査している．[7],[8] では，Web ファイル作成者がその正当性を証明するための署名データを作成し，Web ファイルならびに署名データの両方を Web サーバ上で管理することで，作成者が意図しないファイルの変更，要するに改竄を確実に検出するシステムとなっている．

2.2.3 既存の攻撃検知システムの問題点

上記検知システムに関する基本的な問題点と，改竄攻撃におけるページの構成ファイルを網羅的に監視するときの問題点をまとめる．

(問題点 1) 管理下にない外部の DNS に対する攻撃を検知できないことや，DDoS 攻

撃を受けた場合の影響度を閲覧者の立場から評価することができない.

(問題点 2) いざ侵入されたときには，常駐している検知システム自体に攻撃を受けてしまう.

(問題点 3) 複数のホストを個別管理することにより，運用コストが増大する.

(問題点 4) ホストの OS 毎にシステムを用意しなければならない.

(問題点 5) ホストに検査のための処理負荷を与える.

(問題点 6) ファイル管理者が，全てのファイルを抜けなく登録する必要があり，作業は煩雑なものとなる.

2.3 提案監視システム

2.2 節では，各地の Web サーバシステム内で攻撃を監視するときの問題点についてまとめてきた．ここでは，これらの問題を解決するために，Web サーバを SOC からリモート監視するシステムを提案する.

2.3.1 リモート監視

従来のホスト常駐型 Web 監視システムにおける上記 1)，2)，3)，4) の問題点を本質的に解決するために，監視機能を検知対象 Web サイトから切り離し，遠隔から

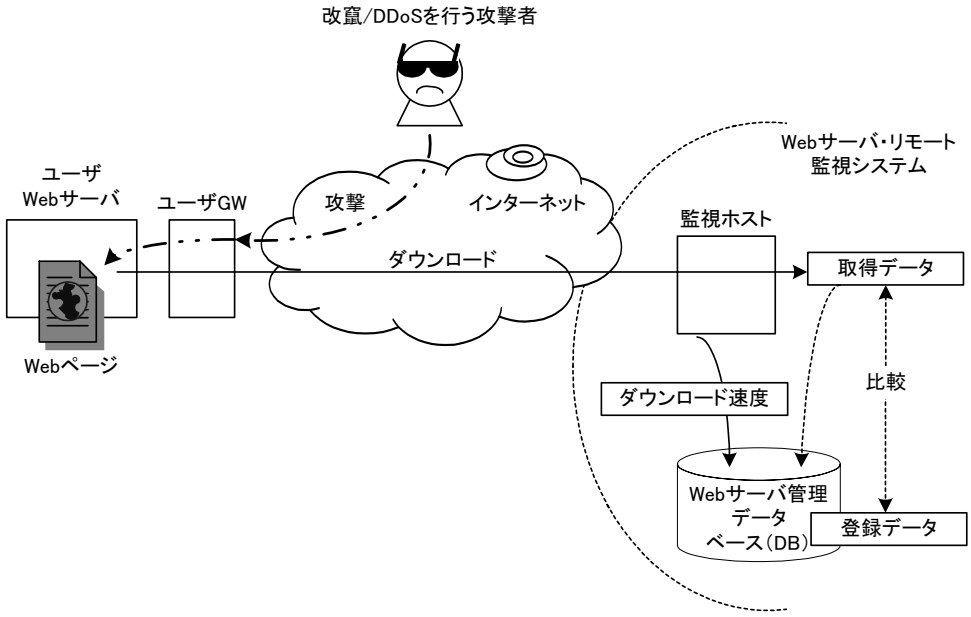

図 2.2: Web サーバリモート監視システム

Web サーバへの攻撃を検知するシステムを提案する（図 2.2）．具体的に本システムは，定期的にデータを取得して，あらかじめ登録されているデータと比較することで，改竄攻撃を検知する．ここで，多くの Web サイトにおいて HTML ファイルの管理者と作成者は異なる．作成者によって頻繁に更新される HTML ファイルを，更新のたびに管理者を通じてオフラインで収集することは煩雑である．よって本システムでは，監視中にダウンロードされた HTML ファイルを基準ファイルにして，次回ダウンロードされるファイルを検査することにする．監視ホストをリモートへ設置することで，DNS スプーフ攻撃を検知できるのみならず，ファイルを取得するときの取得速度を測定することで，DDoS 攻撃やトラヒックの増加による Web サーバレスポンスの低下や停止を検知できるようになる（2.2.3 節の問題点 1 の解決）．また，監視システムそのものに対する攻撃への耐性も保証できる（2.2.3 節の問題点 2

の解決）．さらに，複数のサイトをまとめて管理できるようになり監視のための運用コストを下げることができると共に（2.2.3 節の問題点 3 の解決），監視を受けるホストの OS ごとにシステムを用意する必要がなくなる（2.2.3 節の問題点 4 の解決）．

2.3.2 リモート監視における課題

2.2.3 節の問題点 5) で挙げられる負荷について，監視機能をホストの外部へ設置することで，新たにネットワークに与える負荷も問題となる．

課題 A) 監視先のネットワークへ与える負荷を軽減すること．

リモートから監視することで，複数の Web サーバを一括して管理できるようになる反面，ネットワーク輻輳の影響を受けやすくなり，定期的な監視を行う上でその輻輳による処理時間の増加が問題となりうる．ネットワーク遅延の大きなサイトに，定期的な監視処理が引きずられるべきではなく，また，ネットワーク待ち時間中の CPU リソースを無駄に消費すべきではない．

課題 B) ネットワーク輻輳が発生した場合においても，定期的な監視処理がスムーズに開始されること．

課題 C) ネットワーク遅延に影響されることなく，監視ホストにおける監視処理の高速化を図ること．

2.3.3 監視手法

本システムでは，2.3.2 節の課題 A) を考慮した簡易検査と，高度な手法による改竄を検知するための厳密検査を提案する．この様子を図 2.3 に示す．

簡易検査は，できるだけネットワーク負荷を与えないために，ファイルのコンテン

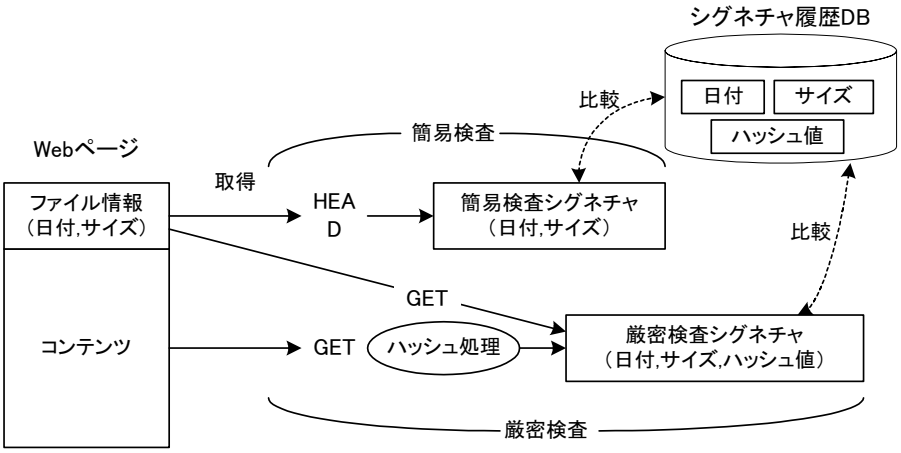

図 2.3: 簡易検査と厳密検査

ツを取得することなく，ファイル情報（日付・サイズ）をシグネチャとして，HTTP-HEAD リクエストで取得して検査を行う．本手法は，昨今のインターネットワームや愉快犯による改竄の成功を広く知らしめることを目的とした攻撃に対して有効であり，現在被害が多発している改竄攻撃の殆どを検知することができる．

厳密検査は，高度な手法による改竄を検知するために，HTTP-GET リクエストでファイル情報とコンテンツの両方を取得した後に，コンテンツについてはハッシュ処理を行い，ファイル情報とハッシュ値をシグネチャとして検査する．本手法は，ファイルの日付やサイズはそのままに，コンテンツのみを改竄する攻撃，例えば図 2.1 のような E コマースサイトの価格表を狙った攻撃に対して有効である．

本システムは，Web ページを構成するファイル毎に，簡易検査と厳密検査を必要に応じて組み合わせることができる監視機能を提供することとした．例えば，価格表のような高度な改竄攻撃のターゲットとなりうるファイルについては毎回の監視で厳密検査を設定して，その他のファイルについては，2.3.2 節の課題 A) を考慮して，簡易検査を 5 回実施した後に厳密検査を 1 回実施する周期で監視を行う．

2.3.4 リンク解析

本システムでは 2.2.3 節の問題点 6) を解決するために，基準となる HTML ファイル（以後，マスタ HTML ファイル）から，リンクを探索して，監視すべきファイル（以後，監視ファイル）を自動的に抽出するリンク解析エンジンを設ける．ここで，監視ファイルはマスタ HTML ファイルも含む．これにより，煩雑になりがちな監視ファイルの設定が容易になり，設定ミスを防ぐことができるようになる．

リンクの探索で注意すべき点は，外部ホストのファイルへリンクされているもの，リンクの関係がループ状になっているものを除くことである．本リンク解析エンジンによりマスタ HTML ファイルから抽出される監視ファイルの関係を図 2.4 に示す．リンク機能を持つタグには，＜ FRAME ＞タグ，＜ IMG ＞タグ，＜ A ＞タグ，＜ EMBED ＞タグ，＜ BGSOUND ＞タグ，＜ FRAME ＞タグ，＜ OBJECT ＞タグがある [12]．以下に本章で検討したリンク解析エンジンについて示す．

＜ FRAME ＞タグ 本タグは，Web ページにフレームを指定するタグである．マスタ HTML ファイルに本タグが記述されている場合，本タグが指定する HTML ファイルと，その HTML ファイルからリンクされている下記＜ IMG ＞タグ，下記＜ A ＞タグの探索条件に合ったファイルを，監視ファイルとする．

＜ IMG ＞タグ 本タグは，画像へのリンクを指定するタグである．マスタ HTML ファイルもしくは＜ FRAME ＞タグで指定された HTML ファイルの中から，本タグでリンクされているファイルを，監視ファイルとする．

＜ A ＞タグ 本タグは，テキストや画像へのリンクを指定するタグである．本タグの場合，リンク探索時にリンクのループ関係が問題となる．このため，マスタ HTML ファイルもしくは＜ FRAME ＞タグで指定された HTML ファイル

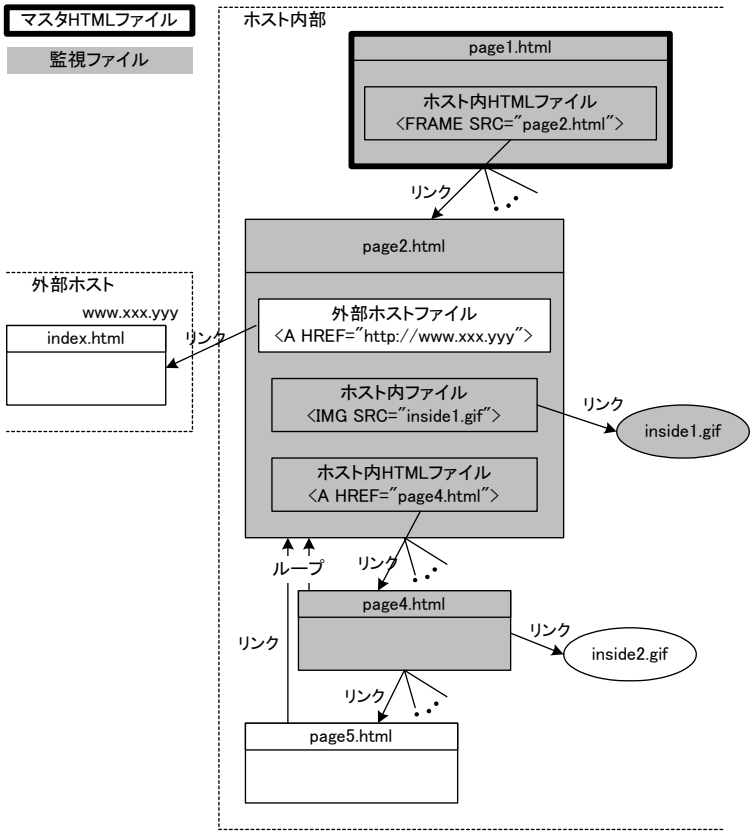

図 2.4: 検査対象ファイル

から，1 階層のみ探索を行い，そこで検出された＜ A ＞タグでリンクされているファイルを，監視ファイルとする．ただし，外部ホスト上のファイルへリンクされているものは除く．

2.4 設計・実装

2.2 節では，Web サーバに対する攻撃監視を行うときの課題について述べ，これを解決するたに 2.3 節で SOC によるリモート監視手法を提案した．ここでは，輻輳

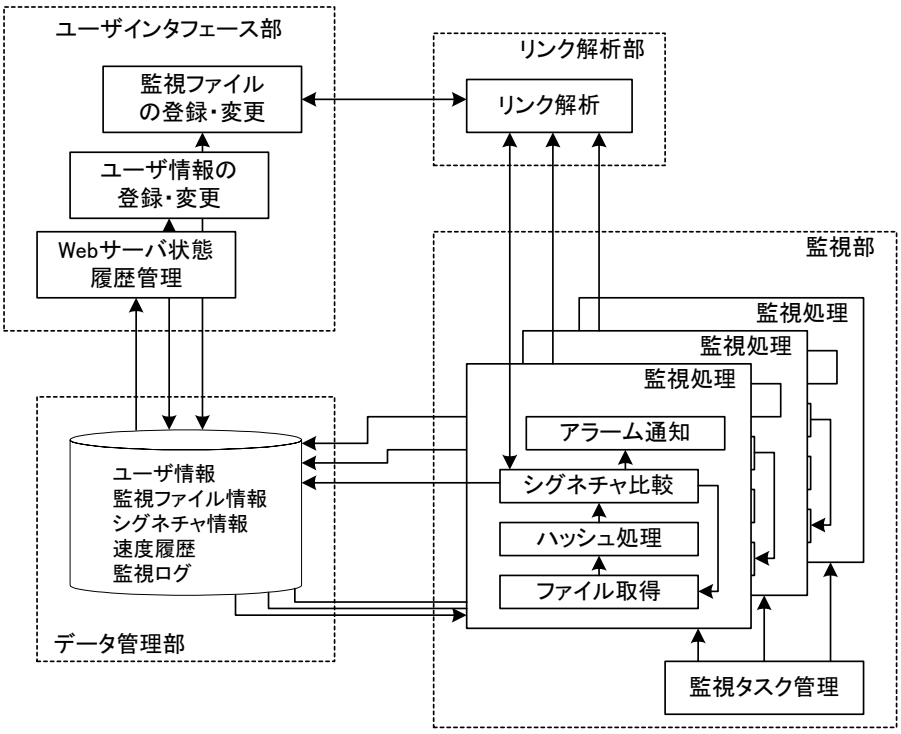

図 2.5: システム機能構成

しがちなネットワーク経由で監視を行う際に，安定した処理を実現するためのシステム設計ならびに実装を行う．

Web サーバリモート監視システムの機能モジュール構成を図 2.5 に示す．本システムは，ユーザインタフェース部，リンク解析部，データ管理部，監視部に大別される．

2.4.1 ユーザインタフェース部

ユーザインタフェース部は，監視対象となるユーザ情報，監視ファイル情報を，Web ブラウザを用いて登録，変更するための機能である．また，監視履歴を閲覧す

図 2.6: リンク自動解析例

るためのインターフェースでもある．

2.4.2　リンク解析部

3.4 節で説明したリンク解析エンジンとして，＜FRAME＞タグ，＜IMG＞タグ，＜A＞タグの抽出機能を実装した．その出力結果例を図 2.6 に示す．抽出されたファイル毎に，簡易検査と厳密検査の監視間隔を任意に設定できる．

2.4.3　データ管理部

データ管理部では，ユーザ情報，監視ファイル情報，シグネチャ情報，速度履歴，監視ログを保存・管理する．

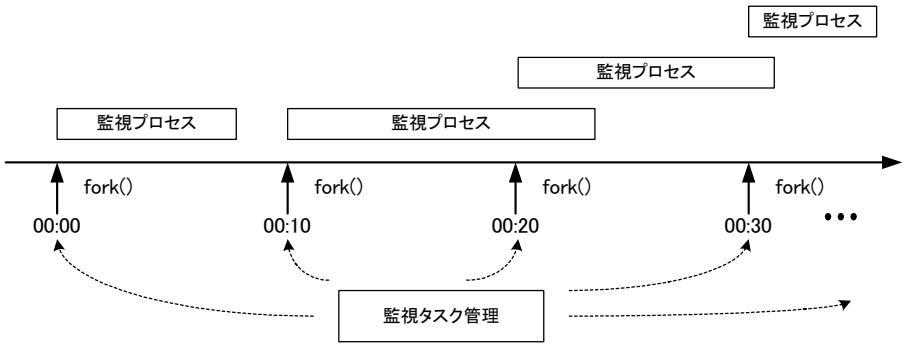

図 2.7: 監視プロセスの独立起動

2.4.4 監視部

監視プロセスの独立起動

2.3.2 節の課題 B) を考慮して，ネットワーク輻輳などの原因によりある時刻の監視処理に遅延が生じた場合でも，その遅延が次の監視時刻における処理の開始に影響を与えない設計が要求される．そこで本システムでは，定期的な監視時刻になると，独立した監視プロセスを起動して，その時刻に監視するファイル（以下，タスク）を割り当てる機能を設ける．これにより，監視間隔で処理しきれないタスクが，次の監視プロセスに引き継がれることがなくなり，遅延の積み重ねを防ぐことができるようになる．この様子を図 2.7 に示す．

監視スレッドの並列化

2.3.2 節の課題 C) を考慮して，CPU リソースの効率的な利用による監視処理の高速化を図るために，監視処理のスレッドの並列化を行う．これは，データ管理部から取得したタスクを，一つの監視タスク待ち行列へと取りまとめ，空いているスレッドへ割り当てる並列処理型の待ち行列モデルである．この様子を図 2.8 に示す．こ

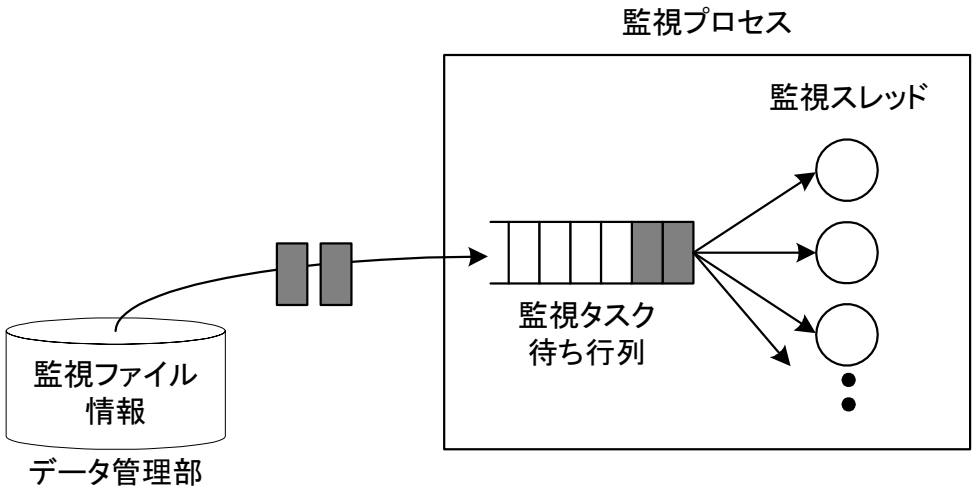

図 2.8: 監視スレッドの並列化

れにより，ネットワーク越しにファイルを取得するときに発生する CPU の空き時間を，他のスレッドへと割り当てることができるようになり，処理効率の向上へと繋がる．

監視処理

取得したファイル情報やコンテンツから求められたハッシュ値からシグネチャを生成し，データ管理部に登録されているシグネチャと比較することで，ファイルの変更を検知する．

本処理では，Web ページ作成者による正当なファイルの変更と侵入者による改竄を区別していない．このため本システムは，Web ファイルの更新確認にも利用できる．

アラーム通知

監視中に,

- ネットワークの障害によりレスポンスを受け取れない
- ファイルが削除されて存在しない
- ファイルが変更されてシグネチャが異なる

の状況が検知された場合，アラーム通知を行う.

2.4.5 監視処理の流れ

上記実装方針に従った本システムにおける改竄監視処理フローを図 2.9 に示す．定期的な監視時刻になると，データ管理部よりその時刻に検査を行う監視ファイル情報とそのシグネチャを一括取得する．そして，監視プロセスを起動し，スレッドを生成する．処理の空いているスレッドは，監視タスク待ち行列からタスクを取得する.

取得したタスクが簡易検査の場合，HTTP-HEAD リクエストでファイル情報を取得して，DB に登録されているシグネチャと比較する．取得したタスクが厳密検査の場合，HTTP-GET リクエストでファイル情報とコンテンツを取得して，コンテンツについてはハッシュ処理を行い，DB に登録されているシグネチャと比較する.

比較の結果，変更が検出されて，かつ，マスタ HTML ファイルでない場合は，そのままアラーム通知を行う．マスタ HTML ファイルの場合は，そのファイルを取得して，新規に追加されたリンクの有無を解析し，解析結果を含めてアラーム通知を行う．この新規リンクに関する通知は，新たにリンク付けされたファイルに対する監視の登録抜けを防ぐことを目的としている.

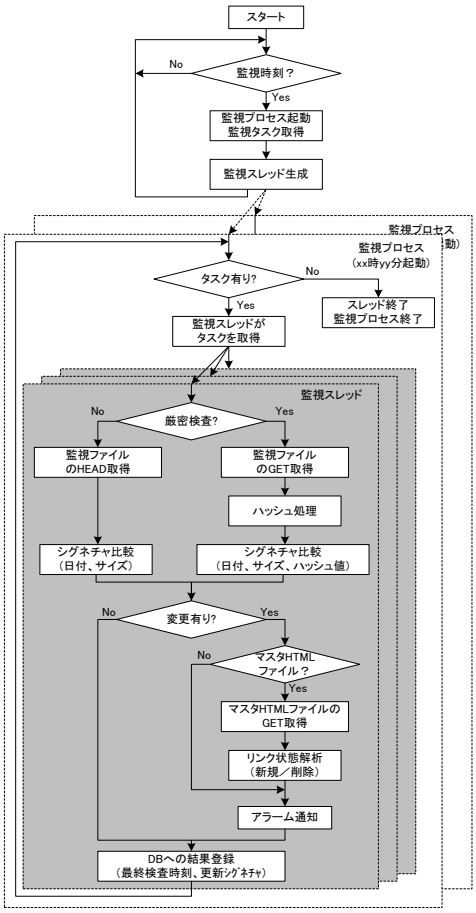

図 2.9: 監視処理フロー

検査の終了した監視ファイルについては，最新の検査時刻を DB へ登録する．このとき，変更があった監視ファイルについては，更新シグネチャも併せて登録する．

2.4.6 実装環境

3 台のホストをネットワーク接続して本システムを構築した．ユーザインタフェース部を，CPU が 400MHz-UltraSPARC Ⅱe，メモリが 384Mbyte，OS が Solaris8 の

表 2.1: 監視ファイル数とサイトサイズ

	32 サイト合計	1 サイト平均
監視ファイル数	1008	32
サイトサイズ	6,822 Kbyte	213 KByte

ホスト上で実装した．データ管理部を，CPU が 500MHz-UltraSPARC Ⅱe ，メモリが 512Mbyte，OS が Solaris8 のホスト上で実装した．リンク解析部と監視部を併せて，CPU が 400MHz-UltraSPARC Ⅱ e ，メモリが 384Mbyte，OS が Solaris8 のホスト上で実装した．

2.5 性能評価

2.3.2 節の課題 B)，課題 C) に関する本システムの処理能力について評価するために，インターネット上に公開されている 32 の企業のトップページをマスタ HTML ファイルとして試験を実施した．本システムから Internet Service Provider までのネットワーク上には，本システム以外のトラヒックも重畳している．

2.5.1 監視ファイルの概要

リンク解析エンジンを用いて，32 の企業サイトの監視ファイル数とその総サイズを調査した（表 2.1）．

表 2.2: 監視処理時間構成

処理	詳細
ホスト内処理	DB からのタスク取得
	監視タスク待ち行列処理
	ハッシュ（厳密検査）
	シグネチャ比較
	DB の更新
	アラーム通知
ネットワーク処理	HTTP-HEAD/GET アクセス処理

2.5.2 監視処理時間構成

本測定における監視処理時間は，大きく分けてホスト内処理時間とネットワーク処理時間から構成される．各処理に含まれる詳細な処理を表 2.2 にまとめる．

このホスト内処理時間とネットワーク処理時間について，監視スレッド数を 64 としたとき，表 2.1 の全ての監視ファイルに対して簡易検査および厳密検査を平日 6 時から 12 時の時間帯で実施したときの処理時間について，それぞれ 5 回測定した．この測定結果の平均値を図 2.10 に示す．

図 2.10 より，ホスト内処理時間の方がネットワーク処理時間よりも大きな割合を占めている様子がわかる．表 2.2 ならびに図 2.10 より，厳密検査を実施しているときの平均トラヒック量は，高々0.9Mbit/s であり，ネットワーク処理よりもホスト内処理の方が大きくなっている．ホスト内処理の DB 更新処理には，監視ファイル毎に最終検査日時情報を DB に記録する処理が含まれており，これが大きな割合を占めて

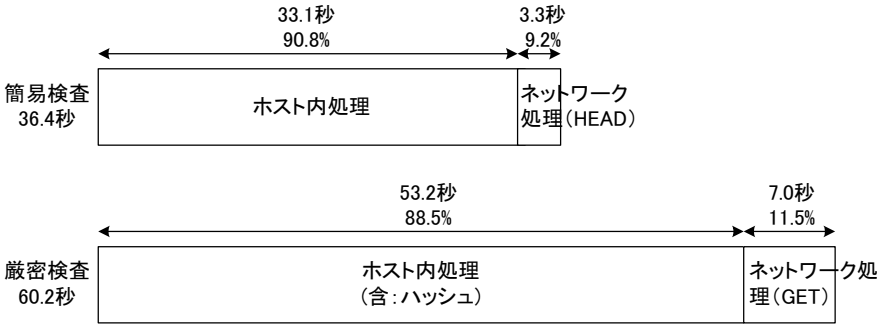

図 2.10: 監視処理時間と割合

いる．ネットワーク処理時間に注目すると，厳密検査のそれは簡易検査に比べて約 2 倍の時間が掛かっており，HTTP-GET によるコンテンツの取得は HTTP-HEAD によるファイル情報の取得の約 2 倍の負荷をネットワークへ与えていることがわかる．

2.5.3 時間帯に対する監視処理時間評価

簡易検査と厳密検査の監視処理時間について，平日 3 日分および休日 3 日分について測定した．測定は簡易検査ならびに厳密とも同じ日に実施しており，簡易検査は毎時 10 分に，厳密検査は毎時 20 分に測定した監視処理時間を，その時間帯の代表値とした．このときの監視スレッド数は 64 と設定した．

図 2.11 に，32 サイト全ての監視ファイルについて HTTP-HEAD による簡易検査を実施したときの時間帯に対する監視処理時間を示す．

図 2.12 に，32 サイト全ての監視ファイル について HTTP-GET による厳密検査を実施したときの時間帯に対する監視処理時間を示す．

図 2.11，2.12 より，殆どの簡易検査による処理時間は 36 秒程度であり，厳密検査による処理時間は 60 秒程度となっており，コンテンツの取得とハッシュ処理を実施

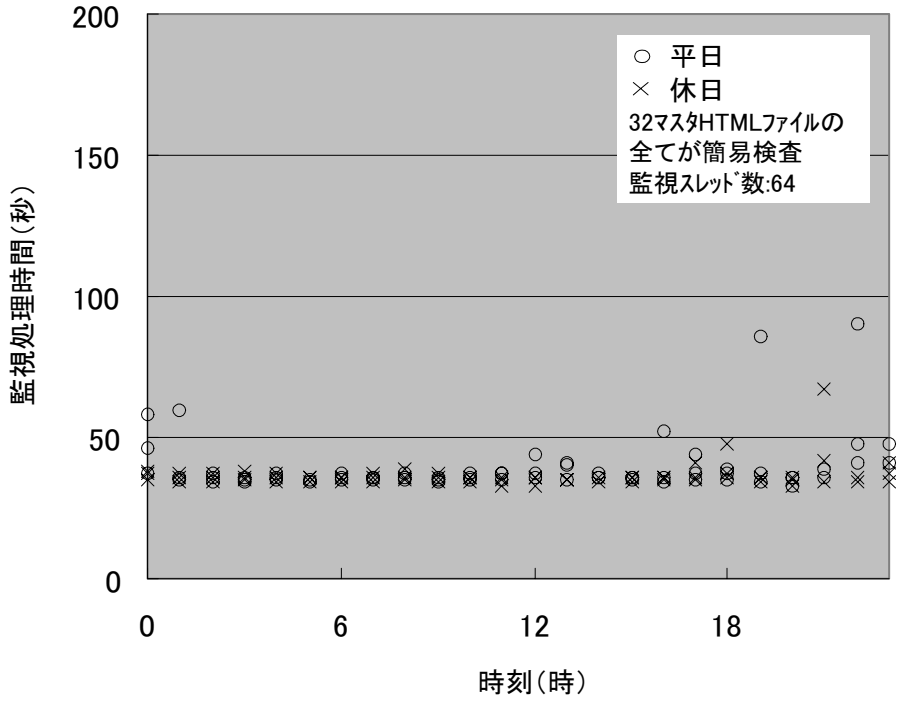

図 2.11: 簡易検査における時間帯別監視処理時間

しない簡易検査処理の方が短時間で完了している様子が分かる．夕方から深夜にかけて，多少なりとも監視処理時間が大きくなっているが，ほぼ一日を通じて安定した監視処理時間を実現しており，監視プロセスの独立起動，および，監視スレッドの並列化の効果が現れている（2.3.2 節の課題 B，C の解決）．平日ならびに休日の監視処理時間の差は殆どなかった．

2.5.4 監視スレッド数に対する監視処理時間評価

平日 6 時から 12 時の時間帯で，32 サイト全ての監視ファイルが簡易検査もしくは厳密検査の場合において，監視スレッド数を変化させたときの監視処理時間を 5

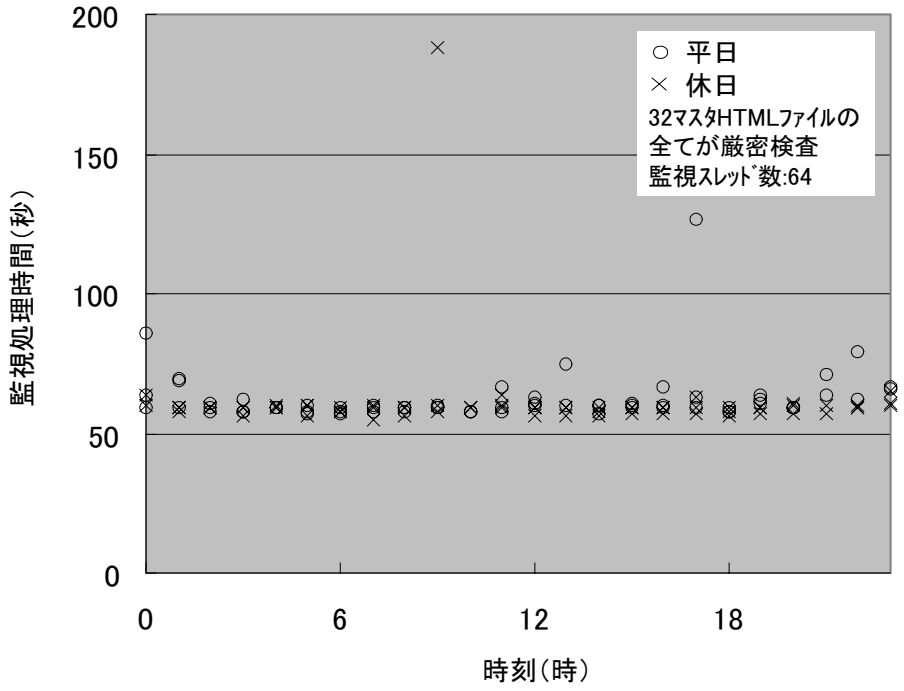

図 2.12: 厳密検査における時間帯別監視処理時間

回測定した．このときの結果を図 2.13 に示す．

今回の試験では，スレッド数が 64 のとき最小の監視処理時間を記録しており，ホスト内処理の並列化とネットワーク処理の並列化におけるスレッド数の最適値が存在することが分かる．これは，スレッド数が小さな場合では，ネットワーク処理における HTTP-HEAD/GET アクセス処理の間に CPU の空き時間が発生して，処理効率が低下しているためである．逆に，スレッド数が大き過ぎるときには，監視タスク待ち行列の排他制御や，DB の更新処理における排他制御の影響が現れているためである．スレッド数の最適値は，ネットワーク輻輳と CPU 負荷の分布に依存しており，時刻や曜日などの様々な要因によって変動する．本システムを効率良く運用するには，ネットワーク輻輳や CPU 負荷などの分布調査を行いながら，最適なス

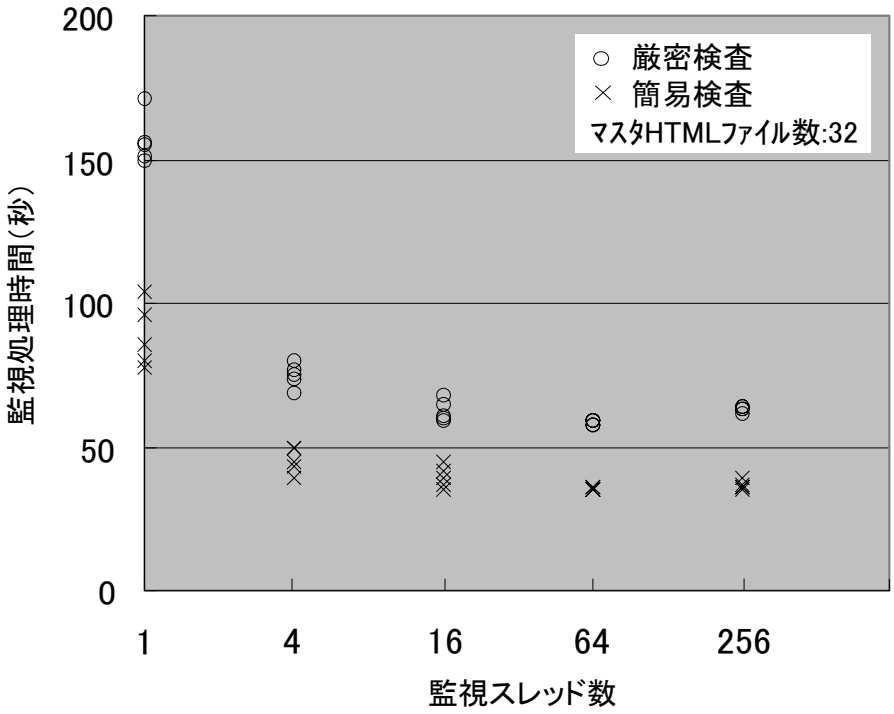

図 2.13: 監視スレッド数に対する監視処理時間

レッド数を設定・制御する必要がある．ちなみに図 2.10 は，今回の試験における最も処理効率の高い監視処理時間の割合を示している．

スレッド数が小さなときには，ネットワークの輻輳の影響を受けやすく，監視処理時間のばらつきが大きくなっている．一定間隔での監視を行うためにも，2.3.2 節の課題 C) を考慮してばらつきがなくなる程度の複数スレッドを設定する必要がある．

2.5.5 マスタ HTML ファイル数に対する監視処理時間評価

平日 6 時から 12 時の時間帯で，監視スレッド数が 64 で，32 サイト全ての監視ファイルが簡易検査もしくは厳密検査の場合において，監視するマスタ HTML ファイル

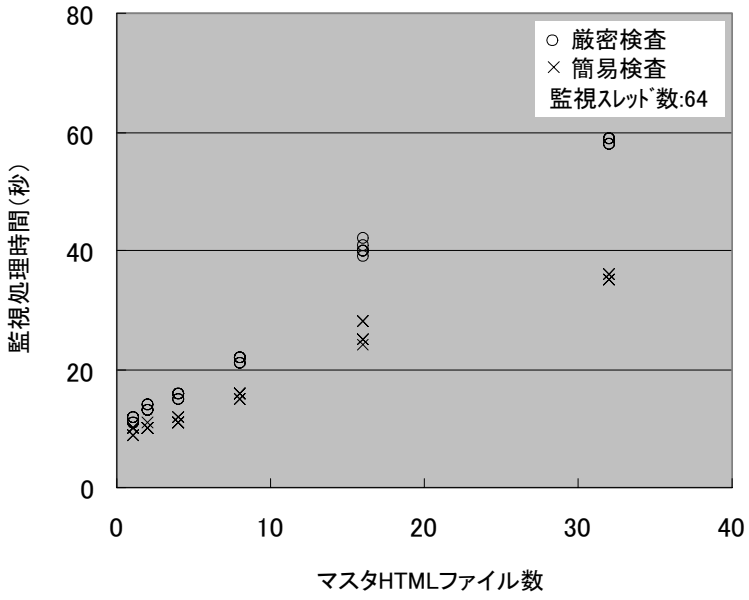

図 2.14: マスタ HTML ファイル数に対する監視処理時間

数を変化させたときの監視処理時間を 5 回測定した．このときの結果を，図 2.14 に示す．

図 2.14 より，マスタ HTML ファイル数が増えるに従い，ほぼ直線的に監視処理時間が増加していることが分かる．例えば本システム一式を用いた監視を行う場合，今回の 32 マスタ HTML ファイルの全てについて厳密検査を実施するならば 1 分間隔で監視を行うことができ，10 分間隔ならば約 300 マスタ HTML ファイルを監視することができ，一定の性能を確保できることが判明した．

2.6 総括

本論文では，Web サーバが管理するファイルをリモートから監視する，Web サーバリモート監視システムを提案した．監視機能をリモートへ設置することで，従来

の検知システムで問題となっていた点を解決し，

- DNS スプーフ攻撃，DDoS 攻撃を検知できる
- いざ Web サーバへ侵入された場合でも監視システム自体に対する攻撃を受けない
- 監視結果を統合管理できる
- Web サーバの OS 毎に監視システムを用意する必要がない
- 複数のファイルから構成される Web ページを簡単な操作で抜けなく監視できる

などを実現することができた．リモートから監視を行うに際して，ネットワークに与える負荷の軽減を目的とした簡易検査機能を設けると共に，高度な改竄攻撃の検知を目的とした厳密検査機能も設けて，これら二つの検査を，ファイル毎に必要に応じて組み合わせて監視を行える機能を用意した．また，本システムは，輻輳の起こりうるネットワーク上で高速な監視を実現するために，監視プロセスの独立化と，監視処理の並列化を図っている．これに関して，インターネット上で評価を行った結果，本手法が監視処理時間の短縮とばらつきを抑えることができる様子を確認した．

今回設計したリンク解析エンジンは，マスタ HTML ファイルにリンク付けされる 1 ないし 2 階層までのファイルを自動的に抽出することはできるが，Web サーバ全体のファイルを抽出するまでには至っていない．リンクのループを正確に判別しながら全てのファイルを抽出する機能については今後の課題とする．また，昨今の Web サービスで利用が進んでいる動的に変化するファイルについても，区別無く抽出しており，正常な変化を頻繁に検知しないためにも，運用者が手動で監視対象から外す必要がある．動的ファイルの自動削除については，マスタ HTML ファイルを

複数回取得して，その中から動的に変化しているファイルを検出するメカニズムが必要になり，これについても今後の課題とする．

第2章参考文献

[1] Web 改竄の防止を目的として行う価値のある対策,
http://www.ipa.go.jp/security/ciadr/webjack.htm,
情報処理振興事業協会.

[2] ワーム sadmind/IIS による Web 改竄インシデントの対策について,
http://www.ipa.go.jp/security/ciadr/200105sadmindiis.html,
情報処理振興事業協会.

[3] E. G. Amoroso, “Intrusion Detection : An Introduction to Internet Surveillance, Correlation, Trace Back, Traps, and Response”, Intrusion. Net Books, 1999.

[4] 竹森敬祐, 力武健次, 田中俊昭, 清本晋作, 中尾康二, “Intrusion Trap System の実装および評価”, 情報処理学会, 2001 コンピュータセキュリティシンポジウム (CSS 2001), pp.415-420, Oct. 2001.

[5] Tripwire for Web Pages Apache Edition,
http://www.tripwire.co.jp/product/web.html,
トリップワイヤ・ジャパン.

[6] 伊原 秀明, “ TRIPWIRE for LINUX”, オライリー・ジャパン, Apr. 2001.

[7] 可部孝二, 曽我正和, 西垣正勝, 田窪昭夫, “ ホームページ改竄パトロール方式”, 情報処理学会, コンピュータセキュリティ研究会, CSEC-8-30, pp.173–178, Mar. 2000.

[8] 板垣晋, 曽我正和, 西垣正勝, 田窪昭夫, “ 強化型ホームページ改竄パトロール方式”, 情報処理学会, 2001 コンピュータセキュリティシンポジウム (CSS 2001), pp.403–408, Oct. 2001.

[9] Anonymous, “ Linux 版クラッカー迎撃完全ガイド”, インプレス, Apr. 2000.

[10] 三輪, 白井, 白濱, 又江原, 柳岡, “ 不正アクセスの手法と防御”, ソフトバンク パブリッシング, 2001 年 7 月.

[11] R. Rivest, “ The MD5 Message-Digest Algorithm”, IETF, RFC1321, April 1992.

[12] アンク, “ HTML タグ辞典 第 4 版”, 翔泳社, Jan. 2001.

第3章

Intrusion Trap Systemにおける安全で有効なログ収集のための動的誘導機能の実装

1章では，SOCにおけるセキュリティ監視の要件として，

(i) 各地のネットワークシステムの状態をリモート監視すること

(ii) 未知の攻撃情報を収集する基盤技術を確立すること

(iii) 各地のセキュリティシステムから出力されるログを統合分析すること

について述べてきた．2章では，(i) について解決しており，セキュリティ侵害を受けた後の被害を最小限に抑えるための迅速な検知を目指した事後対策の技術であった．(ii) は，侵入者の攻撃情報を積極的に収集して防御対策へと役立てる事前対策の要件である．本章では，(ii) の要件を達成するためのシステム提案ならびに実装・評価を行う．

3.1 概要

ネットワークシステムを構成する機器には，日々セキュリティホールが報告されており，これらを狙った侵入者による攻撃やインターネットワーム [1] による攻撃など，侵入に対する脅威が拡大している [2]．一般的な防御システムとして，ファイアウォールや IDS [3, 4] などがあるが，ファイアウォールの場合，通過を許可された通信により引き起こされる攻撃を防げないことや，IDS の場合，シグネチャ情報を持たない未知の攻撃やサイト独自のアプリケーションに対する攻撃を検知することができないという課題が存在する．

これら日々多様化する攻撃手法の把握は，セキュリティ確保の面で重要である．近年，侵入手法やツールさらには侵入者の意図などを分析するためのプラットホームとして，脆弱性を持つおとりのシステムを利用した Honeypot [5, 6] と呼ばれる手法が注目を集めている．本手法を適用した Honeynet Project[7] では，インターネット上の各所におとりのシステムを設置して監視を行い，一般にみられる攻撃手法の分析を進めている．しかしながら，おびき寄せることを前提とした Honeypot は，サービスを提供しているシステム自体を防御するものではないため，IP アドレスをランダムに選択して攻撃してくる侵入者やインターネットワームから本来のシステムを守ることができない．また，おとりを用いることでおびき寄せによる犯罪の誘発に関する危惧など，その適用性について議論がやまない．

これに対して，IDS でトラヒックを監視しておき，疑わしい行為が検知された IP アドレスからのアクセス先を，正規システムからおとりシステムへ強制的に誘導する Internet Trap [5, 8, 9, 10] が提案されている．これは，Honeypot で達成できる機能に加え，正規システムを防御できる点や，犯罪の予兆が検知されたアクセスを対象に誘導するため，犯罪の誘発に繋がらない点で有効である．また，おとりとして

正規システム上の重要なデータ以外のデータや各種設定情報をミラーリングしたシステムを用意することで，サイト独自のアプリケーションに対する攻撃についても収集できるプラットホームとなる．しかし，これまで提案，開発されてきた Internet Trap [8, 9, 10] の誘導は TCP コネクションの開始時点のみに行われるため，疑わしい行為が検知された TCP コネクションが継続される限り，正規システムに攻撃が加えられる恐れがある．また，侵入者の行動ログをおとりシステム上で収集するため，これを改竄，消去されかねない問題も残っている．

そこで本章では，TCP コネクションの開始時点のみならず継続中の TCP コネクションについても，IDS からの不審な挙動を知らせるトリガをきっかけに直ちにおとりへと誘導することで，正規システムを守りつつ情報収集を行える ITS について，

1. 一つのホスト上に正規領域とおとり領域を設ける手法
2. 正規ホストと独立した外部おとりホストを設ける手法

の二つの設計手法を提案してきた [11]．これら二つの ITS は，継続中の TCP コネクションを誘導するときに通信シナリオに矛盾が生じないように，正規領域（正規ホスト）とおとり領域（おとりホスト）の通信状態の同期を図る機能を設けており，機能面から継続的なサービスを提供する TCP コネクション管理を行っている．本論文では，正規ホストをそのまま利用できる 2. の ITS モデルに注目し，FTP と HTTP サービスに適用するときの誘導機能の設計および実装を行う．そして処理性能に関する測定を行い，本誘導機能を適用した場合でも本来のサービスに与える影響を小さく抑えることができ，誘導も迅速に行われることを確認する．ITS は，TCP コネクションで提供される様々なアプリケーションサービスに適用できるため，サイト上で侵入者の行動を分析するためのプラットホームとして広く利用されることが期

待される．

以降，3.2 節では既存の Internet Trap の概要とその問題点について述べ，ITS に必要とされる機能をまとめる．3.3 節で ITS の全体構成を示して，本論文で実装する諸機能の位置付けを説明する．3.4 節において中心的な役割を果たす誘導機能について説明し，3.5 節で FTP と HTTP サービスに適用する場合の ITS を構成する各モジュールの設計を行う．3.6 節で処理性能に関する評価を行い，そして最後に 3.7 節でまとめる．

3.2 既存システムの概要と問題点

3.2.1 既存の Internet Trap の概要

Internet Trap は，正規の利用者のみならず侵入者にもサービスを提供するシステムである．ただし侵入者に対しては正規システムに見せ掛けたおとりシステムからサービスが提供される．このサービス中に収集された侵入者の行動ログを分析することで，システムへの侵入手法や利用ツール，サイト上の脆弱性，侵入目的などを把握でき，ファイアウォールや IDS，その他の機器の設定を見直すことができるようになる．また，おとりシステムに接続させることで，追跡のための時間稼ぎが可能になる．

図 3.1 に，既存の Internet Trap [8, 9, 10] のシステム構成を示す．Internet Trap は，侵入検知部，侵入者リスト，アクセス制御部，正規システム，おとりシステムの 5 つのモジュールから構成される．アクセス制御部は，クライアントからアクセス要求を受け取ると，侵入検知部からの報告が記録されている侵入者リストを参照し，もし記録があればそのアクセス先をおとりシステムへと誘導する．このときの

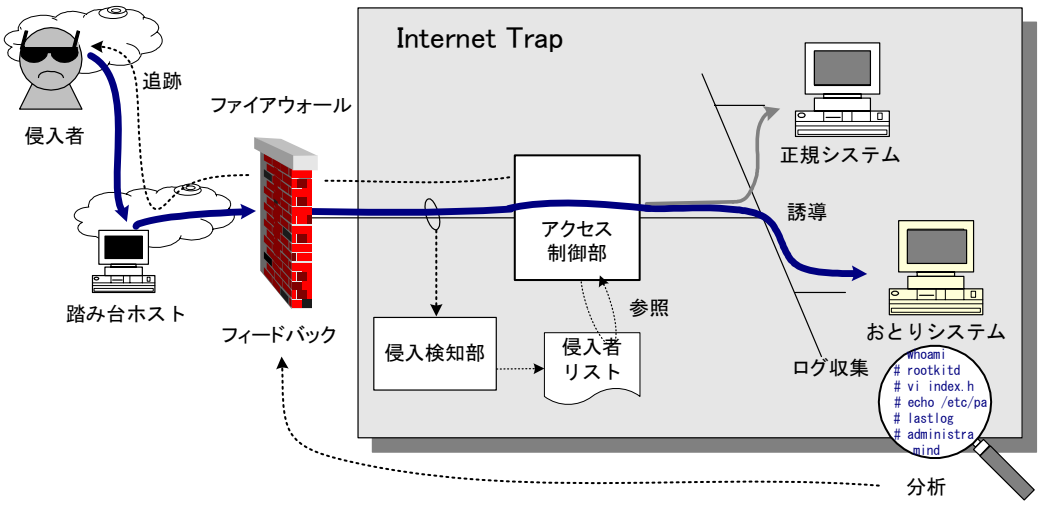

図 3.1: 既存の Internet Trap の構成

侵入者からの攻撃は，おとりシステムへ加えられるため，正規システムは守られる．

3.2.2　問題点

既存の Internet Trap は，TCP コネクションの開始時点で誘導を行っており，不正が検知された後の TCP コネクションから誘導される．もし，不正の検知された TCP コネクションが継続されれば，正規システムに接続されたままになり攻撃を受けてしまう可能性がある．既存の Internet Trap では，クライアントからの TCP コネクションはアクセス制御部を経由して，正規システムで終端されている．このため，通信中の TCP コネクションを継続したまま正規システムからおとりシステムへと誘導することは，TCP のシーケンス処理上の不整合が発生するため，不可能である．

既存の Internet Trap で継続中の TCP コネクションからの攻撃を防ぐには，TCP

コネクションを一旦切断して再接続を促すことになり，このとき侵入者に逃げられてしまうことも考えられる．たとえ再接続に来たとしても，誘導までに行われた正規システムへの行為を，おとりシステムへ反映させる機能は無く，誘導時の通信シナリオに矛盾が生じる．

ここで通信シナリオの矛盾とは，TCP コネクションにより提供されるサービスを例に説明すると，正規システムへの TCP コネクション上で行ったログイン処理やディレクトリ間の移動，ファイルの生成，変更，削除などの作業の結果が，おとりシステムに反映されていないことである．このように既存の Internet Trap では，TCP コネクションを切断したり通信シナリオに矛盾が生じることで，侵入者に誘導機構の存在に気付かれてしまう可能性が高く，情報収集を円滑に行うことができない．

その他，侵入者のログはおとりシステム上で収集されるため，侵入者がおとりシステムの管理者権限の取得に成功してログの改竄や削除を行った場合，収集した情報が信頼できないものになる．

3.2.3 必要とされる機能

正規システムを保護しながら情報収集を円滑に行うためには，不正が検知されると TCP コネクションの途中であっても TCP コネクションを継続したままおとりシステムへと誘導する機能が必要である．このとき，誘導前後で通信シナリオの継続性を保つ機能も必要になる．また，一連の誘導処理は迅速に行われなければならない．その他，行動ログはおとりシステム以外でも収集すべきである．

3.3 提案システムの概要

図 3.2 に，TCP コネクションの動的誘導機能を持った ITS のシステム構成を示す．

ITS の目的は，

目的 1. 正規システムの安全確保

目的 2. 収集した行動ログの活用

であり，本論文では，目的 1. の安全確保と，目的 2. の前段階となる有効なログ収集を達成するために，侵入者の TCP コネクションを開始時点のみならず継続中にも，おとりシステムへと誘導する手法として，図 3.2 の処理 1. から処理 4. までの機能を提案・実装する．

ここで有効なログを収集するためには，誘導前後において侵入者に継続的なサービスを提供することが重要である．継続的なサービスは，

- 攻撃中のログを収集できること
- 多くの侵入者をおとりシステムに繋ぎ止めること

などが期待され，注意深い侵入者に誘導後のある時点で逃げられてしまった場合でも，それまでの行動ログを収集できるようになる．

ITS は，既存の Internet Trap が持つ各処理部と正規・おとりシステムに加え，おとりシステム上のログへの攻撃対策としてアクセス制御部に侵入者の通信記録を管理するログ DB を設置する．以後の章において，誘導機能と各処理部の詳細を説明する．

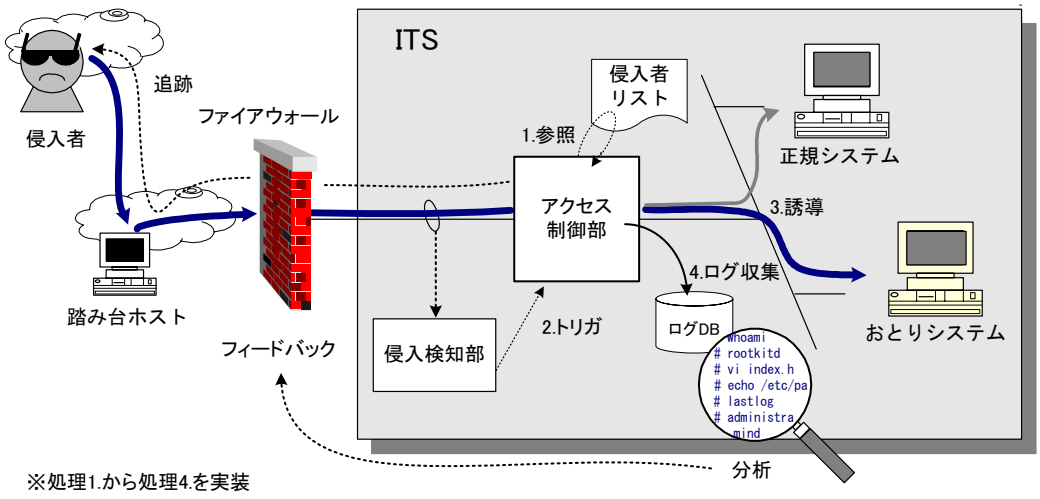

図 3.2: 提案する ITS の構成

3.4　誘導手法

3.4.1　静的誘導と動的誘導

本論文で提案する ITS は，侵入者の TCP コネクションを正規システムからおとりシステムへと誘導する手法として，静的誘導と動的誘導の 2 つの機能を持つ．静的誘導は，TCP コネクションの開始時点で誘導する手法であり，既存の Internet Trap が持つ唯一の誘導手法である．動的誘導は，侵入検知部にて不審な挙動が検知されると直ちに継続中の TCP コネクションを誘導する手法である．このときの誘導には，通信シナリオの継続性とその処理の迅速性が要求される．

図 3.3 に，TCP コネクションにより提供されるサービスを例に誘導の様子を示す．

過去に不審な挙動が検知されていないユーザが接続しにきた場合，正規システムへログインする．その後，正常利用していた途中で管理者権限の取得攻撃などを試みた場合，侵入検知部がそれを検知してトリガを発行し，おとりシステムへと強制

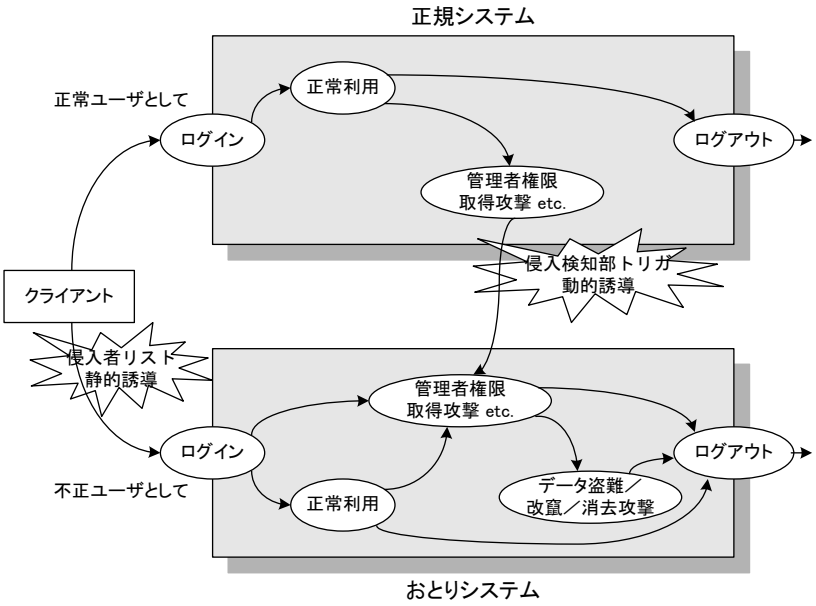

図 3.3: 静的誘導と動的誘導

的に誘導する．誘導時には，正規システムとおとりシステム間の通信シナリオの同期を図っておくことで，ログイン処理や作業ディレクトリの移動などの処理は必要ない．このため，多くの侵入者は誘導されたことに気付くことなく，データの盗難，改竄，消去など，おとりシステム上で様々な攻撃へと発展していく．

過去に不審な挙動のみられたクライアントの場合は，ログイン時点からおとりシステムへ誘導する．

初めの一つ目のパケットから未知の攻撃を仕掛けてくる侵入者については，提案する ITS は正規システムを守ることができない．しかし，手動による侵入者やインターネットワームの多くは，事前にターゲットホストを探すことが多く，IP スイープや Port スキャンなどの予兆が検知されるため，静的誘導が可能になる．

UDP プロトコルについては，TCP におけるコネクションの概念が無いため，静的誘導のみ適用される．

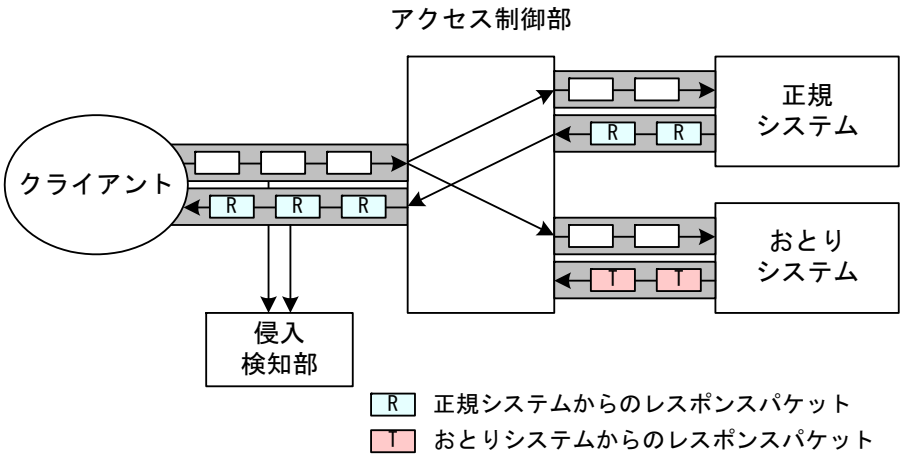

図 3.4: トリガ受信前の TCP コネクション状態 -非誘導モード-

3.4.2　動的誘導の詳細

誘導はアクセス制御部が主体となって行う．クライアントとアクセス制御部，アクセス制御部と正規システム，アクセス制御部とおとりシステムは，3 つの別々の TCP コネクションが張られており，アクセス制御部はアプリケーションレベルのデータ中継のみを行っている．正規システムからおとりシステムへと誘導するときには，クライアントとアクセス制御部間の TCP コネクションは継続されたままであり，アクセス制御部と正規システム間の TCP シーケンス番号と，アクセス制御部とおとりシステム間の TCP シーケンス番号の違いを気にすることなく誘導を行うことができる．

図 3.4 に，動的誘導前の TCP コネクションの様子を示す．アクセス制御部は，クライアントからのリクエストパケットを正規システムとおとりシステムの両方へ同時に中継する．ただしクライアントへの応答は，両方のサーバからレスポンスパケットを受信するタイミングで，正規サーバから受信したものをクライアントへ返信す

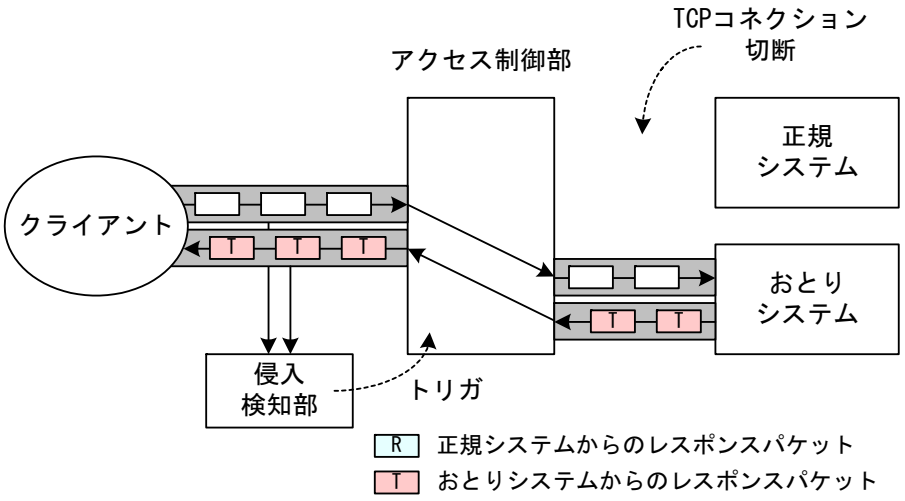

図 3.5: トリガ受信後の TCP コネクション状態 -誘導モード-

る．これにより，正規システムとおとりシステム間の通信シナリオの同期を図っている．

図 3.5 に，侵入検知部において不審な挙動を検知してトリガが発行されたときの TCP コネクションの様子を示す．アクセス制御部が危険を知らせるトリガを受け取ると，直ちに正規システムとの TCP コネクションを切断する．以後のクライアントとの通信はおとりシステムとの間で行われる．このように，アクセス制御部とおとりシステム間の TCP コネクションをあらかじめ起動して正規システムと同じ通信を行っておくことで，誘導時の通信シナリオの継続性を保つことができると共に，誘導を迅速に行うことができる．

3.4.3　アクセス制御部の処理フロー

侵入者リストによる静的誘導と侵入検知部からのトリガによる動的誘導を行うアクセス制御部の処理フローを図 3.6 に示す．アクセス制御部の処理は，クライアン

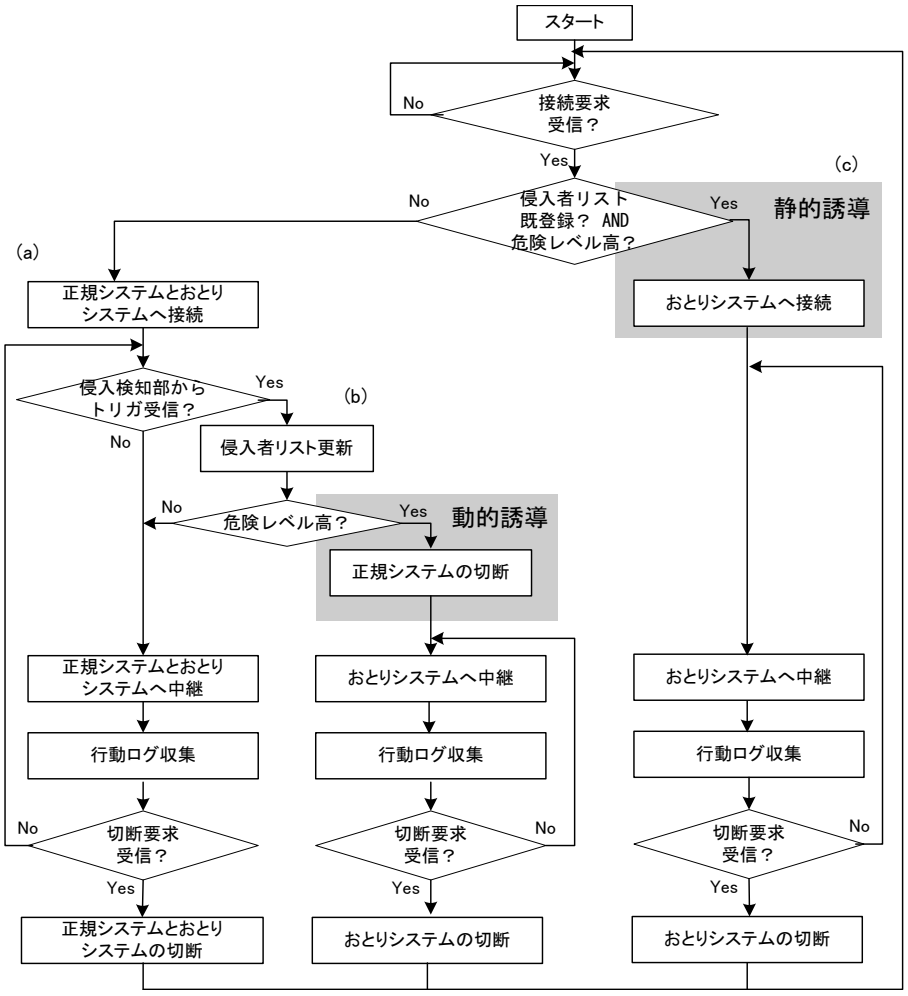

図 3.6: アクセス制御部の処理フロー

トの性質により大きく 3 つに分けられる．

(a) **クライアントが常に正常なアクセスを行う場合** アクセス制御部は，クライアントから接続要求を受信すると，まず侵入者リストを確認する．ここで過去に不審な行為が無いかもしくはあらかじめ設定されている危険レベルよりも低いことが判明すると，TCP コネクションを正規システムとおとりシステムの両方へ確立し，行動ログ

を収集する．

(b) **クライアントが通信中に不正を行った場合** (a) の状態で，通信中に侵入検知部からトリガを受け取ると，トリガ情報を基に侵入者リストを更新する．そして，トリガの危険レベルを確認して，あらかじめ設定されている危険レベルよりも高い場合には，正規システムとの TCP コネクションを切断する．以後の TCP コネクションはおとりシステムとのみ確立され，行動ログを収集する．

(c) **あらかじめクライアントが侵入者リストに登録されている場合** もし通信の開始時点で侵入者リストに IP アドレスが記録されていて，かつあらかじめ設定されている危険レベルよりも高い場合には，TCP コネクションをおとりシステムとのみ確立し，行動ログを収集する．

アクセス制御部は，上記 3 つのケースの全てにおいて TCP コネクションの切断要求を受け取ると，クライアントとそれぞれのシステム間の TCP コネクションを終了する．

3.4.4 通信シナリオの継続性における課題

ITS の誘導機能は，正規システムの安全を確保しつつ，できる限り多くの侵入者をできる限り長い時間，おとりシステムに繋ぎ止めることで，有効な行動ログを収集することを目的としている．そのためにも，正規システムとおとりシステムの双方が提供するサービスに違いが出ないように，誘導前後において正規システムとおとりシステムの状態を可能な限り一致させている．このことは，悪意の無いクライアントが誤検知によりおとりサーバに誘導された場合でも，サービスを受けることを可能にしている．

しかし，両システムの状態を完全に一致させることは不可能である．ここでは，本論文でのシステム状態の整合性は，誘導前後において再ログインやディレクトリの移動が不要なことなど，継続的なサービス提供を可能にするレベルの整合性までを目標とし，プロセス制御やファイル管理などの非決定的要素により，排除しきれない状態の不整合については今後の課題として以下に列挙しておく．

プロセス・メモリの不整合

初期状態において，おとりシステムの OS やファイルを正規システムと同じものを用意していた場合でも，プロセス ID，メモリ使用状況において差異が生じる．ここで，本誘導機能を telnet サービスに適用した場合，注意深い侵入者が，正規システムからおとりシステムへと誘導される前後において ps コマンド等を用いることで，システム状態の不整合に気付く可能性がある．

ファイルの不整合

一時的に生成されるファイルの名前には乱数的な要素があるため，正規システムとおとりシステムでこれらのファイル名の同期を図ることはできず，ファイル状態の不整合に気付く可能性がある．

IP アドレスの不整合

正規システムとおとりシステムには，アクセス制御部と通信を行うために別々の IP アドレスが付与されている．ここで，本誘導機能を telnet サービスに適用した場合，注意深い侵入者が，正規システムからおとりシステムへと誘導される前後において ifconfig コマンドを用いることで，IP アドレスの不整合に気付く可能性がある．

アクセス元 IP アドレスの違いによる改竄ファイルの不整合

ある IP アドレスからアクセスしてくる侵入者が，おとりシステムへと誘導されておとりシステム内のファイルを改竄した後に，確認のために別の IP アドレスからアクセスしてきた場合，ファイルの不整合に気付くことになる．

3.5 構成機器の設計

動的誘導は，各種 TCP コネクションにより提供されるサービスに適用できる誘導機能である．ここでは ISP が提供している代表的なサービスとして，複数のユーザが Web サーバ上のコンテンツを FTP でメインテナンスするシステムを例として，各モジュールの具体的な設計について述べる．

3.5.1 侵入検知部

侵入検知部は図 3.7 に示すように，IDS，IDS が出力するアラートログ，侵入者リスト，トリガモジュールから構成される．

IDS として Snort[3] を利用した場合，Snort のアラートログには侵入者の IP アドレス，検知日時，攻撃名，危険レベルなどが記録される．表 3.1 に侵入者リストを示す．これは，トリガ発行の制御に利用するものであり，検知された IP アドレス，その IP アドレスが初めて検知されたときの危険レベルと日時，その IP アドレスにおける過去最高危険レベルと検知日時を記録する．危険レベルについては，利用する IDS により異なるため，あらかじめ IDS が出力する危険レベルと侵入者リストへ記録する危険レベルの対応表を作っておく．

侵入検知部は，クライアントとアクセス制御部の間でトラヒックを監視しており，

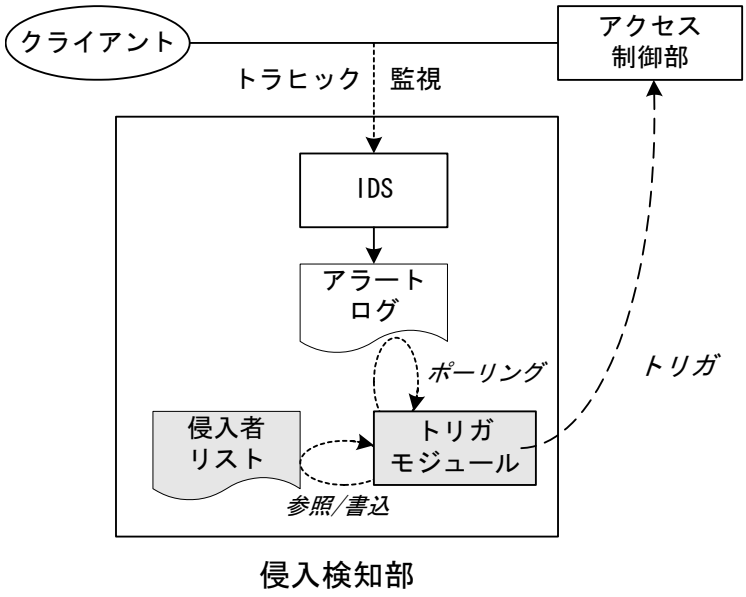

図 3.7: 侵入検知部の設計

IDS が攻撃を検知すると，アラートログへ各種情報を出力する．トリガモジュールは，アラートログをポーリング監視しており，新たなログが出力された場合には，その IP アドレスが以前に記録されていないか，もしくは，危険レベルが以前に検知されたレベルよりも高いかについて，侵入者リストに問い合わせる．初めての IP アドレス，もしくはより高い危険レベルの IP アドレスの場合には，アクセス制御システムへトリガを発行すると共に，侵入者リストを更新する．トリガの内容は，IP アドレス，危険レベル，検知日時であり，アクセス制御部の侵入者リストを更新するための差分データとなる．

表 3.1: 侵入者リスト

侵入者	最大危険		初回危険	
IPアドレス	レベル	検知日時	レベル	検知日時
192.168.0.10	中	2002.11.17−10:23:43	中	2002.11.17−10:23:43
192.168.2.23	高	2002.11.20−01:34:52	中	2002.11.18−10:51:38
192.168.10.20	低	2002.11.21−04:45:01	低	2002.11.21−04:45:01
192.168.32.8	中	2002.11.21−13:45:01	低	2002.11.21−12:57:29
…	…	…	…	…

3.5.2 アクセス制御部

アクセス制御部は，クライアントとサーバ間に設置する．図 3.8 に，FTP ならびに HTTP サービスを中継するアクセス制御部の構成を示す．アクセス制御部は，中継モジュール制御デーモン，FTP ならびに HTTP 中継モジュール，侵入検知部のものと同じフォーマットを持つ侵入者リスト，ログ収集モジュールから構成される．

中継モジュール制御デーモンは，クライアントからの接続要求を一括して受け付ける．ここで，クライアントからの接続要求が FTP の場合には FTP 中継モジュールに，HTTP の場合には HTTP 中継モジュールに，それぞれ正常もしくは不正クライアントとしての中継処理の指示を行う．FTP もしくは HTTP 中継モジュールは，各プロトコルを理解して制御を行うプロキシとして実現する．ここで，クライアントからの接続要求を迅速に処理するために，各中継モジュールは中継モジュール制御デーモンのスレッドとして，あらかじめ起動しておく．例えば FTP の場合，クライアントから同時に接続される数は少ないと見積もって 10 程度を，HTTP の場合，ある程度接続が集中することを見積もって 100 程度を起動しておく．

中継モジュール制御デーモンから各中継モジュールへの処理の割当てやトリガ指示は，共有メモリを通じて行う．この通信用の共有メモリとして，FTP 中継モジュール制御用と HTTP 中継モジュール制御用をそれぞれ用意して，各中継モジュールで

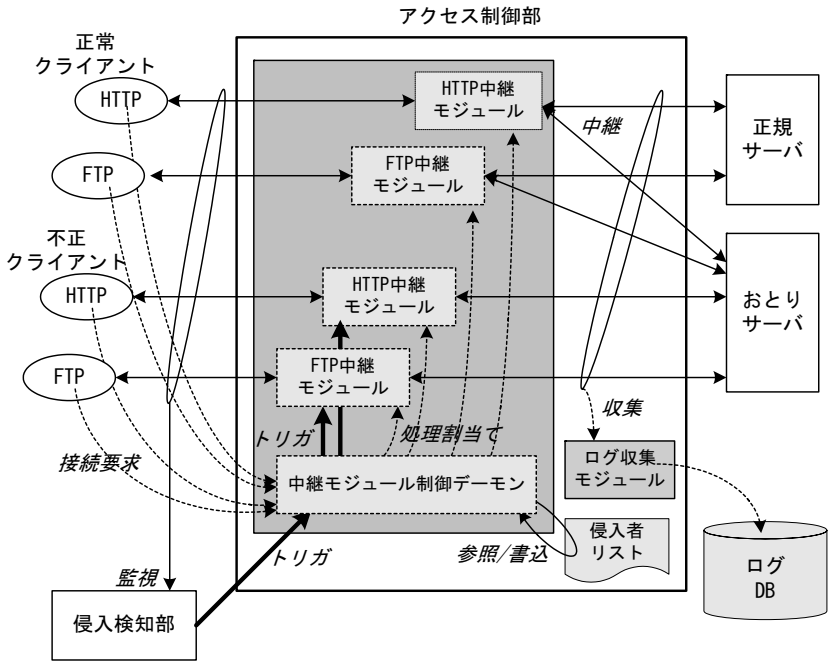

図 3.8: アクセス制御部の設計

あるスレッドを制御する．

アクセス制御部では，おとりサーバ上のログが攻撃された場合に備えて，各クライアントの通信ログを外部のログ DB へと記録する．ここで対象とするログは，不正なクライアントのみならず，正常なクライアントと判定されていたものについても収集する．これは，正常なクライアントと判定されていた場合でも，TCP コネクションの途中で誘導された時の誘導前後の挙動を分析するためである．

3.5.3　正規サーバとおとりサーバ

正規サーバはそのまま利用できる．おとりサーバは，正規サーバと同じデータ，サービス，セキュリティ対策を施す．ただし，個人情報を扱うシステムに適用する場

合，正規システムで扱う全てのファイルをおとりシステムにミラーリングすることは問題である．システムの初期設定においては，おとりシステムにはダミーのデータを用意しておく．継続的な運用においては，正規ユーザの個人情報がおとりシステムに送られる際にダミーのデータへと変換する必要がある．ここで，3.4.4 節のおとりシステム上のファイルが改竄された場合には，正規システムにも同様な問題があるため，直ちに改竄されたときの行動ログの分析を行い，両システムのセキュリティ対策を図る．そして，次に誘導される侵入者にファイルの不整合を気付かれる前に，改竄前の状態に戻す必要がある．

3.5.4 周辺機器

ITS が外部システムへ攻撃するような踏み台に利用されてはならない．このため，外部への接続要求を拒否するように，ファイアウォールなどの機器でフィルタリングを行うようにする．

3.6 性能評価

ここでは ITS のシステムへの適用性に関して，処理性能面から評価を行う．具体的には，ITS を適用したときの本来のサービスに与える影響と，不審な挙動が検知されたときの動的誘導速度について評価，考察を行う．

3.6.1 評価環境

評価に使用した機器のハードウェアスペックを表 3.2 に示す．また，測定項目について，図 3.9 から図 3.11 に示す．

表 3.2: ハードウェアスペック

モジュール	CPU	メモリ
正常&不正クライアント	PentiumIII 1GHz ×1	256MByte
侵入検知部	PentiumIII 1GHz ×1	256MByte
アクセス制御部	PentiumIII 1.13GHz×2	256MByte
正規&おとりサーバ	PentiumIII 1GHz ×1	256MByte

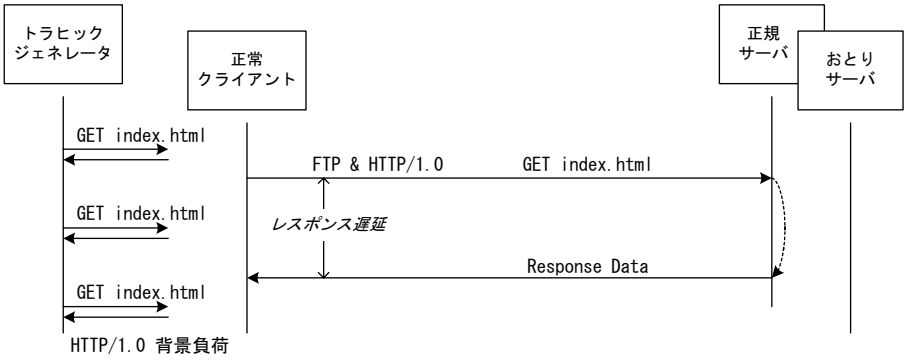

図 3.9: 直結時のサーバレスポンス遅延

各機器は，100Base-TX の LAN で接続されている．ハードウェアのトラフィックジェネレータを使用して，サーバに対して一定間隔で HTTP/1.0-GET の背景負荷を与えた．このとき取得する index.html ファイルのサイズは，2.9KByte である．応答時間の測定は，ネットワーク上のパケット時間をモニタすることで行っており，5 回の測定の平均値を求めた．

性能測定は，次の 3 つの遅延について行った．

- 直結時のサーバレスポンス遅延（図 3.9）

  クライアントとサーバを直接接続したときの，FTP ならびに HTTP サービスのレスポンス遅延を測定する．レスポンス遅延は，クライアントが

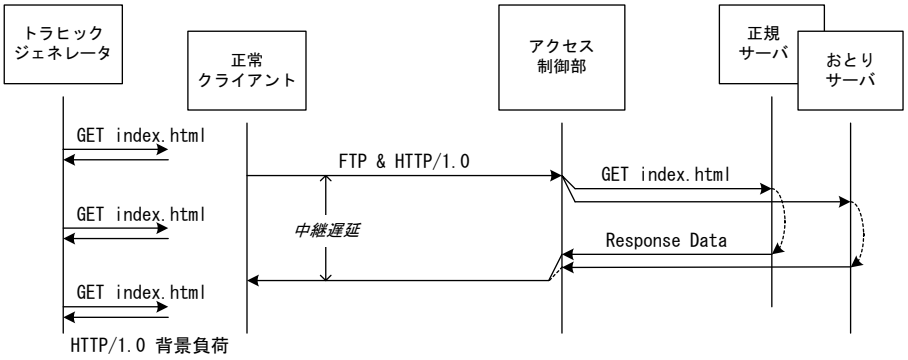

図 3.10: ITS による中継時のサーバレスポンス遅延

ファイルの取得要求パケットを送信してから，クライアント側へレスポンスパケットが返信されるまでの時間を測定している．

- ITS による中継時のサーバレスポンス遅延（図 3.10）

  ITS を設置したときの，FTP ならびに HTTP サービスの中継遅延を測定する．中継遅延は，クライアントがファイルの取得要求パケットを送信してから，クライアント側へレスポンスパケットが返信されるまでの時間を測定している．

- ITS による動的誘導遅延（図 3.11）

  ITS による，FTP ならびに HTTP サービスにおける継続中の TCP コネクションの動的誘導遅延を測定する．動的誘導遅延は，クライアントが不審パケットを送信してから侵入検知部がトリガを発行するまでのトリガ遅延と，アクセス制御部がトリガを受信してから正規サーバへ TCP コネクションの切断要求を送信するまでの切断遅延の合計時間を測定することで求めている．

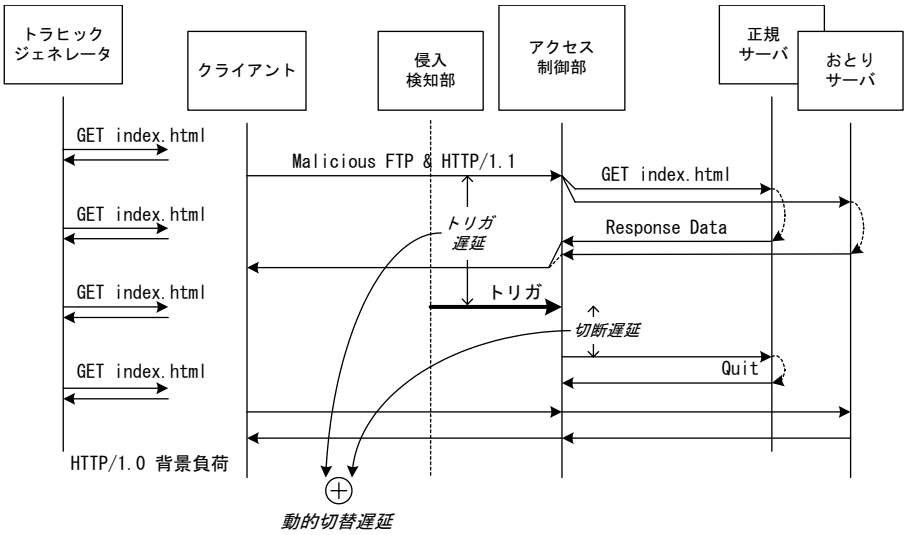

図 3.11: ITS による動的誘導遅延

表 3.3: ITS 適用時と未適用時の FTP/HTTP 遅延

クライアント側		HTTP/1.0 背景負荷	
プロトコル	遅延種別	none	50 GETs/sec
FTP	直結 (図9)	2.3 msec	2.3 msec
	中継 (図10)	3.5 msec	3.5 msec
HTTP/1.0	直結 (図9)	1.7 msec	1.5 msec
	中継 (図10)	81.2 msec	86.4 msec

3.6.2　ITS 適用時のサービスに与える影響に関する評価結果

表 3.3 に，直結時の FTP ならびに HTTP サービスと ITS 適用時の FTP ならびに HTTP サービスのレスポンス遅延を示す．測定は，背景負荷が無いときと HTTP/1.0 による 50 GETs/sec の状況で実施した．

FTP サービスについては，背景負荷の無いときと HTTP/1.0 の背景負荷を与えたときの両状況において，ITS 適用時のレスポンス遅延は直結時のレスポンス遅延と

同じ結果となっており，FTP 本来のサービスへ与える影響は測定できない程，小さなものであることが判明した．これは，背景負荷が HTTP/1.0 であるために，3.5.2 節で説明した中継モジュール制御デーモンと FTP 中継モジュール間の通信用共有メモリにおいて，FTP 中継モジュール間のメモリ参照の競合が発生しないためである．また，TCP コネクションが確立された後のコマンド中継は，TCP コネクション開始時点で行われる侵入者リストの確認処理が省かれるために処理時間が小さくなっている．

HTTP/1.0 サービスについては，負荷の無いときと与えたときの両状況において，約 80msec 程の遅延が発生している．これは，HTTP/1.0 プロトコルの TCP コネクション制御が原因となっている．HTTP/1.0 では，ファイル取得要求ごとに新たに TCP コネクションを確立する．これにより，アクセス制御部は TCP コネクションが確立される毎に，侵入者リストを参照して要求元 IP アドレスが正常クライアントからのものなのか不正クライアントからのものなのかを判断して，中継モジュールにその旨を割当てるというオーバヘッドが発生している．また，背景負荷が HTTP/1.0 であるために，複数の HTTP 中継モジュールが動作中となり，中継モジュール制御デーモンと HTTP 中継モジュール間の通信用共有メモリにおいて競合が発生してしまうことも原因となる．しかしながら，ITS 適用時の HTTP/1.0 の中継遅延は 100msec より小さく，ネットワークを経由して利用される HTTP サービスへの影響は小さいと言える．

表 3.3 の FTP サービスの結果を基に HTTP/1.1 サービスにおける遅延について推定してみると，HTTP/1.1 では TCP コネクションを継続したままファイル取得要求を送信できるため，FTP と同じくファイル取得要求時に侵入者リストの確認処理が省かれる．よって，中継処理は FTP サービスのときと同じくらい高速に行われることになる．逆に，HTTP/1.0 サービスの結果を基に FTP サービスおけるログイン処

表 3.4: ITS における FTP/HTTP 動的誘導遅延

クライアント側		HTTP/1.0	背景負荷
プロトコル	遅延種別	none	50 GETs/sec
FTP	トリガ	154.5 msec	186.3 msec
	切断	3.1 msec	5.3 msec
（図11）	動的誘導	157.6 msec	192.6 msec
HTTP/1.1	トリガ	102.3 msec	135.3 msec
	切断	6.0 msec	11.6 msec
（図11）	動的誘導	108.3 msec	146.9 msec

理遅延についても推定してみると，ログイン処理の際には新たに TCP コネクションが確立されるため HTTP/1.0 プロトコルの場合と同様なオーバヘッドによる遅延が生じることになる．

3.6.3 動的誘導速度に関する評価結果

表 3.4 に，ITS 適用時の FTP ならびに HTTP サービスにおいて，正常クライアントが不審なパケットを送信してからアクセス制御部へトリガが送信されるまでの遅延，アクセス制御部がトリガを受信してから正規サーバへ切断要求を送信するまでの遅延，これら二つの遅延を合計した動的誘導遅延を示す．測定は，背景負荷が無いときと HTTP/1.0 による 50 GETs/sec の状況で実施した．

FTP サービスについては，負荷の無いときと与えたときの両条件において，侵入検知部が不審なパケットを検知してからトリガを送信するまでのトリガ遅延が 150 から 200msec 程度となっている．これは，IDS が不審パケットを検知してからアラートファイルに書き出すまでの遅延が大きなためである．アクセス制御部がトリガを受信してから正規サーバへの TCP コネクションを切断するまでの切断遅延は数 msec

となっていた．これらトリガ遅延と切断遅延の合計となる動的誘導遅延は，200msec 以下であり，手動でコマンドを送信する侵入者に対しては十分高速に誘導されていることがわかる．

HTTP/1.1 サービスについても，負荷の無いときと与えたときの両条件において，トリガ遅延が 100 から 140msec 程度に，切断遅延が 10msec 程度に，その合計の動的遅延は 100 から 150msec になっていた．HTTP/1.1 サービスについても，手動でコマンドを送信する侵入者に対しては十分高速に誘導されていることがわかる．

3.7 総括

本章では，不審な挙動の TCP コネクションを強制的に正規システムからおとりシステムへ誘導することで，正規システムを保護しながら侵入者の行動ログを収集する ITS の誘導機能を提案した．誘導は，TCP コネクションの開始時点のみでなく継続中の TCP コネクションについても可能であり，また正規システムとおとりシステムの通信状態の同期を図っておくことで誘導時の通信のシナリオを継続できるため，侵入者に気付かれることのない手法となっている．提案した手法について，FTP ならびに HTTP サービスに適用する場合の各構成機器の設計を行った．これに従い実装を行い，コマンドの中継遅延ならびに TCP コネクションの動的誘導遅延について測定した．その結果，ITS を適用しても本来のサービスに与える影響を小さく抑えることができ，かつ誘導についても手動で攻撃してくる侵入者に対しては十分迅速に行えることを確認した．

3.5.3 節で，継続的な運用において，おとりシステムに正規ユーザからの重要な情報が蓄積されてしまうことを防ぐためのダミーデータへの変換機能が必要であることを述べてきた．このダミーデータへの変換機能については，どのデータの，どこ

を，いかに変換するかといった課題が残る．今後は，正規ユーザのデータを守るための機能について検討を進める．

第3章参考文献

[1] 新種ウィルス「W32/Nimda」に関する情報，情報処理振興事業協会，
http://www.ipa.go.jp/security/topics/newvirus/nimda.html

[2] コンピュータ緊急対応センタ（JPCERT/CC)，
http://www.jpcert.or.jp/

[3] Snort，
http://www.snort.org/

[4] P. E. Proctor, "Practical Intrusion Detection Handbook"， Prentice Hall ，NJ, 2001.

[5] E.Amoroso: "Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps and response", Intrusion.Net Books，Sparta, NJ, 1999.

[6] 武田圭史，磯崎宏: ネットワーク侵入検知，ソフトバンクパブリッシング，2000.

[7] Honeynet Project，
http://project.honeynet.org/project.html

[8] 宮川明子，稲田徹，後沢忍: 不正侵入者を外部ネットワークに設置したおとりサーバへ誘導するセキュリティシステムの検討，情報処理学会, コンピュータセキュリティ研究会，CSEC，pp.225-230，Jul. 2001.

[9] Decoy Server Solution, http://www.atsweb.it/Images/Documenti/TOP_DS_Decoy%20Server%20Solution.pdf, Top Layer Networks Product.

[10] Man Trap, http://enterprisesecurity.symantec.com/products/products.cfm?ProductID=157, Symantec Product.

[11] 竹森敬祐，田中俊昭，中尾康二: 不正侵入者に探知されない通信セッションのおとりサーバへの引継ぎ方式の検討，情報処理学会, 第 61 回全国大会，4F-3, Oct. 2000.

[12] 竹森敬祐，田中俊昭，清本晋作，中尾康二: 不正侵入者に探知されることなくおとりのデータ領域へと誘導するおとりシステムの実装評価，情報処理学会，コンピュータセキュリティ研究会，CSEC，pp.79-84，Feb. 2001.

[13] 竹森敬祐，力武健次，清本晋作，田中俊昭，中尾康二: Intrusion Trap System の設計および実装，情報処理学会, 第 63 回全国大会，2G-1，Sep. 2001.

[14] 竹森敬祐，力武健次，田中俊昭，清本晋作，中尾康二: Intrusion Trap System の実装および評価，情報処理学会，コンピュータセキュリティシンポジウム 2001 (CSS 2001)，pp.415-420，Oct. 2001.

第4章

Security Operation Centerのための IDSログ分析支援システム

1章では，SOCにおけるセキュリティ監視の要件として，

(i) 各地のネットワークシステムの状態をリモート監視すること

(ii) 未知の攻撃情報を収集する基盤技術を確立すること

(iii) 各地のセキュリティシステムから出力されるログを統合分析すること

について述べてきた．2章では，(i) について解決しており，セキュリティ侵害を受けた後の被害を最小限に抑えるための迅速な検知を目指した事後対策の技術であった．3章では，(ii) について解決しており，侵入者の攻撃情報を積極的に収集して防御に役立てる事前対策の基盤技術であった．(iii) は，各地のネットワークから出力されるセキュリティログを統合管理・分析することで，新たな攻撃の兆候を把握すること，セキュリティ侵害を迅速に発見することを目的とした，事前・事後対策の要件である．本章では，(iii) の要件を達成するためのシステム提案ならびに実装・評価を行う．

4.1 概要

近年，各地のネットワークに IDS を設置して，サイバーテロを監視する SOC の設置が進められている [1]．広範囲を監視する SOC は，新たな攻撃予兆の発見やネットワーク間の攻撃比較などを行える可能性を持っており，期待が高まっている．しかし，複数の IDS を纏めて運用するには解決すべき 4 つの大きな問題がある．

1 つ目は，設置されている IDS の種別ごとにログフォーマットや運用手順が異なることで，簡単に統合運用できない問題である．これにより，広域や個々のネットワークを同じ基準で監視することができない．2 つ目は，既存の IDS に付随する分析支援ツールには，攻撃傾向の把握に適している Port や Country に関する分析機能が無いことによる，多角的な分析を行えない問題である．3 つ目は，管理者に必要な情報は全て提供するという思想で，たとえ誤検知であっても疑わしいイベントは全て検知する False Positive に設計されがちな監視ポリシによって多量のログが出力されてしまい，微かな痕跡を見逃してしまう問題である．この冗長なログを削減する試みとして，運用パラメータの最適化手法があるが [2, 3, 4]，予備作業に掛かるコストや最適化パラメータを他の IDS に転用できない等の問題がある．視覚的に異常を強調することで冗長なログを目立たなくする試みもあるが [5, 6]，運用者の経験や主観に依存する痕跡検出作業への信頼性に疑問が残る．4 つ目は，出力されるログの特徴が，監視対象のネットワーク構成の変化や新たな攻撃の出現によって日々変動していることで，IDS に付随する既存の頻度分析機能だけでは，異常なイベントを的確に検出できない問題である．

そこで本章では，様々な IDS から出力されるログを統合管理して，時間軸上における異常なイベントを客観的な数値として出力する IDS ログ分析支援システムを提案する．分析対象のパラメータとして，Attack Signature，Source/Destination Port，

Source/Destination IP，Source/Destination Country とし，各パラメータの過去の長期プロファイルを基準データとして最近の短期プロファイルの変化の程度を異常率として評価する．各地で運用されている IDS からのログを統合したデータを用いて評価を行い，従来からの頻度分析結果の中から冗長なイベントを特定できること，発見が困難であった微かな痕跡を検出できることを確認する．本システムは，複数の IDS からのログを統合分析する機能と個別分析する機能を設けており，SOC における広域監視と個々のネットワーク監視において，分析作業の信頼性の向上と効率化に寄与する．

以下 4.2 節において，現状の IDS を用いた SOC 監視における問題点を整理して，要件を述べる．4.3 節で IDS ログ分析支援システムを提案し，4.4 節で分析アルゴリズムの適用手法を説明する．4.5 節で実データを用いた評価を通じて本システムの有効性について考察を行い，最後に 4.6 節でまとめる．

4.2 IDS を用いた SOC 監視の問題と要件

4.2.1 問題点

ログフォーマットと分析基準の不一致

SOC では，インターネットの広域に及ぶ攻撃傾向の把握や，ネットワーク単位の攻撃被害の把握のために，各地の IDS から送られてくるログを監視している．ここで，監視対象のネットワークには様々な IDS が導入されており，そのログフォーマットや分析手順が異なる問題がある．既存の IDS 管理システムには，各種 IDS を統合管理する機能はなく [1]，勿論，異なるログ間を跨って全てのネットワークの状況を一括把握するための統合分析機能や，個々のネットワークを同じ基準で評価するた

めの個別分析機能もない.

分析パラメータの不足

既存の IDS には，Attack Signature や Source/ Destination IP に注目した頻度分析機能はあるが，攻撃の傾向把握 [7] に役立つ Source/Destination Port に関する分析機能が無い．また，Distributed Denial of Service(DDoS) に代表される攻撃では，ある特定の地域から集中的に攻撃を受ける場合や，ある特定の地域へ集中的に向かう場合があり，このような攻撃の傾向把握に役立つ Source/Destination Country に関する分析機能が無い.

冗長なログ

一般に，IDS から出力されるログは，誤検知，多重検知，対策済み検知など，冗長なログが多量に出力される [1].

ここで誤検知とは，かすかな痕跡を見逃すことを防ぐために False Positive に設計された監視ポリシにより，正常なトラヒックを誤って検知してしまう問題である．多重検知とは，一つの攻撃の中に複数の特徴を持つような攻撃に対して，一致する Attack Signature 毎にアラームを出力してしまう問題である．対策済み検知とは，パッチが適用されたシステムに対する無効な攻撃を検知してしまう問題である．このようなログは，同じ攻撃を繰り返し受けるたびに出力されるため，さらに冗長なものとなる.

多量のログは，傾向把握には適しているが，出現頻度の低いログを見落としがちになり，新たな攻撃の兆候を的確に検出することを困難にする．実際に筆者らは，おとり [8] を用いて侵入手法を収集する実験を行っていた際に，シェルの奪取に成功

した痕跡として，直接関係の無い Port へのアクセスが 1 件だけ記録されていた経験がある．頻度分析によるログ監視では，見落としてしまう微かなイベントであった．

冗長なログをフィルタリングする手法として，ニューラルネットワークで誤検知ログを削減する研究 [2]，監視する必要のない Attack Signature を削除するポリシーチューニング [3]，セキュリティ監査の結果を用いて対策の施されたシステムに対する攻撃ログを排除する対策済みフィルタリング [4] などが試みられている．しかしこれらの手法では，予備作業に掛かるコスト，微かな痕跡を誤って削除してしまう懸念，最適化したパラメータを他の IDS に転用するときの課題等において問題がある．

ログ出力特性のネットワーク依存

一般に，監視するネットワークの構成や Firewall の設定によって，ログの出力特性は異なってくる．多数のネットワークを監視する SOC において，全てのネットワークの構成を把握しておくことや日々更新される設定情報を収集することは困難であり，各ネットワークの状況に応じた的確な分析を行うことができない．また，監視しているネットワーク内のホストが頻繁に利用される時間帯とあまり利用されない時間帯，曜日についても同じく利用頻度が異なり，結果としてログの出力特性も変動する．既存の IDS に付随する頻度分析機能に，こうしたログの出力特性を考慮した機能はなく，運用者は頻度値の大きなログに注目しがちになる．既存の IDS では，4.2.1 節で説明した出力されやすいログと新たな兆候を示すログが，同じ画面上に頻度順で表示されるため，経験や主観による判断で異常を検出する現在の運用手順では，その結果に対する信頼性が運用者の技量に依存してしまう．

SOC のなりすまし攻撃

悪意の侵入者は，SOC に侵入するか外部の DNS サーバを騙して利用者を偽の SOC に誘導する等の，SOC のなりすまし攻撃が考えられる．この場合，本来信頼すべき SOC の情報が改竄されることで，利用者に大きな被害を与えてしまう．

4.2.2 要件

以上に述べてきた問題点から，SOC における IDS ログ分析のための要件をまとめる．

要件 1 複数種類の IDS を統合管理できること

要件 2 広域監視のための統合分析を行えること

要件 3 詳細監視のための個別分析を行えること

要件 4 各種 IDS ログに共通するパラメータの分析を行えること

要件 5 冗長なログを排除できること

要件 6 出現頻度の低いログを見逃さないこと

要件 7 要件 5 のための特別な作業が不要であること

要件 8 ログ出力特性を考慮できること

要件 9 異常を客観的な数値で把握できること

要件 10 SOC から発信される情報の正当性を証明できること

4.3 提案システム

ここでは 4.2.2 節の要件 1-4 を達成するために，各種 IDS のログに共通して含まれる多角的な分析に必要なパラメータを統合管理する DB を設けて，この DB を用いて広域監視と個別監視を行う IDS ログ分析支援システムを提案する．

4.3.1 構成

各ネットワークに導入された IDS からのログを収集，管理，分析する IDS ログ分析支援システムの構成を図 4.1 に示す．ログ収集部は，各地の IDS から出力される様々なログをネットワーク経由で定期的に収集する．ログ保存部は，4.2.2 節の要件 1 を考慮して，収集したログに共通して含まれるパラメータを抽出し，統一されたフォーマットを持つ IDS ログ統合管理 DB に保存する．ログ分析部は，運用者からの要求に従い各種イベントに関する異常率を算出する．インタフェース提供部は，運用者に対して Web ブラウザによるインタフェースを提供する．このとき 4.2.2 節の要件 10 を考慮して，SOC からの発信情報の正当性を証明するために，公開鍵基盤を用いた署名技術を利用する．これは，署名に利用する秘密鍵と対をなす公開鍵を世界的に信頼されている認証局に登録しておき，情報発信の際には秘密鍵で発信情報に署名を施しておくことで，情報を受けた利用者が認証局に登録されている公開鍵を用いてその正当性を確認できる技術である．

ログ分析の単位は，4.2.2 節の要件 2, 3, 8 を考慮して，全ての IDS ログを一括して処理する統合分析機能と，個々に処理する個別分析機能を設ける．

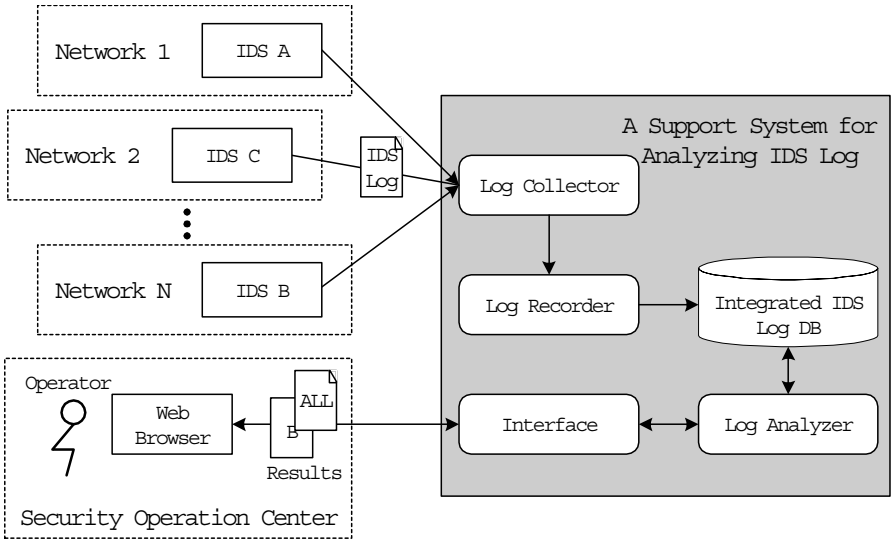

図 4.1: IDS ログ分析支援システムの構成

4.3.2 分析パラメータと DB 設計

代表的な IDS のログとして，Snort [9] の Alert ファイルを図 4.2 に示す．各イベントは，[**] で囲まれたシグネチャID とシグネチャ名で始まり，次の [**] の前までである．各種 IDS から出力されるログには，検知日時，Attack Signature，Source/Destination Port，Source/Destination IP，通信プロトコルなどの情報が共通して含まれている．

これらのパラメータのうち，本システムでは 4.2.2 節の要件 4 を考慮して，以下の 7 種類を分析対象とする．

- Attack Signature
- Source/Destination Port
- Source/Destination IP
- Source/Destination Country

```
[**] [1:1418:2] SNMP request tcp [**]
[Classification: Attempted Information Leak] [Priority: 2]
11/21-19:23:53.643852 192.168.10.34:1086 ->192.168.20.36:161
TCP TTL:128 TOS:0x0 ID:164 IpLen:20 DgmLen:48 DF
******S* Seq: 0x480DBF7C  Ack: 0x0  Win: 0xFAF0  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
[Xref => cve CAN-2002-0013][Xref => cve CAN-2002-0012]
[**] [1:1420:2] SNMP trap tcp [**]
[Classification: Attempted Information Leak] [Priority: 2]
11/21-19:23:53.644145 192.168.10.34:1087 -> 192.168.20.36:162
TCP TTL:128 TOS:0x0 ID:165 IpLen:20 DgmLen:48 DF
******S* Seq: 0x480E4F26  Ack: 0x0  Win: 0xFAF0  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
[Xref => cve CAN-2002-0013][Xref => cve CAN-2002-0012]
```

図 4.2: Snort の Alert ファイル

ここで，Attack Signature に関しては，IDS ごとに独自に銘々されており，統一された名称になっていない．そこで，IDS 間の Attack Signature の統一化を目的とした Common Vulnerabilities and Exposures(CVE) [10] や WHITEHATS [11]，その他，脆弱性情報を公開しているサイト [12, 13] の情報を基に，様々に銘々された Attack Signature の統合を図る．例えば Snort の Attack Signature の場合，図 4.2 の最終行に，

[Xref => cve CAN-2002-0012]

という CVE の参照 ID が記されており，こうした参照情報を基に，統一した Attack Signature を付与する．Source/Destination Country については，IANA [14] で管理されている IP アドレスとドメイン表から Country を割り出している．

7 つの分析パラメータを管理するにあたり，時間や場所，各イベントの検知数などの付随情報が必要になる．

- いつ Time

表 4.1: イベントテーブル

Event	IDS	Signature	Time	Source IP : Port	Destination IP : Port	Protocol	Event-count
1	1	sig1026	2002/5/3 5:42:41	192.168.1.33 : 5963	192.168.2.42 : 53	UDP	1
2	1	sig2012	2002/5/3 5:45:08	192.168.1.33 : 1766	192.168.2.42 : 80	TCP	1
3	2	sig1016	2002/5/3 5:45:08	192.168.2.40 : 1767	192.168.1.33 : 111	TCP	1
4	1	sig1003	2002/5/3 5:45:08	192.168.1.36 : 1935	192.168.2.40 : multi	TCP	793
5	1	sig1102	2002/5/3 5:45:08	192.168.1.37 : 1972	192.168.2.40 : 135	TCP	8
6	1	sig1102	2002/5/3 5:45:08	192.168.1.38 : 1977	192.168.2.41 : 135	TCP	8
7	1	sig2301	2002/5/3 5:45:08	192.168.1.39 : 3333	192.168.2.40 : 137	TCP	7
8	1	sig1102	2002/5/3 5:45:09	192.168.1.38 : 2222	192.168.2.41 : 135	UDP	1

- どこ IDS-ID
- どのくらい Event-count

以上のパラメータを管理するイベントテーブルを表 4.1 に示す．次章の分析は，本表の Event-count の異常性に注目する．

ここで，IDS によっては表 4.1 に示されるパラメータ以外にも，パケットのヘッダ情報や検知理由なども出力される．これら付加的なパラメータについては，別途テーブルを設けて情報の欠落がないように管理する．

4.4 分析手法の適用

ここでは，本システムは，必要な情報を多角的に判断するための支援を目的としており，従来システムが持つ頻度分析に加えて，4.2.2 節の要件 5-9 を達成するために，出力されるイベントの変化量に注目した分析手法の適用を提案する．これは，出力数が様々に変動するイベントの中から異常に変化したイベントを検出するために，過去の長期間の出力特性と最近の短期間の出力特性について，出力数の平均値を比較する比率分析と，平均値と標準偏差を用いて評価する稀率分析を適用するものである．2 つの提案手法は，IDS から出力された冗長なイベントを削除することなく，イベントの変化の程度を順位付けする手法である．よって，IDS 側では冗長なログが出力される IDS の設定はそのままでも構わない．

4.4.1 比率分析

比率分析は，出力数の変化量に注目する分析手法である．比率分析の様子を図 4.3 に示す．横軸が期間を，縦軸がログに含まれるイベント数を表している．

期間 T_s で表す短期プロファイルのイベント数を E_s とし，T_s を含まない過去の期間 T_l で表す長期プロファイルのイベント数を E_l としたときに，長期プロファイルに対する短期プロファイルの比率 D は，

$$D = (E_s/T_s)/(E_l/T_l)$$

と求めることができる．ここで，短期プロファイルで検知されたイベントのうち，長期プロファイルで 1 件も検知されていないイベントについては，比率を算出できないため，初検知イベントとして別に列挙する．

比率分析を Attack Signature に適用する場合，短期プロファイルで検知された

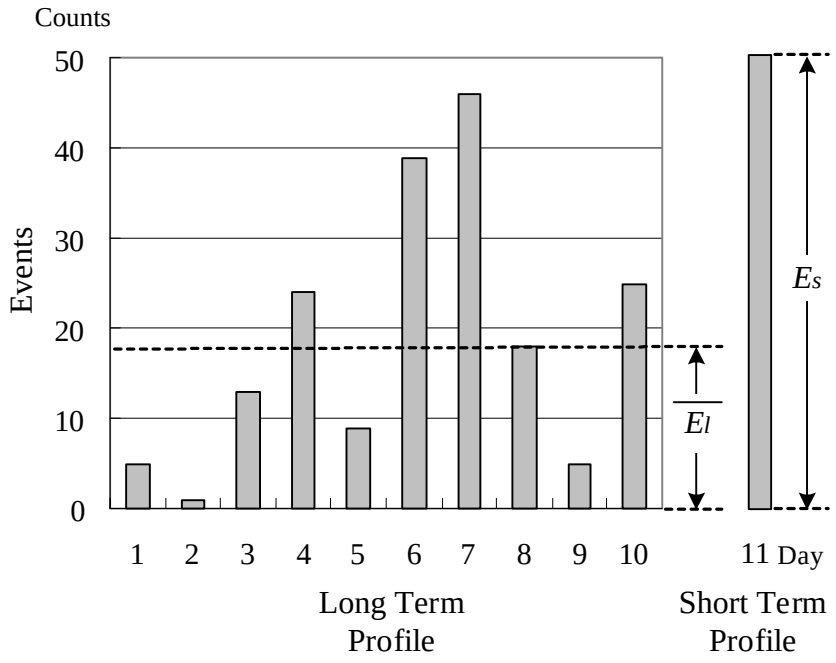

図 4.3: 比率分析モデル

Attack Signature の各イベントに対して一つずつ比率を算出する．出力結果として，比率の大きなイベントから順に，もしくは，小さなイベントから順に表示する．

4.4.2 稀率分析

SOC で収集されるログの規模は膨大な量に上る．この規模が大きなことから，検知されるイベントは正規分布に近づくものと仮定する．

稀率分析は，出力数の平均と標準偏差を用いて，様々に変動するイベントの異常性を評価する手法である．例えば，Attack Signature の中に比率分析の結果が同じイベントがあった場合でも，変動の大きなイベントの比率と小さなイベントの比率では，その値の重みが異なってくる．統計分析に，95%信頼区間という指標がよく

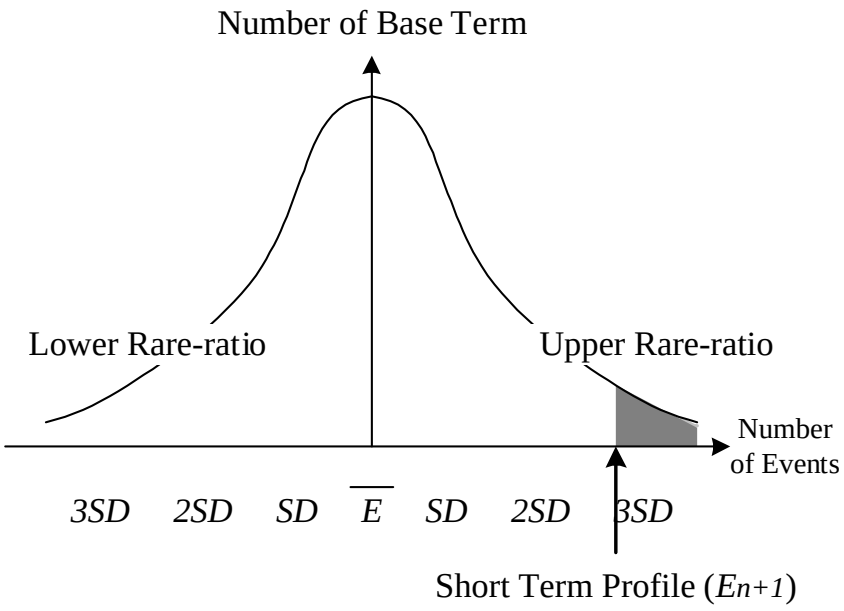

図 4.4: 稀率分析モデル

利用されているが，ここでの稀率分析とはこの信頼区間の補集合を算出する手法である．稀率分析の様子を，図 4.4 に示す．横軸は長期プロファイルのイベント数の平均 $\overline{E}$ とその標準偏差 SD の距離を，縦軸はイベント数に該当する単位期間数を表している．短期プロファイルのイベント数が長期プロファイルのイベント数の平均値よりも大きな場合の上側信頼限界の補集合を上側稀率と呼び，小さな場合の下側信頼限界の補集合を下側稀率と呼ぶことにする．上側稀率ならびに下側稀率の最大値は 50%であり，この 50%付近のときは通常のイベントの状態と変わりがないことを表している．

$n+1$ 個の単位期間があるときに，期間 T_{n+1} で表される短期プロファイルのイベント数を E_{n+1} とし，期間 $T_1, T_2, T_3, ..., T_n$ で表される長期プロファイルの各期間に含まれるイベント数を $E_1, E_2, E_3, ..., E_n$ とする．このとき，長期プロファイルのイ

ベント数の平均 $\overline{E}$ は,

$$\overline{E} = \frac{\sum_{i=1}^{n} E_i}{n}$$

であり，長期プロファイルのイベント数の標準偏差 SD は,

$$SD = \sqrt{\frac{\sum_{i=1}^{n}(E_i - \overline{E})^2}{n}}$$

となる．これより長期プロファイルの正規分布の密度関数 $f(E)$ は,

$$f(E) = \frac{1}{\sqrt{2\pi}SD} exp\left\{\frac{-(E-\overline{E})^2}{2SD^2}\right\}$$

となる．よって，短期プロファイルの E_{n+1} が長期プロファイルの $\overline{E}$ よりも大きなときの上側稀率 R_u は,

$$R_u = \int_{E_{n+1}}^{\infty} f(E)dE$$

と求めることができ，小さなときの下側稀率 R_l は,

$$R_l = \int_{-\infty}^{E_{n+1}} f(E)dE$$

と求めることができる．ここで，短期プロファイルで検知されたイベントのうち，長期プロファイルで 1 件も検知されていないイベントについては，稀率を算出できないため，初検知イベントとして別に列挙する．

稀率分析を Attack Signature に適用する場合，短期プロファイルで検知された Attack Signature の各イベントに対して一つずつ稀率を算出する．出力結果として，平均値に対してプラス側の稀なイベントから順に，もしくは，平均値に対してマイナス側の稀なイベントから順に表示する．

4.4.3 長期プロファイルの指定

新たな攻撃傾向を把握するには，直近の連続した時間を長期プロファイルとすることが妥当である．しかし，数ヶ月に渡る長い期間出力され続けているイベントについては，短期プロファイルで指定した時刻や曜日を考慮して長期プロファイルを選択することで，より妥当な分析結果を得られると考えられる．そこで本システムでは，長期プロファイルの期間の指定方法として，4.2.2 節の要件 8 を考慮して以下の 3 つを設ける．

(a) を基本条件として，(b) もしくは (c) を AND 条件で指定する．

(a) 連続時間 指定した開始日時分秒から終了日時分秒までの連続する期間を長期プロファイルとする．

(b) 時刻指定 (a) で指定された期間のうち，短期プロファイルで指定する時間帯を 24 時間周期で抽出したものを長期プロファイルとする．例えば，短期プロファイルで 12 時から 18 時までの 6 時間を指定した場合には，長期プロファイルもその時間帯のイベントを用いる．

(c) 曜日指定 (a) で指定された期間のうち，短期プロファイルで指定する曜日帯を 7 日間周期で抽出したものを長期プロファイルとする．例えば，短期プロファイルで平日（月〜金）を指定した場合には，長期プロファイルも平日のイベントを用いる．

4.4.4 分析結果の判断

$D > 1.00$ 倍 **もしくは** $R_u \simeq 0.00\%$ 新たな攻撃が出回り始めたとき，内部ホストがワームに感染したとき，DDoS 攻撃を受けたときなどの異

常に注目できる．攻撃が活発な段階であり，警戒する必要がある．

$D \simeq 0.00$ 倍 **もしくは** $R_l \simeq 0.00\%$ 普段から検知され続けていたログが減少もしくは無くなってしまう異常に注目できる．攻撃が収束に向かっている段階，もしくは，システムが停止した状態である．

$D \simeq 1.00$ 倍 **もしくは** $R_t, R_l \simeq 50.00\%$ 普段から検知され続けているログを見分けることができる．既に対策が施されている攻撃，もしくは，誤検知されやすいイベントであり，特に注意する必要はない．

初検知 新たな攻撃で残る微かな痕跡の可能性があり，注意すべきイベントである．

4.5 評価と考察

ここでは，分析手法の有効性の確認のために Attack Signature に注目して評価を行う．

4.5.1 評価用データ

評価に用いるログとして，インターネットに接続された日本国内の 3 箇所のネットワークで実運用されている IDS から収集したログを利用する．IDS は，ネットワークによって Firewall の内側や外側に設置されており，その設定は日々更新されている．ネットワークの IP アドレス規模と一日あたりに出力されるログの規模を表 4.2 に示す．ログの規模に関しては，ログ A を基準にした概算倍率で示してある．

評価を行うにあたり，短期プロファイルと長期プロファイルのデフォルト条件を

表 4.2: ネットワークとログの規模

	ネットワーク規模	ログ規模
ログ A	クラス B ネットワーク	1.00
ログ B	クラス C ネットワーク	0.15
ログ C	8IP ネットワーク	0.10

表 4.3: 評価用プロファイルのデフォルト条件

プロファイル	ログ	期間
短期	統合ログ A-C	2003 年 8 月 20 日の 1 日間
長期	統合ログ A-C	2003 年 6 月 20 日から 2003 年 8 月 19 日の 2ヶ月間 時刻・曜日を考慮しない

表 4.3 の通りとする.

4.5.2 比率分析と稀率分析の評価

表 4.4 に，Attack Signature に関する従来からの頻度分析の上位 10 位までの結果に対して，本論文で提案した比率分析と稀率分析の結果をマッピングした．”-”印は，各分析で上位 10 内に入らなかったものを表す（以下の全てで同じ）．”上”印は上側稀率を，”下”印は下側稀率を表す.

表 4.4 より，頻度分析による 1,4,5,7,8,10 位は比率分析や稀率分析では 10 位内に

表 4.4: 頻度分析を基準にした Attack Signature の結果

Attack Signature	頻度分析		比率分析		稀率分析	
検知 47(初 0)	順	件	順	倍	順	%
Ping sweep	1	9300	-	1.60	-	上 46.81
HTTP port probe	2	8959	-	1.95	10	上 20.90
MSRPC port probe	3	7975	1	47.44	2	上 0.00
TCP port scan	4	7506	-	0.44	-	下 33.64
TCP port probe	5	3296	-	0.84	-	下 43.25
SMTP port probe	6	3093	8	3.30	8	上 7.64
FTP port probe	7	2603	-	1.37	-	上 30.15
DNS port probe	8	1544	-	0.70	-	下 41.29
FTP PORT bounce	9	1230	7	3.52	7	上 5.16
Echo reply wo req	10	787	-	1.19	-	上 31.20

入ることもなく，普段から検知され続けているイベントであることがわかる．特に，頻度分析の 4,5,8 位については，頻度値としては大きな値だが，長期プロファイルと比べると減少していることから，不要なイベントであることがわかる (節の要件 5 の達成)．頻度分析の 3 位については，比率分析と稀率分析で 1,2 位になっており，急増している危険なイベントである．ちなみにこれは，2003 年 8 月に世界的に被害の拡がった MSBlaster ワーム [15] による攻撃先ホストの探索イベントであり，実際には IDS の設置されたネットワーク内部で感染が広がっていた．このように，従来の頻度分析では頻度値の大きなイベントに注目しがちであったが，比率分析と稀率分析をマッピングすることで，実際には多くのイベントが日頃から出力され続けて

いる様子がわかるため，運用者の分析作業の省力化に繋がる．また，運用者にログ A,B,C のネットワーク情報や分析経験が無い場合でも，急増するイベントを客観的な数値として評価できるようになり (節の要件 9 の達成)，信頼性の高い判断が可能になる．

表 4.5 に，比率分析の上位 10 位までの結果に対して，頻度分析と稀率分析の結果をマッピングした．表 4.5 より，比率分析の 2-6,9 位は，頻度分析では 10 位内に入っていない．ここで比率分析の 2 位について詳細に調査したところ，HTTP サービスで独自に開発した文字列検索機能を提供する CGI プログラムに対して，手動らしき時間間隔で，”AAAA...A” という長い文字が入力されており，バッファオーバフローを狙った侵入の試みであった．この CGI プログラムに対して特殊文字列を入力する追加調査を実施したところ，サニタイジングされないケースが判明し [16]，プログラムの修正が必要であることがわかった．また，5 位についても調査したところ，組織から離れてメールアカウントを削除されたユーザが，過去に利用していたアカウントへ繰り返しアクセスしているイベントであった．このように，頻度分析では見落としがちな微かな痕跡の中から，異常なイベントに注目できることがわかる (節の要件 6 の達成)．

今回，短期プロファイルで検知された Atatck Signature の総数は 47 種類で，初検知 Attack Signature は無かった．

この他，MS Blaster ワームの感染活動について，Destination Country に視点を変えて分析を行ったところ，Japan のイベント数が比率分析では 2.75 倍増加して 10 位圏外であったにも関らず，変動を考慮した稀率分析では 6.43%で 8 位に入っていた．Japan イベントについては，長期プロファイルの検知数は多いが検知数の変動は小さい特徴があった．このように，ばらつきの少ない安定したイベントが少しでも変化するときの異常性に注目するには，稀率分析の結果が有効である．

表 4.5: 比率分析を基準にした Attack signature の結果

Attack Signature	頻度分析		比率分析		稀率分析	
検知 47(初 0)	順	件	順	倍	順	%
MSRPC port probe	3	7924	1	47.44	2	上 0.00
HTTP repeated char	-	6	2	36.61	1	上 0.00
TCP small segment	-	94	3	12.29	3	上 0.01
SMTP pipe mailaddr	-	1	4	10.16	5	上 1.88
POP3 login failed	-	27	5	7.59	4	上 0.06
HTTP field binary	-	417	6	5.44	6	上 3.07
FTP PORT bounce	9	1230	7	3.52	7	上 5.16
SMTP port probe	6	3093	8	3.30	8	上 7.64
HP Remote watch	-	1	9	2.77	9	上 14.46
FTP PASV DoS	-	2	10	2.39	-	上 33.72

4.5.3 長期プロファイルの評価

長期プロファイルとして比較的長い期間を指定すると，過去に発生した多くのイベントを学習できる．しかしながら，攻撃傾向は日々変化している [7]．比較的長い期間を指定した場合，短期プロファイルを評価するための基準データとして，掛け離れた長期プロファイルとなりうる．ここでは，長期プロファイルの期間を変化させたときの比率分析と稀率分析について検証する．

図 4.5 に，表 4.3 のうち長期プロファイルを 1 週間から 2ヶ月まで変化させたときに，短期プロファイルとして 2003 年 8 月 18 日から 20 日の 3 日間の頻度分析の 10 位

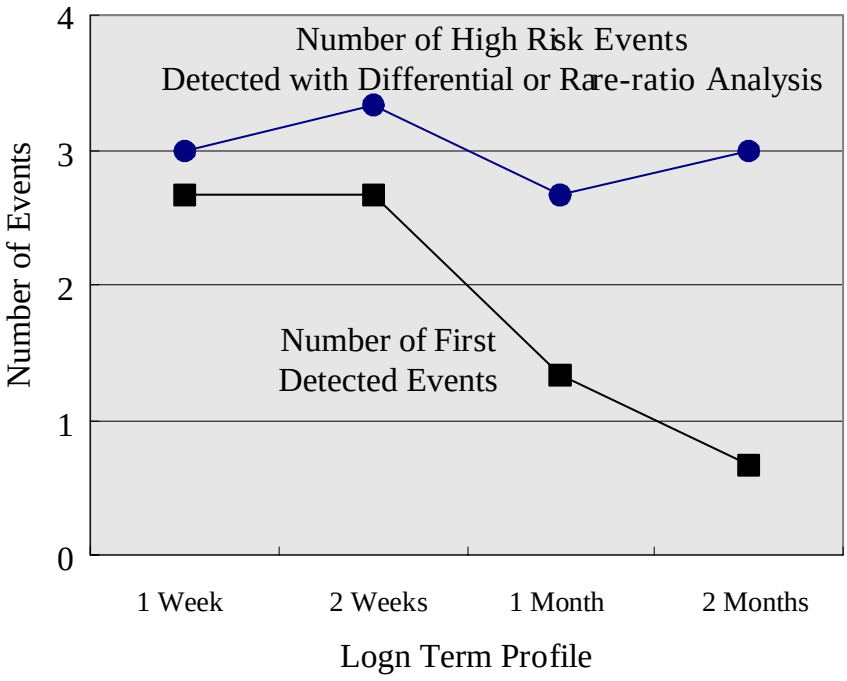

図 4.5: 長期プロファイルに関する評価

内のうち，比率分析と稀率分析のいずれか 10 位内に入った Attack Signature の 1 日あたりの数を示す．また，初検知された Attack Signature の 1 日あたりの数も示す．

図 4.5 より，長期プロファイルを 1 週間から 2ヶ月まで変化させても，危険なイベントとして指摘される Attack Signature の数にあまり変化はなく，長期プロファイルを 1 週間としても差し支えない様子がわかる．実際には，頻度分析で”HTTP port probe”が 3 日間とも 10 位内になっていたものの，長期プロファイルを変化させたときの比率分析と稀率分析で 10 位前後を変動していた．これ以外の指摘されたイベントについては，長期プロファイルを変化させても常に指摘されていた．初検知された Attack Signature の数からみると，長期プロファイルが 1,2 週間において複数種類が初検知されており，調査のための作業工数が余計に掛かることがわかる．

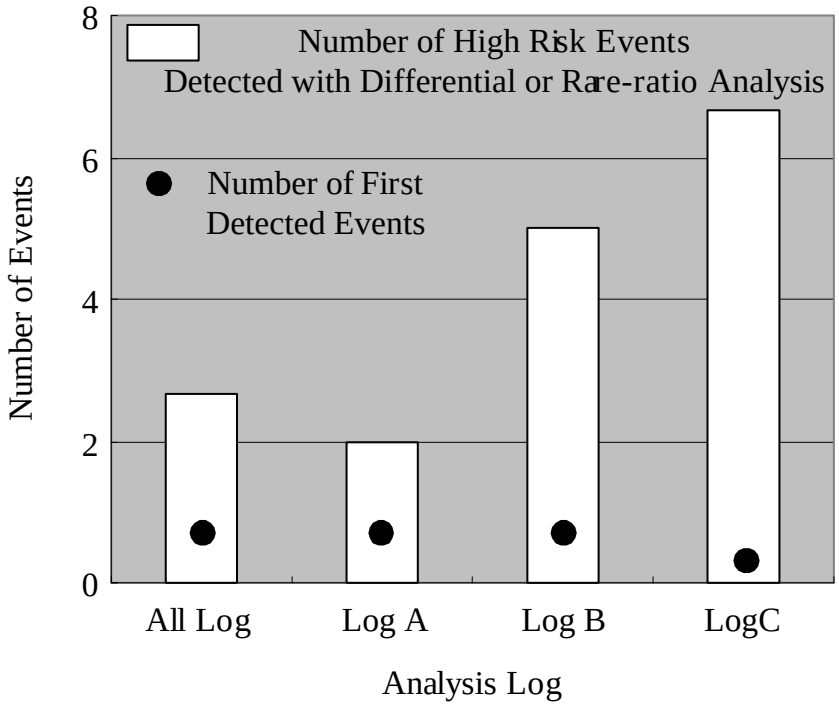

図 4.6: 統合分析と個別分析

4.5.4 統合分析と個別分析の評価

ここでは，ログ A-C を統合して分析する場合と個別に分析する場合の比率分析と稀率分析について評価する．

図 4.6 に，短期プロファイルとして 2003 年 8 月 18 日から 20 日の 3 日間の頻度分析の 10 位内までのうち，比率分析と稀率分析のいずれか 10 位内に入り，かつ増加傾向にある Attack Signature の 1 日あたりの数を示す．また，初検知された Attack Signature の 1 日あたりの数も示す．

図 4.6 より，個別分析で危険なイベントと指摘された Attack Signature の平均数は，ネットワーク規模が小さくなるにつれて大きくなっている．これは，ネットワー

ク規模が小さなログほど，過去の学習データ規模も小さくなり，短期プロファイルの頻度分析値が，比率分析や稀率分析で敏感に評価されるためである．統合分析によって指摘された Attack Signature の平均数は比較的小さくなっている．これは，個別分析で危険とされた Attack Signature でも，他のネットワークで頻繁に検知されている場合があり，異常なイベントではないと判断されるためである．ここで，ログ A の平均数よりも統合分析の平均数の方が，若干大きくなっていることがわかる．これは，規模の小さなログ B やログ C で増加したイベントの中でも，特に危険なイベントがログ A に埋もれることなく指摘されているためであり，SOC での監視において，効率的に危険なイベントに注目できる様子がわかる．

初検知件数は，統合ログでも個別ログでもほぼ同じ数になっている．詳細を確認したところ，統合ログで初検知されたイベントとログ A で初検知されたイベントは一致していた．逆に，ログ B やログ C で初検知されたイベントは，ログ A において過去に検知されていた．このように，初検知イベントについても他のネットワークと比較することができ，周囲の状況を把握できる統合分析機能は，SOC における支援機能として重要な役割を果たす．

4.5.5 頻度分析・比率分析・稀率分析の運用指針

本システムは，従来手法としての頻度分析，提案手法としての比率分析と稀率分析を併設しており，これらの分析結果を相互比較することで，注目すべき異常なイベントを迅速に発見できることを目指している．頻度分析では頻度値の大きなイベントを，比率分析では急増したイベントを，稀率分析では長期間のばらつきの程度から逸脱して増加したイベントを抽出できる．逆に，頻度分析では頻度値の小さなイベントを，比率分析と稀率分析では定常的に行われるイベントを抽出できない．

ここで，出現頻度が高くかつ定常的に出現しているイベントの中に急増する攻撃が混在している場合には，比率分析，稀率分析において抽出が困難である．例えば，HTTP アプリケーションに対する新たな攻撃が出現した場合，Destination Port 80 のみに注目した分析では，これを見逃してしまう．しかし，ここで抽出できなかったイベントでも，プロトコル違反などを検知する Attack Signature や送信元の Source Country など，他の分析パラメータにおいて急増している可能性がある．日々の運用では，分析対象の 7 つのパラメータについて視点を変えながら監視を行うことで，埋もれがちなイベントを抽出できる．

また，一定の間隔で少しずつ実行されるイベントについても，変化の程度は小さく，頻度分析，比率分析，稀率分析を併用して，各種パラメータに関する分析を行っても的確に抽出できない．こうしたイベントを見逃さないためにも，初めて検知されたイベントを列挙する機能も備えており，運用者に注意を促す機能となっている．

4.6 総括

本章では，各地の IDS から出力されるログを統合管理して，イベントの出力特性の異常性を比率分析と稀率分析で評価する IDS ログ分析支援システムを提案した．実際のログを用いて評価を行った結果，従来からの頻度分析結果の中から減少傾向の不要なイベントを特定できること，発見が困難であった頻度値が小さいながらも急増するイベントを抽出できることを確認した．

SOC では 24 時間継続的に複数のネットワークを監視しており，個々のネットワークのイベントの推移を把握しておくことや，こうした情報を運用者間で引き継ぐ作業は煩雑である．本システムは，各ネットワークを個別にもしくは一括して，比率分析や稀率分析の結果を頻度分析にマッピングする機能を有しており，異常に変化

するイベントの抽出作業において頻度値の履歴を追う必要がなく省力化に寄与する．また，客観的な数値で異常性を評価できるため，監視経験の浅い運用者にも画一的な判断基準を提供することができ，SOC 運用における支援システムとして期待される．

本論文の稀率分析では，検知される全てのイベントを正規分布と仮定している．正規分布と仮定するにあたり，我々が調査した経験では，日々検知され続けているイベントを正規分布で評価するには適している知見を得ている．しかしながら，攻撃の種類や流行の度合いによっては，異なる分布を仮定することが妥当な場合も考えられる．本稀率分析は，別の分布関数を求めることができれば，正規分布に従わなくてもその分布関数を利用して評価できる．今後の課題として，長期プロファイルを用いてイベントごとの分布関数を動的に導出して，その式から稀率分析を行う手法について検討を進める．

本評価の中では，IDS ログの統合管理によって新たに検出された異常はなかった．今後，検出が期待される攻撃として，ネットワークを伝播して広がるワームや DDoS 攻撃がある．これらの攻撃を検出するためには，より広域の IDS ログを統合管理して，分析時間単位の微小化を図り，かつ攻撃伝播波形を追跡する機能が必要になる．今後は，統合化の特徴を生かした分析手法に関して検討を進める．

第4章参考文献

[1] 沢田篤史，高倉弘喜，岡部寿男：開放型大規模ネットワークのためのIDSログ監視支援システム，情報処理学会論文誌, Vol.44, No.8, pp.1861-1871，Aug. 2003.

[2] 宮地玲奈，小宅宏明，川口信隆，岡田謙一，重野寛：機械学習によるネットワーク型IDSのfales positive削減手法の提案，情報処理学会，コンピュータセキュリティ研究会，CSEC，May. 2003.

[3] 新IDS導入の真価を探る，ソフトバンク パブリッシング，N+I Network Guide，Vol.21, pp.76-95, Jan. 2003.

[4] Internet Security Systems，http://www.isskk.co.jp/

[5] 大谷尚通, 小迫明徳, 桑田喜隆, 井上潮, 岩田恵一：広域不正アクセスに対する侵入検出状況把握システムに関する検討，情報処理学会, 第63回全国大会, Sep. 2001.

[6] 高田哲司, 小池英樹：見えログ：情報視覚化とテキストマイニングを用いたログ情報ブラウザ，情報処理学会論文誌, Vol.41, No.12, pp.3265-3275, Dec. 2000.

[7] Internet Storm Center，http://isc.incidents.org/

[8] 竹森敬祐，力武健次，三宅優，中尾康二：Intrusion Trap System における安全で有効なログ収集のための動的切替え機能の実装，情報処理学会論文誌, Vol.44, No.8, pp.1838-1847， Aug. 2003.

[9] Snort，http://www.snort.org/

[10] Common Vulnerabilities and Exposures(CVE),
http://www.cve.mitre.org/

[11] WHITEHATS Network Security Resource,
http://www.whitehats.com/

[12] Security Focus(Bugtraq),
http://www.securityfocus.com/bid/

[13] Computer Emergency Response Team/Coordination Center(CERT/CC) Advisories,
http://www.cert.org/advisories/

[14] The Internet Assigned Numbers Authority(IANA),
http://www.iana.org/

[15] IPA(情報処理振興事業協会) セキュリティセンター
http://www.ipa.go.jp/security/

[16] 日経ネットワークセキュリティ2003，日経 BP，Nov. 2002.

[17] 松原望，縄田和満，中井検裕：統計学入門，東京大学出版会，Jul. 1991.

[18] Edward G. Amoroso：Intrusion Detection，Intrusion.Net Books，NJ，1999.

[19] 武田圭史, 磯崎宏：ネットワーク侵入検知，ソフトバンク パブリッシング，Jun. 2000.

第5章

結論

近年，様々な通信プロトコル，ホストOS，サーバアプリケーションが急速に広がるにつれ，開発サイクルの短期化と設計の低コスト化が図られるようになってきた．これらのプログラムの開発において，機能の利便性ばかりに注目が集まる中，セキュリティを重視した設計は見落とされがちになっている．その結果，多くのホストで利用されているプログラムでさえも脆弱性が潜在するようになり，ひとたび未知の攻撃手法を用いたツールやコンピュータウィルスが現れると，各種インターネットサービスの停止に追い込まれてしまう脅威が顕著になってきた．

このような背景の中，広域のネットワークシステムを監視するSOCの構築が始まった．SOCにおけるサイバーテロ監視技術の要件として，各地のネットワークシステムの状態をリモート監視できること，各地のネットワークシステムから報告されるセキュリティログを統合分析して異常をいち早く把握できること，未知の攻撃情報を収集して新たな脅威を把握できること等が挙げられる．

本論文では，このような背景と要件から，広域ネットワークにまたがるサイバーテロ監視技術を提供することを目的として，広く普及しているインターネットアプリケーションの一つであるWebサーバシステムのリモート監視技術，未知の攻撃情報を収集できるトラップ方式のおとりシステムの制御技術，各種IDSログを統合管理して客観的に異常なイベントを抽出する分析支援技術について研究を行った．

第 1 章は，緒論であり研究の背景と本論文の概要について述べた．第 2 章は，Web サーバシステムのリモート監視技術に関する提案を行った．第 3 章は，未知の攻撃を収集するためのトラップ方式のおとりシステムの誘導制御手法の提案を行った．第 4 章は，各地に設置された IDS ログを統合管理して時間軸上での異常を客観的に評価する IDS ログ分析支援技術について提案した．本研究の主な成果を以下に列挙する．

第 2 章において，Web サーバが管理するホームページ用のファイルをリモートから監視することで，改竄攻撃のみならずネットワークサービス停止攻撃による異常を検知できることを示した．提案システムは，リンクページを辿ることにより監視対象となるファイルを自動的に抽出する機能と，ファイルのヘッダ情報やハッシュ値の変化に注目して改竄を判定する機能を有しているため，運用者の負担がほとんど無い状態において高い確率で Web サーバに対する攻撃を検知できる．大規模監視の実現性検証のために，インターネット上での監視処理速度に関する実験を行い，ネットワークへ与える負荷を軽減しつつ，局所的な輻輳に影響を受けない安定したリモート監視システムであることを定量的に明らかにした．

第 3 章において，正規システムとおとりシステムの通信状態の同期をとっておくことで，誘導処理の高速性と通信シナリオの継続性を確保でき，侵入者におとりシステムの存在を気付かれない通信コネクションの誘導制御手法を設計している．そして FTP サービスならびに Web サービスへの実装を行い，実験用ネットワークで評価を行った結果，侵入者に気付かれないレベルの高速な誘導を実現していることを確認した．このシステムにより，本来のサービスを提供しつつも侵入者の挙動・攻撃手法を容易に収集することが可能になる．

第 4 章において，長期間のイベント傾向を比較対象にして短期間のイベントの発生状況から，ログに含まれる各種イベントの異常性について順位付けする分析手法を設計した．各地で運用されている IDS の攻撃検知ログを用いて評価を行った結果，

多量のイベント情報の中から，従来では発見が困難であった異常なイベントを特定できること，検証不要なイベントを排除できることを確認している．客観的な数値で評価する分析手法により，監視経験の浅い運用者にも画一的な判断基準を提供することができ，IDS ログ分析における支援システムとして期待される．

本研究の成果を集めたサイバーテロ監視技術により，SOC における 24 時間継続的に複数のネットワークを監視する運用者の作業負担を大幅に軽減すると共に，個々のネットワーク上で発生している攻撃の挙動を容易に把握することが可能になった．

今後の課題

2 章で述べた Web サーバリモート管理システムの今後の課題として，更新によって送られるアラームの中に，改竄を知らせるアラームが埋もれてしまう問題がある．正規の更新と改竄を区別して検知する技術の確立が必要であり，変更のあったファイルの構文を解析して，改竄に見られる特徴の有無を普遍的なルールで検査する手順の検討を進めていく．

3 章で述べたトラップ方式の今後の課題として，収集した行動ログの中から攻撃情報を自動的に抽出する技術の確立と，おとりシステムが攻撃を受けた際に元の状態に復元させる技術の確立が必要である．また，個人情報を扱うシステムに適用する場合，運用中に正規システムとおとりシステムの両方に，重要な情報が保存されてしまう問題がある．今後は，秘密を守るべき情報の取り扱いにおいて，おとりシステム側に守るべき重要な情報が保存されないように，ダミーの情報に置き換える手法について検討を進める．

4 章で述べた IDS ログ分析支援システムの今後の課題として，Router や Firewall のログに関しても統合管理する技術の確立，ネットワーク単位の異常を検出するた

めのネットワーク間ログを比較分析する技術の確立，収集されるログの保全のために署名ならびに暗号化を施して管理する技術の確立，ネットワークの内側と外側の攻撃状態を比較分析する技術の確立，攻撃増減に関する未来予測を行う技術の確立がなど必要である．

今後の展望

本研究で述べてきたサイバーテロ監視の今後の展望として，本技術を通じて収集した攻撃手法や被害に関する情報を基に，

・セキュリティに関する啓蒙活動

・攻撃を仕掛けているサイト情報の公開やネットワークからの締出し

・ネットワークとホストが協調した自動防御（Prevention）

などの体制整備が挙げられる．

本研究により，的確なサイバーテロの検出や被害の把握が可能になり，これらの情報をネットワーク管理者や一般ユーザに広く公開することで，セキュリティ意識の向上に努める必要がある．また，セキュリティ対策に改善の見られないサイトに関しては，厳しく対処すべきであり，サイト情報の公開を通じてネットワークサービスの停止などの処置を図っていく体制作りを進める必要がある．さらに，本監視技術と Router や Firewall などのトラヒック制御機器やホストとが連携を図る技術の確立を進める必要がある．

SOC による監視情報を防御へと活用することは重要であり，ネットワーク運用者ならびに利用者にとって安心・安全なネットワーク社会の実現に向けた発展が期待される．

謝辞

本研究を遂行するにあたり，大変多くの方々からの御協力，御支援を頂き，ここに深く感謝の意を表します.

本論文の執筆にあたり, 御懇切な御指導, 御鞭撻, ならびに様々なご配慮を賜った慶應義塾大学理工学部情報工学科 笹瀬巌教授に深謝致します. また, 有益な御助言, 御示唆を賜った慶應義塾大学理工学部情報工学科 中川正雄教授，岡田謙一教授，山本喜一助教授に深く感謝の意を表します.

また，本研究の機会を与えて頂き，研究の途上で終始御熱心，御懇切に御指導，御助言を頂きました株式会社 KDDI 研究所 浅見徹所長，菅谷グループリーダ，田中俊昭グループリーダ，三宅優主任研究員，ならびに KDDI 株式会社 中尾康二セキュリティ室長に深く感謝致します．ここに心より御礼申し上げます.

さらに，多くの御助言, 御討論を頂いた, 慶應義塾大学理工学部情報工学科 荒川豊氏, 島田英一氏, 石田千枝氏, 北田夕子氏をはじめ笹瀬研究室の卒業生ならびに在学生の皆様に心より感謝の意を申しげます.

最後に長年私を支えて頂いた家族に心より感謝致します.