情報処理学会論文誌 Vol.56 No.9 1716–1729 (Sep. 2015)

LPCケプストラム分析を利用したマルウェアの感染検知

岩野 透[1,a)] 吉浦 裕[1] 畑田 充弘[2] 市野 将嗣[1,b)]

受付日 2014年12月5日, 採録日 2015年6月5日

概要：トラヒックデータを利用したマルウェア感染検知において，検知精度の向上が課題となっている．そこで，本研究では，特徴量抽出部にLPCケプストラム分析を適用したマルウェア感染検知手法を提案する．LPCケプストラム分析で抽出されるスペクトル包絡に着目することで，時間的な変化パターンを考慮した特徴量抽出を行う．この提案手法の有効性を検証するため，検知精度の評価実験を行った．感染時通信はD3MとCCCを，正常時通信は2種類の異なるイントラネットを流れるトラヒックデータを用いた．提案手法と従来手法の検知精度を比較し，提案手法の有効性を示す．

キーワード：LPCケプストラム分析，トラヒック，MWS Datasets，クラスタリング

Applying LPC Cepstrum Analysis on Malware Infection Detection

Toru Iwano[1,a)] Hiroshi Yoshiura[1] Mitsuhiro Hatada[2] Masatsugu Ichino[1,b)]

Received: December 5, 2014, Accepted: June 5, 2015

Abstract: We propose a method to detect malware infection using LPC Cepstrum analysis. LPC Cepstrum analysis enables variations of temporal features to be applied to malware detection, can also possibly improve the accuracy. To evaluate this method, we used D3M and CCC as infected traffic data, two intranet traffic as normal traffic. Then we calculate the accuracy by TPR and TNR, and compare to previous works.

Keywords: LPC Cepstrum analysis, Traffic, MWS Datasets, clustering

1. はじめに

近年，インターネットを利用したサービスが拡大し，必要不可欠な存在となっている．その一方で，マルウェアへの感染による犯罪被害が社会問題となっている [1]．

このようなマルウェア感染への主な対策として，感染前にマルウェアを検知し，被害を防ぐ侵入検知がある．この侵入検知の一般的な手法として，マルウェアのバイナリパターンの特徴を利用する，シグネチャ型の検知手法がある．しかし，シグネチャ型の検知手法は，事前にマルウェア検体を解析する時間と労力が必要であり，短期間で大量に発生する未知のマルウェアには対応できない [2]．実際，大手セキュリティベンダのマルウェア検知率が45%程度だという報告もある [3]．そのため，マルウェアへの感染を前提とし，感染後に被害を拡大させないためのマルウェア対策手法（感染検知）が重要となっている．

マルウェアに感染したPCは，感染後に特有な通信を行うと考えられる [4]．具体的には，C&Cサーバとの通信やマルウェアのダウンロード，インターネットへの接続確認などである．これらは，未知のマルウェアに共通する通信と考えられる．よって，この通信をとらえることにより，未知のマルウェアでも検知できる可能性がある．また，トラヒックデータの取得環境は，感染PCの外部に設置可能であり，検知対象と分離することができる．そのため，ルートキットなどマルウェアに検知結果を妨害されず，検知精度の信頼性を担保できる．以上より，本研究ではトラヒックデータに着目した感染検知手法を検討する．

[1] 電気通信大学大学院情報理工学研究科
The University of Electro-Communications Graduate School of Informatics and Engineering, Chofu, Tokyo 182–8585, Japan
[2] NTTコミュニケーションズ株式会社
NTT Communications Corporation, Minato, Tokyo 108–8118, Japan
[a)] iwano-toru@uec.ac.jp
[b)] ichino@inf.uec.ac.jp

トラヒックデータに限らず，挙動を利用したマルウェア検知手法は，誤検知が多いという課題がある．本研究でのマルウェア検知システムは，パターン認識の問題に基づいており，システムは前処理部，特徴抽出部，識別部で構成される [5]．感染時と正常時で差異のある特徴量を抽出できれば検知精度が向上するため，本研究では特徴量抽出部に LPC ケプストラム分析を適用する．これにより時間的な変化パターンを考慮した特徴量を抽出でき，マルウェアの検知精度を向上させることができる可能性がある．

以下，2 章で既存研究に利用されている特徴量を整理し，3 章で本研究における特徴量抽出手法を述べ，4 章で検知精度の評価実験の内容を説明し，5 章でこの結果を述べる．6 章で利用したトラヒックデータの挙動調査の結果をふまえ，提案手法の検知結果について考察する．7 章で本稿のまとめを行う．

2. 先行研究における利用特徴量

トラヒックデータを利用したマルウェア検知手法としては，ペイロード情報に着目する手法とヘッダ情報に着目する手法がある．ペイロード情報を利用する場合，プライバシが問題にならず，暗号化もされていない通信を適用領域としている．それに対して，ヘッダ情報を利用する場合，プライバシの問題を考慮[*1]しつつ，暗号化された通信も適用領域とすることができる．そのため，本研究ではパケットのヘッダ情報に着目する．本章では既存研究で用いられたヘッダ情報を利用した特徴量について整理する．

宮本らは，ヘッダ情報の種類ごとのパケット数を特徴量とし，SVM を利用した異常検出手法を提案した [6]．利用したヘッダ情報は，パケットサイズとポート番号情報，TCP パケットのフラグ情報である．種類ごとのパケット数を 1 分ごとに算出し，SVM を利用して異常検知を行った．

川元らは，ヘッダ情報から得られた 36 種類の特徴量について，感染検知における有効性の評価を行った [7]．この実験では，TPR，TNR の観点から評価を行い，4 種類の特徴量が感染検知に有効であることを示した．

Soniya らは，ヘッダ情報を利用したボットネット通信の検知手法を提案した [8]．手法としては 30 分ごとに，送信先 IP アドレスやポート番号から，ユーザが通常の用途では利用しない通信を抽出する．この通信に対して，パケット到着間隔や syn スキャンの有無などを解析し，ボットネット通信を検知した．

鈴木らは，正常通信と異常通信について，特徴量の取得時間を変化させ，検知精度の変化を調査した [9]．特徴量はパケットサイズの平均など 3 種類のヘッダ情報を利用し，学習には SVM を利用した．

これらの先行研究では，ヘッダ情報から取得可能な特徴量の有効性を示している．しかし，マルウェアの感染後の通信挙動は日々変化している．実用を考えると，より検知精度の高い手法を検討する必要がある．

また，本章で述べた先行研究は，特徴量の時間的な変化パターンを考慮していない．しかし，通信挙動は一定の手順に従い実行されることが予想されるため，感染時と正常時の特徴量は時間的な変化のパターンが異なると考えられる．そのため，特徴量の時間的な変化パターンに着目することで，正常時と感染時の通信の差異をより明確にとらえられる可能性がある．

3. 本研究における提案手法

本章では提案手法であるトラヒックデータからの時間波形の作成方法と，時間波形への LPC ケプストラム分析の適用について説明する．

3.1 特徴量抽出単位

特徴量の抽出単位はフローごと，タイムスロットごと，パケットスロットごとの 3 種類がある．

フローごとでの特徴量抽出は 5-tuple（送受信 IP アドレス，送受信ポート番号，プロトコルの 5 項目が同一のパケット群）を 1 単位として特徴量を抽出する．しかし，通信が終了するまで特徴量の抽出ができないため，早期検知には不向きである．

タイムスロットごとでの特徴量抽出は一定の時間ごとに特徴量を抽出する．タイムスロットごとでの特徴量抽出は，通信の終了を待つ必要がない．そのため，フローごとでの特徴量抽出よりもマルウェアの早期検知の可能性がある．しかし，マルウェアは短時間で感染挙動を終える検体もあり，この場合はさらなる早期検知を行う必要性がある．

パケットスロットごとでの特徴量抽出は一定のパケット数ごとに特徴量を抽出する．パケットスロット幅での特徴量抽出は，通信の終了を待つ必要がない．そのため，フローごとでの特徴量抽出よりもマルウェアの早期検知の可能性がある．また，パケットスロットの間隔を調整することで，特徴量の取得時間を短縮できる．これにより，タイムスロットごとでの特徴量抽出よりも，さらに早期にマルウェアを検知できる可能性がある．しかし，取得単位が小さいため，誤検知を引き起こす可能性がある．

マルウェアの感染挙動の期間は検体によって異なる．そのため，それぞれのトラヒックデータに適した特徴量の抽出単位を利用する必要がある．本研究では，短時間のトラヒックデータの感染検知に対しては，パケットスロットを利用する．また，長期間のトラヒックデータの感染検知に対しては，タイムスロットとパケットスロットを併用することを検討する．

[*1] 厳密な解釈では，ポート番号などのヘッダ情報を監視することも通信の秘密に触れると考える場合もある．

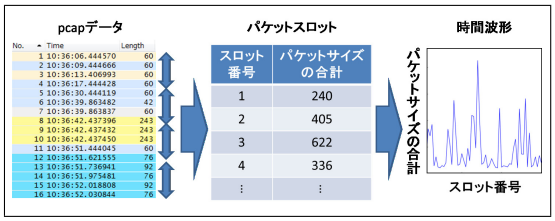

図 1 スペクトル包絡作成までの概要
Fig. 1 Outline of spectrum envelope.

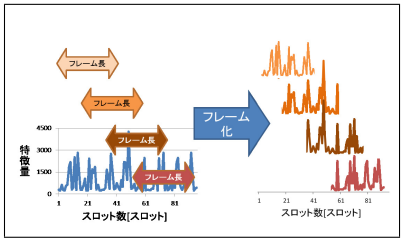

図 2 フレーム作成の概要
Fig. 2 Outline to prepare frames.

3.2 特徴量の時間波形化

本研究では，取得した特徴量から時間波形を作成する．具体的には，取得した特徴量の値を時系列順にプロットし，折れ線グラフ化する．この特徴量の推移を時間波形として利用する．一例として，パケットスロットでの特徴量の抽出から時間波形の作成までの流れを**図 1** に示す．

3.3 LPC ケプストラム分析

本節では LPC ケプストラム分析の概要とその利点について説明する．

3.3.1 分析フレームの作成

LPC ケプストラム分析は，フレームという一定の区間（フレーム長）ごとに行う．このフレームを重複するように徐々に移動させ，繰り返し処理を行う．時間波形からフレーム作成の概要を**図 2** に示す．以降の LPC ケプストラム分析は，この 1 フレームに対しての処理について説明する．

3.3.2 ケプストラム分析

ケプストラムは波形から短時間振幅スペクトルを算出し，このスペクトルの対数の逆フーリエ変換によって生成される [10]．ケプストラム分析を行うことで，時間波形からスペクトル包絡とスペクトル微細構造を近似的に分離して抽出できる．スペクトル包絡は，時間波形における大局的な特徴を表す．また，微細構造は，時間波形における基本周期を表す．

時間波形 $x(t)$ を 2 つの信号で表すことを考える．このとき，$x(t)$ は周期信号 $g(t)$ とインパルス応答 $h(t)$ の畳込みで表すと，式 (1) のようになる．

$$x(t) = G(\omega)H(\omega) \tag{1}$$

ここで，$X(\omega)$，$G(\omega)$，$H(\omega)$ は $x(t)$，$g(t)$，$h(t)$ のフーリエ変換である．式 (1) の対数変換を逆フーリエ変換すると式 (2) となる．

$$x(t) = \mathcal{F}^{-1}\log|G(\omega)| + \mathcal{F}^{-1}\log|H(\omega)| \tag{2}$$

この式 (2) が LPC ケプストラムであり，第 1 項はスペクトル微細構造を，第 2 項はスペクトル包絡を表す．スペクトル包絡とスペクトル微細構造はフィルタで分離することが可能であり，本研究ではスペクトル包絡に着目する．スペクトル包絡とスペクトル微細構造の概形を**図 3** に示す．

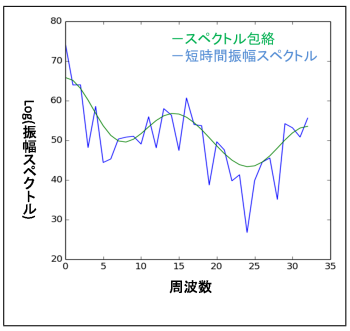

図 3 スペクトル包絡とスペクトル微細構造
Fig. 3 Spectrum envelope and spectrum fine-structure.

3.3.3 LPC ケプストラム

LPC ケプストラム分析では，LPC（線形予測分析）による予測波形モデルに対して，ケプストラム分析を行う．予測波形を利用することで，元の波形よりもスペクトル包絡のピーク特性を抽出しやすいという利点がある [10]．

全体トラヒック波形の任意の標本値 x_l（l：整数）は，これに隣接する過去の p 個の標本値の間に，ある係数 α_i $(i = 1, 2, \cdots, k)$ をかけ，足し合わせた次のような線形一次結合が成り立つとする．

$$\hat{x} = \alpha_1 x_{l-1} + \alpha_2 x_{l-2} + \cdots + \alpha_k x_{l-k} \tag{3}$$

このときの α_i を線形予測係数と呼び，標本値 x_l と線形予測値 $\hat{x}_l$ の間の誤差が最小となるように線形予測係数 α を定めることを LPC という．この LPC に基づいて予測した波形モデルは，α を用いて次のように表される．

$$H(z) = \frac{1}{1 + \sum_{i=1}^{k} \alpha_i z^{-1}} \tag{4}$$

LPC ケプストラム分析では，式 (4) の予測波形モデルを信号のスペクトル密度と見なし，

$$X(\omega) = H(z)|_{z=e^{j\omega}} \tag{5}$$

とおいたときのケプストラムを算出する．ケプストラム係数を C_n とすると，C_n は式 (6) のように表される．

$$C_n = \frac{1}{2\pi}\sum_{0}^{2\pi}\log|X(\omega)|e^{jn\omega}d\omega \tag{6}$$

本研究ではこの C_n を特徴量として利用する．

3.3.4 LPC ケプストラム分析のトラヒックデータへの適用

マルウェアは感染後に特徴的な通信を行う．具体的には，DDoS 攻撃や外部への情報送信などである．これらの挙動は，突発的なパケットの増加や，周期性のあるパケット送信を行う．そのため，時間的な変化のパターンが正常時の通信とは異なる．よって，正常時と感染時のトラヒックデータを時間波形化すると，振幅情報に差異が生じると考えられる．

時間波形に LPC ケプストラム分析を行うと，時間波形の振幅情報の差異は，スペクトル包絡に影響を与える．そのため，スペクトル包絡の差異は，振幅情報の差異を反映したものだと考えられる．さらに，LPC ケプストラム分析はスペクトル包絡のピーク特性を強調して抽出する．ピーク特性を強調することにより，時間波形の振幅情報の差異が明確になる．そのため，LPC ケプストラム分析で算出したスペクトル包絡は，時間波形の振幅情報を強調して表している．

以上より，LPC ケプストラム分析で算出したスペクトル包絡を利用することで，トラヒックデータの時間的な変化パターンを強調して抽出できる．時間的な変化パターンには正常時と感染時の差異が現れるため，感染検知の検知精度を向上させることができる可能性がある．そのため，本研究ではスペクトル包絡を表す LPC ケプストラム係数を特徴量として利用する．

4. 評価実験

提案手法の検知精度を評価するため，評価実験を行った．

4.1 提案手法の概要

提案手法における感染検知システムの流れを**図 4** に示す．この検知システムは特徴量抽出部分，クラスタリングによる学習部分，ユークリッド距離を利用した識別部分に分けられる．

特徴量抽出部分は特徴量抽出単位ごとに特徴量を取得した．この特徴量を時間波形化し，LPC ケプストラム分析により算出した LPC ケプストラム係数を特徴量として利用した．

学習部分はそれぞれの学習用データからコードブックを作成した．ここで，コードブックとは学習データの代表パターン群を指す．クラスタリングのアルゴリズムには，LBG+Splitting アルゴリズム [12] を用いた．本実験において正常時通信の学習データは 1,146 フレーム，感染時通信の学習データは 43 フレームである．感染時通信の学習データのフレーム数と文献 [12] に基づいて，コードブック数（レベル数）は 2，4 の 2 種類とした．

識別部分では，正常時通信と感染時通信の識別のために，感染（正常）のテストデータの特徴量と各コードブックとのユークリッド距離を利用した．このユークリッド距離を感染時と正常時で比較し，感染（正常）コードブックとの距離の方が近ければ感染（正常）と識別した．

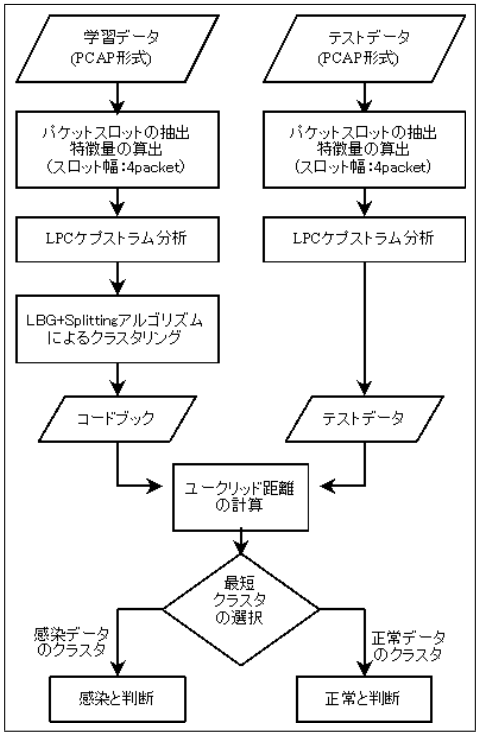

図 4 提案手法におけるフローチャート
Fig. 4 Flowchart for the proposed method.

4.2 比較実験の概要

検知精度の比較のため，提案手法と同一の実験データに対して，時間的な変化パターンを考慮しない場合の検知精度を算出する必要がある．そのため，パケットスロットの値を特徴量として利用した場合の検知精度もあわせて算出した．以降では，この手法を比較手法と表記する．パケットスロットとフレームの長さは値を変えて評価を行い，TPR と TNR が平均して一番大きくなった 1 パケットスロット=4 パケット，1 フレーム=64 スロットに設定した．タイムスロットについては文献 [7] において最良と判断された 1s とした．

4.3 実験データ

評価実験は正常，感染ともに 2 種類のトラヒックデータを利用し，それぞれの検知精度を評価した．これは，異なる環境で取得したトラヒックデータに対しても，LPC ケプストラム分析が有効であるかを検証するためである．

4.3.1 CCC dataset

感染時トラヒックデータは，MWS Datasets 2014 の CCC [11] を利用した．正常時トラヒックデータは，とあるイントラネット A のトラヒックデータを利用した．感染時，正常時ともに 2009 年のトラヒックデータを学習デー

タとして利用した．また，テストデータには 2009，2010，2011 年のトラヒックデータを利用した．なお，2009 年のトラヒックデータは，学習データとテストデータに分割して利用した．これは，検知精度を正確に評価するためである．

正常時通信のデータ量について，2009 年はクライアント数 30 前後で 196,212 パケット（WAN から LAN 方向へ 122,230 パケット，LAN から WAN 方向へ 73,982 パケット），総データ量に対する TCP の割合は 98.8%である．2010 年はクライアント数 30 前後で 362,563 パケット（WAN から LAN 方向へ 136,646 パケット，LAN から WAN 方向へ 225,917 パケット），総データ量に対する TCP の割合は 99.6%である．2011 年はクライアント数 20 前後で 305,514 パケット（WAN から LAN 方向へ 183,913 パケット，LAN から WAN 方向へ 121,601 パケット），総データ量に対する TCP の割合は 98.9%である．ユーザ数は 3 年間を通じて 20 名前後である．時間とともにユーザ数が異なるため，クライアント数にはばらつきがある．また，1 名で複数台の端末を利用している場合もあるため，ユーザ数よりもクライアント数が多くなっている．この通信の大部分は SSH 通信であり，総パケットの中に 90%程度含まれていた．ssh 通信以外では，2009，2010，2011 年の 3 年分ともに windows update，Google や yahoo サービスなどのウェブブラウジング，メールの送受信などが含まれている．2011 年のみに含まれていたサービスとしては，Dropbox や twitter などがあり，恒常的かつ定期的に通信するものや，多くの人が利用しているサービスの通信が含まれている．さらに総データ量に対する TCP の割合も 3 年間いずれもブロードバンドトラフィックレポートに記載されている平均値 [13] と大きな差はない．SSH 通信は LAN 内のサーバへアクセスするのに利用されており，セキュリティを考慮した環境とするとイントラネット A は比較的よく見られる設定であると考えられる．

特徴量は，タイムスロットごととパケットスロットごとの 2 つの抽出単位で取得した．これは，CCC が長時間の観測データであり，データ量が豊富なためである．

4.3.2 D3M データセット

感染時トラヒックデータは，MWS Datasets 2014 の D3M [11] を利用した．正常時トラヒックデータは，とあるイントラネット B のトラヒックデータを利用した．感染時，正常時ともに 2012 年のトラヒックデータを学習データとして利用し，テストデータには 2012，2013，2014 年のトラヒックデータを利用した．なお，2012 年のトラヒックデータは，学習データとテストデータに分割して利用した．これは，検知精度を正確に評価するためである．

正常時通信のデータ量について，2012 年はクライアント数 25 前後で 812,250 パケット（WAN から LAN 方向へ 307,904 パケット，LAN から WAN 方向へ 504,346 パケット），総データ量に対する TCP の割合は 95.3%である．2013 年はクライアント数 24 前後で 335,318 パケット（WAN から LAN 方向へ 194,005 パケット，LAN から WAN 方向へ 141,313 パケット），総データ量に対する TCP の割合は 97.7%である．2014 年はクライアント数 26 前後で 529,063 パケット（WAN から LAN 方向へ 313,047 パケット，LAN から WAN 方向へ 216,016 パケット），総データ量に対する TCP の割合は 99.1%である．ユーザ数は 3 年間を通じて 15 名前後である．ユーザの行う Web ブラウジングなどの通信が主体であり，TCP 通信，中でも SSL 通信と HTTP 通信が総パケットの 20%程度含まれていた．2012，2013，2014 年の 3 年分ともに windows update，Dropbox，Google や yahoo サービスなどのウェブブラウジング，youtube などの動画ストリーミング，twitter，メールの送受信などが含まれており，恒常的かつ定期的に通信するものや，多くの人が利用しているサービスの通信が含まれいる．さらに総データ量に対する TCP の割合も 3 年間いずれもブロードバンドトラフィックレポートに記載されている平均値 [13] と大きな差はない．したがってイントラネット B は昨今のインターネット環境を反映したものであり，一般的な正常時通信データと考えられる．特徴量はパケットスロットごとでの抽出単位で取得した．これは，D3M が短時間の観測データであり，タイムスロットごとでの特徴量抽出には不向きだからである．

表 1 利用特徴量の一覧表
Table 1 List of feature.

番号	利用する特徴量
1	パケットサイズの総数 [byte]
2	パケットサイズの平均 [byte]
3	パケットサイズの最小 [byte]
4	パケットサイズの最大 [byte]
5	パケットサイズの標準偏差 [byte]
6	到着間隔の平均 [ミリ秒]
7	到着間隔の最小 [ミリ秒]
8	到着間隔の最大 [ミリ秒]
9	到着間隔の標準偏差 [ミリ秒]
10	SYN パケット数 [packet]
11	FIN パケット数 [packet]
12	PSH パケット数 [packet]
13	ACK パケット数 [packet]

4.4 評価特徴量

先行研究 [7] で利用されていた特徴量の中から，13 種類の特徴量を選択し，**表 1** にまとめた．この特徴量は，時間波形を作成した際，LPC ケプストラム分析が適用できることを基準に選択したものである．

4.5 評価指標

評価指標としては，TPR（True Positive Rate）と TNR

（True Negative Rate）を利用した．TPR は感染時テストデータを感染であると正しく識別できた割合のことを指す．また，TNR は正常時テストデータを正常と正しく識別できた割合のことを指す．

5. 実験結果

実験結果は精度向上した特徴量の数と，最も高い検知精度の 2 つの観点から検証した．精度向上した特徴量の数を調査することで，LPC ケプストラム分析全体の効果を検証できる．また，最も高い検知精度を検証することで，LPC ケプストラム分析の実用性を検証できる．

5.1 CCC dataset の検知精度

タイムスロットで特徴量抽出時の TPR，TNR を**表 2**，**表 3** にまとめ，パケットスロットで特徴量抽出時の TPR，TNR を**表 4**，**表 5** にまとめた．以降の表の検知精度は，クラスタリングのレベル数 2，4 の検知精度の平均値である．LPC ケプストラム次元数は値を変えて評価を行い，TPR と TNR の平均値が一番大きくなった 3 に設定した．各手法における TPR と TNR で，最も検知精度の良い特徴量を太字で表記した．

特徴量抽出単位がタイムスロットごとのとき，TPR では，13 種類中 8 種類の特徴量で検知精度が向上した．TNR では，11 種類の特徴量で検知精度が向上した．最も高い検知精度は，TPR と TNR ともに比較手法の方が高かった．

パケットスロットで特徴量を抽出したとき，TPR では，13 種類中 9 種類の特徴量で検知精度が向上した．TNR では，13 種類中 6 種類の特徴量で検知精度が向上した．最も高い検知精度では，TPR と TNR ともに比較手法の方が高かった．

表 2 CCC における各特徴量の TPR（タイムスロット抽出時）
Table 2 TPR on each feature in CCC dataset (timeslot).

番号	提案手法				比較手法			
	2009	2010	2011	平均	2009	2010	2011	平均
1	**94.6**	**97.3**	**96.8**	**96.3**	97.2	72.2	84.3	84.6
2	92.2	96.2	95.3	94.5	94.9	93.2	89.5	92.5
3	72.5	75.1	73.1	73.6	94.8	90.2	86.6	90.5
4	95.1	93.5	92.1	93.6	97.8	90.5	86.0	91.4
5	94.1	92.3	89.1	91.9	94.9	91.4	87.3	91.2
6	85.8	82.7	76.8	81.7	94.9	84.4	75.3	84.9
7	81.9	72.6	71.7	75.4	38.9	32.6	32.2	34.6
8	85.3	80.2	87.5	84.3	88.1	61.0	50.4	66.5
9	90.7	83	85.3	86.3	93.9	72.5	66.9	77.8
10	98.0	89.0	79.4	88.8	84.6	76.1	64.4	75.0
11	84.3	92.1	96.1	90.8	98.2	96.5	96.4	97.0
12	84.4	90.8	91.7	89.0	97.6	96.4	96.7	96.9
13	89.5	95.5	97.1	94.0	**98.3**	**98.5**	**98.7**	**98.5**

表 3 CCC における各特徴量の TNR（タイムスロット抽出時）
Table 3 TNR on each feature in CCC dataset (timeslot).

番号	提案手法				比較手法			
	2009	2010	2011	平均	2009	2010	2011	平均
1	38.9	55.6	36.1	43.5	3.2	29	8.7	13.6
2	64.8	69.5	85.1	73.1	37.9	36.1	39.2	37.8
3	75.3	34.1	60.6	56.7	95.4	92.9	99.4	95.9
4	85.3	67.1	56.6	69.6	41.2	22.1	30.1	31.2
5	88.3	72.2	90.7	83.7	35.3	54.3	40.7	43.4
6	99.8	46.4	88.8	78.3	72	27.9	46.5	48.8
7	52.1	26.8	48	42.3	41.9	10.1	29.4	27.2
8	87.4	39.7	75.3	67.5	63.5	61.3	42.1	55.6
9	100	44.9	85.1	76.7	66.7	30.4	47.3	48.1
10	**99.4**	**99.3**	**97.4**	**98.7**	**99.3**	**99.7**	**98.8**	**99.3**
11	55.2	65.8	37.7	52.9	15.7	21.3	12.3	16.4
12	80	91.3	40.4	70.5	22.6	37.5	17.2	25.8
13	47.2	75.4	37.1	53.2	5.4	25.8	12.9	14.7

表 4 CCC における各特徴量の TPR（パケットスロット抽出時）
Table 4 TPR on each feature in CCC dataset (packetslot).

番号	提案手法				比較手法			
	2009	2010	2011	平均	2009	2010	2011	平均
1	98.4	98.7	97.7	98.3	98.7	98.8	96.6	98.0
2	**98.6**	**98.8**	**97.8**	**98.4**	98.7	98.8	96.6	98.0
3	96.7	95.6	94.8	95.7	**99.9**	**99.6**	**98.7**	**99.4**
4	97.1	96.8	94.7	96.2	99.0	98.9	97.2	98.4
5	72.2	86.6	79.2	79.3	98.7	98.8	96.8	98.1
6	82.3	57.5	66.0	68.6	65.7	34.9	37.3	46.0
7	75.2	42.2	37.2	51.5	12.3	66.4	65.3	48.0
8	81.0	53.0	67.7	67.2	65.3	41.4	42.0	49.6
9	85.7	43.1	64.7	64.5	65.3	47.7	43.8	52.3
10	98.7	98.7	96.6	98.0	97.1	97.2	92.6	95.6
11	57.8	38.1	29.0	41.6	0.1	0.1	0.3	0.2
12	58.4	44.2	43.3	48.6	1.4	0.7	1.7	1.3
13	74.5	85.5	73.8	77.9	97.3	98.0	94.4	96.6

表 5 CCC における各特徴量の TNR（パケットスロット抽出時）
Table 5 TNR on each feature in CCC dataset (packetslot).

番号	提案手法				比較手法			
	2009	2010	2011	平均	2009	2010	2011	平均
1	92.8	98.5	89.7	93.7	61.3	60.7	63.5	61.8
2	92.2	98.4	89.1	93.2	61.3	60.7	63.5	61.8
3	88.5	99.2	68.8	85.5	45.3	46.6	42.9	44.9
4	92.2	98.7	94.2	95	77.1	18.3	76.3	57.2
5	95.6	99.5	93.7	96.2	52.8	42.6	54	49.8
6	58.5	34.6	61.4	51.5	48.7	59.8	50.3	52.9
7	59.4	62.7	60.3	60.8	76.8	87.7	79.4	81.3
8	54.1	33.0	60.9	49.3	48.3	59.8	46.8	51.6
9	52.7	29.2	62.8	48.2	48.1	60.3	46.9	51.8
10	**97.0**	**98.5**	**95.3**	**96.9**	**99.0**	**99.9**	**99.4**	**99.4**
11	78.4	56.9	72.8	69.4	97.2	99.7	97.3	98.1
12	74.6	92.5	76.1	81.1	82.5	63	77.9	74.5
13	96.2	97.2	95.4	96.2	94.9	99.2	97.4	97.2

表 6 D3M における各特徴量の TPR（パケットスロット抽出時）
Table 6 TPR on each feature (D3M dataset).

番号	提案手法				比較手法			
	2012	2013	2014	平均	2012	2013	2014	平均
1	75.0	94.4	48.6	72.7	69.4	93.4	82.5	81.8
2	43.8	41.7	48.9	44.8	**69.4**	**93.6**	**82.5**	**81.9**
3	18.8	33.3	31.5	27.8	84.7	49.5	95.8	76.7
4	71.9	66.7	66.9	68.5	64.4	91.5	76.8	77.6
5	75.0	87.5	58.4	73.6	66.5	93.1	76.4	78.7
6	37.5	36.1	25.7	33.1	42.8	82.1	13.5	46.1
7	46.9	19.4	40.6	35.6	49.4	49.0	46.3	48.2
8	28.1	8.3	21.7	19.4	45.3	48.0	44.7	46.0
9	43.8	66.7	27.5	46.0	51.5	49.6	45.5	48.9
10	82.1	78.6	58.9	73.2	20.3	1.9	27.1	16.4
11	75.0	42.9	55.2	57.7	19.7	2.1	17.2	13.0
12	**96.4**	**100.0**	**76.8**	**91.1**	42.4	50.0	49.0	47.1
13	32.1	71.4	25.4	43.0	34.6	90.3	5.0	43.3

表 7 D3M における各特徴量の TNR（パケットスロット抽出時）
Table 7 TNR on each feature (D3M dataset).

番号	提案手法				比較手法			
	2012	2013	2014	平均	2012	2013	2014	平均
1	39.2	37.3	37.3	37.9	30.8	51.7	71.8	51.5
2	51.3	51.1	48.1	50.1	30.8	51.6	71.8	51.4
3	43.3	42.9	52.1	46.1	12.5	16.1	17.7	15.4
4	40.6	38.8	35.7	38.4	40.7	64.2	80.5	61.8
5	50.2	51.5	58.3	53.4	37.3	61.6	67.7	55.5
6	4.4	64.9	54.5	61.2	**75.7**	**77.9**	**91.1**	**81.5**
7	**90.3**	**89.4**	**88.0**	**89.2**	54.1	53.9	52.1	53.3
8	84.6	83.8	76.8	81.7	57.7	56.9	53.6	56.0
9	65.6	66.5	56.5	62.9	57.4	56.7	53.5	55.9
10	32.1	33.6	28.2	31.3	4.0	3.4	1.3	2.9
11	46.8	47.1	37.8	43.9	3.0	3.5	1.3	2.6
12	79.3	78.1	73.9	77.1	39.6	50.0	22.0	37.2
13	60.3	59.7	49.4	56.5	50.2	50.4	49.8	50.1

5.2 D3M dataset の検知精度

特徴量ごとの TPR を**表 6** に，TNR を**表 7** にまとめた．LPC ケプストラム次元数は値を変えて評価を行い，TPR と TNR の平均値が一番大きくなった 7 に設定した．

特徴量ごとで比較したとき，TPR では，13 種類中 3 種類の特徴量で検知精度が向上した．TNR では，13 種類中 8 種類の特徴量で検知精度が向上した．最も高い検知精度の比較では，TPR と TNR の両方で提案手法の方が高かった．

表 6 より，TPR で最も検知性能の良い特徴量は，PSH パケット数と分かる．各検体ごとの PSH パケット数の TPR を算出し，**表 8** に示した．なお，検体の接頭語とファミリー名は，TrendMicro [14] と Kaspersky [15] のデータベースを調査した．

表 8 より，マルウェアの接頭語（検体の種類）やファミリー名と検知精度には，関連性がないことが分かる．

表 8 PSH パケット数におけるテストデータ検体ごとの TPR
Table 8 PSH-based TPR on each sample.

年度	接頭語	ファミリー名	検知率［%］
2012	トロイの木馬	Ransom	88.9
		Generic	100
2013	バックドア	Win32	100
		Win32	100
		Win32	100
		Win32	100
		Win32	100
	トロイの木馬	Dropper	100
2014	トロイの木馬	VILSEL	100
		VILSEL	100
		Kryptik	83.3
		Kryptik	59.5
	スパイウェア	Fareit	100

6. 考察

本章では 5 章で得られた実験結果について考察する．

6.1 CCC における検知結果の考察

本節では，CCC を利用した場合の結果について考察する．考察は SYN パケット数とパケットサイズの平均に着目する．SYN パケットは，提案手法と比較手法の両方で，検知精度の高い特徴量であった．また，パケットサイズの平均は，提案手法でのみ，検知精度の高い特徴量だった．2 種類の特徴量を検証することで，提案手法と比較手法の特徴量抽出手法の差異を考察する．

6.1.1 SYN パケット数についての考察

正常時学習データの SYN パケットの時間波形を**図 5** に示す．また，感染時学習データの SYN パケットの時間波形を**図 6** に示す．

図 5，図 6 より，正常時の SYN パケットは 1 パケットスロットに 2 パケット前後であった．それに対して，感染時の SYN パケットは 1 パケットスロットに 4 パケットのものがほとんどだった．トラヒックデータを調査したところ，感染時通信のほとんどが SYN flood による DoS 攻撃だった．このため，感染時通信では SYN パケットが非常に多いデータ分布となった．CCC の取得環境はサーバ型ハニーポットであり，攻撃者からの積極的な攻撃の収集を目的としている．そのため，このような DoS 攻撃の挙動が大量に収集されたと考えられる．

以上より，時間的な変化パターンを考慮しない場合でも，正常時と感染時でデータの分布が明確に異なっている．そのため，SYN パケット数は比較手法で高い検知精度となったと考えられる．

また，時間波形の概形も正常時と感染時で明確に異なることが分かる．そのため，時間的な変化パターンを考慮す

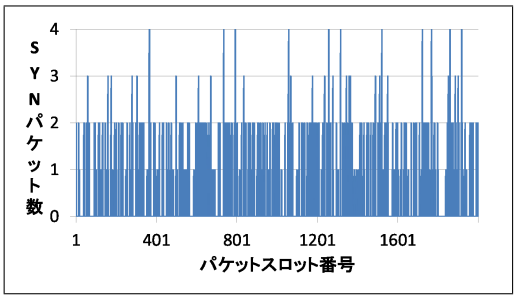

図 5 正常時学習データの SYN パケット数の時間波形
Fig. 5 Temporal waveform for occurences of SYN packet (normal).

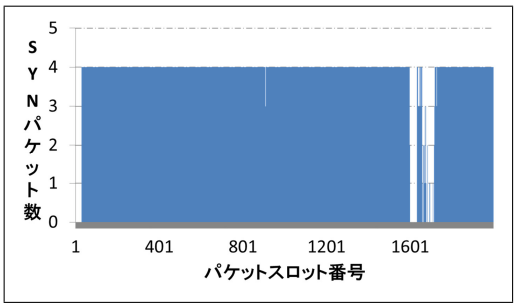

図 6 感染時学習データの SYN パケット数の時間波形
Fig. 6 Temporal waveform for occurences of SYN packet (malicious).

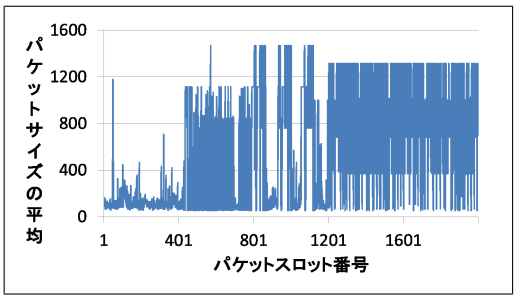

図 7 正常時学習データのパケットサイズの平均の時間波形
Fig. 7 Temporal waveform for average packet size (normal, training data).

る提案手法でも正常と感染を明確に分離できた．よって，SYN パケット数は提案手法でも高い検知精度となったと考えられる．

6.1.2 パケットサイズの平均についての考察

正常時学習データのパケットサイズの平均の時間波形を**図 7** に示す．また，感染時学習データの時間波形を**図 8** に，正常時テストデータの時間波形を**図 9** に示す．

正常時学習データでは，パケットサイズの平均値が 1 kByte 以上のパケットスロットの数が，全スロットの 90%以上を占めていた．これは，サーバとの SSH 通信が頻繁に実施されていたためである．

感染時学習データでは，パケットサイズの平均値が 60 Byte 程度のパケットスロットの数が，全体の 98%以上を占めていた．これは，前述の SYN flood 攻撃で送信されていたパケットのパケットサイズが 60 Byte 程度であるからである．

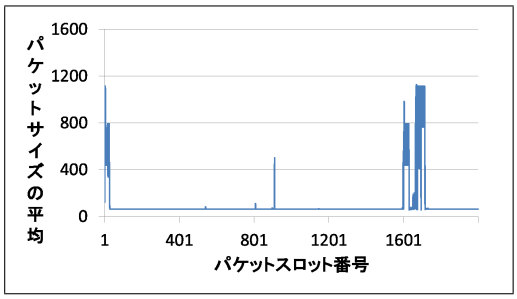

図 8 感染時学習データのパケットサイズの平均の時間波形
Fig. 8 Temporal waveform for average packet size (malicious, training data).

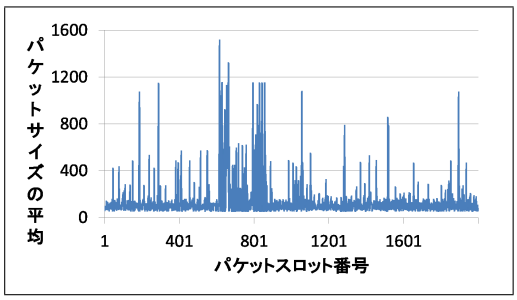

図 9 正常時テストデータのパケットサイズの平均の時間波形
Fig. 9 Temporal waveform for average packet size (normal, test data).

正常時テストデータでは，パケットサイズの平均値が 1 kByte 以上のパケットスロットの数が，全体の 60%以上を占めていた．このデータ分布は，正常時学習データとは異なっていることが分かる．これは，正常時学習データではつねに行われた SSH 通信が，テストデータでは長時間行われなかったことが原因と考えられる．

以上より，正常時学習データと感染時学習データは明確に異なることが分かる．しかし，正常時学習データと正常時テストデータでもデータ分布が異なっていたため，比較手法の TNR では誤検知が多発したと考えられる．

それに対して，時間波形の概形を比較すると，図 8 は図 9，図 7 とは明確に異なることが分かる．これは，時間的な変化パターンを考慮した場合，感染時と正常時の差異をより明確にとらえられるということを指す．そのため，提案手法では，TPR・TNR ともに高い検知精度となったと考えられる．

以上より，比較手法はデータが明確に分離できない場合，誤検知が大量に発生した．これに対して，提案手法では時間的な変化パターンを考慮することで，データの差異をより強調することができる．これにより，マルウェアをより正確に検知できる可能性がある．

6.2 D3M における TPR の特徴量の有効性考察

提案手法の TPR において，最も優秀な検知精度だった PSH パケット数について考察する．比較手法において，本特徴量の 3 年間の平均 TPR は 47.1%だった．それに対し，提案手法の平均 TPR は 91.1%と大幅に向上した．この理由を調査するため，感染時トラヒックデータを調査し，考察する．考察は時間的な変化パターンを考慮しない場合，時間的な変化パターンを考慮する場合，2 つの観点から行う．

6.2.1 提案手法による特徴量の差異について

PSH パケット数について，次の 2 つの視点から，提案手法によりどの程度影響があったかを分析した．

(a) 正常時通信のコードブックと感染時通信のコードブックが LPC ケプストラムを用いた提案手法によりどのくらい改善されたか．

(b) 感染時通信のテストデータと正常時・感染時通信のコードブックとのユークリッド距離がどのくらい改善されたか．

(a) について説明する．提案手法は，正常時と感染時それぞれについて LPC ケプストラム係数を並べたベクトルによってコードブックを求めた．比較手法は，スロットから得られる PSH パケット数をベクトルとしてコードブックを求めた．正常時通信と感染時通信のコードブック間のユークリッド距離の最小値を調べ，分散で正規化した．比較手法によるデータの分散値は 1.2 となり，提案手法によるデータの分散値は 0.39 となった．これらの数値を基に計算すると，提案手法における正常と感染のコードブック間のユークリッド距離は 0.64 となり，比較手法における正常と感染のコードブック間のユークリッド距離は 0.12 となった．提案手法により正常と感染のユークリッド距離は 5.3 倍に拡大したことが分かる．

(b) について説明する．感染時通信のテストデータと感染時通信のコードブックとのユークリッド距離を I，感染時通信のテストデータと正常時通信のコードブックとのユークリッド距離 N として，感染時通信のテストデータと正常時・感染時通信のコードブックとのユークリッド距離の比率 R_{TPR}

$$R_{TPR} = \frac{I}{N}$$

を計算し，比率 R_{TPR} がどのように改善されたかを示す．R_{TPR} が小さいほど手法の信頼性があり，$R_{TPR} < 1$ のとき，正しく検知できる．

図 10 に 3 年分の D3M データセット（33,952 スロット，530 フレーム）を用い，提案手法と比較手法の R_{TPR} を求め，ヒストグラムを示す．横軸が R_{TPR} で，縦軸が各比率 R_{TPR} における頻度の割合（全度数に対する該当比率の頻度の割合）を表す．

比較手法における 1 より小さい R_{TPR} の頻度の割合は 48.8%であった．提案手法の 1 より小さい R_{TPR} の頻度の割合は 84.2%であった．提案手法の頻度の割合が比較手法の頻度の割合より 35.4%増加しているので，比較手法より提案手法のほうが正しく検知できるようになっていることが分かる．

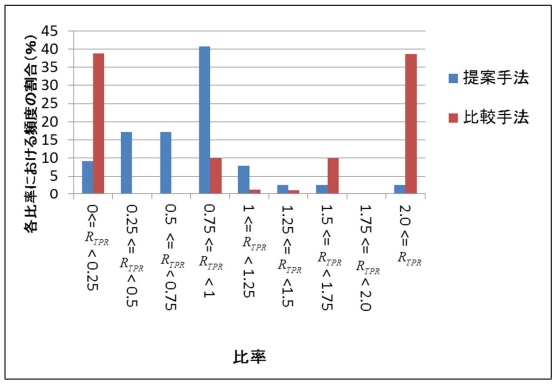

図 10 R_{TPR} のヒストグラム
Fig. 10 Histogram of R_{TPR}.

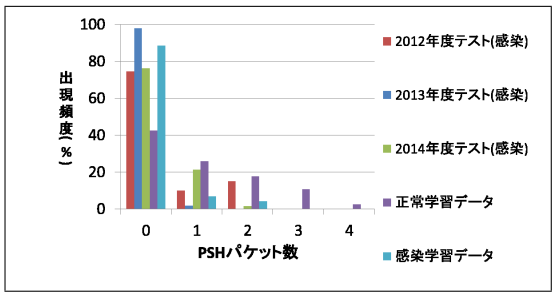

図 11 PSH パケット数のヒストグラム
Fig. 11 Histogram of PSH packet.

以上の 2 つの分析により，提案手法と比較手法の差異を示すことができると考える．

6.2.2 時間的変化を考慮しない場合のデータに関する考察

学習データと利用した感染時テストデータについて，PSH パケット数のヒストグラムを作成し，**図 11** に示す．ヒストグラムを利用することで，時間的な変化パターンを考慮しないデータ分布を知ることができる．

図 11 より，すべてのトラヒックデータにおいて，PSH パケット数が 0 のスロットが大部分を占めていると分かる．データの分布に差がないため，正常と感染を分離することはできない．そのため，比較手法では，多数の誤検知が発生したと考えられる．

6.2.3 時間的変化を考慮した場合のデータに関する考察

時間的な変化パターンに着目した場合のデータの差異を調査するため，各トラヒックデータの時間波形を調査した．その結果，各データの時間波形の傾向を 3 種類に分類できることが判明した．学習データの時間波形の分類を**表 9** に示す．また，テストデータの時間波形の分類を**表 10** に示す．

表 10 より，時間波形の分類と検知精度には関連性があることが分かる．正常時と感染時それぞれの時間波形を調

表 9 学習データの時間波形分類
Table 9 Temporal waveform classfication for training data.

接頭語	ファミリー名	時間波形分類
トロイの木馬	Generic	(3) ランダムな時間波形
	IRCBRUTE	(2) 定期的な時間波形
	FakeAV	(1) 散発的な時間波形

表 10 テストデータの時間波形分類と TPR
Table 10 Temporal waveform classfication and TPR for test data.

年度	ファミリー名	時間波形分類	TPR [%]
2012	Generic	(1) 散発的な時間波形	100
	Ransom	(2) 定期的な時間波形	88.9
2013	Win32	(1) 散発的な時間波形	100
	Win32	(1) 散発的な時間波形	100
	Win32	(1) 散発的な時間波形	100
	Win32	(1) 散発的な時間波形	100
	Win32	(1) 散発的な時間波形	100
	Dropper	(1) 散発的な時間波形	100
2014	VILSEL	(1) 散発的な時間波形	100
	VILSEL	(1) 散発的な時間波形	100
	Kryptik	(2) 定期的な時間波形	83.3
	Kryptik	(3) ランダムな時間波形	59.5
	Fareit	(2) 定期的な時間波形	100

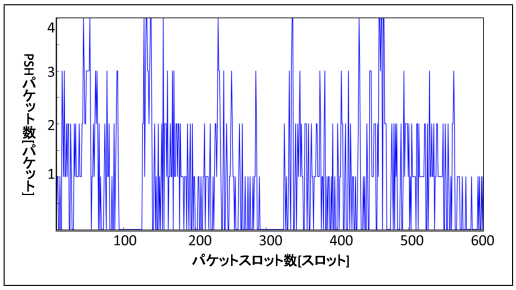

図 12 PSH パケット数の時間波形（正常時）
Fig. 12 Temporal waveform for occurences of PSH packet (normal).

査し，挙動の分析を行った．

6.2.3.1 正常時の時間波形

学習データにおける PSH パケット数の時間波形の一部を図 **12** に示す．このとき，正常時通信は http request 要求や SSL 通信，FTP 通信などを行っていた．これらの通信は，パケットに PSH フラグを立てる通信であり，ユーザが日常的に利用する通信である．特に http request 要求は，Web ページの画像ファイルの表示などに利用される通信である．そのため，正常時通信の PSH パケット数は，連続的な時間波形になったと考えられる．

6.2.3.2 感染時の散発的な時間波形

この時間波形の例として，2013 年の Dropper 検体の時間波形を図 **13** に示す．この検体を調査したところ，散発的に http request 要求を行っていた．この http request 要求では，拡張子を偽造した js ファイルのダウンロードなどを行っていた．これは，他の悪性コードのダウンロードだと考えられる．

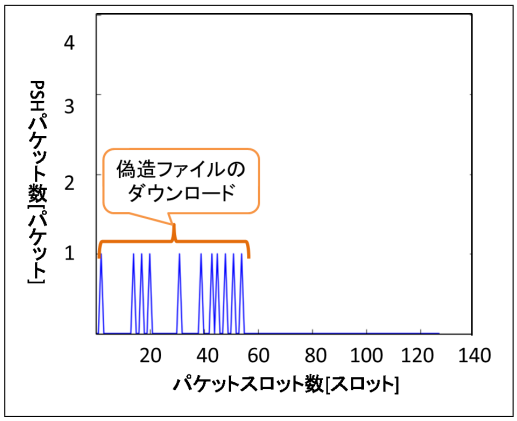

図 13 感染時の散発的な時間波形の一例
Fig. 13 Example of sporadic temporal waveform (infected).

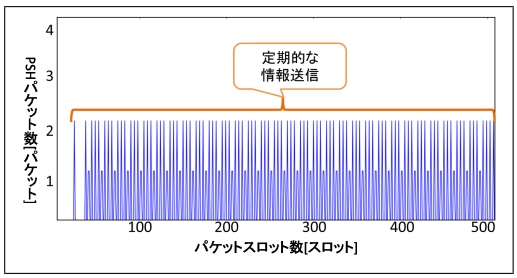

図 14 感染時の定期的な時間波形の一例
Fig. 14 Example of regular temporal waveform (infected).

このような時間波形は，ユーザが通常の操作をしても，発生しないと考えられる．実際，今回の正常時通信において，このような時間波形は確認できなかった．そのため，散発的な時間波形は，感染時に固有の時間波形であると考えることができる．以上より，散発的な時間波形に分類される検体は，正常とは異なる時間波形であるため，高い TPR になったと考えられる．

6.2.3.3 感染時の定期的な時間波形

この時間波形の例として，2012 年の Ransom 検体の時間波形を図 **14** に示す．この検体を調査したところ，定期的に http request 要求を行い，外部に対して情報を送信していた．

このような周期性のある時間波形をユーザが発信することは難しい．これは，プログラムで自動的に通信を行う，感染時に固有の時間波形であると考えられる．以上より，定期的な時間波形に分類される検体は，正常とは異なる時間波形であるため，高い TPR になったと考えられる．

しかし，表 8 より一部の検体では，定期的な時間波形と分類される検体でも，誤検知が発生していた．この原因を調査するため，検体ごとに検知に失敗したフレームの時間波形を調査した．その結果，該当するフレームでは，実行

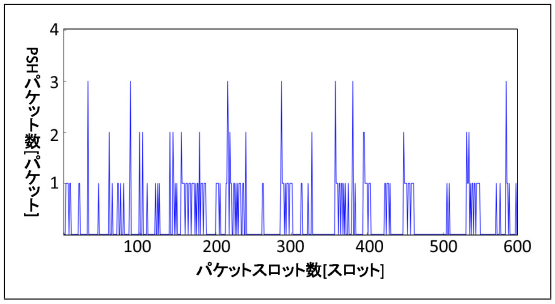

図 15 感染時のランダムな時間波形の一例
Fig. 15 Example of random temporal waveform (infected).

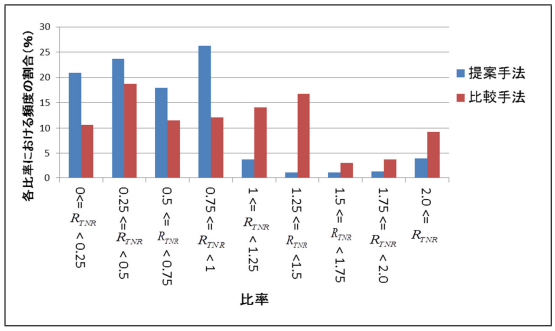

図 16 R_{TNR} のヒストグラム
Fig. 16 Histogram of R_{TNR}.

ファイルのダウンロードなど，非定期的な通信挙動が行われていた．これにより，定期的な波形とは異なる時間波形となってしまい，誤検知が発生したと考えられる．

6.2.3.4 感染時のランダムな時間波形

この時間波形の例として，2014 年の Kryptik 検体の時間波形を**図 15** に示す．この検体の挙動を調査したところ，複数の IP アドレスに対して，同時に http request 要求を行っていた．このため，通信が重なりあい，複雑な時間波形になったと考えられる．

このような時間波形には一定の規則性がないため，学習データとは異なる時間波形となる．そのため，2014 年の Kryptik 検体はきわめて低い TPR になったと考えられる．なお，この 2014 年の Kryptik 検体の総パケットは，2014 年の他のマルウェア検体よりも非常に多い．Kryptik 検体の TPR（59.5%）が低いため，2014 年の TPR が低くなった．

以上より，時間的な変化パターンを考慮しない場合，PSH パケット数の感染時通信と正常時通信でデータ分布の差異が少なかった．そのため，比較手法では誤検知が多い結果となった．

それに対して，時間的な変化パターンを考慮した場合，正常時と感染時の時間波形の概形が大きく異なっていた．これは，マルウェアが特有の通信挙動を行っていたためである．LPC ケプストラム分析により，この時間的な変化パターンの差異をとらえることができたので，提案手法では TPR が向上したと考えられる．

6.3 D3M における TNR の特徴量の有効性考察

提案手法の TNR において，最も優秀な検知精度だった到着間隔の最小値について考察する．本特徴量の比較手法の TNR は平均で 53.3%だった．それに対して，提案手法の TNR は平均で 89.2%と大幅に向上した．この理由を調査するため，6.2 節と同様にトラヒックデータの解析を行った．

6.3.1 提案手法による特徴量の差異について

到着間隔の最小値について，次の 2 つの視点から，提案手法によりどの程度影響があったかを分析した．

(a) 正常時通信のコードブックと感染時通信のコードブックが LPC ケプストラムを用いた提案手法によりどのくらい改善されたか．

(b) 正常時通信のテストデータと正常時・感染時通信のコードブックとのユークリッド距離がどのくらい改善されたか．

(a) について説明する．提案手法は，正常時と感染時それぞれについて LPC ケプストラム係数を並べたベクトルによってコードブックを求めた．比較手法は，スロットから得られる到着間隔の最小値をベクトルとしてコードブックを求めた．正常と感染のコードブック間のユークリッド距離の最小値を調べ，分散で正規化した．比較手法によるデータの分散値は 428 となり，提案手法によるデータの分散値は 0.079 となった．図 18，図 20 に示しているように到着間隔の最小値は分散が大きいデータである．これらの数値を基に計算すると，提案手法における正常時通信のコードブックと感染時通信のコードブック間のユークリッド距離は 8.1 となり，比較手法における正常時通信のコードブックと感染時通信のコードブック間のユークリッド距離は 2.8×10^{-5} となった．提案手法により正常時通信のコードブックと感染時通信のコードブック間のユークリッド距離は 2.9×10^{5} 倍に拡大したことが分かる．

(b) について説明する．正常時通信のテストデータと正常時通信のコードブックとのユークリッド距離を N，正常時通信のテストデータと感染時通信のコードブックとのユークリッド距離 I として，感染時通信のテストデータと正常時・感染時通信のコードブックとのユークリッド距離の比率 R_{TNR}

$$R_{TNR} = \frac{N}{I}$$

を計算し，比率 R_{TNR} がどのように改善されたかを示す．R_{TNR} が小さいほど手法の信頼性があり，$R_{TNR} < 1$ のとき，正しく検知できる．

図 16 に 3 年分の D3M データセット（101,278 スロット，1,582 フレーム）について，提案手法と比較手法の R_{TNR} を求め，ヒストグラムを示す．横軸が R_{TNR} で，縦軸が各比率 R_{TNR} における頻度の割合（全度数に対する該当比率

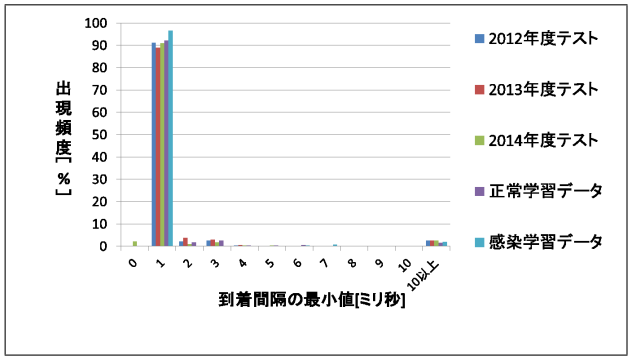

図 17 到着間隔の最小値のヒストグラム
Fig. 17 Histogram of minimum arrival time.

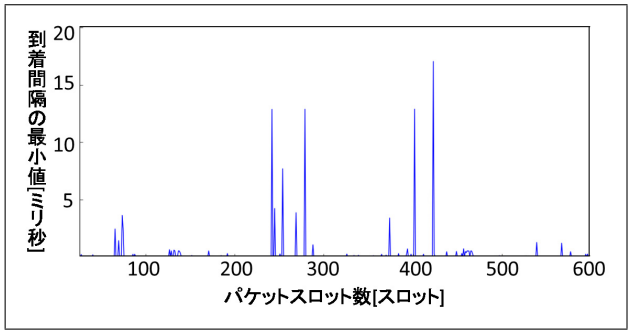

図 18 正常時学習データの到着間隔の最小値の時間波形
Fig. 18 Temporal waveform for minimum arrival time (normal, training data).

の頻度の割合）を表す．

比較手法における 1 より小さい R_{TNR} の頻度の割合は 52.9%であった．提案手法の 1 より小さい R_{TNR} の頻度の割合は 88.8%であった．提案手法の頻度の割合が比較手法の頻度の割合より 35.9%増加しているので，比較手法より提案手法のほうが正しく検知できるようになっていることが分かる．

以上の 2 つの分析により，提案手法と比較手法の差異を示すことができると考える．

6.3.2 時間的変化を考慮しない場合のデータに関する考察

学習データと利用した正常時テストデータについて，到着間隔の最小値のヒストグラムを作成し，**図 17** に示す．

図 17 より，すべてのトラヒックデータで，到着間隔の最小値が 1 ミリ秒程度のとき，最も出現頻度が高いことが分かる．データに差異がないため，正常時と感染時の分離を行うことはできない．そのため，比較手法では誤検知が多発したと考えられる．

6.3.3 時間的変化を考慮した場合のデータに関する考察

正常時学習データ到着間隔の最小値の時間波形の一例を**図 18** に示す．また，感染時学習データの時間波形の一例を**図 19** に，正常時テストデータの時間波形を**図 20** に示す．

図 18，図 19，図 20 より，感染時の時間波形の概形は，正常時とは異なることが分かる．具体的には，正常時通信

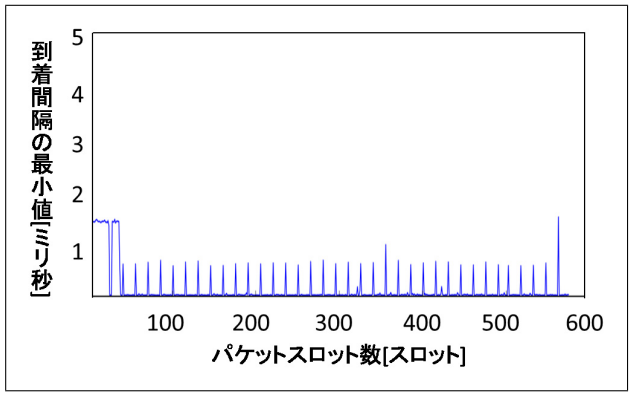

図 19 感染時学習データの到着間隔の最小値の時間波形
Fig. 19 Temporal waveform for minimum arrival time (infected, training data).

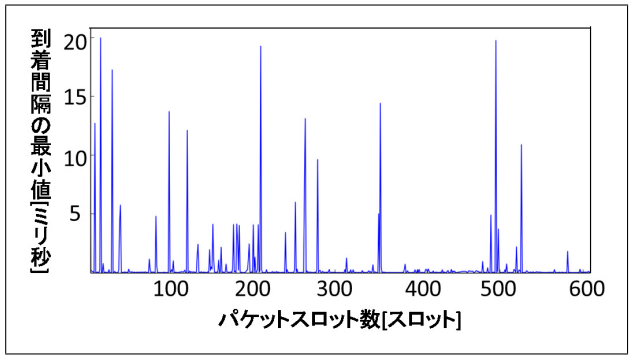

図 20 正常時テストデータの到着間隔の最小値の時間波形
Fig. 20 Temporal waveform for minimum arrival time (normal, test data).

は突発的に到着間隔の最小値が大きくなった．それに対して，感染時は到着間隔の最小値が 2 ミリ秒を超えることはなかった．これは，正常時の通信がユーザの行動次第で変化するのに対し，マルウェアの通信は攻撃者によって定義されたプログラムに従い，自動的に行われたためだと考えられる．

以上より，時間的な変化パターンを考慮しない場合の PSH パケット数では，感染時と正常時のデータ分布において差異が確認できなかった．そのため，比較手法では誤検知が多い結果となった．それに対して，時間的な変化パターンを考慮した場合，正常時と感染時の時間波形の概形が大きく異なっていた．LPC ケプストラム分析により，この時間的な差異をとらえることができたので，提案手法では TPR が向上したと考えられる．

6.4 使用する特徴量の選択について

トラヒックデータは環境に依存する部分があるため，他の環境では最良となる特徴量は異なる可能性がある．今回の評価においても，5.1 節で示した表 4，表 5（CCC での評価）と 5.2 節で示した表 6，表 7（D3M での評価）は異な

る環境でのデータであるが，最良となった特徴量は異なっている．それをふまえ，以下に特徴量の選び方について考え方を説明する．

他の環境で感染検知をする場合には，あらかじめ当該環境で正常時通信を取得し，別途 MWS や MALWARE-TRAFFIC-ANALYSIS.NET [16] などが提供する感染時通信を入手する．その際，長期的に安定して検知することを考慮して 4.3 節で示していているようにできるだけ長期にわたって通信データを取得する．たとえば，4.3.2 項では正常時通信，感染時通信ともに 2012 年，2013 年，2014 年に取得した．

取得したデータを用いて有効な特徴量を選ぶ．4.5 節で示した TPR，TNR を指標として 5.1 節，5.2 節で示したように各特徴量の TPR，TNR を網羅的に調べる．各取得時期の TPR，TNR の平均値（5.2 節では 3 年間の平均値に相当）が一番高いものを最良の特徴量として選ぶ．

7. まとめ

本研究では，LPC ケプストラム分析を利用した特徴量抽出手法を提案し，検知精度を評価した．実験は，異なる環境の感染時通信と正常時通信を用い，様々なデータに対しての LPC ケプストラム分析の有効性を検証している．

実験手法によって有効な特徴量は変化したものの，検知精度の向上を複数の特徴量で確認することができた．これは，時間的な変化パターンを考慮したことにより，正常時と感染時の通信の差異をより明確に分離することができたことを示している．

今後は，さらなる検知精度の向上を目的として，複数の特徴量を組み合わせた検知手法や，通信挙動ごとに学習を行い，それぞれの検知精度の評価を行っていく．

参考文献

[1] 独立行政法人情報処理推進機構：標的型攻撃メールの傾向と事例分析 (2013)，入手先 ⟨https://www.ipa.go.jp/security/technicalwatch/20140130.html⟩（参照 2014-11-20）．

[2] Kaspersky: Kaspersky Security Bulletin 2013. Overall Statistics for 2013, available from ⟨http://media.kaspersky.com/pdf/KSB_2013_EN.pdf⟩ (accessed 2014-11-10).

[3] the guardian: Antivirus software is dead, says security expert at Symantec, available from ⟨http://www.theguardian.com/technology/2014/may/06/antivirus-software-fails-catch-attacks-security-expert-symantec⟩ (accessed 2014-11-23).

[4] 畑田充弘，稲積考紀，有川隼ほか：サンドボックス解析結果に基づく URL ブラックリスト生成方式に関する事例調査，電子情報通信学会技術研究報告，Vol.114, No.117, pp.309–314 (2014).

[5] 石井健一郎，上田修功，前田英作ほか：わかりやすいパターン認識，pp.1–2, オーム社 (2008).

[6] 宮本貴朗，小島篤博，泉正夫ほか：SVM を用いたネットワークトラヒックからの異常検出，電子情報通信学会論文誌，Vol.J87-B, No.4, pp.593–598 (2004).

[7] 川元研治，市田達也，市野将嗣ほか：マルウェア感染検知のための経年変化を考慮した特徴量評価に関する一考察，コンピュータセキュリティシンポジウム 2011 論文集，Vol.2011, No.3, pp.277–282 (2011).

[8] Soniya, B. and Wilscy, M.: User Traffic Profile for Traffic Reduction and Effective Bot C&C Detection, International Journal of Network Security, Vol.16, No.1, pp.46–52 (2014).

[9] 鈴木男人，小池愛理，鈴木孝之ほか：サイバー攻撃検知に関わるパラメータ抽出法の検討，情報処理学会第 76 回全国大会講演論文集，Vol.2014, No.1, pp.625–626 (2014).

[10] 吉井貞熙：ディジタル音声処理，pp.37–48，東海大学出版会 (1985).

[11] 秋山満昭，神薗雅紀，松木隆宏ほか：マルウェア対策のための研究用データセット〜MWS Datasets 2014〜，情報処理学会研究報告コンピュータセキュリティ（CSEC），Vol.2014-CSEC-66, No.19, pp.1–7 (2014).

[12] Linde, Y., Buzo, A. and Gray, R.: An Algorithm for Vector Quantization, *IEEE Trans, Comm.*, Vol.28, No.1, pp.84–95 (1980).

[13] IIJ ブロードバンドトラフィックレポート，Vol.8, 12, 16, 20, 24（参照 2015-03-29）．

[14] TrendMicro: Threat Encyclopedia, available from ⟨http://www.trendmicro.com/vinfo/us/threat-encyclopedia/#malware⟩ (accessed 2014-10-25).

[15] Kaspersky: Viruslist.com, available from ⟨http://www.viruslistjp.com/⟩ (accessed 2014-10-25).

[16] MALWARE-TRAFFIC-ANALYSIS.NET, available from ⟨http://www.malware-traffic-analysis.net/⟩ (accessed 2015-03-29).

岩野 透

2014 年電気通信大学情報理工学部総合情報学科卒業．現在，同大学大学院情報理工学研究科総合情報学専攻博士前期課程在学中．

吉浦 裕 （正会員）

1981 年東京大学理学部情報科学科卒業．日立製作所を経て，2003 年より電気通信大学勤務．現在，情報理工学研究科教授．情報セキュリティ，プライバシ保護の研究に従事．博士（理学）．日立製作所社長技術賞（2000 年），情報処理学会論文賞（2005 年，2011 年），日本セキュリティ・マネジメント学会論文賞（2011 年），システム制御情報学会産業技術賞（2005 年），IEEEIIH-MSP Best Paper Award（2006 年）等受賞．電子情報通信学会，日本セキュリティ・マネジメント学会，システム制御情報学会，人工知能学会，IEEE 各会員．

畑田 充弘 （正会員）

1978 年生．2003 年早稲田大学大学院理工学研究科修士課程修了．同年 NTT コミュニケーションズ（株）入社．以来，マルウェア対策をはじめとするネットワークセキュリティの研究開発に従事．早稲田大学大学院基幹理工学研究科博士後期課程在学中．電子情報通信学会会員．

市野 将嗣 （正会員）

2003 年早稲田大学理工学部電子・情報通信学科卒業．2008 年同大学大学院理工学研究科博士課程修了．2007 年日本学術振興会特別研究員．2009 年早稲田大学大学院基幹理工学研究科研究助手．2010 年同大学メディアネットワークセンター助手．2011 年電気通信大学大学院情報理工学研究科助教．バイオメトリクス等パターン認識，ネットワークの品質と安全性を考慮したトラヒック分類に関する研究に従事．博士（工学）．電子情報通信学会，IEEE 各会員．