第 V 部

フローベースのネットワーク
トラフィック計測

第5部
フローベースのネットワークトラフィック計測

第1章　はじめに

roft（Research Of Flow Trend）ワーキンググループ（略称roft WG）は、フローベースのネットワークトラフィック計測手法について議論をするワーキンググループとして、2004年3月に創設された。フローベースのネットワークトラフィック計測とは、Cisco Systems社 [42] のNetFlow[41]、InMon社 [134] のsFlow[278] といったトラフィック観測技術を用いたトラフィック計測を指す。これらの観測技術を用いることにより、低コストに詳細なトラフィック情報を取得可能であり、特定トラフィックの検出やトラフィックエンジニアリングへの応用が期待されている。

roft WGでは、具体的な活動目標として、次の4つを挙げている。

1. フローベース計測関連ツール群の開発
2. 他のトラフィック計測技術との比較検討
3. 目的や対象に合わせた適正サンプリングレートの調査
4. 収集トラフィック情報に対する応用技術の検討

まず、フローベースのトラフィック計測に必要なツール群を開発し、計測に必要な敷居を下げることで、多くの計測ポイントへの導入を促す。次に、他のトラフィック計測技術とフローベース計測を比較検討し、それぞれの利点、欠点を明確にする。また、トラフィック計測結果の利用目的、または計測対象のネットワークの規模に応じた適切なパケットサンプリングレートを調査することで、広帯域ネットワークの計測に向けたスケーラビリティを評価する。そして、ポリシ違反トラフィックの検出や、トラフィックエンジニアリングなど、収集したトラフィック情報に対する応用技術について検討する。

本報告書では、roft WGの活動の一環として、まず、2003年WIDE春合宿での実験を報告する。この報告では、WIDE合宿における実トラフィックに対して、NetFlowを用いたフローベースのトラフィック計測をした手法と、利用傾向分析結果について述べる。次に、収集トラフィック情報に対する応用技術の例として、ユニークな通信先ホスト数に注目した異常トラフィック検知手法について報告する。提案手法は、ワームの攻撃やP2Pファイル共有アプリケーションによるトラフィックを異常トラフィックとし、ユニークな通信先ホスト数という指標に注目することで、それらの自動検出を目指している。この指標は、従来のトラフィック計測手法では取得が困難であるが、フローベースのトラフィック計測を用いることで低コストに取得可能である。本報告では、提案手法が従来から広く用いられている転送バイト量や転送パケット数などの監視では検出できなかった異常トラフィックが検出可能なことを示す。

第2章　flow実験チーム2004年WIDE春合宿実験報告

2.1 実験の主旨、期待していた成果

roft ワーキンググループ（http://www.roft.org/）（以下roft WG）は、フローを用いたトラフィック観測システムを提案し、研究開発している。2004年WIDE春合宿において、roft WGはflow実験チームとして実験を行った。当該実験では、フローと呼ばれる観測単位で合宿のネットワークトラフィックを観測することで、次の成果を目指した。

1. 得られたトラフィック情報をネットワーク管理者に提供することにより、合宿ネットワークの安定運用に貢献する。
2. 動作デモを通じて提案システムの有用性をアピールすることで、さまざまな地点での計測許可をいただく。また、既存トラフィック計測ツールで困っている点や、追加機能案、改良案などを広く収集し、今後の研究開発に生かす。

2.2 プライバシポリシ

ネットワークトラフィックを計測するにあたり、トラフィックに含まれる個人情報の扱いについて十

分考慮する必要がある。本実験では、パケットのレイヤ 3 およびレイヤ 4 ヘッダに書かれた情報、転送パケット数、転送バイト数のみを収集し、パケットのペイロード部分に関しては一切収集していない。また、一般ユーザに対するトラフィック可視化のデモンストレーションにおいては、アドレス情報など、個人の識別につながる情報は全て暗号化して提供した。今回収集したトラフィックデータは研究・運用目的でのみ利用する。特に、アドレス情報などを除いた統計データは今後の合宿ネットワーク運用の資料と本報告書上および Web アプリケーション経由で公開する。

2.3 実験の構成

2.3.1 合宿ネットワークトポロジ

2004 年 WIDE 春合宿のネットワークは図 2.1 に示すように、

- noc
- wired
- wireless-1
- wireless-2

の 4 つのセグメントで構成された。noc セグメントにはいくつかのインターネットへつながる回線とサーバ群が配置され、その他 3 つのセグメントは一般参加者の生活用セグメントとなった。

2.3.2 flow 実験チーム、システム構成

flow 実験チームは、前項で述べた 4 つのセグメントの間に設置されたルータ、Cisco7204VXR にて計測し、各セグメント間のトラフィック、各セグメントからインターネットへと向かうトラフィックは全て観測対象とした。noc セグメントからインターネットへ向かうトラフィックも、一時 Cisco7204VXR を経由するよう設定され、観測対象となった。Cisco7204VXR で観測されたトラフィックはフローと呼ばれる単位（Cisco の定める NetFlow に基づく）で集約され、NetFlow パケットとして、フロー収集用 PC（flow-NOC1）に対して出力された。flow-NOC1 は、フローデータ用データベースサーバと可視化されたトラフィック情報を Web アプリケーション経由で提供する Web サーバとを兼ねた。flow-NOC1 にて収集されたフローデータは Web アプリケーションを介して可視化されネットワーク管理者、一般参加者に提供された。

2.3.3 Web アプリケーションの構成

Web アプリケーションに、トラフィックを指定するためのさまざまなパラメータを与えることで、該当するトラフィックの情報を可視化することができる。指定できるパラメータとして、日時、入力インタフェース、出力インタフェース、IP プロトコルバージョン、プロトコルタイプ、送信元ポート番号、宛先ポート番号などがある。

トラフィックの可視化方法は 2 種類用意した。可視化方法 A では 24 時間の時系列で転送量または転送パケット数の変化を表示する。可視化方法 B では 5 分の時間枠の中で大量のトラフィックを発生させ

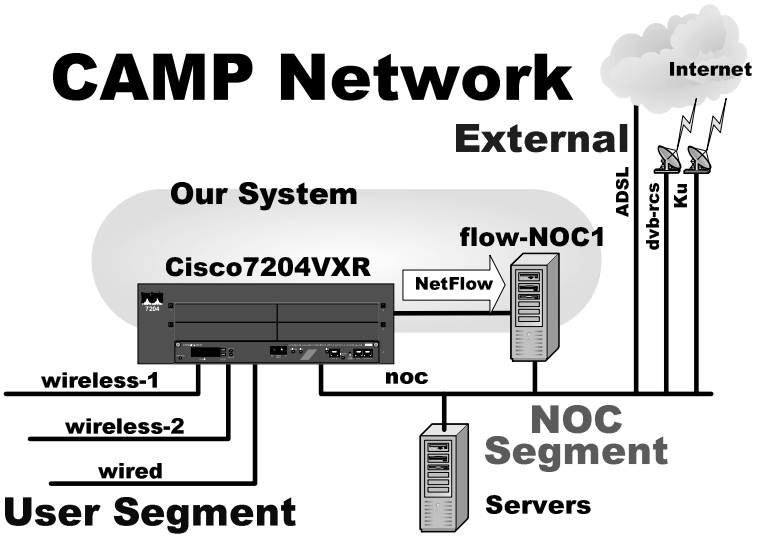

図 2.1.　合宿ネットワークの構成と flow 実験チーム機器の構成

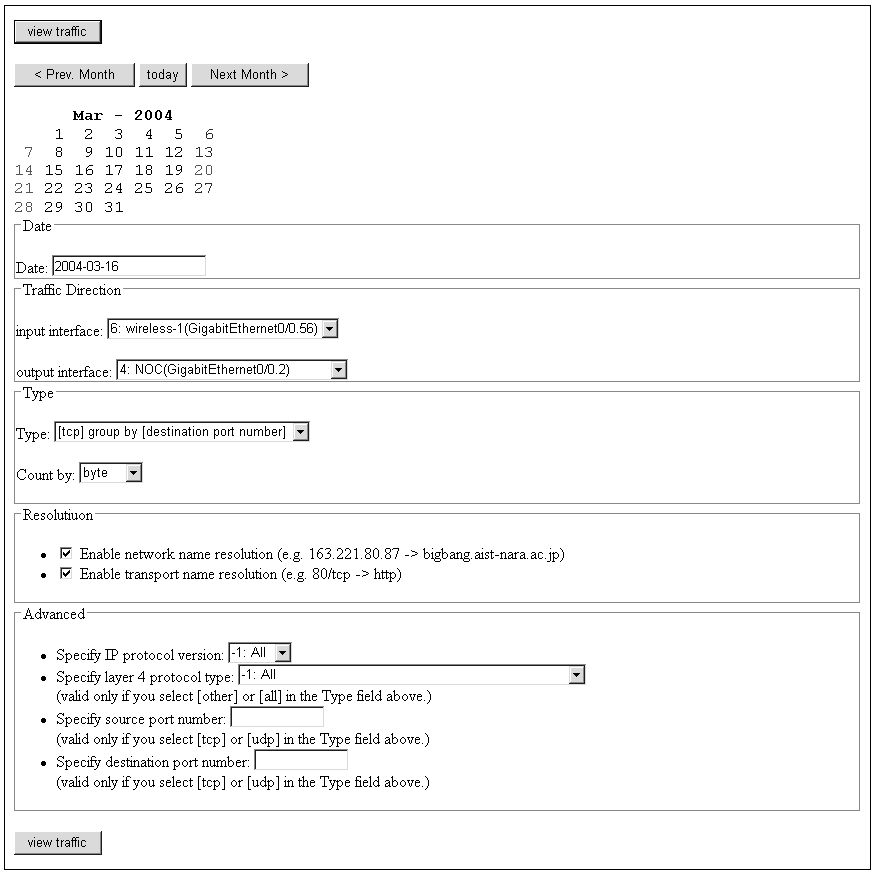

図 2.2. Web アプリケーショントップ画面

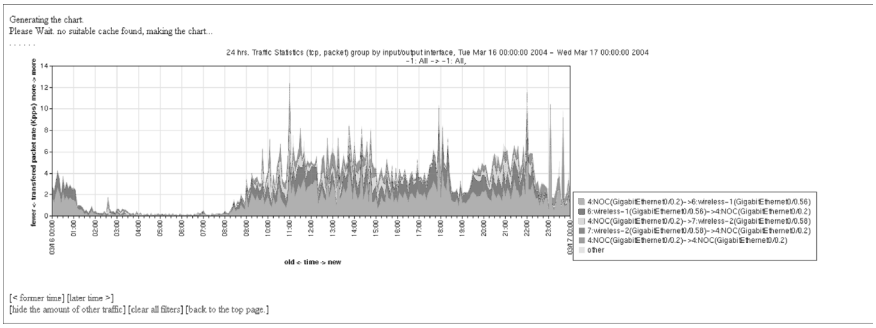

図 2.3. 可視化方法 A の例—入力・出力インタフェースごとに分類したトラフィック量変化

ている送信元ホストと宛先ホストを表示する。各可視化方法では、指定したパラメータに該当するトラフィックを種類ごとに分類（色分け）して表示できる。分類方法には、「送信元ポート番号や宛先ポート番号ごと」、「入力・出力インタフェースごと」、「IP プロトコルバージョンごと」、「プロトコルタイプごと」を用意した。

Web アプリケーションのトップページからパラメータと分類方法を指定することで、可視化方法 A のページへ遷移する。このページ上のトラフィック量変化のグラフから時間枠を指定することで可視化方法 B のページへ遷移する。図 2.2〜図 2.5 は実際の Web アプリケーションの動作画面である。

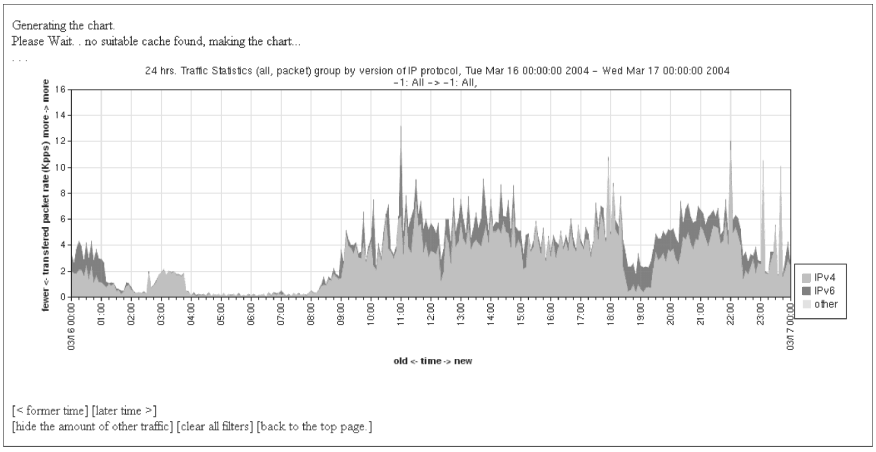

図 2.4.　可視化方法 A の例―IP プロトコルバージョンごとに分類したパケット量変化

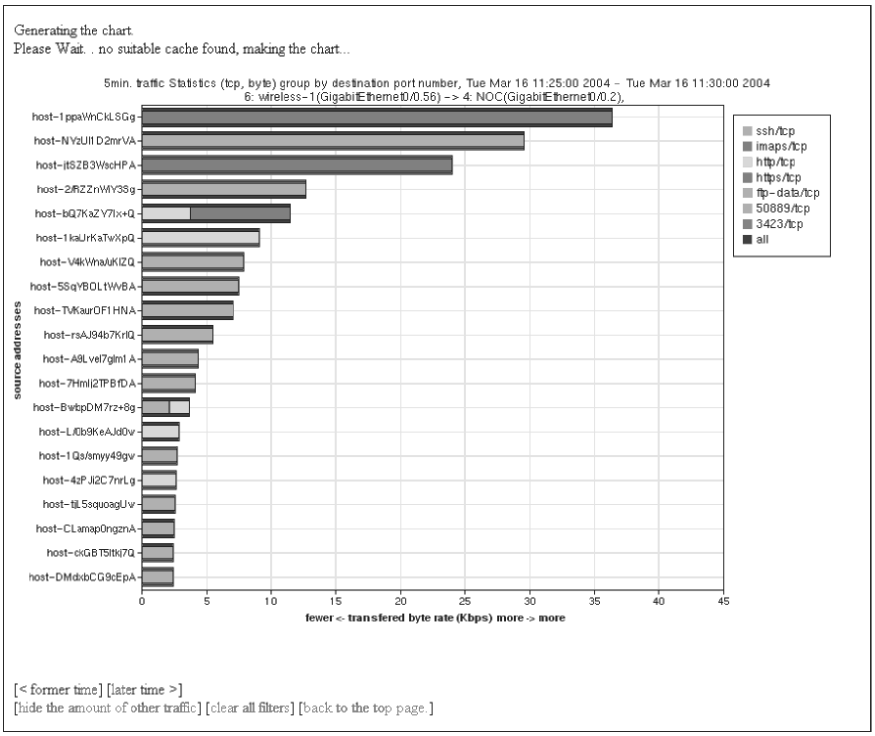

図 2.5.　可視化方法 B の例―宛先ポート番号ごとに分類したトラフィック発生量の多い送信元ホスト（ホスト名を暗号化済）

2.4 実験の結果と考察、得られた知見

トラフィック計測においてとくに問題は発生せず、計測は成功した。実験システムの構成で問題なくトラフィックの計測が可能なことが証明された。

計測の結果、合宿の 4 日間で 58,873 件のトラフィックレコードが観測された。これは、パケット数で見ると 93,998 個、転送バイト数で見ると約 39 MB 分に相当する。本実験では、サンプリングレートをパケット単位で 1/1000 とした。従って、実際に合宿期間中に Cisco7204VXR を通過したパケット数はその約 1,000 倍と推測され、約 9 千 4 百万パケット、39 GB となる。平均秒間転送レートに直すと、329 kpps、1.1 Mbps となる。1/1000 というサンプリングレートはデータベースサーバの負荷を考慮し、Web アプリケーションが現実的な時間で処理できるレコード数から逆算した値である。

2.4.1 トラフィック内訳

得られたトラフィック情報の代表的な統計値を示す。

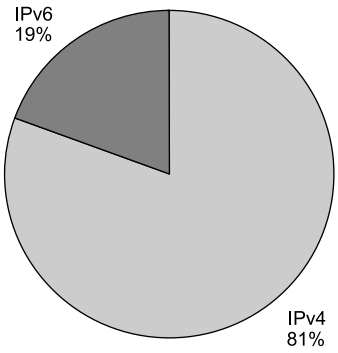

図 2.6. IP プロトコルバージョン別トラフィック量（パケットカウント）

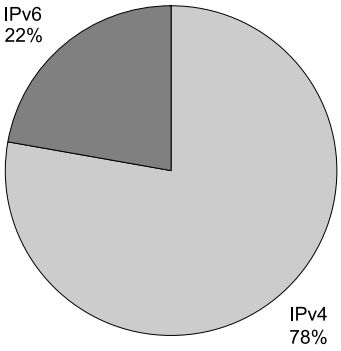

図 2.7. IP プロトコルバージョン別トラフィック量（転送量）

- IP プロトコルバージョン別トラフィック量

図 2.6、2.7 は全体トラフィックに対する IPv4、IPv6 それぞれの内訳を、パケットカウント単位、転送量単位で示している。IPv4 トラフィックが首位であるが、IPv6 トラフィックも 20%程度と、かなりの量を占めていることが分かる。

- ポート番号別トラフィック量

図 2.8 は合宿内セグメントから外部（インターネット）へ向かう上り TCP トラフィックの転送量を宛先ポート番号別に示している。22（ssh）が最も多く、80（http）、443（https）、25（smtp）と続いている。ただし、当実験ではレイヤ 4 ヘッダ上のポート番号情報のみからアプリケーションを判断しており、たとえばポート番号が 80（http）であるからといって、実際に http のトラフィックであるという保証はできない。

図 2.9 は同様に下りトラフィックの転送量を送信元ポート番号別に示している。ただしインターネットからの下りセグメントのほかに、noc セグメントからユーザセグメントへの下りトラフィックも含んでいる。80（http）が最も多く、22（ssh）やメール関連と続いている。

- プロトコル別トラフィック量

図 2.10 は全体トラフィックに対するプロトコル別のトラフィック量を示している。全体の 95%が TCP によるトラフィックであり、4%が UDP によるトラフィックである。トンネリングに使われる gre、esp、ipv6 がそれぞれ 1%以

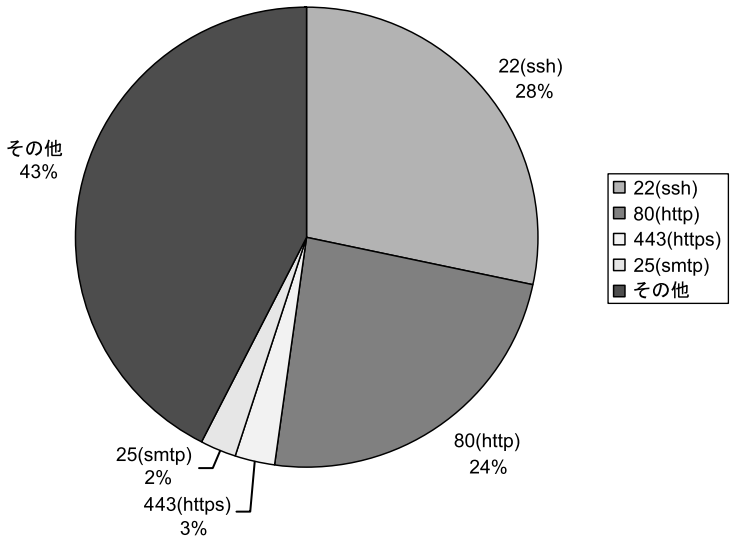

図 2.8. 宛先ポート番号別上り TCP トラフィック量（転送量）

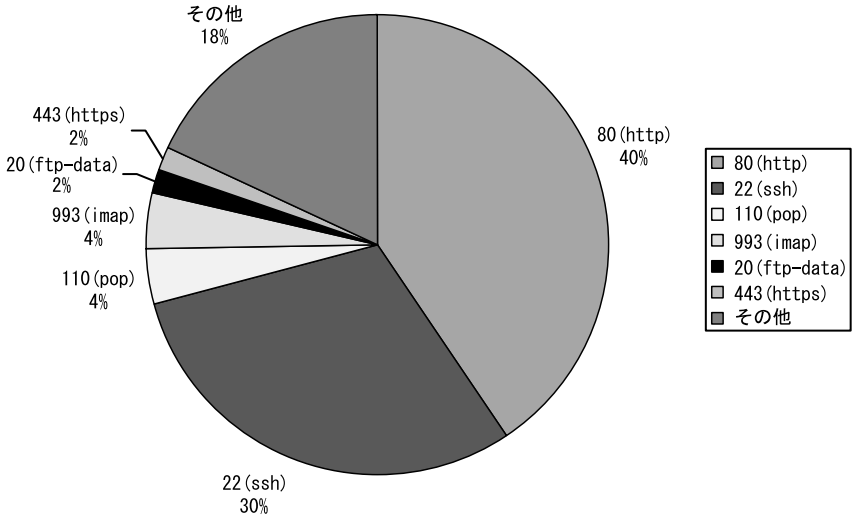

図 2.9.　送信元ポート番号別下り TCP トラフィック量（転送量）

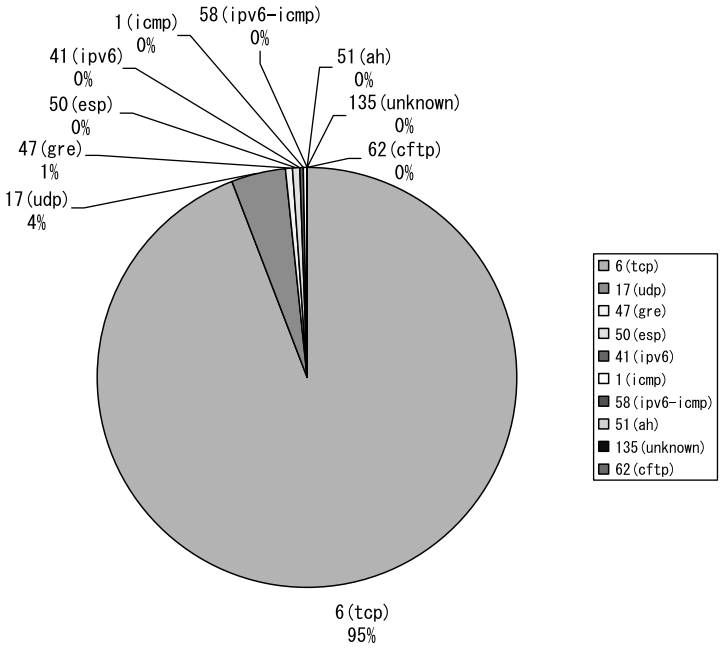

図 2.10.　プロトコルタイプ別トラフィック量（転送量）

下含まれている。icmp 関連トラフィックも全体トラフィック量と比較すると 1%未満である。

2.4.2 ワーム検知

本実験では、ネットワークの安定運用に貢献するためにワームに感染したホストの検知も目指した。その方法は、ワームが原因で発生するトラフィックの特徴を利用し、該当トラフィック量の変化を監視することで、ワームに感染しているホストを特定するというものである。合宿期間中、ワームが原因で大量発生することが多い ICMP、宛先ポートが 135/tcp であるトラフィックを実験システムによって監視した。しかし、そのようなトラフィックはほぼ 0 であり、感染ホストも検出できなかった。合宿では、IPtraceback 実験チームも同様の目的で IDS を設置していたが、その IDS でもワームが原因と推測されるトラフィックは観測されていない。このことから、本実験でのワームの検知方法に問題があったのではなく、実際にワームに感染したホストが発生しなかったものと考える。

2.4.3 トラフィック量変化

ネットワークの安定運用に必要な指標として、トラフィック量の変化がある。本実験では、回線障害などを原因とする、急激なトラフィックの減少や増加を監視した。その結果、各日におけるトラフィック量の周期的な変化は観測できたが、急激な変化は観測できず、障害検知には応用できなかった。

2.4.4 Web アプリケーションへのアクセスとアンケート結果

アクセスログからの推定によると、合宿期間中にのべ 100 人ほどの人に本実験の Web アプリケーションを試用して頂いたようである。

アンケートの結果には、実験自体や実験の有用性に対する否定的な意見はなく、提案システムの有用性のアピールは成功した。一方、Web インターフェースの操作性や、計測結果の応用方法に対する疑問も多く寄せられた。

2.5 結論

まず、本実験の 1 つめの目標である、「得られたトラフィック情報をネットワーク管理者に提供することにより、合宿ネットワークの安定運用に貢献する。」という点について述べる。トラフィック情報の管理者への提供については、Web アプリケーションを通じて正常にでき、問題はなかった。しかし、今回、想定していたワームによる異常トラフィックや回線障害によるトラフィック量の急激な変化は観測できず、それらの管理者への通知もできなかった。その原因の 1 つは、合宿ネットワークがそもそも安定していたことが挙げられる。また、合宿期間が 4 日間と非常に短く、正常なネットワークの状態の定義が困難であったことも原因である。

次に、2 つめの目標である、「動作デモを通じて、提案システムの有用性をアピールすることで、さまざまな地点での計測許可をいただく。また、既存トラフィック計測ツールで困っている点や、追加機能案、改良案などを広く収集し、今後の研究開発に生かす。」という点について述べる。Web アプリケーションの動作デモは合宿参加者のおよそ 35%、100 人前後の人に閲覧して頂いた。このパーセンテージは十分とは言えず、実験に関する宣伝が不足していたと考えられる。合宿参加者用メーリングリストやポスターなどを用いた宣伝は盛んに行ったが、BoF 枠などを活用して、宣伝すべきであった。追加機能案や改良案も、口頭やアンケート結果を通じて頂いたが、これについても BoF 枠を設定すべきであったと感じている。計測拠点の拡充については、WG 内で具体的な計測目的と必要機器、計測担当者などの詳細がまとめきれておらず、具体的な提案に持って行くことができなかった。WG 内でさらなる議論をし、これらの詳細を取り決める必要がある。

2.6 今後の方向

実験システムの構成で問題なくトラフィックの計測が可能なことが証明され、その応用事例（Web アプリケーション）の有用性についてもアピールできた。今後、roft WG は、フローベース計測であることの優位性をアピールするとともに、ほかの応用方法も検討する必要がある。

フローベース計測であることの優位性を示すには、広帯域なネットワーク上でも低コストで計測が可能なことを示すほかに、サンプリングレートの妥当性についても示す必要がある。計測データの目的に応じた妥当なサンプリングレートを検討するほか、目的によってはサンプリングをせずともスケーラブルに計測が可能なシステムの検討が必要であると考えている。

計測結果のほかの応用事例としては、過去のトラフィック状況を考慮した上で異常なトラフィックを自動的に検出する手法、多地点で計測することによって直接計測していない地点のトラフィックを予測する手法などを検討している。

第 3 章　通信先ホスト数の変化に注目した異常トラフィック自動検出手法の提案と評価

概要

本章では、一般的な組織の運用ポリシを鑑み、ワームによる攻撃および P2P ファイル交換ソフトウェアによるトラフィックを異常トラフィックと定義する。ネットワークがインフラとして広く利用されるようになるにつれ、ネットワーク管理者は組織の運営効率、健全性の維持のため、異常トラフィック発生の有無を常に監視する必要がある。しかし、ネットワーク上で展開されるサービスが多様化し、従来のよう

に転送バイト量やパケット量から異常トラフィックを検出する手法、シグニチャによる異常トラフィック検出には限界が生じている。本章では、既存の異常トラフィック検出手法とその問題点を挙げ、優位なトラフィック検出手法に必要な機能要件を挙げる。本章では、それらの機能要件に適合する異常トラフィック検出手法を提案する。提案手法を用いることにより、従来指標の監視では検出不可能な異常トラフィックを、低コストで、シグニチャに非依存で自動検出が可能であることを示す。また、提案手法による異常トラフィック検出の検出精度、および、広帯域への適応性、そして、リアルタイムな異常トラフィック検出への利用可能性を評価する。

3.1 はじめに

多くの企業や学校で、コンピュータネットワークは日々の活動に必要不可欠なものとなってきている。ネットワークへの依存度が高まるにつれ、ネットワークの障害が組織に及ぼす悪影響も大きくなる。組織のネットワーク管理者は、ネットワークの安定性を維持するため、障害を引き起こすトラフィックを検出し、障害を未然に防止することが求められる。また、障害の結果となるトラフィックを迅速に検出し、速やかに障害に対処することが求められる。

また、組織は、組織内ネットワークについて、利用ポリシを定めているのが通常である。たとえば、企業のネットワークであれば一般的に、私的な利用は認められない。また、学校であれば営利目的の利用が禁じられている場合が多い。こうした利用ポリシに反するトラフィックを検出し、該当ユーザに警告することも組織の健全性を保つ上で必要である。

このように、ネットワーク管理者は、組織の運営効率、健全性の維持のため、常にネットワークを監視し、さまざまなトラフィックを検出する必要がある。

ここで、本章ではワームによる攻撃と P2P ファイル共有アプリケーションによるトラフィックを異常トラフィックと定義し、検出対象とする。その理由は次の通りである。

まず、ワームによる攻撃トラフィックは、障害の結果生じたトラフィックであり、新たな障害を引き起こすトラフィックの典型例である。もし、組織内のホストがワームに感染すると、ネットワークのスループットの低下、感染ホストのパフォーマンスの低下が生じ、効率的な組織運営に支障をきたす。また、ワーム感染はインターネットを通じて組織外にも広がり他組織にも迷惑をかける。そうなると、ワームの感染は組織の自己責任だけではすまされず損害賠償問題に発生するケースもある。したがって、ネットワーク管理者はワームに感染したホストがないかを常に調査し、感染ホストが発生したら速やかに対処し、感染の蔓延を抑えなければならない。

また、P2P ファイル交換アプリケーションの普及により、組織内のユーザが著作物を違法に交換するという事例も後を絶たない。こうしたユーザの存在は、組織の倫理問題だけでは済まされず、訴訟に発展し、組織の名誉が傷つけられる可能性もある。ネットワーク管理者は、P2P ファイル交換アプリケーションの利用状況を把握し、違法な利用がないことを確認することが求められる。こうした背景から、ワームによる攻撃と P2P ファイル共有アプリケーションによるトラフィックは、多くの組織の運営ポリシ上、検出が求められる異常トラフィックの典型であるといえる。

しかし、こうした異常トラフィックの検出は容易ではない。

現在、多くの組織ではシグニチャ型 IDS[1] を用いてワームによる攻撃トラフィックを検出している。一方、ワームは悪意のある者により次々に亜種や新種が製造される。そのため、シグニチャ型 IDS はシグニチャ（検出パターン）の更新により新たなワームの検出に対応する。シグニチャ型 IDS の開発元は、ワームのサンプルを入手し、その動作の解析により最新のシグニチャを作成する。したがって、組織内にワームが蔓延する前に、開発元からシグニチャが提供されるとは限らない。つまり、従来の手法でワームを完全に検出することは不可能である。

また、従来はネットワーク上で提供される多くのサービスは使用するポート番号が固定的に決まっていた。ポート番号によるアクセス制御はパケットフィルタリングを用いることで容易に実現できる。したがって、管理者が利用ポリシに反する特定のサービスの使用を把握することは容易であった。しかし、多くの P2P ファイル共有アプリケーションはランダムなポート番号を使用する。つまり、特定のポート番号によるトラフィックを監視することで、これら

1　本章では、ネットワーク型 IDS とインライン型 IDS をまとめて、単に IDS と呼ぶ（詳細は、3.2.1 項参照）。

のアプリケーションの利用を検出することは不可能である。また、通信内容を暗号化するP2Pファイル共有アプリケーションも多い。アプリケーション固有のペイロード（通信内容）を観測することによる、これらのアプリケーションの利用の把握も不可能である。

ネットワークの広帯域化も、異常トラフィックの検出を難しくする要因の1つである。監視対象ネットワークが広帯域になればなるほど、監視しなければならないパケットレート（単位時間あたりのパケット数）は増加する。しかし、IDSが扱えるパケットレートには限界がある。また、対応するパケットレートが高くなるにつれIDS製品は高価となり、導入コストが高くなる。Snort[284]など一般のPC上で動作する無料のIDSも存在する。しかし、こうしたIDSの処理性能はPCの処理性能に依存し、高いパケットレートは扱えない。

その中で、広帯域下でも、シグニチャに依存せずに異常トラフィックを検出可能な手法が求められている。

こうした背景を踏まえ、本章では、異常トラフィックの共通した特徴を利用し、単位時間あたりのユニークな通信先ホスト数に注目した異常トラフィック検出手法を提案する。そして、提案手法が従来指標の監視では検出不可能な異常トラフィックを、低コストで、シグニチャに非依存で自動検出が可能であることを示す。また、提案手法による異常トラフィック検出の検出精度、および、広帯域への適応性、そして、リアルタイムな異常トラフィック検出への利用可能性を評価する。

3.2節では既存の要素技術と異常トラフィック検出に広く用いられる手法を挙げ、その特徴と問題点を明らかにする。3.3節では、既存手法の問題点を踏まえ、優れた異常トラフィック検出手法に必要な機能要件をまとめる。3.4節では提案手法の目的および想定環境とその詳細、および実装について述べる。3.6節では提案手法の評価と考察を行う。3.7節では、評価と考察の結果をまとめ、今後の課題を述べる。

3.2 既存技術

本節では、既存の要素技術と、広く用いられている異常トラフィック検出手法を挙げ、その特徴と問題点を明らかにする。

3.2.1 要素技術

本項では、異常トラフィック検出に用いられている要素技術について述べる。

トラフィック情報取得技術

トラフィック情報とは、通信トラフィックを構成するパケットの属性情報、パケットの集合体であるコネクションおよびフローの属性情報である。また、トラフィックの流量（転送バイト量や転送パケット数）もトラフィック情報と呼ぶ。異常トラフィック検出のためには、まず、これらトラフィック情報を取得し、分析する必要がある。ここでは、トラフィック情報を取得するために広く利用されている技術として、SNMP（Simple Network Management System）[34]、ミラーリング、NetFlow、sFlow[278]について述べる。

SNMP

SNMPは、ネットワーク上の機器が持つ情報の提供方法、およびその取得方法を定めた規格である。提供される情報は機器によって異なる。たとえば、ルータは、インタフェース情報（インタフェースタイプ、通過したトラフィック量、パケット数）やルーティングテーブルを提供している。一般的な利用法は、インタフェースを通過したトラフィック量、パケット数から、それらの値の時間変化を知ることである。SNMPの利点は、多くのネットワーク機器に実装されていて、提供される情報を容易に利用可能な点である。その反面、SNMPで提供される情報は限られており、個々のパケットに関する詳細な情報を取得したい場合は、別の手法をとる必要がある。

ミラーリング

ミラーリングはSNMPより詳細な情報を取得する手法の1つである。スイッチ上を流れるパケットをコピー（ミラーリング）して観測用の機器に転送する方法である。この手法を用いることで通常の通信に影響を与えることなく、その通信の全内容を取得することができる。ただし、この手法は全パケットの全内容をコピーするため、多くのトラフィックが流れる広帯域リンクを監視した場合にスイッチや観測機器への負荷が高くなるという欠点がある。

NetFlow と sFlow

SNMP では取得不可能な情報が必要だが、ミラーリングはコストが高すぎるという場合、NetFlow や sFlow が選択肢となる。

NetFlow は Cisco Systems 社が策定したトラフィック情報取得技術であり、同社のルータ製品に実装されている。近年は、アラクサラ社 [5] や Juniper Networks 社 [154]、Extreme Networks 社 [93] などのルータ製品にもこの機能が実装されている。Cisco Systems 社のルータ製品では、ルーティング用のキャッシュからトラフィック情報を取得することで、トラフィック情報取得によるルータの負荷を低く抑えている。NetFlow では、表 3.1 に示した値が同一の一連のパケットを集約して、フローと定義する。TCP におけるコネクションに近い概念であるが、UDP や ICMP のパケットについても定義される点が異なる。

表 3.1.　同一フローと見なされるための値

ID	information
1	SNMP index of input interface
2	IP Protocol Type
3	Source IP Address
4	Destination IP Address
5	TCP/UDP source port number
6	TCP/UDP destination port number
7	IP type of service (ToS)

フロー情報は、

- TCP において、FIN または RST フラグが立ったパケットを観測した場合
- そのフローに属するパケットが一定時間以上流れなかった場合
- そのフローに属するパケットが一定時間以上流れ続けた場合

のいずれかの条件を満たすと、NetFlow パケットと呼ぶ UDP パケットとして出力される。

NetFlow は、Version 1 から Version 9 まで各種バージョンが存在しており、取得できる属性情報は NetFlow のバージョンによってやや異なる。基本的にはレイヤ 3、レイヤ 4 のヘッダ情報、入出力インタフェース番号、バイト長などが取得可能である。たとえば、NetFlow Version 5 では各フローに対して表 3.2 に示した属性情報が取得できる。これらの情報は 1 フローあたり 48 byte とコンパクトであり、長期観測での利用が可能である。

表 3.2 における、ID 1、2 は、フローの送信元と送信先の IP アドレス、3 は次ホップの IP アドレスである。4、5 は観測ルータやスイッチにおける入出力インタフェースの番号である。6、7 は観測されたフローに含まれるパケット数と、転送されたバイト数、8、9 はフロー中の最初のパケットを受信した時刻（ミリ秒）と最後のパケットを受信した時刻（ミ

表 3.2.　NetFlow version 5 で取得できる情報

ID	information
1	Source IP Address
2	Destination Address
3	IP address of next hop
4	SNMP index of input interface
5	SNMP index of output interface
6	number of packets in the flow
7	cumulative bytes in the flow
8	system up time when the first packet of the flow was received
9	system up time when the last packet of the flow was received
10	TCP/UDP source port number
11	TCP/UDP destination port number
12	IP protocol type
13	IP type of service (ToS)
14	Cumulative logical OR of TCP flags
15	Source AS number
16	Destination AS number
17	Source Mask
18	Destination Mask

リ秒）である。観測したフローがTCPまたはUDPに関するフローであった場合、10、11の送信元、送信先のポート番号が取得できる。12は、IPプロトコルタイプ（ICMP/TCP/UDPなど）、13はIPのType of Service値である。TCPに関するフローの場合は、14のTCPフラグの累積論理和が取得できる。ルータがEGPを用いてルーティングをしている場合は、15、16の送信元と送信先のAS番号を取得できる。17、18は送信元IPアドレス、送信先IPアドレスのアドレスマスク長である。Version 5は、Version 1を元に、取得可能な情報を増やしたものである。また、Version 9では、NetFlowパケットの構造が可変長となり、取得できる情報が拡張可能となっている。

一方、sFlowはInMon Corporation社[134]が提唱したトラフィック観測技術であり、RFC3176[248]で標準化されている。パケットの集約はせず。サンプリングした個々のパケットについてその情報をsFlowパケットとして出力する点がNetFlowとは異なる。得られる情報はおおむねNetFlowで得られるものと同様である。オプションとしてパケットのペイロードの一部を取得することができ、シグニチャマッチング方式との併用などについても研究がなされている。

NetFlow、sFlowともに、サンプリングを組み合わせることにより広帯域のネットワークリンク上でもトラフィック情報取得が可能であるという利点がある。また、規格の標準化により対応機器が増えている。NetFlowやsFlowを用いることで、既存のネットワークに変更を加えることなく、低コストでトラフィック情報の取得ができる可能性が高い。

IDS (Intrusion Detection System)

IDSとは、侵入検出システムという意味である。IDSにはネットワーク型IDS、インライン型IDSおよび、ホスト型IDSの3種類がある。ネットワーク型IDSおよびインライン型IDSの監視対象はトラフィックであり、ホスト型IDSの監視対象はホストの状態である。本章では、異常トラフィックを検出するという観点から、ホスト型IDSについては言及せず。ネットワーク型IDSおよびインライン型IDSを単にIDSと表記する。

IDSは、狭義では悪意のある攻撃者によるコンピュータやネットワークへの侵入を検出するシステムである。しかし、ワームによる攻撃トラフィックや特定のアプリケーションによるトラフィックの検出にも広く利用される。IDSにはシグニチャ型IDSとアノマリ検出型IDSの2種類がある。

シグニチャ型IDS

シグニチャ型IDSは、ミスユース型IDSと呼ばれることもある。シグニチャ型IDSは、事前にワームや特定のアプリケーション固有のペイロード、異常なTCPフラグなどの知識を検出シグニチャとして与えることで、該当するトラフィックが観測された際に管理者に通知することができる。このIDSは、シグニチャにマッチする異常トラフィックであれば確実に検出できるという利点がある。

アノマリ検出型IDS

アノマリ検出型IDSは、プロファイル型IDSと呼ばれることもある。アノマリ検出型IDSは、本稼働前に学習期間が必要である。学習期間中に観測したトラフィックの状態を正常と仮定し、その状態から外れたトラフィックが発生したときに管理者に通知する。

異常認知手法

ネットワーク管理者が、異常トラフィックに対処するためには、まず、異常トラフィックの発生を知る必要がある。ここで、管理者が異常トラフィックの発生を知るための手法を異常認知手法と呼ぶ。ユーザからの報告以外の異常認知手法として能動的手法、受動的手法がある。能動的手法とは、システムが提供する情報に対して管理者が何らかの思考をし、管理者が異常かどうかを判断する手法である。管理者がトラフィックパターンの推移グラフを見て、通常量を超えた上昇や下降、スパイク（急上昇部分）などを探す方法がこの手法の例である。

一方、受動的手法では異常であるか否かの判断主体はシステムであり、システムが異常と判断した出来事のみを管理者に通知する。IDSで使われる管理者への通知はこの手法にあたる。

能動的手法では、管理者が定期的にシステムの出力を調査する必要があり、管理者への負荷が高い。また、調査間隔を短くすればするほど管理者の負担が増すため、リアルタイムな異常検出は難しい。さらに、判断主体は管理者であり、どの程度の急激な変化やスパイクを異常とするかは、管理者個人の経験や

知識に依存するという欠点がある。受動的な手法は、管理者の負担は小さい。しかし、システムによる異常判断の元となるしきい値の設定が難しい。しきい値を低く設定すると False-Positive 率[2]が上昇する。しかし、逆に高くすると、False-Negative 率[3]が上昇する。一般に、False-Positive 率も False-Negative 率も最小限になる適切なしきい値の算出には長期の試行錯誤が必要となる。

しきい値算出技術

先に挙げた受動的異常認知手法を実現するためには、システムは正常、異常の判断機能を有してなければならない。シグニチャ型 IDS のように、トラフィックがシグニチャにマッチするかどうかの判定であれば容易に実現できる。しかし、トラフィックパターンの量的変化に対して正常か異常かの判断を下すことは容易ではない。

トラフィックの正常状態と異常状態のしきい値を自動的に算出し、精度の高い通知を実現するためにさまざまな手法が検討されている。文献 [14] では、Wavelet 変換と Holt-Winters 法を用いて異常トラフィックの自動検出を試みている。Wavelet 変換を用いて転送バイト量の時間変化を各周波数成分に分離し、成分ごとの偏差からしきい値を算出する手法と Holt-Winters 法を用いて算出する手法を比較している。いずれの手法も転送バイト量の時間変化を指標としており、転送バイト量に変化を生じさせる現象については高い精度で検出ができている。しかし、検出された現象が必ずしも管理者が必要とする異常の通知であるかどうかは十分に評価されていない。

3.2.2 既存手法

既存の異常トラフィック検出手法は、大きくわけ、以下の 4 種類がある。

(i) 可視化された転送バイト量、パケット数から、管理者が能動的に異常を発見
(ii) しきい値から外れた転送バイト量、パケット数を異常として受動的に発見
(iii) シグニチャ型 IDS を利用して受動的に発見
(iv) アノマリ検出型 IDS を利用して受動的に発見

(i) および (ii) の手法は、MRTG[216] や FlowScan[100] を利用することで実現できる。MRTG と FlowScan は、観測対象ネットワーク上のルータを通過した転送バイト量、転送パケット数の時間変化をグラフ化できる。図 3.1、図 3.2 に MRTG と FlowScan の動作例を示す。P2P ファイル共有アプリケーションは転送バイト量に、ワームの攻撃は転送パケット数に変化をもたらすことがある。したがって、管理者が MRTG の出力結果を目視し、通常時との変化や急上昇を発見することで異常トラフィックを発見できる可能性がある。管理者はこれらのグラフをから、3.2.1 項にて述べた能動的な手法を用いることで、異常トラフィック発生の検出に利用できる。

FlowScan は NetFlow で取得可能な通信元／通信先のポート番号の情報を利用し、使用ポート番号別の転送バイト量やパケット数の変化をグラフ化することができる。つまり、特定のポート、たとえば HTTP で使用されることが多い 80/tcp ポートのみに注目し、その流量の変化から異常の有無を探ることができる。

(ii) の手法は手法 (i) の通知方法を能動的から受動的に変更したものである。MRTG と FlowScan は図 3.3 のように、あらかじめ上下限しきい値を設定することにより、観測値がその値を上回ったとき、または下回ったときに管理者に通知を送信することができる。管理者が目視や判断をすることなく異常トラフィックの候補を管理者に通知することができるため、運用コストを削減することができる。

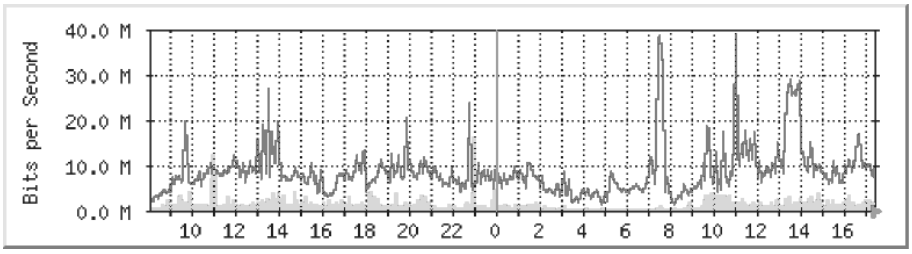

図 3.1. MRTG の動作例

2　正常状態を異常と通知する現象の発生率
3　異常状態を正常と判断し通知しない現象の発生率

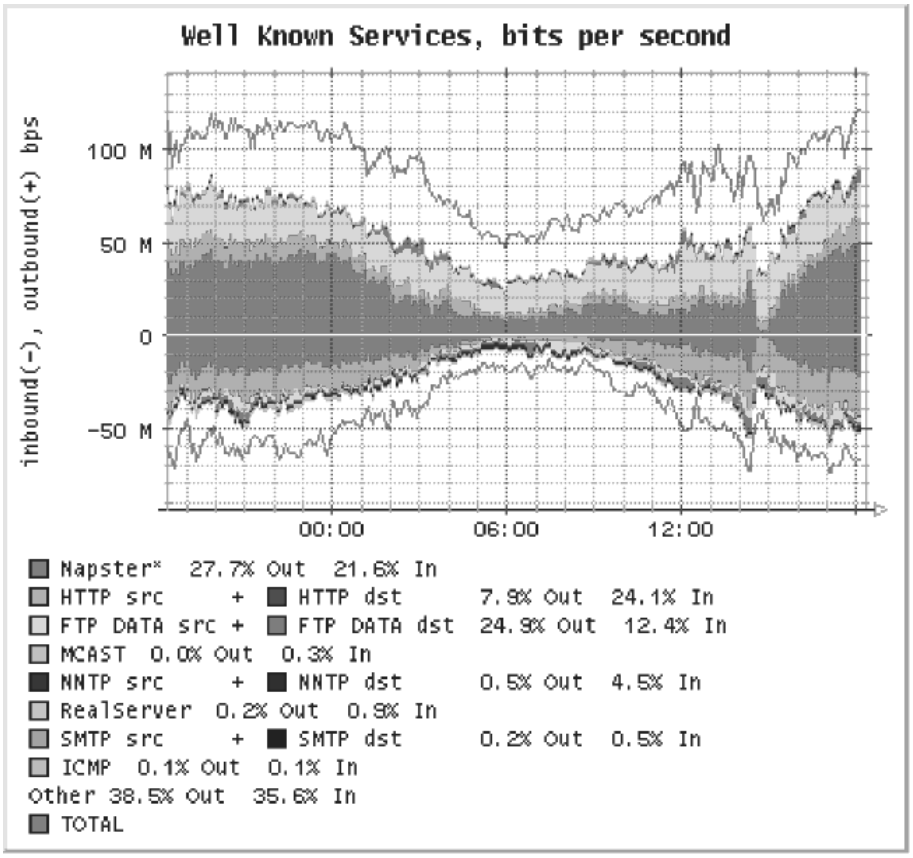

図 3.2. FlowScan の動作例

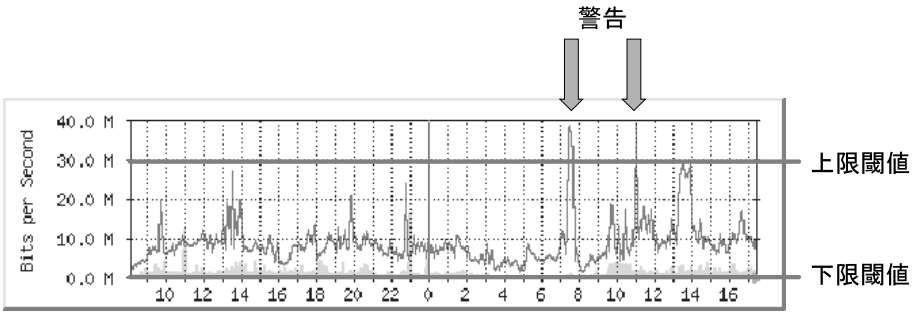

図 3.3. MRTG のしきい値設定と通知

(iii) および (iv) の手法は、3.2.1 項で述べた IDS による通知を利用し、異常トラフィックを検出する手法である。

3.2.3 既存手法の問題点

3.2.2 項で挙げた手法 (i) では、転送バイト量、または転送パケット数に注目し、その変化から異常トラフィックの検出を試みている。MRTG はトラフィック情報の取得に SNMP を用いており、ほかのトラフィック情報を用いることはできない[4]。

異常トラフィックが転送バイト量やパケット数に変化を与えることはある。しかし、必ず識別可能な変化を与えるとは限らない。また、広帯域なネットワークでは、単に 1 人のユーザがファイルをダウンロードしただけで転送バイト量やパケット数は大きく変化する。つまり、これらの指標を用いた異常ト

4 厳密にはほかの情報を取得可能である。しかし、個々のパケットに関する情報など、異常トラフィックの検出に利用可能な情報は提供されない。

ラフィック検出は精度が低いという問題点がある。異常トラフィックが利用するポート番号が事前にわかっている場合は、FlowScan を用いて特定のポートを監視することで、全体の転送バイト量やパケット数を見た場合より、比較的高い精度の異常トラフィック検出ができる可能性がある。しかし、ランダムなポート番号を使う P2P ファイル共有アプリケーションョンや、未知のワームの検出などには利用できない。また、管理者が能動的にシステムの出力を監視しなければならず、システムの日々の運用コストは高くなる。

手法 (ii) では、しきい値の設定により通知の送信が自動化される。管理者が能動的にシステムの出力を監視する必要はなく、運用コストは削減できる。また、異常の発生から認知までの時間を短縮することができる。しかし、しきい値を管理者が一元的に設定しなければならない。適切な上下限しきい値の設定には、管理者の知識や経験、さらに事前の長期観測データが必要となり、導入コストは高くなる。また一元的に設定したしきい値では、トラフィックの周期性などの特徴が考慮されず、早朝のトラフィックでも昼間のトラフィックも同じ基準で判断するという問題点がある。

手法 (iii)、(iv) は、IDS の問題点をそのまま継承する。まず、シグニチャ型 IDS は、3.1 節で述べたようにシグニチャが対応しない未知の異常トラフィック、暗号化されたペイロードを用いるアプリケーションの検出は不可能という欠点がある。また、アノマリ検出型 IDS は、正常なトラフィック状態をどのパラメータを用いて定義し、正常状態と異常状態のしきい値をどのように設定するかによって False-Positive 率、False-Negative 率は大きく変化する。しかし、一般に False-Positive 率、False-Negative 率が最小になるアルゴリズムが知られているわけではなく、特定のトラフィックを高精度に検出するためには、対象ネットワークに合わせた専門家によるチューニングが必要という欠点がある。正常と仮定された状態と外れたトラフィックが、必ずしも管理者が検出したい異常とは限らないという問題点もある。

また、IDS に共通した問題点もある。ネットワーク型 IDS はトラフィック情報取得のためにミラーリングを用いる。一方、インライン型 IDS は観測対象トラフィックが流れる経路上に設置され、直接対象トラフィックを検査する。いずれのタイプでも、観測対象トラフィックに含まれるパケットすべてを検査し、シグニチャに該当するトラフィック、または、

表 3.3.　Snort 性能評価 PC スペック

CPU	PentiumIII 1.2 GHz
Memory	512 MB
Network Interface	1 Gbps (DP83820 Chip)
OS	Slackware9.1 (Linux2.4.22)
Snort	snort2.20 ruleset 2004/11/11 snapshot

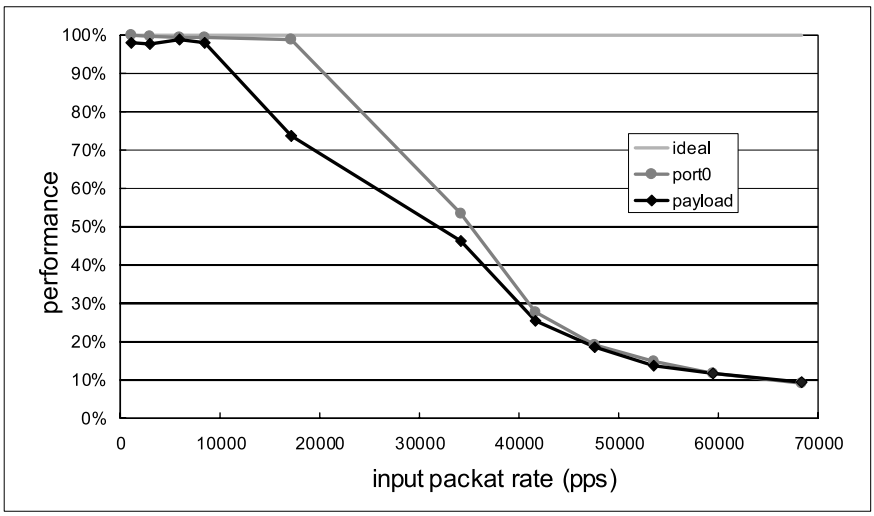

図 3.4.　IDS の限界例

正常状態から外れたトラフィックがないかを調査する。その結果、観測対象のパケットレートと比例してIDS装置の負荷は上昇する。限界以上の負荷がかかると、調査処理が間に合わず正確な検出が期待できない。つまり、IDS装置は低帯域ネットワークから広帯域ネットワークまで、スケーラブルに対応できない。

図3.4は、表3.3に示したPC上での、代表的なネットワーク型、シグニチャ型IDSであるSnortの性能評価結果を示している。x軸が入力パケットレートを示す。y軸は、入力したシグニチャにマッチするパケット数に対して、実際に検出できたパケットの率（検出率）を示している。スケーラブルに異常が検出できれば、idealに示したように検出率は100%となることが期待される。しかし実測値は、ヘッダのみの検査で検出できる異常（port0）、ペイロードの検査が必要な異常（payload）のいずれも、入力パケットレートが増加するにつれ検出率が下がることが分かる。

3.3 機能要件と提案手法の目標

本節では、前節で述べた既存手法の問題点を踏まえ、優位な異常トラフィック検出手法に必要な要件を述べる。

3.3.1 機能要件と既存手法

3.2.3項で述べたように、シグニチャを用いた手法や広帯域にスケーラブルに対応できない異常トラフィック検出手法には問題がある。また、転送バイト量やパケット数に注目した手法は検出精度が低く優れていない。導入コストや運用コストが高い手法も問題がある。

この点を考慮すると、異常トラフィック検出手法には次の要件が必要となる。

(a) 低コストで利用を開始できる
(b) 検出シグニチャに依存しない
(c) 自動で異常トラフィックを検出可能である
(d) 高い精度で異常トラフィックを検出可能である
(e) 広帯域環境下で使用可能である
(f) リアルタイムに異常トラフィックを検出可能である

まず、要件(a)として、「低コストで利用を開始できる」という要件を挙げる。異常トラフィックの検出に際し、ネットワークトポロジの変更や高価な機器の導入が必要な手法、あるいは、管理者に特殊な技能が必要な手法は、利用開始にかかるコストが高く問題である。

要件(b)として、「検出シグニチャに依存しない」という点が挙げられる。3.1節で述べたように、シグニチャに依存する手法には問題点が多い。シグニチャなどの事前知識に依存せずに、異常トラフィック検出ができる必要がある。

要件(c)として、「自動で異常トラフィックを検出可能である」という要件を挙げる。管理者が能動的手法を用いて異常トラフィックを検出しなければならない手法も適切でない。なぜなら、管理者の能動的手法に依存する手法は、特定の経験や知識を持つ管理者にしか利用できず、管理者の手間も大きいからである。

要件(d)に、「高い精度で異常トラフィックを検出可能である」という要件を挙げる。検出精度が悪く、False-Positiveや、False-Negativeが多発する手法は、信頼できず、まったく役に立たない。

次に要件(e)の「広帯域環境下で使用可能である」という要件について述べる。今日、一般ユーザにギガビットネットワークが普及し、10 Gbpsのバックボーンネットワークも一般的になりつつある。そして、今後も、ネットワーク帯域の広帯域化は促進されると考えられる。こうした環境下では、低いパケットレートしか扱えない異常トラフィック検出手法は役に立たない。異常トラフィック検出手法は、低帯域環境下でも広帯域環境下でもスケーラブルに対応できることが望まれる。

最後の要件(f)に、「リアルタイムに異常トラフィックを検出可能である」という要件を挙げる。異常の発生から管理者がそれを認知するまでに長い時間を要する手法は、その異常への対処が遅れる結果となり不適切である。

3.2.2項に挙げた既存手法と機能要件を比較すると表3.4のようになる。

まず、要件(a)を見た場合、導入時に適切なしきい値を管理者が算出する必要のある手法(ii)は優れていない。また、製品の導入やミラーリングのためのネットワークトポロジ変更が必要となる手法(iii)、(iv)も優れていない。次に要件(b)を見ると、異常トラフィックの検出をシグニチャに依存している手法(iii)は不適切である。要件(c)を見ると、管理者が能動的に異常トラフィックの発見および判断をし

表 3.4.　既存手法と機能要件

		既存手法			
		(i)	(ii)	(iii)	(iv)
機能要件	(a)	○	×	×	×
	(b)	○	○	×	○
	(c)	×	○	○	○
	(d)	×	×	△	△
	(e)	○	○	×	×
	(f)	×	○	○	○

なければならない (i) の手法は優れていない。次に、要件 (d) を見ると、(i) および (ii) の手法は、転送バイト量やパケット数に変化を与えるトラフィックしか発見できず、その変化が必ずしも異常トラフィックとは限らないので不適切となる。(iii) の手法はシグニチャにマッチする異常は確実に検出できる反面、未知の異常や暗号化されたトラフィックに対しては効果を発揮できず、△とした。また、高精度な検出には専門家によるチューニングが必要な (iv) の手法も△とした。要件 (e) について述べると、ミラーリングなどの手法とともに、すべてのパケットを精査する必要のある (iii) および (iv) の手法は優れていない。要件 (f) について言及すると、異常トラフィックの発生から、管理者がそれを認知するまでに時間を要する (i) の手法は不適切である。

3.3.2 提案手法の目標

提案手法は、前節に挙げた機能要件をすべて満たす異常トラフィック検出手法を目指す。

まず、提案手法では、既に多くのルータやスイッチに実装されている NetFlow や sFlow を用いてトラフィック情報を取得する。この特徴により、低いコストで検出を開始できる（要件 (a)）。また。NetFlow や sFlow はパケットサンプリングと併用できるため、広帯域への適応性を実現する（要件 (e)）。そして、異常トラフィックの一般的な特徴を考慮した指標を用いることで、シグニチャに依存しない検出を実現する（要件 (b)）。そして、Holt-Winters 法（3.4.5 項参照）を利用し、異常トラフィックと正常トラフィックのしきい値を自動算出することで、要件 (c) を実現する。

高い精度で異常トラフィックを検出可能（要件 (c)）か否か、また、サンプリングレートを下げ、広帯域に適応させた時にその検出制度を維持できるか（要件 (d)）、そして、リアルタイムな異常トラフィック検出を実現できる（要件 (f)）か否かは、3.6 節にて明らかにする。

3.4 提案手法

本節では、提案する異常トラフィック検出手法について、その対象環境と詳細な特徴、および、アルゴリズムについて述べる。

3.4.1 対象環境

提案手法が異常トラフィック検出の対象としている環境は、組織内ネットワークである。ここでいう組織とは、企業や学校を想定しており、以下対象組織と呼ぶ。これは、3.1 節で定義した異常トラフィックの検出を必要としているのは、対象組織のネットワーク管理者であるからである。一方、インターネットサービスプロバイダや IX（Internet eXchange：インターネットサービスプロバイダの相互接続ポイント）が管理するネットワークは、一般に、トラフィックの転送だけに気を配り、そのトラフィックが異常であるかどうかについては関知しない。したがって、企業や学校外のネットワークは対象環境としない。

3.4.2 異常トラフィックの一般的な性質

3.1 節で定義した、異常トラフィックの一般的な性質を示す。

ワームの一般的な動作を図 3.5 に示す。ワームはより多くのホストに感染しようとする性質を持つ。ワームに感染したホストは、ほかのホストの脆弱性を攻撃パケットにより攻撃し、そのホストに感染する。新たに感染したホストは同様にほかのホストの脆弱性を攻撃する。この動作が再帰的に行われることで、ワームの感染は爆発的に広まる。攻撃先ホストの選択方法はワームによって異なり、ランダムに選んだアドレスに対して攻撃するワームもあれば、自ホストのアドレスの周辺アドレスから攻撃を開始するワームもある。しかし、より多くのホストに感染しようと、多数のホストに攻撃パケットを送信する点は共通である。

P2P ファイル共有アプリケーションの一般的な動作を図 3.6 に示す。一般的な P2P ファイル共有アプリケーションは、P2P ネットワークに接続されているホストのアドレス情報、各ホストが提供している共有ファイルの情報などを各ホストに分散管理して

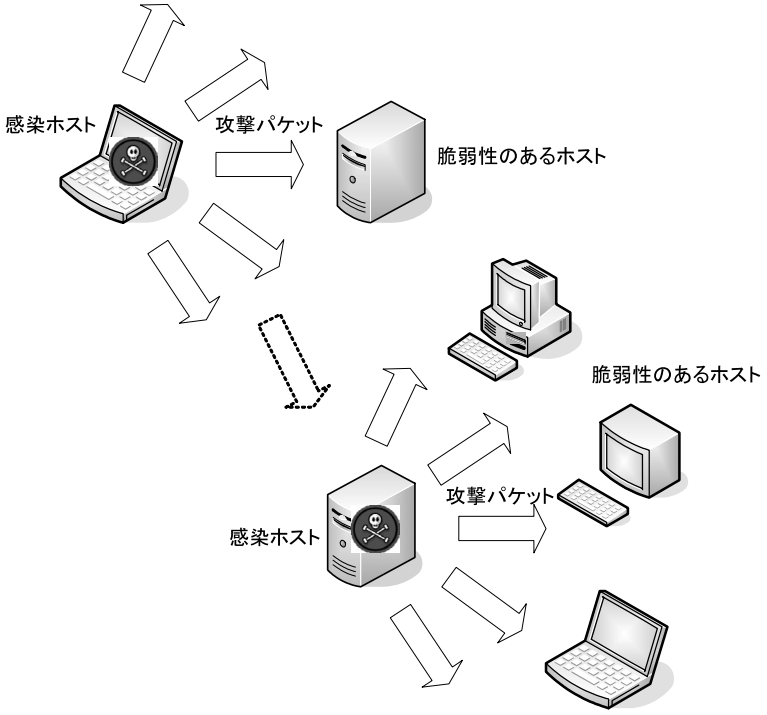

図 3.5. ワームの一般的な動作

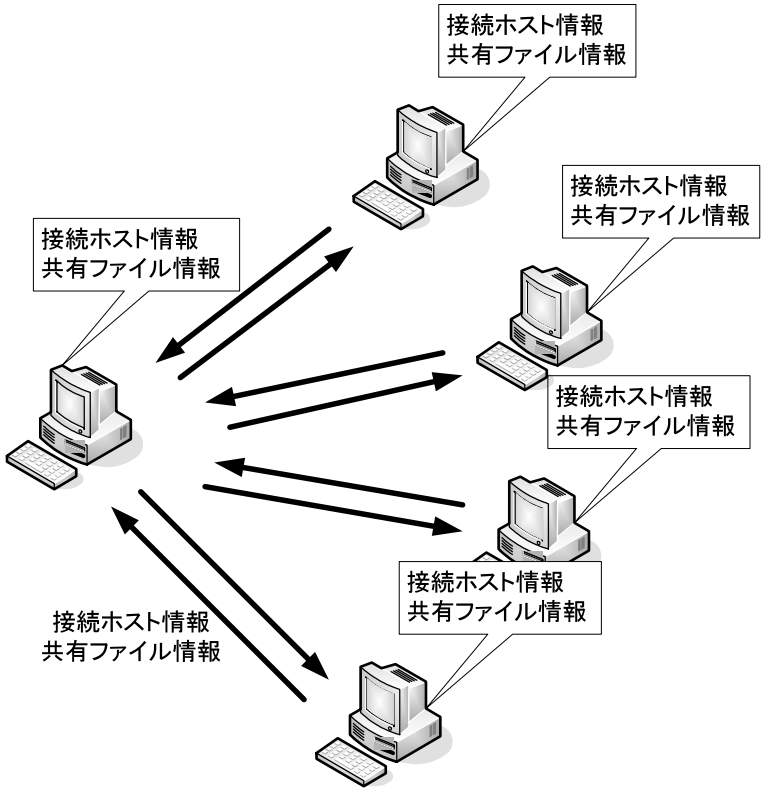

図 3.6. P2P ファイル共有アプリケーションの一般的な動作

いる[5]。また、共有されているファイルの実体が複数のホストに分散配置される P2P ファイル共有アプリケーションもある。こうした中、P2P ファイル共有アプリケーションはユーザが要求したファイルを検索し、ダウンロードするために分散管理された情報を収集する必要がある。そのため、P2P ファイル共有アプリケーションは多数のホストと通信をするという性質がある。

3.4.3 注目指標

IDS 以外の異常トラフィック検出手法および関連研究は、通過トラフィックのバイト流量、パケット流量の変化に注目している。この理由は、歴史的にこれらの値が SNMP などを用いて低コストで取得ができたからである。しかし、3.2.3 項で述べたように、この値に注目することが異常トラフィック検出に最適であるとはいえない。

一方、はじめに定義した異常トラフィックは 3.4.2 項で示した性質を持つため、共通して単位時間あたりのユニークな（重複を除いた）通信先ホスト数（以下、注目指標）が上昇する。この注目指標は、単位時間に観測された重複を除いた通信先ホスト数である。

たとえば、表 3.5 に示した始点（source）と終点（destination）を持つ、4 つのトラフィックが観測されたとする。注目指標は観測された重複を除いた通信先ホスト（destination host）が、「192.168.0.1」、「192.168.0.3」の 2 つなので、2 と求められる。

表 3.5.　注目指標のカウント例

Flow ID	source(host:port)		destination(host:port)
1	10.0.0.1:1025	→	192.168.0.1:80
2	10.0.0.2:1025	→	192.168.0.1:80
3	10.0.0.1:1026	→	192.168.0.3:80
4	10.0.0.1:1025	→	192.168.0.1:443

注目指標はファイルのダウンロード、メールのやりとりなど、サーバクライアント型サービスによる変動は少ない。組織内のホストが、Web サイトや FTP サイトから巨大なファイルをダウンロードしても、値の変化は 1 である。また、多くの組織内ホストからアクセスされる Web サイトやメールサーバなどが及ぼす影響も（アドレスベースの負荷分散が行われてない限り）たかだか 1 である。

本章では、異常トラフィックによって大きく変動し、ほかのトラフィックによる変動が少ない注目指標の時間変化を用いて異常トラフィックの検出を試みる。ただし、本手法は、3.4.1 項で示したように、対象組織内の異常トラフィックを発生させているホストを見つける目的での利用が想定される。したがって、提案手法は、上り（対象組織内から組織外へ向かう）トラフィックに対して注目指標を利用する。

3.4.4 指標の周期性と適合モデル

注目指標の時間変化は、観測対象ネットワークを利用する人々の生活周期と深く関連し、周期性を持つ。奈良先端科学技術大学院大学（以下本学）で観測された注目指標の時間変化は図 3.7 のようになる。昼間にピークを迎え早朝が谷となり、土日は平日に比べて値が小さい傾向があることがわかる。また、その自己相関関数は図 3.8 のようになり、24 時間周期、1 週間周期に強い周期性が見られる。一般的な組織の活動形態を考慮すると、多くの組織で似た傾向が見られる。しかし、周期性があるとはいえ注目指標にはばらつきがある。たとえば、先週の注目指標と今週の注目指標は似ている可能性は高いがまったく同じではない。また、注目指標は個々のアプリ

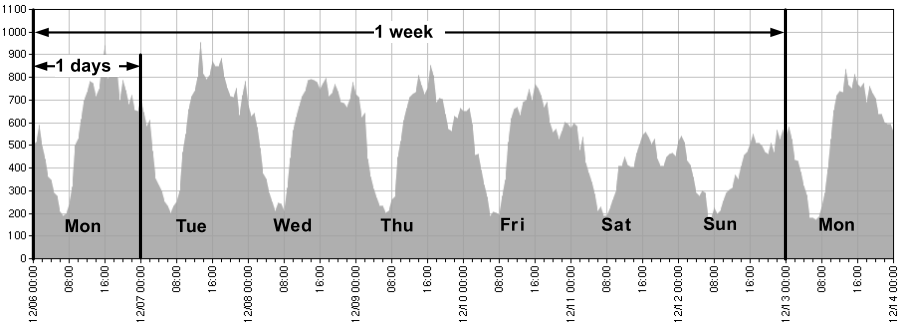

図 3.7.　注目指標の時間変化の周期性

5　一部の情報が特定のサーバホストで集中管理されるハイブリッド型と呼ばれる形態もある。

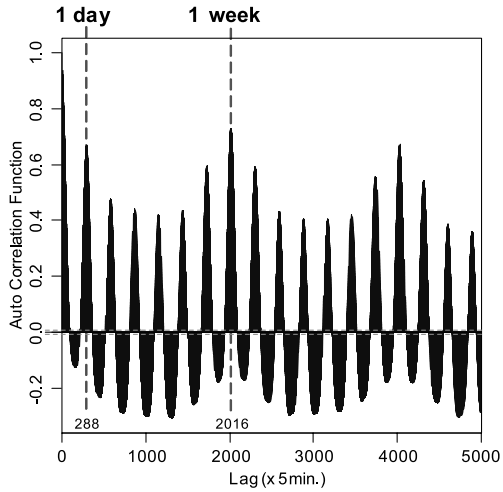

図 3.8. 注目指標の時間変化に対する自己相関関数

ケーションの動作により支配され、アプリケーションには通信継続時間がある。したがって、必然的に現在の注目指標は、1 時間前の注目指標よりも 5 分前の注目指標により似ている可能性が高い。そして、数ヶ月以上の長期を観察した場合、利用されるアプリケーションのパラダイム変化により注目指標が変化する。こうした特徴を持つ注目指標に対するしきい値計算には Holt-Winters 法が適しており [28]、本手法では Holt-Winters 法と呼ばれる手法を用いる。

3.4.5 Holt-Winters 法

Holt-Winters 法は、需要予測分野で実績のある予測手法であり、季節変動のある需要予測に最適である。過去の実測値と各種パラメータを与えることで、次の単位時間における予測値を算出することができる。

まず、Holt-Winters 法は、指数平滑法（Exponential Smoothing）という手法を元に拡張したものである。指数平滑法では、ある時刻（$t+1$）における予測値（$\hat{y}_{t+1}$）を次の式で与える。

$$\hat{y}_{t+1} = \alpha y_t + (1-\alpha)\hat{y}_t \tag{12}$$
$$= \hat{y}_t + \alpha(y_t - \hat{y}_t) \tag{13}$$

y_t は直前時刻（t）における実測値を、$\hat{y}_t$ は直前時刻（t）に対する予測値を示す。n 単位時間前の観測値が最新の予測値に与える影響が n に対して指数関数的に減少するという特徴から、指数平滑法と呼ばれる。

パラメータ α は、$0 < \alpha < 1$ の値をとる。α を 1 に近づけるほど直前の観測値を、0 に近づけるほど長期の観測値の優先度を重視して予測値が算出される。図 3.9 に α による予測値の差を示す。実線が観測値、点線が予測値を示す。α が 1 に近づくにつれ直前値を重視した予測が、0 に近づくにつれ長期の変動を重視した予測がなされていることが分かる。1 に近づけすぎると、特異点が予測値に大きな影響を与え、適切な予測ができない場合がある。また、0 に近づけすぎると、観測値の変動に対する予測値の追随が遅れるという問題点がある。

Holt-Winters 法は、指数平滑法に周期変動の概念を導入したものである。Holt-Winters 法では、ある時刻（$t+1$）における予測値（$\hat{y}_{t+1}$）を次の式で与える。

$$\hat{y}_{t+1} = a_t + b_t + c_{t+1-m} \tag{14}$$
$$a_t = \alpha(y_t - c_{t-m}) + (1-\alpha)(a_{t-1} + b_{t-1}) \tag{15}$$
$$b_t = \beta(a_t - a_{t-1}) + (1-\beta)b_{t-1} \tag{16}$$
$$c_t = \gamma(y_t - a_t) + (1-\gamma)c_{t-m} \tag{17}$$

ここで、m は 1 周期の単位時間数である。たとえば、単位時間が 10 分で、周期が 1 週間であれば $m = 1008$ である。c_t（Seasonal Trend：周期傾向）は、周期ごとの傾向を把握するために用いられる。数

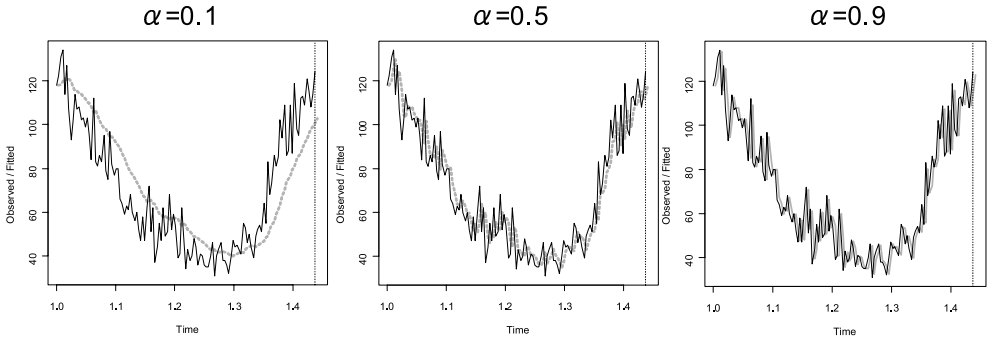

図 3.9. α による予測値の差

式 17 では、周期間（周期が 1 週間の場合、同曜日同時刻）の観測値間で指数平滑法を適応している。b_t（Liner Trend：線形傾向）は、観測値が上昇傾向か下降傾向かを把握するために用いられる。a_t（Baseline：基線）は、周期傾向、線形傾向による影響を除外した変動の把握に用いられる。

α は基線において、β は線形傾向において、γ は周期傾向において、それぞれ直近の観測値を重視してしきい値を求めるか、長期の観測値を重視して求めるかを設定するパラメータである。1 に近いほど直近の実測値を重視し、0 に近いほど長期の実測値を重視する。

また、Holt-Winters 法は信頼水準を与えることで、単一の予測値だけではなく、実測値が確率的にとりうる範囲を算出することができる。

3.4.6 Holt-Winters 法の注目指標への適応

Holt-Winters 法の入力として、注目指標の過去の実測値と信頼水準を与える。そうすることで、次の単位時間で実測値がとりうる予測範囲を算出し、その予測範囲をしきい値の内側とする。次の単位時間の実測値がしきい値外であった場合、異常トラフィック発生の候補時間とする。

また、需要予測分野では周期を 1 年と設定することが多い。しかし、注目指標の周期は、3.4.4 項に示した特徴から、1 日または 1 週間を周期と設定することが妥当である。1 日を周期と設定した場合、平日と土曜日、日曜日という特性が異なる周期が混在し、予測精度が悪化する。一方で、1 週間周期とするとこの点は改善されるが、しきい値計算に必要な事前の実測値の量が増え、計算に必要な時間が長くなる。しきい値計算に必要な時間は 3.6 節で明らかにするとおり、リアルタイムなトラフィック異常検出に十分利用可能な短さである。したがって、本章では、周期を 1 週間と設定した。

3.4.7 しきい値計算アルゴリズムと異常トラフィック検出

しきい値計算と異常トラフィック検出の具体的なアルゴリズムは次のようになる。

単位時間を N 秒とした場合、現在時刻から過去 M 秒に対して N 秒ごとの注目指標の変動を取得する。取得した値に Holt-Winters 法を適応し、次の N 秒に対する上限しきい値、下限しきい値を設定する。次の N 秒間に実際に観測された注目指標が、設定されたしきい値内に収まらなかった場合。つまり、上限しきい値を上回った場合、または下限しきい値を下回った場合に、異常トラフィック発生の可能性ありとして管理者に通知する。この作業を N 秒ごとに実行し、しきい値内に収まらなかった実測値を通知することで、リアルタイムな異常トラフィック検出を実現する。この過程を図 3.10 に示す。

ここで、M はトラフィック計測を開始してから提案手法が利用可能になるまでの期間となる。あまりにも M を長くすると、運用現場において実用的ではない。しかし、周期変動を正しくとらえるために、設定した周期の数倍以上の長さは必要となる。この点を考慮し、本章では、M を 28 日（2419200 秒、4 周期）とした。

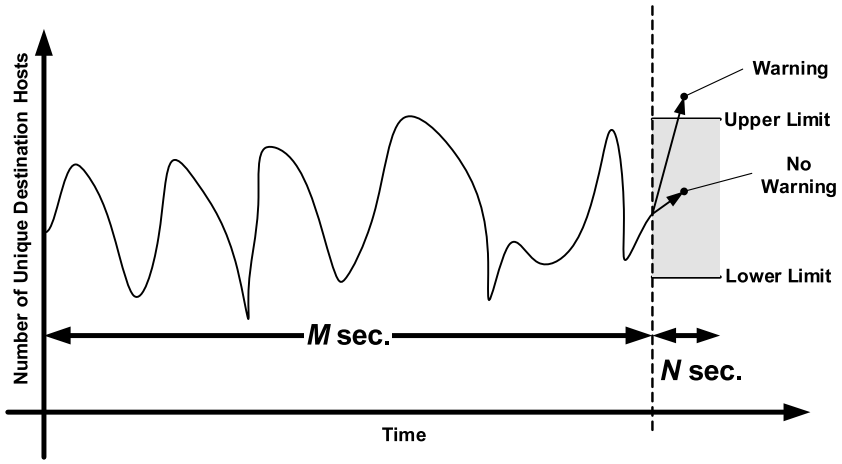

図 3.10.　異常トラフィック検出手法

3.4.8 Holt-Winters 法のパラメータ

Holt-Winters 法によるしきい値設定には、信頼水準 CL、α、β、γ の 4 つのパラメータが必要となり、$0 < CL$、α, β, $\gamma < 1$ である。CL はしきい値内に次の実測値が含まれる確率を示し、1 に近いほど上限しきい値と下限しきい値の差は大きくなる。信頼水準を 1 に近づければ近づけるほど、False-Positive 率が減り、False-Negative 率が上昇する。

α、β、γ は、直近の観測値を重視してしきい値を求めるか、長期の観測値の変動を重視して求めるかを設定するパラメータである。1 に近いほど直近の実測値を重視し、0 に近いほど長期に実測値を重視する。1 に近い値を利用すれば、トラフィックパターンの急激な変化にしきい値が順応しやすくなる。これは、異常トラフィックが発生が継続している状況を許容するしきい値が生成される可能性が高いことになる。逆に、0 に近い値を設定すれば正常なトラフィックパターン変化に対するしきい値が順応が遅れ、結果的に False-Positive 率が上昇する。この点を踏まえてパラメータを調整する必要がある。中でも、α は周期より短い変動を、β は周期より長い変動を、γ は周期ごとの変動をとらえるために利用する。

具体的な値を求める式として、

$$\alpha,\ \beta = 1 - \exp\left(\frac{\ln(1 - weight)}{n}\right) \qquad (18)$$

$$\gamma = 1 - \exp\left(\frac{\ln(1 - weight)}{m}\right) \qquad (19)$$

が利用可能である。$weight$ は重み（$0 < weight < 1$）を、n は単位時間数、m は周期数を指す。

たとえば、$N = 300$（10 分）として過去 1 時間（$n = \frac{60}{10} = 6$）の観測値に 50%の重み（$weight = 0.5$）を置く場合、$\alpha \simeq 0.109$ となり、過去 2 週間（$n = \frac{60 \times 24 \times 7 \times 2}{10} = 2016$）の値に 80%の重み（$weight = 0.8$）を置く場合、$\beta \simeq 0.008$ となる。また、周期を 1 週間と取り、1〜2 週間前の同曜日、同時刻（$m = 2$）の値に 80%の重み（$weight = 0.8$）を置く場合、$\gamma \simeq 0.55$ となる。

3.4.9 パラメータの自動算出

統計処理ソフトウェアによっては α、β、γ の値を自動算出できるものもあり、自動算出値を利用する方法もある。本提案手法の実装では統計処理ソフトウェア GNU R[110] の Holt-Winters パッケージを用いている。このソフトウェアでは、α、β、γ の値を明示的に与えなかった場合、過去の予測値と実測値の差の平方の総和（数式 20）が最小になるように自動算出される。

$$\sum_{t=1}^{n} (y_t - \hat{y}_t)^2 \qquad (20)$$

3.5 実装

本節では、前節で示した提案手法の実装について述べる。

3.5.1 ベースとなるシステム

提案手法を実装する土台として、図 3.11 に示すシステム（以下本システム）を実装した [362, 366]。本システムはトラフィック情報取得モジュール（Interface Module）、データベース（Database）、トラフィック情報解析モジュール（Analysis Module）、トラフィック情報可視化モジュール（Visualize Module）、レポートモジュール（Report Module）に分類される。まず、観測対象ネットワークリンク上の NetFlow

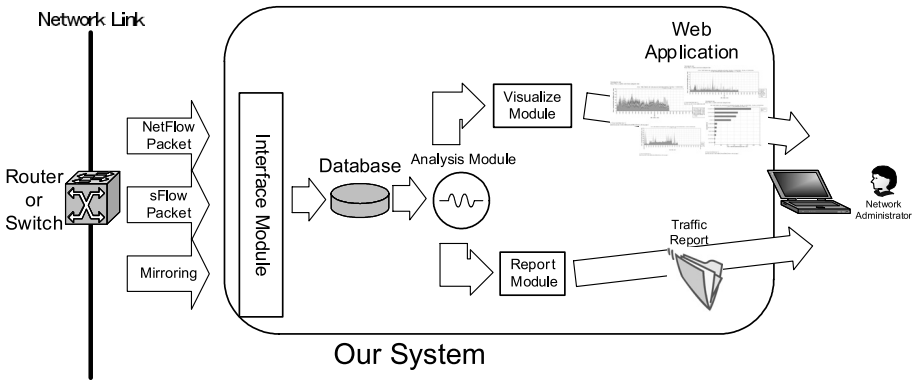

図 3.11. ベースとなるシステムの概要図

を利用可能なルータでトラフィック情報を取得し、NetFlow パケットを本システムに対して出力する。取得モジュールでは、NetFlow パケットに含まれる各フローの属性情報を分離し、データベースへ蓄積する。この動作により、過去に観測されたトラフィックの情報がデータベースに蓄積される。本章では、NetFlow によるトラフィック情報取得を前提とする。しかし、ミラーリングしたトラフィックをそのまま扱える pcNetFlow[241]、sFlow パケットを入力として扱えるモジュールも開発中である。提案手法は、トラフィック情報解析モジュールとして実装する。本モジュールでは、蓄積された情報に対して、さまざまな解析処理を行う。この処理は、一般的な統計処理から、本章で提案するしきい値設定、信号処理などである。トラフィック情報可視化モジュールは Web アプリケーションにより、解析したトラフィック情報を可視化することで、管理者の能動的異常認知をサポートする。レポートモジュールでは、トラフィック情報解析モジュールにより発見された異常トラフィックや各種通知を管理者に主にメールで通知する。

3.5.2 実装環境

本システムは、一般的な PC で動作する Linux 上に実装した。使用した PC のスペックは表 3.6 の通りである。データベースサーバとしても利用することから、ディスク容量は余裕を持って確保されている。しかし、実際のフロー属性データの容量は 10 ヶ月分で 10 GB 程度であり、近年の一般的なコンピュータのハードディスク容量を考慮すると、十分、数年以上の長期観測に利用が可能である。

表 3.6.　実装環境

CPU	Intel Xeon CPU 2.80 GHz × 2
Memory	4 GB
HDD	400 GB
Network Interface	1 Gbps (BCM5703 Chip)
OS	Mandrake Linux 9.2 (Linux2.4.22)

トラフィック情報取得モジュールは C 言語を用い、NetFlow2MySQL として実装した。データベースには MySQL を利用した。トラフィック情報解析モジュールとレポートモジュールは Perl を用いて実装した。また、Holt-Winters 法によるしきい値の算出には、3.4.9 項で述べたように、GNU R の Holt-Winters パッケージを利用した。

3.5.3 トラフィック観測ポイント

トラフィック観測ポイントとして、本学の境界ルータ、Juniper Networks 社の M-20 を用いる。このルータは、インターネットと本学イントラネットの境界に設置され、本学内からインターネットへのトラフィック（上りトラフィック）、またその逆向き（下りトラフィック）はすべてこのルータを経由する。通過するトラフィック量は、上り平均 2.01 Mbps (0.98 kpps)、下り平均 7.93 Mbps（1.06 kpps）となっている。また、観測機器への負荷、データベースのポータビリティを考慮し、パケット単位で $\frac{1}{1000}$ 以下[6]のサンプリングを適応する。

ただし、提案手法は 3.4.1 項で示したように、学内の異常トラフィックを発生させているホストを見つける目的で使用する。したがって、上り（学内からインターネットへ向かう）トラフィックに対して注目指標を利用する。

3.5.4 動作例

提案手法を実測トラフィックに対して適応した例を図 3.12 に示す。注目指標の実測値（Observed Value）に対して、Holt-Winters 法を用いた上限しきい値

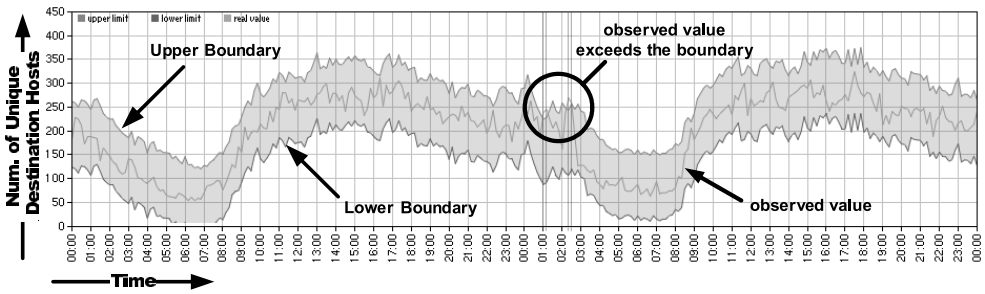

図 3.12.　提案手法の動作例

6　1000 個以上に 1 個の割合でパケットをサンプリングし、サンプリングされたパケットを元にフローの属性情報を作成

(Upper Boundary)、下限しきい値 (Lower Boundary) を設定し、実測値がしきい値から外れた際に管理者に通知する。リアルタイムの異常トラフィック検出においては、しきい値の算出と実測値との比較は N 秒ごとに実行される。

3.6 評価と考察

本節では、提案手法がその設計目標を満たしており、異常トラフィック検出に利用可能かを評価する。

3.6.1 評価項目とその意義

提案手法は、3.3.1 項に述べた以下の要件をすべて満たすことを目標に設計した。

(a) 低コストで利用を開始できる
(b) 検出シグニチャに依存しない
(c) 自動で異常トラフィックを検出可能である
(d) 高い精度で異常トラフィックを検出可能である
(e) 広帯域環境下で使用可能である
(f) リアルタイムに異常トラフィックを検出可能である

ただし、(a) および (b) の機能要件は提案手法の設計段階で満たしている。したがって、本節では評価対象としない。ここで、(c) から (e) の目標が達成されているかを評価するため、次の (1) から (4) の評価項目を挙げる。

(1) パラメータごとの False-Positive、False-Negative 率に与える影響
(2) サンプリングレートが False-Positive、False-Negative 率に与える影響
(3) 従来指標に注目した場合との比較
(4) しきい値計算に必要な時間

まず、(1) の評価項目について述べる。提案手法は、3.4.8 項に述べたように各種パラメータを必要とする。一度パラメータを与えてしまえば、あとは管理者の判断を必要としない。つまり、妥当なパラメータが決定できれば要件 (c) を満たすことになる。また、目的 (d) を満たすためには、提案手法が多くの異常トラフィック検出をでき、かつ、正常なトラフィックを異常と通知することが少ない必要がある。そこで、評価項目 (1) として、単位時間 N と Holt-Winters 法のパラメータごとに提案手法の False-Negative の発生数と、False-Positive 率を評価し、パラメータの設定指針について考察する。同時に、自動算出されたパラメータの利用が妥当かどうかを評価する。ただし、本章では、ほかの手法で異常トラフィックが認められなかった時刻に提案手法が異常トラフィック発生の通知を発した場合、提案手法の False-Positive とする。また、ほかの手法により確実に異常トラフィックの発生と認められた出来事をリファレンスとし、そのリファレンス異常が提案システムによって検出できなかった場合、提案システムの False-Negative とカウントする。

次に、(2) では、パケットのサンプリングレートが、提案手法の False-Positive、False-Negative 率に与える影響を評価する。パケットのサンプリングレートを下げても False-Positive、False-Negative 率に変化が生じなければ、本提案手法はスケーラブルに広帯域に適応可能であり、(e) の要件を満たす。

そして、評価項目 (3) では、提案手法で用いた注目指標が妥当性を評価するために、ほかの指標を用いて異常トラフィック検出を試みた場合と比較する。

最後に、評価項目 (4) について述べる。提案手法によるしきい値計算に必要な時間を計測し、要件 (f) に適合したリアルタイムな異常トラフィック検出が実現可能かを評価する。

3.6.2 リファレンス異常

提案手法の False-Negative 率を評価するために、次の 3 つの出来事を異常トラフィックのリファレンス（以下リファレンス異常）として利用する。

(i) ワーム感染の拡大
(ii) P2P ファイル共有アプリケーションの実行 A
(iii) P2P ファイル共有アプリケーションの実行 B

リファレンス異常 (i) は、Sasser と呼ばれるワームに感染した PC が学内に持ち込まれ、学内ホスト約 40 台に感染が広まったケースである。リファレンス異常 (ii) は 1 人のユーザが P2P ファイル共有アプリケーション WinMX を起動し、ファイルの検索と共有をした事例である。リファレンス異常 (iii) は別のユーザが同様に P2P ファイル共有アプリケーション e-Donkey を使用した事例である。いずれのケースも本提案手法の稼働前に起きた出来事であるが、提案手法を過去の観測データに対して適応し、発見可能であったかどうかを評価する。実際の運用現場において、リファレンス異常 (iii) はワームが攻撃に使用する 445/udp ポートを監視することで発見され、リファレンス異常 (ii) と (iii) はシグニチャマッチング型 IDS で発見された。提案手法においては、使用

するポート番号やシグニチャなどの事前知識を用いずにこれらのリファレンス異常の検出を試みる。

3.6.3 Holt-Winters 法のパラメータ評価と考察

まず、False-Negative の評価のため、Holt-Winters 法での CL、α、γ を変化させた場合に、各リファレンス異常が検出できたかどうかを評価する。次に、False-Positive 率の調査のため、ほかの手法で異常トラフィックが認められなかった時刻に提案手法が異常トラフィック発生の通知を発した率を評価する。多くのリファレンス異常を正常に検出できる、言い換えれば False-Negative 率が低く、正常時を異常と通知しない（False-Positive 率が低い）パラメータが適しているといえる。

- α に対して、40 分以内の実測値に 90%の重みを置く場合−(a)、50%の重みを置く場合−(b)、自動算出値（3.4.9 項参照）を利用−(c)
- γ に対して、1 周期前の同曜日、同時刻に 90%の重みを置く場合−(i)、50%の重みを置く場合−(ii)、自動算出値（3.4.9 項参照）を利用−(iii)
- 信頼水準 CL が、99%の場合−(A)、99.9%の場合−(B)
- 単位時間 N を 600 秒（10 分）とした場合、2,400 秒（40 分）とした場合

のそれぞれの組み合わせについて、リファレンス異常が正常に検出できたかどうかと、False-Positive 率について評価した。その結果を表 3.7 に示す。また、パケット単位のサンプリングレートは $\frac{1}{1000}$[7]とした。表 3.7 中の○は、そのパラメータの組み合わせで該当リファレンス異常が検出できたことを、×は検出できなかったことを示す。

リファレンス異常 (i) はすべてのパラメータの組み合わせにおいて正常に検出できており、提案手法によるワームの感染行為の検出は妥当であるといえる。しかし、リファレンス異常 (ii) は $N = 300$ の時はすべて検出できているが、$N = 2400$ の時は、検出できるパラメータの組み合わせと、検出できない組み合わせがある。リファレンス異常 (i) は、発見され対処されるまで数日の時間を要し、異常状態の継続状態が長時間にわたった。一方、リファレンス異常 (ii) の継続時間は 20 分（1200 秒）程度と短時間であった。この結果、単位時間を 2400 秒とした時のリファレンス異常 (ii) が、注目指標へ与える影響も半減し、検出できなかったと考えられる。その一方で、リファレンス異常 (iii) についてはいずれのパラメータでも検出に成功していない。これは、リファレンス異常である、P2P ファイル共有アプリケーションのトラフィックが注目指標に与える変化が小さすぎ、全体量としてみた注目指標をしきい値内に止めたためである。

表 3.7.　Holt-Winters 法のパラメータの評価

ID	N (sec.)	CL	α	γ	リファレンス異常 (i)	(ii)	(iii)	False-Positive
1	600	A	a	i	○	○	×	3.47%
2				ii	○	○	×	5.90%
3			b	i	○	○	×	4.51%
4				ii	○	○	×	4.17%
5			c	iii	○	○	×	4.17%
6		B	a	i	○	○	×	1.39%
7				ii	○	○	×	2.43%
8			b	i	○	○	×	1.39%
9				ii	○	○	×	1.39%
10			c	iii	○	○	×	1.74%
11	2400	A	a	i	○	×	×	16.67%
12				ii	○	×	×	18.06%
13			b	i	○	○	×	2.78%
14				ii	○	○	×	8.33%
15			c	iii	○	○	×	6.94%
16		B	a	i	○	×	×	6.94%
17				ii	○	×	×	6.94%
18			b	i	○	×	×	2.78%
19				ii	○	×	×	4.17%
20			c	iii	○	○	×	2.78%

ここで、α、γ の設定について考察する。3.4.9 項の手法で自動算出される α、γ が最適な False-Positive 率をもたらすとは限らない。また、高い False-Positive 率をもたらすわけでもない。一方、この評価結果から、特定の α と γ の値が一般に最適とは言うことはできない。たとえば、表 3.7 の ID1 と ID11 は、α と γ の設定方針が同等であるが、ID1 の False-Positive 率は低く、ID11 の False-Positive 率は高い。このように、同等の α と γ を与えても、N が変化すると、False-Positive 率が急激に変化することもある。この点を考慮すると、自動算出値の利用も妥当である。

また、N および、CL の設定について考察する。N が大きいと、短時間の変化しかもたらさない異常トラフィックは検出できていない。この点を考慮すると、対象とする異常トラフィックの一般的な継続時間より短い N を設定する必要があることがわかる。

7　1000 個に 1 個の割合でパケットをサンプリングし、サンプリングされたパケットを元に作成されたフローの属性情報を利用

また、ID14 と ID19 の比較でわかるように、高い CL の値は False-Positive 率を下げ、検出できない異常トラフィックを増やすことがわかる。

これらの指針によって N および CL を設定し、自動算出された α、γ を利用することで 3.6.1 項で述べた目的 (a) を達成することができる。

3.6.4 サンプリングレートごとの評価と考察

次に、N を 10 に、α を (a) に、β を (i) に、信頼水準 CL を (A) に固定し、サンプリングレートを低下させる。3.6.3 項で用いたサンプリングレート $\frac{1}{1000}$ を元に、$\frac{1}{2000}$、$\frac{1}{4000}$、$\frac{1}{8000}$、$\frac{1}{16000}$ と低下させた時の、リファレンス異常の検出の可否、および False-Positive 率は表 3.8 のようになった。表 3.8 中の○は、そのサンプリングレートで該当リファレンス異常が検出できたことを、×は検出できなかったことを示す。

リファレンス異常 (i) はサンプリングレートを下げても検出ができているが、リファレンス異常 (ii) については $\frac{1}{4000}$ 以下のサンプリングレートでは検出できていない。これは、サンプリングレートが下がるにつれ、注目指標の正常時の変動係数が大きくなることが原因であると考える。正常時の変動係数が大きくなると、信頼水準が等しくても、異常トラフィックの発生による注目指標の変化がしきい値内に収まる可能性が高くなるからである。

表 3.8. サンプリングレートごとの評価

サンプリングレート	リファレンス異常			False-Positive 率
	(i)	(ii)	(iii)	
1/1000	○	○	×	3.47%
1/2000	○	○	×	2.43%
1/4000	○	×	×	1.74%
1/8000	○	×	×	1.74%
1/16000	○	×	×	3.13%

3.6.5 従来指標との比較

従来から広く用いられている転送バイト量やパケット数の変化に注目し、3.6.3 項で使用したパラメータの組み合わせにより Holt-Winters 法を適応した。その結果を、表 3.9 に示す。◎はすべてのパラメータの組み合わせにおいて検出できた指標、○は一部のパラメータの組み合わせにおいて検出できた指標、×はいずれのパラメータの組み合わせにおいてもまったく検出ができなかった指標を示す。

転送バイト量や転送パケット数を指標に用いた場合、本章で定義した異常トラフィックはまったく検出できていない。提案手法では、リファレンス異常 (i) および、リファレンス異常 (ii) の検出に成功し

表 3.9. 従来指標との比較

	リファレンス異常		
	(i)	(ii)	(iii)
注目指標	◎	○	×
転送パケット数	×	×	×
転送バイト数	×	×	×

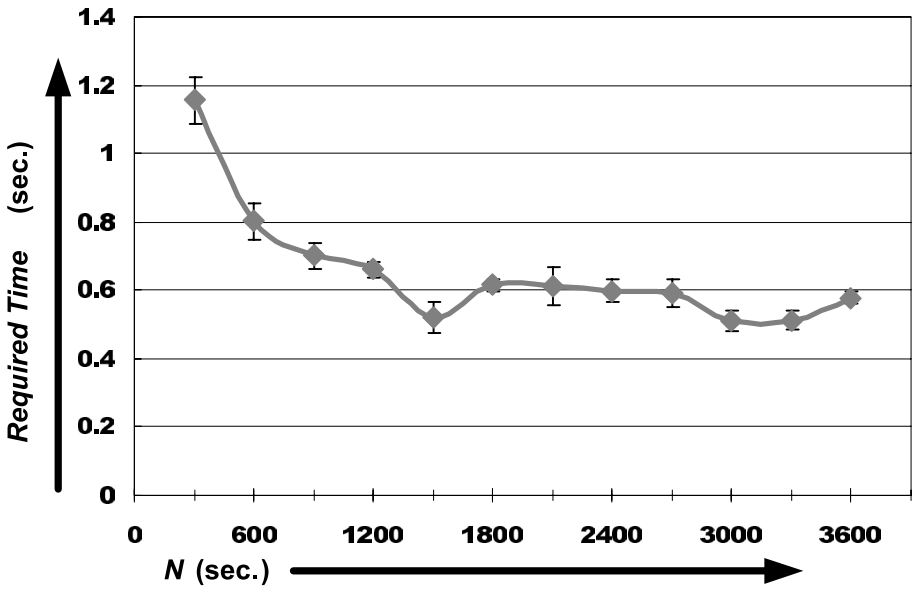

図 3.13. しきい値計算に必要な時間の評価

ており、注目指標は妥当であるといえる。転送バイト量やパケット数に注目した場合に検出されたトラフィックは、http や ftp、scp などによるファイルの転送がほとんどであった。

3.6.6 しきい値計算に必要な時間の評価と考察

リアルタイムな異常トラフィック検出には N 秒間隔でしきい値の計算が必要である。3.5.2 項で挙げた実装環境上でのしきい値計算に必要な時間を評価する。CL、α、β、γ の値は計算時間に影響を与えないため固定値を使用した。N の値が 300（5 分）から 3600（1 時間）の時に、1 回のしきい値計算に必要な時間を評価した。計算時間計測の誤差を減らすため、100 回のしきい値計算をして、1 回のしきい値計算に要した時間（プログラムの起動時間、I/O 処理時間を含む）の平均をとった。その結果を、評価結果を図 3.13 に示す。図 3.13 は、x 軸は N の値、y 軸が 1 回のしきい値計算に要した時間を示している。3.4.7 項で示したように、Holt-Winters 法は、$\frac{M}{N}$ ステップの処理を必要とする。そのため、M を固定して、N を大きくするとしきい値計算に要する時間は短くなる。

しきい値の算出に要する時間が最大の $N = 300$（5 分）の場合でも、その計算時間はたかだか 1.2 秒程度である。$N = 300$ の場合の必要しきい値計算間隔は 300 秒であるから、提案手法は十分リアルタイムなトラフィック検出に用いることができ、機能要件 (f) を満たす。

3.7 おわりに

本提案手法は、3.3 節に挙げた機能要件を一部満たすことができた。

まず、従来広く用いられていた転送バイト量やパケット量の監視では発見できなかった、ワームによる攻撃トラフィック、および、P2P ファイル共有アプリケーションによるトラフィックを、検出シグニチャなどの事前知識を用いずに検出することができた。そして、提案手法はその設計において、要件 (a)、要件 (b) を満たすことができた。次に、α、β の自動算出値が妥当であることを示し、CL および、N の設定指針を示した。これにより、提案手法は管理者の知識や経験に依存しない、自動異常トラフィック検出を実現でき、要件 (c) を満たすことができた。そして、提案手法による異常トラフィック検出は、十分短時間で実現できることがわかり、要件 (f) を満たすことができた。

しかし、要件 (d) および要件 (e) の実現性については十分とはいえない。まず、すべての異常トラフィックが検出できるわけではなく、3.6.2 項で挙げた、リファレンス異常 (iii) のように提案手法では検出できない異常トラフィックも存在する。しかし、検出できなかった異常トラフィックも注目指標に少なからず影響を与えている。検出できなかった原因は、そのトラフィックによる影響が、全体量としてみた注目指標をしきい値外に押し出すために十分でなかったためである。こうした異常トラフィックの検出にはさらなる工夫が必要となる。あらかじめサブネットごとに分割した注目指標について提案手法を適応する手法はその一例であり、その実装と評価が今後の課題である。

そして、数パーセント以上の False-Positive 率も実運用への投入に向けた課題である。今後、注目指標だけでなく、通信先ホストのアドレス、使用ポート番号のばらつきなど、別の指標を組み合わせて異常の評価をした場合の False-Positive 率を検証していく必要がある。

また、3.6.4 項で示したように、サンプリングレートを下げた際の異常トラフィック検出精度が低下している。しかし、$\frac{1}{16000}$ といった低いサンプリングレートでも検出できているリファレンス異常も存在することから、IDS などの全てのパケットを検査する手法と比較すると、広帯域への適応性は高い。

さらに、3.6.1 項で述べたように、本評価では、ほかの手法で異常トラフィックが認められなかった時刻に提案手法が異常トラフィック発生の通知を発した場合、提案手法の False-Positive としている。しかし、この方法では、ほかの手法で検出できなかった異常トラフィックが提案手法のみで検出できた場合も False-Positive とされるという問題がある。この問題の解決のために、さらなる解析結果の分析と実地調査の強化が必要となる。