Vol.2010-IOT-8 No.31
2010/3/2

TCP，UDP トラフィックに着目したファイル共有検出システムの構築と運用評価

元木 伸宏[†1]　泉 裕[†2]

ユーザによるファイル共有が企業や教育機関でも見受けられる一方で，著作権の侵害などの深刻な問題が表面化している．一般的にはデータ・ペイロードを検閲する以外に，ファイル共有ソフトウェアを検出する有効な手段がない．

本研究では，ネットワーク上の TCP，UDP パケットの流れから，同ソフトウェアを検出する汎用的な手法を提案する．さらに，本手法に基づくシステム構築と，和歌山大学における実験結果について述べる．

Building and Evaluation by Operation for File Sharing Detection System based on TCP and UDP Traffic

Nobuhiro Motoki[†1] and Yutaka Izumi[†2]

While file sharing by some users is present at companies and educational institutions, the serious problems such as the infringements of the copyright has come to the front. Generally, there is no effective means to detect file sharing software, except checking payload data.

In this paper, we suggest the versatile method to detect the software by TCP and UDP flows on the network. In addition, this paper describes building systems based on this method and an experimental result in Wakayama University.

†1 和歌山大学大学院システム工学研究科システム工学専攻
Graduate School of Systems Engineering, Wakayama University
†2 和歌山大学システム情報学センター
Center for Information Science, Wakayama University

1. はじめに

情報インフラの遍在性，ネットワークの安定性が高まるにつれて，高品質な通信環境が整いつつあり，新たな通信デバイス・ソフトウェアが次々と登場している．インターネット上にオーバレイ環境を構築する P2P（Peer to Peer）ファイル共有ソフトウェアも上記の一つであり，不特定多数のユーザ間で特定ファイルを容易に共有できるため，多くのユーザに利用されている[1)]．

上記に加えて，近年ではファイル共有ソフトウェア利用によって様々な問題が発生している．主な問題として，著作権侵害の問題，会社・学校などの組織の情報漏洩，音楽・動画といった大容量ファイルの共有による通信帯域の圧迫などが存在する．

現状では，企業・教育機関などでファイル共有ソフトウェアを禁止・制限する組織が増えてきている．しかし，セキュリティポリシの制限規定では利用者を排除できない．さらに，ユーザ空間でのソフトウェア実行権を制限するシンクライアント環境の導入には膨大なコストと運用負荷がかかる．加えて，プロキシなどによるポート制限では有用なアプリケーションの利用に影響を及ぼす．現在，パケットのデータペイロードを検閲する製品が存在するが，特定のファイル共有ソフトウェアに特化しており，シグネチャの更新や検閲の負荷が大きい．

本研究では，パケットヘッダより得られる情報からネットワーク上の挙動に着目し，ファイル共有ソフトウェアを検出する汎用的な手法を提案する．本手法では，全トラフィックを TCP，および UDP の視点に分け，それぞれ固有のトラフィックパターンを用いて検出する．P2P 型のファイル共有ソフトウェアの特徴は検索能力の高さにあるため，組織内の一端末と外部との通信に顕著なトラフィックパターンが見受けられる．さらに，管理者による被疑トラフィックの把握が可能な，本手法に基づく検出システムを構築し，和歌山大学内ネットワークを対象に運用実験を行った．

本稿では，既存の検出手法，提案手法，実験，および評価について述べる．

2. 既存の検出手法

本章では，本研究の背景に存在する P2P ソフトウェア，および同ソフトウェアを検出する既存技術について述べる．

2.1 P2P ネットワーク

P2P 方式では，サーバを介さず機器同士が直接情報をやりとりする．P2P ネットワークは，P2P により構成されたオーバレイネットワークであり，クライアント・サーバネット

表 1 本稿で使用したファイル共有ソフトウェア
Table 1 File sharing software used in this paper

Winny		**LimeWire**	Gnutella クライアント
WinMX	Napster 互換クライアント	**Cabos**	LimeWire の後継
Share(ex2)	Share の TCP 版	**eMule**	eDonkey の後継
Share(nt5)	Share の UDP 版	**BitComet**	BitTorrent クライアント
PerfectDark		**Azureus**	BitTorrent クライアント

ワークに比べて特定機器への負荷集中が発生しづらく，障害時においてもネットワークを維持することが可能である．総務省では，2007 年 6 月に P2P 技術の有効活用へ向けて「P2P に関する報告書」をまとめている[2]．

2.2 P2P ソフトウェア

本稿では，P2P ネットワークを利用するソフトウェアを P2P ソフトウェアとし，そのうちファイル共有を目的としたソフトウェアをファイル共有ソフトウェアとする．

2.2.1 ファイル共有ソフトウェア

Winny をはじめとするファイル共有ソフトウェアは，不特定多数の PC 間で専用のプロトコルを用いて通信し，ファイルを共有する．同ソフトウェアはファイルを共有する際に，P2P 方式を用いて独自のオーバレイネットワークを構成し，論理的に独立したネットワークの中で容易にファイルを共有することが可能である．本研究では，現在利用が多いと考えられる**表 1** のファイル共有ソフトウェアを選出し，検出対象とした．

2.2.2 他の P2P ソフトウェア

P2P ネットワークは，ファイル交換・共有以外に，インスタントメッセージング，インターネット電話などで利用されており，耐障害性，同報性，分散性を要する領域を中心にソフトウェアが存在している．本稿では，後述する戸田らによる手法 6) で false positive の可能性が存在することや，近年の国際通話で占める割合が非常に高くなっていることから，Skype を実験対象に含める．

2.3 既 存 技 術

本節では，検出技術を実装した機器の設置方法により，2 つに分けて述べる．

2.3.1 ゲートウェイ型

ファイル共有ソフトウェアを検出する機器を，トラフィックの通路にゲートウェイとして設置する形式を指す．一度バッファに溜めて処理を行うので，フィルタリングにより異常なパケットを即座に遮断することが可能であるが，機器に大きな負荷がかかる．ゲートウェイ型により実装されているものの多くは，専用のアプライアンスとして販売されている製品群であり，国内で問題になった Winny を検出可能なものが多い．また，多くの製品が，パケットのデータペイロードを検閲し，アプリケーションプロトコルの合致したものをパケット単位で検出する．例として，One Point Wall[3] などが存在する．

2.3.2 トラフィックモニタ型

ファイル共有ソフトウェアを検出する機器を，対象ネットワークの全パケットをコピーしてキャプチャ可能な地点に設置する形式を指す．例として，対象ネットワークのトラフィックが 1 つに収束する L2 スイッチを設置し，ポートのミラーリング機能やリピータハブを用いて監視用の機器へトラフィックを複製する．前述のように，ファイル共有ソフトウェアの検出が可能であるが，トラフィックモニタ型の機器だけでは，トラフィックを遮断できない．一般的なソフトウェアでは，snort[4] などの IDS が存在し，既存の研究では，藤井らによる手法 5)，戸田らによる手法 6)，および松田らによる手法 7) などが存在する．

3. 提 案 手 法

本章では，本研究で提案した手法について述べる．まず，和歌山大学における運用のフレームワーク（以下，運用フレーム），既存の問題点を考慮した研究目的について言及し，提案手法で用いた 2 つの評価指標．および具体的な検出手法について述べる．

3.1 和歌山大学における現在の運用フレーム

ネットワーク管理者は，モニタリング中の異常な TCP セッション・UDP パケットの増加などを確認した上で，ファイル共有ソフトウェアらしき通信を検知する．そして，その被疑トラフィックの送信元 IP アドレスに対して，通信先，IP アドレスの登録情報，および被疑端末の位置を確認する．同端末の位置は，續木らの MATT[8] を用いることで得られ，同端末がファイル共有ソフトウェアである可能性が高いのであれば，端末の存在する部屋の責任者に問い合わせ，利用ユーザに確認を行う．しかし，**図 1** のような現在の運用フレームでは，ネットワーク管理者への負担が大きく，ファイル共有ソフトウェア検出に関して個人差が生じてしまう．

3.2 研 究 目 的

第 1 章で述べたように，一般的には利用端末より送信されるデータよりファイル共有ソフトウェアを検出する．ここで，ペイロードを精査する手法では通信の秘匿性を失うことになり，プライバシ侵害の問題が生じる．本学のような教育機関の中には，プライバシポリシが厳密に定められていない機関が存在し，データペイロードを閲覧することに不満を覚える

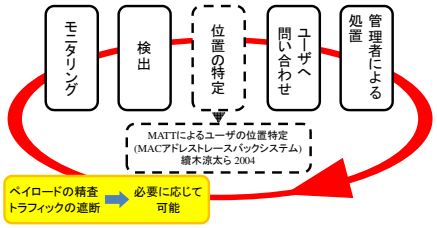

図 1 和歌山大学における現在の運用フレーム
Fig. 1 A current operational framework in Wakayama University

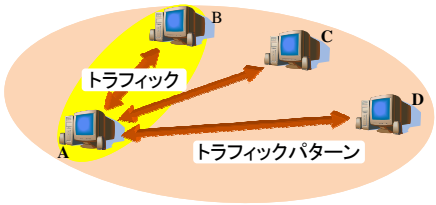

図 2 トラフィックパターン
Fig. 2 Traffic pattern

ユーザが存在する可能性がある．また，ペイロードを精査することで，機器への負担が増大するため，ネットワークの規模によってはペイロードの精査が困難な環境も存在する．

また，トラフィックパターンを用いることによる，長期的なサンプリングや学習の必要性，他の P2P ソフトウェアに対する false positive，および検出ソフトウェアの限定といった問題が第 2.3 節で述べた関連研究には存在する．そこで，前述の問題を解決し，多くのファイル共有ソフトウェアに対応するような汎用的な手法を提案する．さらに，提案手法を実装したシステムを構築し，ネットワーク管理者によるファイル共有ソフトウェア検出を支援する．

3.3 トラフィックパターン抽出

本節では，提案手法で用いるトラフィックパターンという概念について説明し，事前に行ったファイル共有ソフトウェアのトラフィックパターン抽出について述べる．

3.3.1 トラフィックパターン

図 2 は，A〜D の 5 つの端末が通信する様子を表す．まず，レイヤ 4 までのヘッダから得られる送信元 IP アドレス，宛先 IP アドレス，およびポート番号などの情報を用いて表される，図 2 の AB 間のようなデータのやりとりを 1 つのトラフィックとする．本来ならば，双方向のトラフィックに着目しなければならないが，同ソフトウェアが相互にデータを送信することから，便宜上 Outbound トラフィックのみ扱うことにする．トラフィックパターンとは，図 2 全体のように一定時間内に発生したトラフィック全体の様子を指し，本稿では前述のような Outbound のトラフィックパターンに基づいて，ファイル共有ソフトウェアを検出する．

3.3.2 事 前 実 験

まず，事前実験によりトラフィックパターンを抽出する．1 回の実験は，1 つずつソフトウェアを動作させ，10 分間で発生するトラフィックを採取することで完了し，1 つのソフトウェアにつき 5 回実験した．実験対象として，第 2.2.1 項で述べたファイル共有ソフトウェアと，Skype を選択した．Skype は，起動，サーバ接続，通話，およびチャットのような一般的な動作を行った．一方，ファイル共有ソフトウェアに関しては，起動，あるいはサーバへの接続しか行わなかった．

表 2 UDP トラフィック
Table 2 UDP traffic

ソフトウェア	通信先の平均数
BitComet	4643.4
Azureus	996.4
LimeWire	49.8
eMule	47.6
Share(UDP)	42.8
Cabos	17.2
WinMX	10.8
Skype	9

表 3 TCP トラフィック
Table 3 TCP traffic

ソフトウェア	通信先の平均数
Share(TCP)	193.2
Perfectdark	173.4
Winny	168.8
LimeWire	12.6
Cabos	6.8
WinMX	5.6
eMule	4
Skype	2.2

3.4 提 案 手 法

本項では，事前実験の結果と推察される特徴について言及し，提案手法と同手法で用いる 2 つの評価指標について述べる．

3.4.1 評価指標 1

まず，通信先の数に着目すると**表 2** と**表 3** のような結果が得られた．表 2 と表 3 はそれぞれのプロトコルを用いるソフトウェアの結果を表し，ソフトウェアごとに 5 回の実験における通信先ホストの平均数を求めた．ファイル共有ソフトウェアは，主に TCP，UDP，あるいはその両プロトコルを用いて通信していることがわかった．さらに，同結果はすべて送信元ポート番号，および宛先ポート番号に well-known port を使用しない通信であり，一般的なアプリケーションでは見受けられない通信が短時間で発生していると考えられる．

そこで，一定時間内における，1 つの送信元 IP アドレスに対する通信先ホストの数を評価指標 1 と定義する．また，評価指標 1 は，送信元，および宛先ポート番号に well-known port を使用しないトラフィックを対象とする．**図 3** と**図 4** は，事前実験の結果より得られた評価指標 1 の値を，学内の送信元 IP アドレスごとにプロットしたグラフである．まず，図中の 1 番は実験中に得られた学内の一般トラフィックであり，2 番は Skype のトラフィックである．1，2 番のトラフィックが下部に集中しているのに対し，ファイル共有ソフトウェアのトラフィックは上部に集中する傾向がある．そのため，評価指標 1 の値が一定値を超え

Vol.2010-IOT-8 No.31
2010/3/2

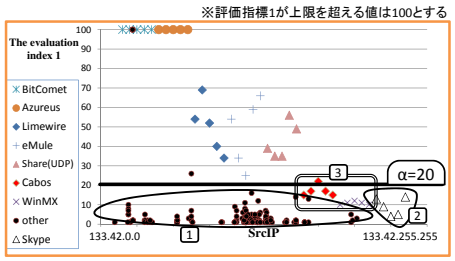

図 3 UDP トラフィックにおける評価指標 1
Fig. 3 The evaluation index 1 for UDP traffic

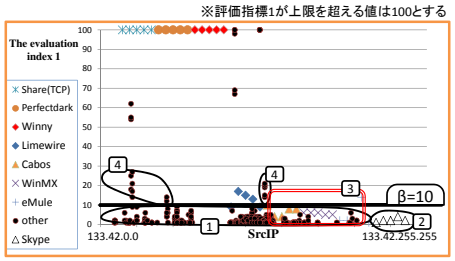

図 4 TCP トラフィックにおける評価指標 1
Fig. 4 The evaluation index 1 for TCP traffic

たものを被疑トラフィックとみなすことが可能であると考えられる．本稿では，UDP トラフィックに対する評価指標 1 の閾値を α，TCP トラフィックに対する評価指標 1 の閾値を β と定義し，図 3 と図 4 より，$\alpha = 20$，$\beta = 10$ とする．ここで，図中の 3 番が閾値を下回っているが，本稿では Skype に対する false positive 回避を優先させる．しかし，図 4 では，4 番のように閾値付近に一般トラフィックが存在し，同トラフィックに関して false positive を引き起こす可能性が考えられるため，次節では TCP トラフィックに用いる評価指標 2 について説明する．

3.4.2 評価指標 2

TCP トラフィックに関しては，他の視点からも評価指標を定める．**表 4** は Winny のサンプルトラフィックである．送信元ポート番号に注目すると，ほぼ連続した値を用いていることが分かる．この挙動が表れるのは，1 つのアプリケーションから複数のアプリケーションへ TCP コネクションを確立する際であると考えられ，他の TCP を用いるファイル共有ソフトウェアに関しても同様の結果が得られた．このポート番号の連続性に比例してファイル共有ソフトウェアの可能性も高まるのでないかと推測できる．そこで，連続性を数値化するため送信元ポート番号を区間ごとに分割し，存在するポート番号の数を計測する．

本稿では，計測した値を評価指標 2 と定義し，同指標が一定の閾値を超えたものをファイル共有ソフトウェアとみなすが，安易に区間の大きさと閾値を決定することはできない．そこで，事前実験のデータに対して，区間を 10，20，および 40 とした時の評価指標 2 のグラフを作成し，**図 5** のように評価指標 2 の軸をそれぞれの区間ごとに正規化した．正規化することで，一般トラフィックの分布が区間の大きさの増大にともなって下部へと移動する傾向があることに対し，ファイル共有ソフトウェアのトラフィックの分布は，上部にとどま

表 4 Winny のサンプルトラフィック
Table 4 Winny sample traffic

SrcAddr	SrcPort		DstAddr	DstPort
133.42.xxx.xxx	1100	⟶	59.xxx.99.47	32367
133.42.xxx.xxx	1102	⟶	117.xxx.40.249	1921
133.42.xxx.xxx	1104	⟶	123.xxx.223.123	3009
133.42.xxx.xxx	1105	⟶	222.xxx.176.238	27399
133.42.xxx.xxx	1107	⟶	124.xxx.211.13	8950
133.42.xxx.xxx	1108	⟶	219.xxx.41.28	14201
133.42.xxx.xxx	1110	⟶	118.xxx.114.168	4000
133.42.xxx.xxx	1111	⟶	211.xxx.169.209	11230

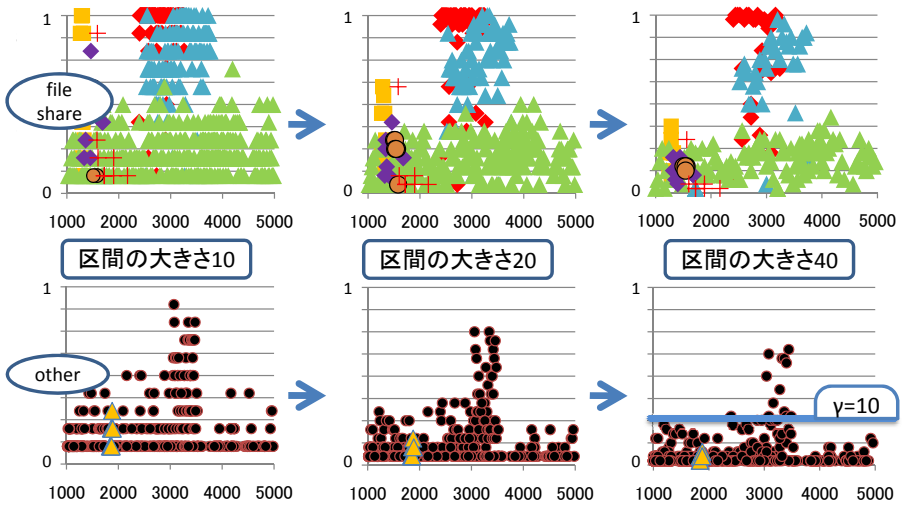

図 5 区間の大きさと評価指標 2 の関係
Fig. 5 Relation between the section size and the evaluation index 2

る傾向があることがわかる．ここで，区間の大きさを 40 よりも大きな値にすると同分布も下部へと移動し始めてしまうため，本稿では区間の大きさを 40 とした．また，区間の大きさ 40 のグラフで一般トラフィックとの境界線が見受けられるため，図 5 のように，評価指標 2 の値を γ とし，$\gamma = 20$ とした．

表 5 実験機器の性能
Table 5 Specification of experimental equipment

CPU	Intel Pentium 4 2.40GHz
Memory	1536MB
Storage	HDD 500GB
NIC	Intel 82550EY 10/100Mbps
OS	FreeBSD 6.2-RELEASE

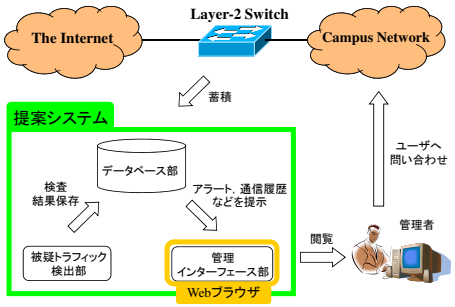

図 6 システム構成図
Fig. 6 System configuration diagram

4. 提案システム

4.1 システム環境

提案システムは，**表 5** のような性能の PC の FreeBSD 上に実装した．市場に流通している最新のハードウェアと性能差があるが，提案システムを動作させるためには十分である．第 3.4 節で述べた提案手法を，FreeBSD 上に Apache と MySQL をインストールした PC に実装した．

4.2 システム構成

本節では，提案システムの構成について述べる．提案システムは**図 6** のように，データベース部，被疑トラフィック検出部，および管理インターフェース部の 3 つから主に構成されており，それぞれの構成要素について動作をふまえて説明する．

まず，データベース部では，対外線接続部に設置されたレイヤ 2 スイッチのポートミラーリング機能を用いることで，本学内外を流通するすべてのトラフィックを収集する．同部分に提案システムを接続し，トラフィックデータのうちレイヤ 3，およびレイヤ 4 ヘッダより得られる情報をデータベースへ常時蓄積する．現在では HDD 容量に合わせて過去 1 カ月分のデータを保持している．

次に，被疑トラフィック検出部は，蓄積されたデータから 10 分間のサンプルを取り出し，10 分ごとに常時検出を行う．被疑トラフィック検出部には，第 3.4 節で述べた提案手法を実装しており，評価指標の閾値を超えると被疑トラフィックとして記録する．

管理インターフェース部は，OS への依存を避けるため Web アプリケーションにより実装されており，Web ブラウザを通じて管理者による被疑トラフィック管理の支援と，ネットワーク運用支援を行う．同インターフェースを用いることで，システム状態の閲覧，パケット量のグラフ閲覧，通信履歴の閲覧，および被疑トラフィックの閲覧が可能である．

5. 実　　験

5.1 ネットワーク環境

和歌山大学内ネットワークは，主に L2，および L3 スイッチにより構成されるスター型トポロジである．本システムは，コアスイッチ付近の L2 スイッチのミラーリングポートへ接続することで，学内の全トラフィックを本システムによって精査することができる．

また，本学には職員，教員，および学生合わせて約 5000 人ほどのユーザが存在する．過去に，学生がファイル共有ソフトウェアを利用していた事例があったことから，一部のユーザはセキュリティポリシへの認識が曖昧であることがわかる．

5.2 実 験 内 容

実験では，本研究で作成したシステムを和歌山大学ネットワークに実際に導入し，実際に管理者が用いることでファイル共有ソフトウェアの検出が可能かどうかを調査する．2009 年 12 月 16 日から 2009 年 12 月 29 日までの 2 週間行い，ネットワーク管理者の監視の下，期間中に無作為のタイミングで動作させたファイル共有ソフトウェアが検出可能か確認する．

関連研究では，特定のファイル共有ソフトウェアに対してのみ行ったものばかりであり，検証が不十分であると考えられるため，本研究では 10 種類のファイル共有ソフトウェアに対して検出可能かを調査する．手順として，事前実験と同様に 10 分間のサンプリングによる検出を 1 回と定義し，1 つのファイル共有ソフトウェアにつき 5 回ずつ検出可能か調査する．ファイル共有ソフトウェアは起動とサーバ接続のみ行い，検索やダウンロードは行わない．同時に，Skype に対して false positive が起きないか調査する．

6. 評価・考察

実験結果は**表 6** の通りである．

6.1 比 較 評 価

実験では，8 種類は 5 回の試行すべてで検出することが可能であった．しかし，Cabos と WinMX は発生させる評価指標の数が少ない傾向にあり，false negative が何度か発生した．Skype に関しては，今回設定した閾値によって，ファイル共有ソフトウェアとの区別が可能であったことから，戸田らによる手法 6) で見受けられた Skype に対する false positive を

表 6 実験結果
Table 6 An experimental result

ソフトウェア	1 回目	2 回目	3 回目	4 回目	5 回目
BitComet	○	○	○	○	○
Azureus	○	○	○	○	○
Share(TCP)	○	○	○	○	○
Share(UDP)	○	○	○	○	○
Perfectdark	○	○	○	○	○
Winny	○	○	○	○	○
LimeWire	○	○	○	○	○
eMule	○	○	○	○	○
Cabos	○	×	×	○	○
WinMX	×	○	×	×	×
Skype	×	×	×	×	×

○:検出　×:未検出

抑制することができたのではないかと考えられる．さらに，松田らによる手法 7) では TCP 接続を用いたピュア型ファイル共有ソフトウェアに限られた検出であったが，本手法では TCP，UDP トラフィック双方の検出が可能であり，藤井らによる手法 5) では検出が困難であった eDonkey の後継ソフトウェアである eMule を検出できた．

また，実験対象を広範囲にしたことで，本手法の汎用性の高さが実証できたと考えられる．そのため，同ソフトウェアと挙動の似ている他のソフトウェアに対しても，解析を要すること無く対応できる．

6.2 考　　察

ファイル共有では，多くのノードへ検索要求を出した方が検索効率が上昇すると考えられるため，検索効率が良いほど発生トラフィックの量が増えるのではないかと推察できる．本手法は，同特徴に着眼した手法であり，未解析のソフトウェアが出現した際も，検索効率が良ければ検出の可能性は大きい．その一例として，実験期間外に，迅雷と呼ばれるファイル共有ソフトウェアを検出した事例があった．現在は，サンプリングに 10 分という試験的な値を選択しており，十分短時間であるが，さらに短くすることも可能であると考えられる．

しかし，問題点もいくつか存在する．まず，NAT 環境のクライアントを識別できないことから，NAT のゲートウェイに対して false positive が起こる可能性がある．また，実験では Cabos と WinMX に対して false negative が生じていることから，現在の手法ではすべてのファイル共有ソフトウェアに対して適用可能であるとは言えない．同問題は重要であるが，今後，ファイル転送時の挙動についても注目すれば十分に検出の可能性はあると考えらえる．

7. おわりに

本稿では，P2P 技術の概要を説明し，同技術を用いるファイル共有ソフトウェアに起因する問題について言及した．そこで，TCP，UDP 双方のトラフィックパターンに着目した汎用的な手法を提案し，同手法を実装した管理者支援システムを構築した．実験では，本手法を用いて多くの種類に対して運用実験を行い，本手法の有効性を示すことができ，既存技術の問題点を解消することができた．しかし，false negative や false positive の可能性など，問題はまだ考えられる．今後は，ファイル転送時を考慮した検出へのアプローチや，評価指標のパラメータに関する検証を行いたい．

参 考 文 献

1) “ファイル共有ソフトの利用に関する調査〜アンケート調査〜報告書（概要版）”
http://www2.accsjp.or.jp/activities/pdf/p2psurvey2009a.pdf（最終アクセス 2010 年 01 月 26 日）
2) “「P2P ネットワークの在り方に関する作業部会報告書」”
http://www.soumu.go.jp/menu_news/s-news/2007/pdf/070629_11_1.pdf（最終アクセス 2010 年 01 月 26 日）
3) “OnePointWall ワンポイントウォール NetAgent Co.,Ltd.”
http://www.onepointwall.jp/（最終アクセス 2010 年 01 月 26 日）
4) “Snort :: Home Page”
http://www.snort.org/（最終アクセス 2010 年 01 月 26 日）
5) 藤井聖，中村豊，藤川和利，砂原秀樹，“通信先ホスト数の変化に注目した異常トラヒック自動検出手法の提案と評価”
電子情報通信学会論文誌 B，Vol.J88-B，No.10，pp.1922-1933
6) 戸田聡，金西計英，矢野米雄，“トラフィックマイニングと可視化による Peer-to-Peer ファイル共有検出支援システムの構築”
情報処理学会研究報告，2007-DSM-45(18)，pp99-103
7) 松田崇，中村文隆，若原恭，田中良明，“相互接続における順逆接続間隔を利用した P2P トラヒック弁別手法”
電子情報通信学会信学技報，NS2006-237(2007-3)，pp415-420
8) 續木涼太，泉裕，齋藤彰一，塚田晃司，“組織内ネットワークにおける MAC アドレストレースバックシステムの開発”
情報処理学会研究報告，2005-DSM-36(3)，pp13-18