マルウェア感染後のHTTP通信の特徴に基づく異常検知

中久木 達哉[1,a)]　青木 茂樹[1,b)]　宮本 貴朗[1]

概要：近年頻繁に観測される標的型攻撃では，組織内ネットワークに侵入するために入念な調査を行い，侵入可能な方法を探し出し，通常の通信に紛れて侵入する．そのため，マルウェアの侵入を防ぐための対策だけでは十分ではない．そこで，マルウェア感染後の活動を検知する手法の重要性が高まっている．標的型攻撃のマルウェアは感染後に C2 サーバと決まった書式で通信を行うことが多く，セッション毎のパケットサイズ等の変化が少ないと考えられる．また，HTTP プロトコルでのマルウェアの通信では，検知を逃れるために User-Agent や Referer に特定の文字列を設定している可能性が高いと考えられる．本研究ではパケットのヘッダからセッション毎のパケットサイズ等の特徴を抽出し GMM で学習する．その後，観測されたセッションの特徴量及び User-Agent，Referer の各クラスにおける出現確率をベイズの定理で統合することで不審な通信を検知する手法を提案する．MWS データセットを対象に実験を実施し提案手法の有効性を確認した．

キーワード：Gaussian Mixture Model，User-Agent，Referer，ベイズ確率，MWS Dataset

Anomaly Detection Based on Characteristics of HTTP Communication after Malware Infection

Tatsuya Nakakuki[1,a)]　Shigeki Aoki[1,b)]　Takao Miyamoto[1]

Abstract: APT(Advanced Persistent Threats) have been observed frequently, in order to infiltrate an organization network, attackers investigate carefully and find ways to penetrate a network. They infiltrate the organization network through normal communication. Thus, it is not enough to prevent malware infiltration. And methods for detecting activities after malware infection are becoming important. Malwares often communicate with C2 server in fixed formats after infection, and it is considered that changes in packet size and so on, between sessions are small. In addition, there is a high possibility that a specific string is set in the User-Agent or Referer to evade detection in HTTP protocol malware communication. In this paper, we propose a method for detecting suspicious communication by integrating session features, occurrence probabilities of User-Agent and Referer in classes using Bayesian theorem. In experiments, we confirmed the effectiveness of our method using MWS dataset.

Keywords: Gaussian Mixture Model，User-Agent，Referer，Bayesian Probability，MWS Dataset

1. はじめに

近年，サイバー攻撃が増加傾向にあり，更に攻撃手法が巧妙化している．従来のサイバー攻撃は，不特定多数のホストの中から，サイバー攻撃に対する対策が不十分なホストを探索して攻撃を行っていた．そのため，組織や団体はファイアウォールや侵入検知システムを設置し，ホストではウイルス対策ソフトを稼働させることで，組織内ネットワークやホストへの侵入を試みる通信やネットワーク内の不審な通信を検知してきた．しかし，近年頻繁に観測されている標的型攻撃は従来のサイバー攻撃とは異なり，これまでの対策では攻撃を防ぐことが難しい．

[1] 大阪府立大学大学院人間社会システム科学研究科
Graduate School of Humanities and Sustainable System Sciences, Osaka Prefecture University
[a)] saa01178@edu.osakafu-u.ac.jp
[b)] aoki@kis.osakafu-u.ac.jp

標的型攻撃は，情報の窃取を目的に特定の組織内ネットワークに侵入した後，ネットワーク内部での攻撃を繰り返しつつ，長期間に亘って機密情報を窃取し続ける攻撃である [1]．従来のサイバー攻撃とは違い，攻撃目標を選定するまでに入念な調査活動を行い計画を立案する．調査活動において，攻撃目標とする組織が行うセキュリティ対策や組織の脆弱性を調査して，攻撃を実行する．

従来から用いられているウイルス対策ソフトや侵入検知システムではパターンマッチングによりサイバー攻撃の侵入を防いでいる．そのため，すでにパターンファイルに記録された特徴を持つ攻撃に対しては高い確率で検知することができるが，新たな攻撃に対してはパターンファイルにその特徴が記録されていないため検知することができない．標的型攻撃を目論む攻撃者は，組織が設置しているウイルス対策ソフトや侵入検知システムを調査し，パターンファイルに存在しない攻撃手法を計画する．そのため，パターンマッチング手法では標的型攻撃を防ぐことは難しい．

そこで，監視対象の異なるセキュリティ対策を何重にも組み合わせることで，侵入や情報漏洩の可能性を下げる多層防御技術が重要になってきている [2]．多層防御技術は，侵入対策，拡大対策，漏洩対策の 3 段階に分けて考えられることが多い．侵入対策では，ファイアーウォールや侵入検知システムを用いて，ネットワークの外部と内部の境界を監視して不正なアクセスを遮断している．拡大対策では，振る舞い検知型のソフトウェアにより，ネットワーク内における不審な通信を監視している．また，各ホストにおけるパスワードの強化やパッチの監視なども拡大対策に含まれる．漏洩対策では，ファイアーウォールや侵入検知システムによる監視に加えて，アクセスログや送信ログの取得・監視などを行うことによって，機密情報が外部に送信されることを防ぐ役割を担っている．

拡大対策として，マルウェア感染後の活動を検知する重要性が高まっている．マルウェア感染後の活動には，外部からの侵入経路 (バックドア) の構築や，外部から端末を遠隔操作するためにボットなどで行われる外部サーバとの定期的な通信 (ビーコン通信) などが含まれる．正規の Web サーバとの通信であれば，同一サーバとのセッションであっても，閲覧するページによってページのサイズ等が異なり，メールサーバーとの通信であれば，送受信するメールのサイズによってセッションに含まれるパケット数やパケットサイズ等が異なる．一方，C2 サーバとのビーコン通信等の場合，情報を決まった書式で送信すると考えられるため，セッション毎のパケット数やパケットサイズ等の変化が少ないと考えられる．

また，HTTP プロトコルでのマルウェアの通信では，検知を逃れるために，特定のブラウザに偽装した User-Agent や独自の User-Agent を用いる可能性がある [3]．通常の HTTP 通信では，HTTP 要求を送る際に Referer に参照元の Web ページの URL を設定するが，マルウェアは Referer に値を設定しなかったり，Referer に特定の文字列を設定していたりする可能性が高いと考えられる．この様な特徴に注目して，Blue Coat Web Filter[4]，McAfee Web Gateway[5] 等では，マルウェアに感染した端末の検知率を向上させるために，User-Agent による異常検知の仕組みが実装されている．Blue Coat Web Filter は，特定の User-Agent 以外の通信をブロックしたり，リンク先のサイト内容から悪性なサイトを検知したりすることで組織内ネットワークのセキュリティを保つシステムである．McAfee Web Gateway は，ウェブゲートウェイが記録する User-Agent から, 安全な User-Agent のリストを作成し，リスト外の User-Agent を用いた通信をブロックすることで組織内ネットワークのセキュリティを保つシステムである．また文献 [6] では，Referer の有無や不審な Referer からホームページのリンクの深さを数値化し，設定値以上の場合に，ブラウザの自動通信を遮断することでマルウェアへの感染を未然に防いでいる．

本研究では, パケットのヘッダから得られた特徴と HTTP ヘッダ中の User-Agent，Referer の特徴を組み合わせることで標的型攻撃を検知する手法を提案する．パケットのヘッダからセッション毎に抽出した特徴を基にクラスタリングした後，各クラスタにおける User-Agent，Referer の出現確率をベイズの定理で統合し標的型攻撃を検知する．実験では，MWS2018 データセットの BOS データセット [7] を用いて本手法の有効性を確認した．以下，2 節で関連研究について述べ，3 節で提案手法について説明する．4 節で実験と考察について述べ，5 節でまとめる．

2. 関連研究

本研究に関連する従来研究として，ヘッダからセッション毎に抽出した特徴を用いた異常検知手法である文献 [8]，[9]，[10] と User-Agent を用いた異常検知手法である文献 [11]，Referer を用いた不審通信分析手法である文献 [12] について述べる．文献 [8] ではマルウェア感染ホストが行う通信と正常な通信の違いに注目し，パケットのヘッダから不正アクセスを検知する手法が提案されている．文献 [9] では，トラフィックデータのヘッダからセッション毎に特徴を抽出してクラスタリングし，各ホストの通信のクラスタ遷移を比較することで未知の攻撃を検知する手法が提案されている．文献 [10] では，パケットのヘッダから得られた特徴を用いて標的型攻撃を検知する手法が提案されている．この手法ではまず，通常通信のトラフィックデータをセッションごとに分割し，パケット数やパケットサイズ等の特徴を抽出する．抽出した特徴を Mean-Shift 法でクラスタリングする．次に，通信相手がどのクラスタのセッションを何回使用しているかを調べ，通信相手の通信挙動を示す特徴ベクトルとし，通信相手をクラスタリングすることで

通常通信を学習する．新たなトラフィックデータについても同様の処理を行い，抽出した通信挙動が学習時の挙動と類似しない場合に，不審な通信挙動として検知する．これらの手法では，パケットのヘッダに特徴が表れる C2 サーバとのビーコン通信などを検知できると考えられるが，ビーコン通信の特徴は同一の Web ページを複数回閲覧している場合と類似するため，これらの特徴のみで高精度に標的型攻撃を検知することは難しいと考えられる．

マルウェアが行う C2 サーバとのビーコン通信は HTTP プロトコルを利用することが多いため，User-Agent，Referer に注目した手法が提案されている．文献 [11] では，マルウェア通信で利用された User-Agent と正常な通信の User-Agent との逸脱度を算出することでマルウェア通信を検知する手法が提案されている．この手法では，マルウェアが行う通信で使用されている User-Agent と正常な通信で使用されている User-Agent の Levenshtein 距離を求めることで逸脱度を算出している．Levenshtien 距離は，編集距離とも呼ばれ，文字列 A から文字列 B を作るために要素の文字を最小で何回「挿入」「削除」「置換」する必要があるかを示す数を表す．文献 [12] では，Referer に特定の文字列が設定されている場合に不正リダイレクトとして検知する手法が提案されている．Referer に設定されているホスト名が，HTTP リクエストの Host ヘッダと一致しない場合や，ホワイトリストに存在しない場合に不正リダイレクトとして検知している．これらの手法では，パケットのヘッダのみに注目した手法では検知できなかった異常を検知することができる．しかし，User-Agent や Referer は容易に擬装できるため，これらの情報のみを利用した手法では，様々な標的型攻撃に対応することは難しい．

本研究では，ヘッダから得られる特徴とペイロードから得られる特徴の両方に注目し標的型攻撃を検知する手法を提案する．本手法により，ヘッダに特徴が表れる攻撃，ペイロードに特徴が表れる攻撃の両方を検知することができ，標的型攻撃を高精度に検知できると考えられる．

3. 提案手法

本稿では，ヘッダからセッション毎に抽出した特徴と User-Agent，Referer を組み合わせた異常検知手法を提案する．図 1 に手法の概要を示す．本手法は学習と検知の 2 つの処理に分かれている．まず学習時の処理では，学習用トラフィックデータから宛先ポート番号 80 番，送信元ポート番号 80 番のパケットを抽出する．パケットから IP アドレスとポート番号の組み合わせを取得しセッションを抽出する．抽出したパケットのヘッダからセッション毎にパケットサイズ等の特徴量を抽出し，更にペイロードから User-Agent，Referer の情報を抽出する．次に，ヘッダからセッション毎に抽出した特徴量のみを用いてセッションをクラスタリングする．クラスタリング後，各クラスタに

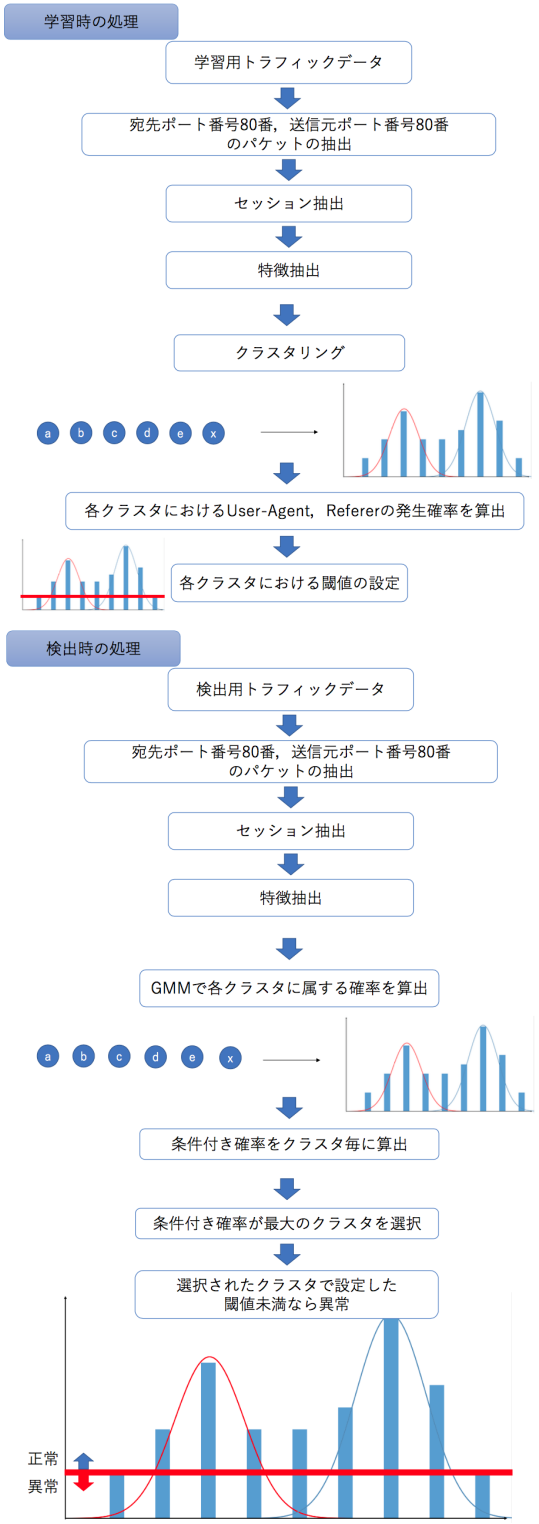

図 1　提案手法の概要

Fig. 1　Outline of Proposed method.

おいて算出した User-Agent，Referer の発生確率を基に各クラスタ毎に閾値を設定する．検知時の処理では，検知用トラフィックデータから，同様の手順で特徴量を抽出し，GMM を用いて各クラスタに属する確率を算出する．その後，各クラスタにおける User-Agent，Referer の発生確率とベイズの定理で統合し，条件付き確率を算出する．各ク

表 1　特徴量の一覧
Table 1　List of feature.

上りパケット数	下りパケット数
上り FIN パケット数	下り FIN パケット数
上り SYN パケット数	下り SYN パケット数
上り RST パケット数	下り RST パケット数
上り PSH パケット数	下り PSH パケット数
上り ACK パケット数	下り ACK パケット数
上り URG パケット数	下り URG パケット数
上り FIN&ACK パケット数	下り FIN&ACK パケット数
上り RST&ACK パケット数	下り RST&ACK パケット数
上り SYN&ACK パケット数	下り SYN&ACK パケット数
上り PSH&ACK パケット数	下り PSH&ACK パケット数
上りパケットサイズ平均	下りパケットサイズ平均
上りパケットサイズ最大	下りパケットサイズ最大
上りパケットサイズ最小	下りパケットサイズ最小
上りパケットサイズ合計	下りパケットサイズ合計
上りパケットサイズ分散	下りパケットサイズ分散
上り TTL 値平均	下り TTL 値平均
上り TTL 値分散	下り TTL 値分散
User-Agent	Referer

ラスタにおける条件付き確率のうち，最大値をとるクラスタを選択し，そのクラスタに属する確率が閾値未満の場合に，不審な通信挙動として検知する．

3.1　学習処理

3.1.1　セッションと特徴量の抽出

学習用トラフィックデータから宛先ポート番号 80 番，送信元ポート番号 80 番の HTTP パケットを取得し，送信元 IP アドレス，宛先 IP アドレス，送信元ポート，宛先ポートの組み合わせを抽出する．抽出した IP アドレスとポートの組からセッション単位でパケットを抽出する．抽出した 4 つの情報では送信元から宛先へのパケットしか抽出できないため，送信元と宛先の IP アドレスとポートを入れ替えたパケットを合わせて抽出する．そして，抽出した双方向のパケットをパケット到着時間の情報を基にソートすることでセッションを抽出する．

抽出した HTTP パケットからパケットサイズや TTL 値等を取得し，HTTP リクエストから User-Agent，Referer を取得し，セッション毎に表 1 に示す 38 種類の特徴量を抽出する．

3.1.2　GMM によるクラスタリング

抽出した特徴量のうち，User-Agent と Referer を除いた 36 種類の特徴量のみを用いてクラスタリングすることで，セッションの特徴を学習する．まず，抽出した 36 種類の特徴量は高次元であるため，次元削減手法の一つである主成分分析を用いて次元を圧縮する．ここでは累積寄与率 85% 以上となる最小の次元数を用いる．次に，次元削減後の特徴量の分布は複数の正規分布の重ね合わせで表すことができると仮定し，得られた特徴量を GMM を用いてモデル化しクラスタリングする．GMM は式 (1) で表すことができ，与えられたデータの分布を複数の正規分布で表すことでデータの確率密度関数を得ることができる手法である．ここで，$\boldsymbol{x}$ はパケットのヘッダからセッション毎に抽出した特徴ベクトルであり，$\boldsymbol{\pi}_k$ は混合係数で各正規分布の重みを表し，K は正規分布の数，$\boldsymbol{\mu}_k$ は各正規分布の中心を表し，$\boldsymbol{\sigma}_k$ は各正規分布の広がり方を表す．

$$P(\boldsymbol{x}) = \sum_{k=1}^{K} \boldsymbol{\pi}_k N(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k), \sum_{k=1}^{K} \boldsymbol{\pi}_k = 1 \tag{1}$$

正確な確率密度関数を得るために $\boldsymbol{\pi_k}$，$\boldsymbol{\mu_k}$，$\boldsymbol{\sigma_k}$ の 3 つのパラメータの最適値を EM(Expectation Maximization) アルゴリズム [14] を用いて求める．EM アルゴリズムはパラメータの最尤推定量を求めるための計算手法である．以下の手順で GMM のパラメータの最適値を EM アルゴリズムで推定する．

Step1：$\boldsymbol{\pi}_k$，$\boldsymbol{\mu}_k$，$\boldsymbol{\sigma}_k$ に初期値を与える．
Step2 (Estep)：ベイズの定理を用いて，式 (2) で $\boldsymbol{x}$ の負担率 γ を求める．負担率は $\boldsymbol{x}$ がどの正規分布にどれくらいの確率で属するのかを表している．

$$\gamma(z_k) = P(z_k = 1|\boldsymbol{x}) = \frac{\boldsymbol{\pi}_k N(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k)}{\sum_{j=1}^{K} \boldsymbol{\pi}_j N(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\sigma}_j)} \tag{2}$$

Step3 (Mstep)：負担率を用いて式 (3)，式 (4)，式 (5) で $\boldsymbol{\pi}_k$，$\boldsymbol{\mu}_k$，$\boldsymbol{\sigma}_k$ を更新する．ここで，N はデータの個数，$\boldsymbol{\mu}_{new\_k}$，$\boldsymbol{\sigma}_{new\_k}$，$\boldsymbol{\pi}_{new\_k}$ はそれぞれ更新後の $\boldsymbol{\mu}_k$，$\boldsymbol{\sigma}_k$，$\boldsymbol{\pi}_k$ を表す．

$$\boldsymbol{\mu}_{new\_k} = \frac{1}{\sum_{n=1}^{N} \gamma(z_k)\boldsymbol{x}_n} \sum_{n=1}^{N} \gamma(z_k)\boldsymbol{x}_n \tag{3}$$

$$\boldsymbol{\sigma}_{new\_k} = \frac{1}{\sum_{n=1}^{N} \gamma(z_k)\boldsymbol{x}_n} \sum_{n=1}^{N} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T \tag{4}$$

$$\boldsymbol{\pi}_{new\_k} = \frac{\sum_{n=1}^{N} \gamma(z_k)}{N} \tag{5}$$

パラメータの値が収束するまで Step2，Step3 を繰り返す．

また，学習の際に，ベイズ情報量基準 (BIC：Bayesian Information Criterion)[15] を用いて最適な正規分布の数を決定する．BIC を算出する際に必要な尤度 L を式 (6) で算出する．BIC は式 (7) で算出する．正規分布の数を 1 から増加させ，BIC が最小値を示す正規分布の数を学習で用いる．M は最適化を行うパラメータの個数を表し，ここでのパラメータは，$\boldsymbol{\pi}_k$，$\boldsymbol{\mu}_k$，$\boldsymbol{\sigma}_k$ の 3 つなので M は 3 である．

$$L = \sum_{j=1}^{N} \{\log \sum_{k=1}^{K} \boldsymbol{\pi}_k N(\boldsymbol{x}_k|\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k)\} \tag{6}$$

$$BIC = -2\log L + M\log N \tag{7}$$

クラスタリング終了後，クラスタ k における User-Agent U と Referer R の発生確率 $p(U,R|k)$ と User-Agent U と Referer R の組み合わせの発生確率 $p(U,R)$ を図 2 に示す様に算出する．図 2 の例では，クラスタ 1 に含まれる User-Agent と Referer の組は 20 組あり，User-Agent A，Referer a の組は 20 組中 2 組存在するため発生確率 $p(A,a|1) = 0.1$ となる．以上の処理を全ての組み合わせ，クラスタで行い各クラスタにおける User-Agent と Referer の発生確率を求める．また，図中下部に示す様に学習データ中の User-Agent と Referer の組が 100 組あり，User-Agent A と Referer a の組が 100 組中 40 組存在した場合，$p(A,a) = 0.4$ となる．以上の処理を全ての User-Agent と Referer の組み合わせで行う．

その後，各クラスタで異常検知時の閾値を決定する．クラスタ k に属する学習データの中で GMM で算出する確率 $p_k(\boldsymbol{x})$ の最小値を選択し，$p_{min}(k)$ とする．User-Agent U と Referer R の組の出現確率と $p_{min}(k)$ を式 (8) に示すベイズの定理で統合して $p(k|U,R)$ を算出する．クラスタ 1 に属するメンバの最小の確率 $p_{min}(1)$ が 0.8，$p(A,a|1) = 0.1$，$p(A,a) = 0.4$ である場合，$p(1|A,a) = 0.2$ となる．そして，全ての User-Agent U, Referer R の組み合わせで $p(k|U,R)$ を算出し，最小のものをクラスタ k における閾値 p_{th_k} とする．以上の処理を全てのクラスタで行ない，クラスタ毎に閾値を設定する．

$$p(k|U,R) = \frac{p(U,R|k)p_{min}(k)}{p(U,R)} \tag{8}$$

3.2 異常検知

3.1.2 節で学習した GMM，User-Agent と Referer の発生確率を基に，ネットワークの異常を検知する．新たな通信が観測された時，まず，新たなトラフィックデータに対して 3.1.1 節で述べた手法に基づきセッションを分割して，特徴量を抽出し，次元を削減する．その後，新たなトラフィックデータから得られたセッションの特徴を，セッションをクラスタリングした空間に投影し，各クラスタに属する確率 $p_k(\boldsymbol{x})$ を求める．

次に，GMM の各クラスタに属する確率と User-Agent と Referer の組の出現確率を式 (9) のベイズの定理で統合し，$p(k|U,R)$ を算出する．ここで，User-Agent と Referer の組の出現確率 $p(U,R|k)$ と $p(U,R)$ は 3.1.2 節で算出した確率を用いている．$p(k|U,R)$ を全ての GMM のクラスタで算出し，算出した $p(k|U,R)$ の中で最大となるクラスタを C_{test}，その確率を p_{test} とする．

$$p(k|U,R) = \frac{p(U,R|k)p_k(\boldsymbol{x})}{p(U,R)} \tag{9}$$

p_{test} がクラスタ C_{test} における閾値 p_{th_test} 以上であれば正常と識別する．一方，閾値未満であれば通常とは異なる不審な通信であると判断し，異常と識別する．

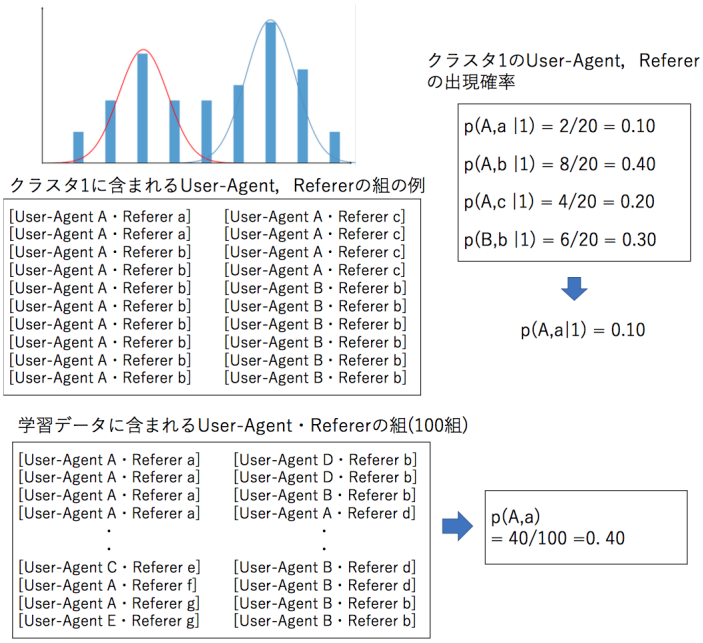

図 2　User-Agent，Referer の発生確率の算出方法

Fig. 2　The Method of Calculating probability of User-Agent and Referer.

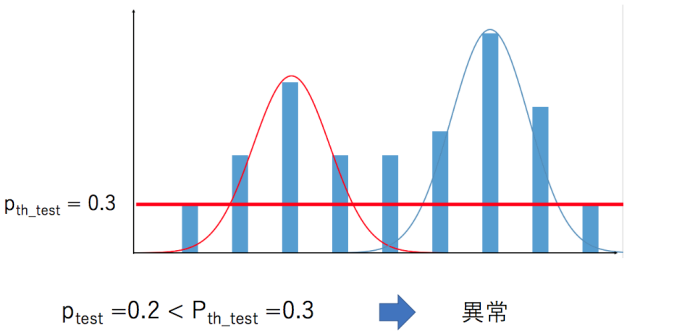

図 3　異常検知の概要

Fig. 3　The Outline of Anomaly Detection.

4. 実験

4.1 実験条件

実験に使用する標的型攻撃の通信を含む例として，MWS Dataset 2018 の一つである，組織内ネットワークへの侵害活動を想定した動的活動観測のデータセット BOS Dataset 2018[7] を用いた．また．BOS データセットにはマルウェアの進行度が示されており，マルウェアにより通信が発生したか，またどのように通信が行われたデータであるかが示されている．進行度ごとの説明を表 2 に示す．本稿では，進行度 2 の 2017 年 8 月 3 日の 24 時間の pcap データを異常を含まない正常なトラフィックデータとして学習データに利用し，進行度 7 の 2017 年 11 月 30 日から 2 日間

表 2　進行度の説明
Table 2　Explanation of Progress.

進行度	説明
1，2	通信発生なし
3，4，5	通信発生したが，C2 サーバとの攻撃通信不成立
6，7，8	通信発生かつ C2 サーバとの攻撃通信成立

表 3　実験データのセッション数
Table 3　The Number of Sessions of Experiment Data.

データ	正常セッション数	異常セッション数	総セッション数
進行度 2	214 個	0 個	214 個
進行度 7	223 個	291 個	514 個
進行度 8	52 個	140 個	192 個

の pcap データと進行度 8 の 2018 年 1 月 23 日から 5 日間の pcap データを異常を含むトラフィックデータとしてテストデータに用いた．各実験データの正常セッション数，異常セッション数，総セッション数を表 3 に示す．テストデータに対するラベル付けは C2 サーバと通信している IP アドレスの通信を異常として実験を行った．

4.2　評価方法

評価方法は Precision, Recall, F-measure を用いた．Precision は異常であると識別されたものの中で，異常であったものの割合を表す．Precision の算出式を式 (10) に示す．

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

Recall は異常を正しく異常であると識別できた割合を表す．Recall の算出式を式 (11) に示す．

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

ここで，TP(True Positive) は異常を正しく異常と識別した数，FP(False Positive) は異常と識別したが正常であった数，TN(True Negative) は正常を正しく正常と識別した数，FN(False Negative) は正常と識別したが異常であった数を表す．

F-measure は Precision と Recall の調和平均である．F-measure の算出式を式 (12) に示す．

$$F\text{-}measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{12}$$

4.3　実験結果・考察

学習データとして進行度 2 のトラフィックデータ，テストデータとして進行度 7 のトラフィックデータを用いた実験を実験 1，学習データとして進行度 2 のトラフィックデータ，テストデータとして進行度 8 のトラフィックデータを用いた実験を実験 2 とする．実験 1，実験 2 における Precision，Recall，F-measure を表 4 に示す．また，User-Agent と Referer の情報を使用することの有効性を評価するために，比較実験として User-Agent，Referer を特徴として用いず，ヘッダのみで実験を行った．実験 1，実験 2 の比較実験をそれぞれ実験 3，実験 4 とする．実験 3，実験 4 における Precision，Recall，F-measure を表 5 に示す．

表 4　検知結果
Table 4　Result of Detection.

実験	学習データ	テストデータ	Precision	Recall	F-measure
実験 1	進行度 2	進行度 7	0.70	0.86	0.82
実験 2	進行度 2	進行度 8	0.69	0.88	0.77

表 5　ヘッダのみを用いた検知結果
Table 5　Result of Detection based only header information.

実験	学習データ	テストデータ	Precision	Recall	F-measure
実験 3	進行度 2	進行度 7	0.55	0.67	0.61
実験 4	進行度 2	進行度 8	0.52	0.70	0.60

表 6　学習データのクラスタリング結果
Table 6　The Result of Clustering train data.

クラスタ	セッション数
1	31
2	29
3	4
4	10
5	90
6	22
7	4
8	22
9	2

4.3.1　学習結果

学習用データのクラスタリング結果を表 6 に示す．214 個のセッションをクラスタリングした結果，9 つのクラスに分類された．クラスタ 5 に多くのセッションが分類され，クラスタ 3，7，9 には少数のセッションしか分類されなかった．クラスタ 5 に分類されたセッションでは，複数の User-Agent と値を設定していない Referer が確認された．クラスタ 3，7，9 に分類されたセッションでは，特定の User-Agent と特定の Referer が確認された．他のクラスタに分類されたセッションでは，セッション毎に User-Agent，Referer が異なっていた．

4.3.2　実験 1 の異常検知結果と考察

実験 1 の異常検知結果について考察する．Recall は 0.86 であり高い結果を得ることができた．実験 3 では，Recall は 0.67 であった．この結果から本手法で，ヘッダに特徴が現れる不審な通信や，ペイロードに特徴が現れる不審な通信を検知できることを確認できた．

パケットのヘッダから抽出した特徴を確認すると，実験 1，3 に用いた進行度 7 のテストデータに含まれる正常セッションと異常セッションのパケットのヘッダから抽出したパケット数やパケットサイズ平均値に差があった．また，テストデータの異常セッション 291 個のうちの 146 個は，学習データの正常セッションに比べてパケット数やパ

ケットサイズ平均値が高い値を示していた．その他のセッションにおけるパケット数やパケットサイズ平均値は，学習データとの大きな差は確認できなかった．

User-Agent，Referer について確認すると，テストデータに含まれる正常セッションと異常セッションでは異なる値が設定されていた．正常セッションでは Referer に値を設定しないセッションが多く，異常セッションでは特定の User-Agent，Referer を設定しているセッションが多く存在した．また，テストデータの異常セッション 291 個のうちの 200 個のセッションが学習データと異なる特定の文字列を User-Agent，Referer に設定していた．その他のセッションは，様々な User-Agent，Referer を設定していた．

パケット数とパケットサイズ平均値の両方で学習データとの差がある場合，どちらかの値で学習データと大きく差がある場合，学習データに含まれない，または出現確率の低い User-Agent，Referer を設定している場合に不審な通信として検知できていた．今回の実験では学習データが全て正常な通信であったため，テストデータ中の異常を正しく識別することができ，Recall は高い結果を得ることができたと考えられる．しかし，学習データに含まれない User-Agent，Referer や学習データにおいて発生確率が低い User-Agent，Referer を設定している正常セッションを誤って不審な通信として検知したセッションも存在した．学習データのセッション数が少なかったために誤検知を招き，Precision は低い結果となったと考えられる．

4.3.3 実験 2 の異常検知結果と考察

実験 2 の異常検知結果について考察する．実験 1 と比較して Precision は 0.69 と少し低下し，Recall は 0.88 と少し上昇したため，F-measure が 0.77 と低下したが，実験 1 の結果と大きな差はなかった．User-Agent，Referer を特徴として用いず，ヘッダから抽出した特徴のみで学習と検証を行った実験 4 では，Recall は 0.70 であった．この結果からも本手法で，ヘッダに特徴が現れる不審な通信や，ペイロードに特徴が現れる不審な通信を検知できると考えられる．

パケットのヘッダから抽出した特徴を確認すると，進行度 7 のデータとは異なり，実験 2，4 に用いた進行度 8 のテストデータに含まれる正常セッションと異常セッションにはパケット数に大きな差はなかったが，パケットサイズ平均値には差があった．また，テストデータの異常セッション 140 個のうちの 98 個は，学習データの正常セッションと比べてパケットサイズ平均値が高い値を示していた．その他のセッションにおけるパケット数やパケットサイズ平均値は，学習データと大きな差はなかった．

User-Agent，Referer について確認すると，テストデータに含まれる正常セッションと異常セッションでは異なっており，正常セッションでは特定の User-Agent A や User-Agent B と値を設定しない Referer を含むセッションが多く存在した．異常セッション 140 個のうちの 112 個のセッションが進行度 7 のデータの異常セッションで使用されていた User-Agent A と特定の文字列 User-Agent C を設定しており，84 個のセッションは特定の文字列を Referer に設定していた．その他のセッションは，様々な User-Agent，Referer を設定していた．

パケットサイズ平均値で学習データとの差がある場合，学習データに含まれないまたは出現確率の低い User-Agent，Referer を設定している場合に不審な通信として検知できていた．実験 1 と同様に，学習データが全て正常な通信であったため，特徴が表れやすくテストデータ中の異常を正しく識別することができ，Recall は高い結果を得ることができたと考えられる．しかし，学習データに含まれない User-Agent，Referer を設定している正常セッションを誤って不審な通信として検知した．また，正常セッションに User-Agent A が含まれていたことや学習データのセッション数が少なかったことから誤検知を招きやすくなり，Precision は実験 1 に比べて少し低い結果になったと考えられる．

以上より，本手法でマルウェア感染後の不審な通信を検知できることを確認した．本手法の F-measure はどの組み合わせにおける実験でも既存手法 [10] を超える結果となった．これは異常検知で，User-Agent，Referer の出現確率を特徴量として追加したためであると考えられる．今回使用した特徴量は，通常の相手と比較して使用するセッション数が多い C2 サーバとの通信や内部調査を行う通信を不審な通信として検知するために有効であることを確認できた．また，マルウェアは特定の文字列を User-Agent，Referer に設定していることがわかった．しかし，マルウェアのセッション数が少ない場合には検知できない可能性がある．今後の課題として，C2 サーバとの通信や内部調査を行う通信を早期に発見するために新たな特徴を追加することなどが挙げられる．

5. おわりに

本稿では，パケットのヘッダからセッション毎に抽出した特徴と User-Agent と Referer の出現確率をベイズの定理で統合することで標的型攻撃を検知する手法を提案した．実験では，MWS データセットを用いて本手法の有効性を確認した．標的型攻撃によるマルウェアの動的観測データを用いた実験では，ヘッダに特徴が現れる不審な通信や，ペイロードに特徴が現れる不審な通信を検知することができた．しかし，マルウェアのセッション数が少ない場合には，異常を検知できない可能性がある．今後の課題として，学習データを増やすことによる検知精度の向上，使用するセッションが少数でも異常検知できるようにするための，新たな特徴量の追加などが挙げられる．

参考文献

[1] Le Blond, S. et al. : A Look at Targeted Attacks Through the Lense of an NGO, Proc. 23rd Usenix Security (2014).

[2] McAfee Blog:なぜ多層防御なのか？リスクを最小限にする最強のセキュリティ対策，入手先 ¡http://blogs.mcafee.jp/defense-in-depth-multilayer-protection¿(参照 2020/07/27)

[3] Fortinet Security Blog http://www.fortinet.co.jp/security_blog/131028-The-Stealthy-Downloader.html (参照 2020/07/23)

[4] BlueCoat ウェブプロキシでの活用例 https://www.bluecoat.com/security-clog/2014-04-29/protecting-your-organization's-web-browsing-new-internet-explorer (参照 2020/07/27)

[5] McAfee ウェブゲートウェイでの活用例 https://www.mcafee.com/enterprise/ja-jp/assets/solution-briefs/sb_05_targetedattack.pdf (参照 2020/07/27)

[6] 安藤槙悟, et al. : 通信の遷移に着目した不正リダイレクトの検知による悪性 Web サイト検知システムの提案，研究報告コンピュータセキュリティ (CSEC) Vol.32,pp.16 (2011).

[7] 高田雄太, 他: マルウェア対策のための研究用データセット MWS 2018 Datasets ，情報処理学会，Vol.2018CSEC-82，No.38，2018 年 7 月.

[8] 蒋 丹，面 和成 : 初期段階における Remote Access Trojan の検知方法，コンピュータセキュリティシンポジウム 2014 論文集，Vol.2014，No.2，pp.719-726(2014)

[9] 蒋 丹，面 和成 : 通信のクラスタ間遷移に基づくサイバー攻撃検知手法，コンピュータセキュリティシンポジウム 2015 論文集，Vol.2015，No.3，pp.1066-1072(2015)

[10] 鍛治一祐, 青木茂樹, 宮本貴朗 : パケットのヘッダに基づく不審な通信挙動の検知, 情報処理学会, Vol.33, pp.17(2018).

[11] 市田達也 : マルウェア通信検知手法における UserAgent の有効性の一考察，コンピュータセキュリティシンポジウム pp.234-241(2015).

[12] 西尾祐哉，et al.:悪性 Web サイトを分析するためにマルチ環境解析における通信ログ解析の効率化，コンピュータセキュリティシンポジウム pp.496-502(2016)

[13] Reynolds, Douglas A : Gaussian Mixture Models, Encyclopedia od biometrics, Vol.741(2009)

[14] Ishihata, Masakazu : Propositionalizing the EM algorithm by BDDs,Transactions of the Japanese Society for Artificial Intelligence,Vol.25,pp.475-484(2010).

[15] Chen, Scott Shaobing, and Ponani S. : Clustering via the Bayesian information criterion with applications in speech recognition,IEEE,Vol. 2(1998).