# エントロピーの変化点に基づく ネットワークパケットの異常検知

関 晃太郎[1,a)]　青木 茂樹[1,b)]　宮本 貴朗[1]

**概要**：近年のサイバー攻撃の増加に伴って, 組織内ネットワークに対するサイバー攻撃を検出する侵入検知システム (IDS:Intrusion Detection System) の研究が盛んに行われている. サイバー攻撃のパケットは通常のパケットとは異なり, 特にフラグなどで表される特徴量の分布に変化が生じると考えられる. そこで本稿では, パケットから抽出した特徴量の出現割合を基にエントロピーを算出し, エントロピーの変化点に注目してサイバー攻撃を検知する手法を提案する. エントロピーは計算式が単純であるため計算コストが低く, 特定の単位時間内で計算できるためリアルタイム処理が容易である. まず, パケットのヘッダからフラグやポート番号など複数の特徴量を抽出する. 次に, 抽出した特徴量の出現割合を基にエントロピーを算出し, 特徴量ごとのエントロピーの時系列データを作成する. その後, 時系列データの変化点を ChangeFinder で検出し, 異常を検知する. 実験では, CICIDS2017 データセットを用いて本手法の有効性を確認した.

**キーワード**：ネットワークの異常検知, エントロピー, ChangeFinder

# Anomaly detection in network packets based on entropy change points

Kotaro Seki[1,a)]　Shigeki Aoki[1,b)]　Takao Miyamoto[1]

***Abstract:*** With the rise in cyber attacks in recent years, research on intrusion detection systems (IDS) for detecting cyber attacks on internal networks has become increasingly active. When a cyber attack occurs, the packets received differ from regular packets, particularly in the distribution of features represented by flags and other indicators. This paper proposes a method to detect cyber attacks by calculating entropy based on the occurrence rates of features extracted from packets and focusing on changes in entropy. Entropy is computationally inexpensive due to its simple formula and can be calculated within specific time units, making real-time processing feasible. First, multiple features such as flags and port numbers are extracted from packet headers. Then, entropy is calculated based on the occurrence rates of the extracted features, and time-series data of the entropy for each feature is created. Subsequently, change points in the time-series data are detected using ChangeFinder to detect anomalies. The effectiveness of this method was confirmed through experiments using the CICIDS2017 dataset.

***Keywords:*** Network Anomaly Detection, Entropy, ChangeFinder

## 1. はじめに

近年, インターネットの普及に伴って, サイバー攻撃が増加している. サイバー攻撃を検知するために, ネットワーク上の不正なトラフィックを監視する侵入検知システム (IDS: Intrusion Detection System) の研究が盛んに行われている.

IDS には Snort [1] や Suricata [2] に代表されるシグネチャ型 IDS とアノマリ型 IDS の二種類が存在する. アノマリ型 IDS は予め正常な通信を記録しておき, 一致しない通信を異常な通信として検知する手法である. そのためアノマリ型 IDS は, データベースに記録されていない未知の

[1] 大阪公立大学大学院情報学研究科
Graduate School of Informatics, Osaka Metropolitan University
[a)] sd23005o@st.omu.ac.jp
[b)] aoki@omu.ac.jp

異常を検知することができる．しかし, 未知の通信であれば正常な通信も異常と検知してしまうため, シグネチャ型 IDS よりも誤検知が多いという欠点がある．代表的なアノマリ型 IDS として, 文献 [3,4] の手法が挙げられる．

ネットワークの異常検知にエントロピーを用いる研究 [5–7] が盛んに行われている．エントロピーは, 複雑な構造を持つデータの急激な変化や不規則性を捉えることができる．一方, 単位時間内で計算できるため, 大規模なデータでもリアルアイムに処理できる．これらの研究では, この特徴に着目してエントロピーの変化を検出することで異常を検知している．

時系列データの変化点を検出する ChangeFinder [9] を利用した異常検知手法が注目されている．ChangeFinder は変化点検出手法の一種であり, 時系列データの変化点をリアルタイムに検出できる．この手法では, 二段階学習を行うため, 他の変化点検出手法に比べ精度が高い．文献 [8] では, ハニーポットで取得したパケットの時系列データの変化点を, ChangeFinder により検出して異常を検知している．

本研究では, ネットワークのパケットから抽出した特徴量の出現割合を基にエントロピーを算出し, エントロピーの変化点に注目して組織に対するサイバー攻撃を検知する手法を提案する．まず, パケットのヘッダからフラグやポート番号など複数の特徴量を抽出する．次に, 抽出した特徴量の出現割合を基にエントロピーを算出し, 特徴量ごとのエントロピーの時系列データを作成する．その後, 時系列データの変化点を ChangeFinder で検出し, 異常を検知する．

以下, 2 節で関連研究について述べ, 3 節では提案手法について説明する．4 節で実験と結果に対する考察を述べ, 5 節でまとめと今後の課題について述べる．

## 2. 関連研究

本研究に関連する従来手法として, パケットのヘッダ情報を利用した異常検知手法である文献 [4] と, エントロピーを用いた異常検知手法である [5–7], ChangeFinder を用いた異常検出手法である文献 [8] について述べる．

文献 [4] では, トラフィックデータから複数の特徴量を抽出し, 教師無し学習と教師あり学習両方を用いて異常を検知し, 機械学習手法ごとの性能を評価している．フロー単位で抽出した特徴を基に, 教師無し学習では PCA(主成分分析) とオートエンコーダを使用し, 教師あり学習ではランダムフォレスト, ロジスティクス回帰, 線形 SVM を使用し, 異常を検知し評価している．実験の結果, 教師無し学習では PCA, 教師あり学習ではランダムフォレストで高精度に異常検知できることを確認している．

文献 [5] では, パケットのエントロピーに基づく異常検知手法が提案されている．この手法では, パケットデータを一定のパケット数で分割し, 分割した各パケットデータに含まれる IP アドレスやポート番号などの特徴量を抽出する．次に, 各パケットデータにおける特徴量の発生確率を求め, 求めた発生確率からエントロピーを算出する．その後, エントロピーの時系列変化に着目した EMMM 法 (EntropyMulti dimensional Maharanobis distance Method) により, エントロピーが大きく変化する時間を異常として検出している．この手法では, 学習及び異常検出にかかる時間の短縮が課題となっている．

文献 [6] では, パケットヘッダから抽出した特徴からエントロピーを算出し, 自己組織化マップ (SOM) で逐次学習しながら異常検知する手法を提案している．この研究ではまず, 正常データのパケットヘッダから特徴量を抽出し, エントロピーを算出する．次に算出したエントロピー値から特徴ベクトルを生成し, SOM で学習する．テストデータからも同様に特徴ベクトルを抽出し, 学習済みの SOM に入力することで異常を検知している．文献 [7] では, 文献 [6] と同様にパケットヘッダとペイロードから抽出した特徴量からエントロピーを算出し, クラスタリングすることで異常を検知する手法が提案されている．

これらの研究では, エントロピーの変化を検出することで高精度に異常を検出できている．しかし, 攻撃の早期検知には主眼が置かれておらず, 早期検知性能は評価されていない．

文献 [8] では, ChangeFinder によりリアルタイムに変化点を検出することで早期に異常を検知する手法が提案されている．この手法ではまず, 複数のハニーポットからログ情報を定期的に集計してデータベースに登録する．次に, ハニーポットごとに通信プロトコル, 送信元ポート番号のアクセス数, 単位時間当たりの受信パケット数の最大値を集計し, 時系列データを作成する．そして, ChangeFinder により変化点検出することでポートスキャン等を検出している．

## 3. 提案手法

提案手法の概要を図 1 に示す．まず, パケットデータから文献 [4] を参考に選択した 12 種類の特徴量を抽出する．次に, 各特徴量の出現割合を基にエントロピーを算出し, 特徴量毎に時系列データを作成する．そして, 作成した時系列データの変化点を Changefinder で検出することにより, 低い処理コストで早期に異常を検知する．

### 3.1 特徴抽出

パケットデータを単位時間で分割し, 各区間でパケットのヘッダから表 1 に示す特徴量を抽出する．ここで, 区間内に TCP パケットが存在しない場合, tcp_sport や tcp_flags のような TCP パケット特有の特徴量が存在しないため, 特徴量の値を 0 とする．また, tcp_opt_MSS は TCP パケットに含まれる MSS(Maximum Segment Size) の設定の有無に着目し, オプションが設定されているパケットは 1, 設定され

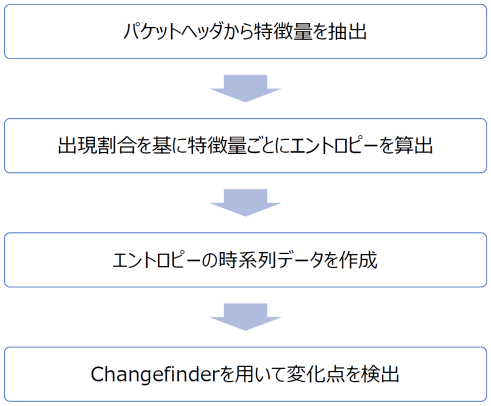



**図 1** 提案手法の概要
**Fig. 1** Outline of Proposed Methods.

**表 1** ヘッダ特徴量の一覧
**Table 1** List of Packet Header Features.

| |
|---|
| ip_tos(IP サービス・タイプ) |
| ip_flags(IP フラグ) |
| ip_totallen(IP データグラム長) |
| ip_ttl(TTL 値) |
| tcp_sport(送信元ポート番号) |
| tcp_dport(宛先ポート番号) |
| tcp_do(TCP データオフセット) |
| tcp_flags(TCP フラグ) |
| tcp_winsize(TCP ウィンドウサイズ) |
| tcp_opt_MSS(TCP オプション MSS) |
| tcp_opt_wscale(TCP オプションウィンドウスケール) |
| tcp_opt_sackok(TCP オプション SACK 許可) |

ていない場合は 0 とする．tcp_opt_wscale, tcp_opt_sackok も同様に処理する．

### 3.2 エントロピー

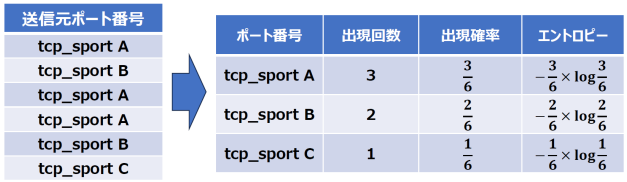
前節で抽出したそれぞれの特徴量 $i$ の出現割合 $p_i$ を基にエントロピーの総和を次式で算出する．

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i \qquad (1)$$

ここで，$n$ は単位時間内での特徴量の種類数であり，特徴量の出現割合 $p_i$ は次式で計算される．

$$p_i = \frac{\text{単位時間内での特徴量 } i \text{ のパケット数}}{\text{単位時間内での全パケット数}} \qquad (2)$$

式 (1) で算出するエントロピーの総和は，単位時間内に出現した特徴量がそれぞれ均等なパケット数だった場合，つまり $p_i$ が均等だった場合は大きくなり，$p_i$ に偏りがある場合は小さい値となる．エントロピーの総和の算出例を図 2 に示す．例では送信元ポート番号として A, B, C の 3 種類のポート番号のパケットが 6 つある．それぞれのポート




総和：$-\left(\frac{3}{6}\times\log\frac{3}{6}+\frac{2}{6}\times\log\frac{2}{6}+\frac{1}{6}\times\log\frac{1}{6}\right)$

**図 2** エントロピーの総和の算出例
**Fig. 2** Example of Entropy Calculation Sum.

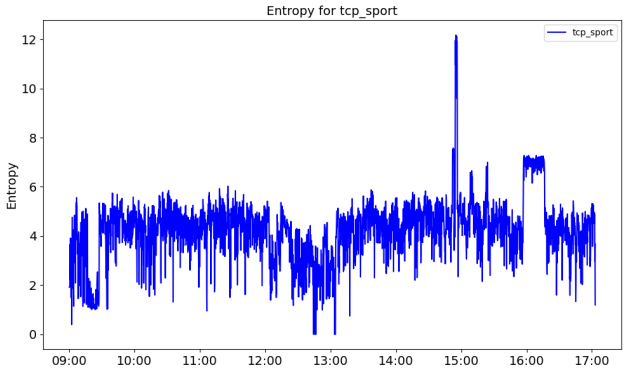



**図 3** エントロピーの総和の例
**Fig. 3** Example of Time Series of Entropy Sum.

番号の出現回数は A が 3 回, B が 2 回, C が 1 回であるため，出現確率は A が 6 分の 3, B が 6 分の 2, C が 6 分の 1 となる．その後，算出した確率を基に送信元ポート番号ごとのエントロピーを求めて，送信元ポート番号のエントロピーの総和を算出する．

図 3 に，あるパケットデータの送信元ポート番号に注目した時のエントロピーの総和の例を示す．横軸が時間，縦軸がエントロピーの総和を表している．図中，15:00 付近と 16:00 付近でエントロピーの総和の急激な変化を確認でき，この時間帯に攻撃されていることが分かる．

### 3.3 変化点検出

抽出した特徴量の時系列データの変化点を ChangeFinder で検出する．ChangeFinder は，SDAR モデルを使用して時系列データを二段階学習し，忘却パラメータ，モデルの次数，平滑化の範囲の 3 つのパラメータを設定することでリアルタイムに変化点スコアを算出する手法である．ChangeFinder の全体の流れを図 4 に示す．

SDAR モデルは，AR(自己学習) モデルにオンライン学習と忘却機能を追加したモデルである．AR モデルは，過去のデータを使って時系列データを予測するために使われる分析手法である．オンライン学習することで時系列データの逐次学習が可能となり，リアルタイムに変化点を検出できる．忘却機能は，過去のデータの影響を少なくすることができ，非定常データへの対応が可能となる．また，SDAR モデルを用いているため，計算量を削減できる．

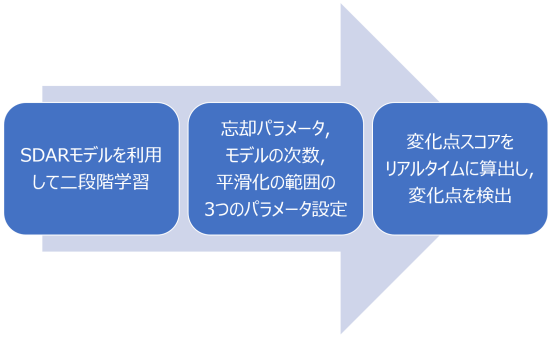



**図 4** ChangeFinder の概要
**Fig. 4** Outline of ChangeFinder.

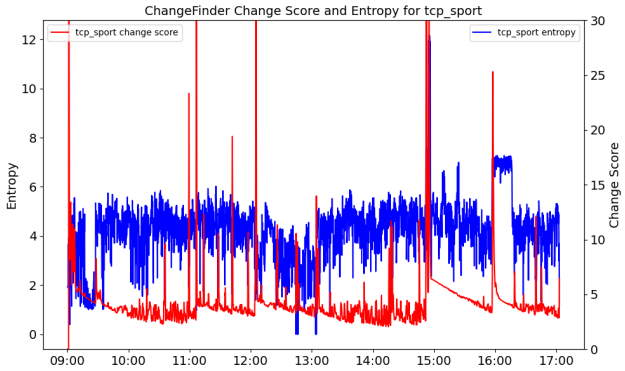



**図 5** ChangeFinder による変化点スコアの算出例
**Fig. 5** Example of ChangeFinder Calculating Change Point Score.

SDAR モデルの学習には忘却パラメータ, モデルの次数, 平滑化の範囲の 3 つのパラメータを使用する．忘却パラメータは, 過去の時系列の影響を調整し, 小さくした場合には変化点のばらつきが大きくなる．また, 平滑化の範囲を大きくすると, 外れ値ではなく変化そのものを検出することが可能になる．

図 3 に示すエントロピーの総和の時系列変化に対して変化点スコアを算出した例を図 5 に示す．横軸が時間, 縦軸左側がエントロピーの値, 右側が変化点スコアを表しており, 青色のグラフがエントロピーの変化, 赤色のグラフが変化点スコアの変化を示している．図中，9:00 付近, 11:00 付近，12:00 付近，15:00 付近，16:00 付近，17:00 付近で変化点スコアが大きくなっていることを確認できる．この中で, 最初の 09:00 付近については，学習が十分でないために変化点が大きくなったと考えられる．それ以外の時刻では攻撃されている可能性が高いと考えられる．そこで, 変化点スコアに対して閾値を設定し, 変化点スコアが閾値を超えた区間に異常ラベルを, それ以外の区間に正常ラベルを付与する．以上の処理を全ての特徴量に対して行い, 異常ラベルが $\theta$ 種類以上付与されている区間を攻撃されている異常区間として検知する．

**表 2** データセットに含まれる攻撃と攻撃時間
**Table 2** Attacks Included in Dataset and Attack Times.

| 攻撃 | 攻撃時間 |
|---|---|
| FTP-Patator | 2017/07/04(火) 9:20-10:20 |
| SSH-Patator | 2017/07/04(火) 14:00-15:00 |
| slow loris | 2017/07/05(水) 9:47-10:10 |
| SlowHTTPTest | 2017/07/05(水) 10:14-10:35 |
| Hulk | 2017/07/05(水) 10:43-11:00 |
| GoldenEye | 2017/07/05(水) 11:10-11:23 |
| Brute Force | 2017/07/06(木) 9:20-10:00 |
| XSS | 2017/07/06(木) 10:15-10:35 |
| Port Scan | 2017/07/07(金) 14:51-15:29 |
| DDoS | 2017/07/07(金) 15:56-16:16 |

## 4. 実験

本手法の有効性を確認するために実験を行った．実験には CICIDS2017 データセット [10] のトラフィックデータを使用した．評価指標として．攻撃が開始されてから異常を検知するまでの時間の差を算出し, 本手法で攻撃を早期に検知できることを確認した．

### 4.1 実験条件

本研究では, エントロピーを算出するための単位時間を 1 分に設定した．また, 異常区間を抽出するときの閾値 $\theta$ を 1 とした時の実験 1 と, $\theta = 2$ とした時の実験 2 を行った．

#### 4.1.1 実験データセット

CICIDS2017 データセットは, 攻撃者端末から組織内ネットワークへのサイバー攻撃を想定した研究用データセットである．データセットは 2017/7/3(月) から 2017/7/7(金) までのトラフィックをキャプチャしており, 月曜日は正常通信のみ, それ以外の日は正常通信と異常通信の両方を含むトラフィックである, 含まれている攻撃は, 火曜日に Brute Force 攻撃 (FTP-Patator, SSH-Patator), 水曜日に DoS 攻撃 (slow loris/SlowHTTPTest/Hulk/GoldenEye), 木曜日に Brute Force 攻撃と XSS 攻撃, 金曜日には Port Scan 攻撃, DDoS 攻撃のトラフィックが含まれている．表 2 に攻撃毎の攻撃時間を示す．

#### 4.1.2 ラベル付与

正解データには各攻撃時間に該当し, かつ攻撃者の送信元 IP アドレスと一致しているパケットを異常ラベル, それ以外のパケットを正常ラベルとして付与した.

### 4.2 実験結果と考察

表 3 に, 実験 1 と実験 2 のそれぞれにおける異常検知時間と実際の攻撃開始時刻との差を示す. また, 図 6, 図 7, 図 8, 図 9 に各曜日で攻撃検出に特に有効であった特徴量の変化点スコアのグラフを示す. 図中, 黄色の背景部分は攻撃時間を表している. 火曜日は tcp_dport, 水曜日は ip_ttl, 木曜日は ip_tos, 金曜日は tcp_opt_wscale が特に有効であった. 表 3 と図 6, 図 7, 図 8, 図 9 に示す結果から, 火曜日 14:00 の SSH-Patator 攻撃, 水曜日 10:43 の DoS Hulk 攻撃, 木曜日 9:20 の Brute Force 攻撃, 金曜日 14:51 の Port Scan 攻撃, 15：56 の DDoS 攻撃ともに攻撃発生時間から短い時間で異常を発見できることを確認した. 一方, 火曜日 9：20 の FTP-Patator 攻撃が検出されなかった理由としては, パケットデータの開始時間から攻撃時間までの間隔が短く, 適切に学習できなかったためであると考えられる. 水曜日 9:47 の slow loris, 11:10 の GoldenEye の異常を検出できなかった要因, 10:14 の SlowHTTPTest の検出に 11 分かかった要因としては, これらが非常に遅い速度でパケットを送り続け, Web サーバとの接続を維持し続ける攻撃であるために, 今回利用した特徴量では攻撃開始時に緩やかにしか変化せずエントロピーの総和が大きく変化しなかったこと，攻撃と攻撃の間隔が短かったために, 正常状態を十分に学習できなかったことが挙げられる. 木曜日 10:15 の XSS 攻撃が検出されなかった理由としては, 有効な特徴量を選択できなかったためであると考えられる. そのため, 各攻撃に有効な特徴量の追加が今後の課題として挙げられる.

また, 図 8 に示す木曜日 12:40 付近と 15:00 付近で攻撃されていないにも関わらず, 変化点スコアが大きくなっている. これは, ip_tos の出現割合が一定でない正常通信を検出したものである. 同時間帯には他の特徴量では変化点スコアが大きくなっていないため, 閾値 $\theta = 1$ とした実験 1 では正常を異常と誤検出しているが, 閾値 $\theta = 2$ とした実験 2 では正しく正常と識別する結果になっていることを確認した. また, 表 3 に示す実験 1 と実験 2 の結果を比較すると, Hulk で 1 分, DDoS で 2 分実験 1 の方が早く検出できているが, 実験 1 の方が誤検出が多くなる傾向にあるため, 今後閾値の決定方法を検討したいと考えている.

次に, 検出した各攻撃において異常と識別した特徴量を表 4 に示す. 実験の結果, SSH-Patator では主に TCP プロトコル関連の特徴量が有効であることを確認した. これは SSH-Patator が SSH 認証を対象にした Brute Force 攻撃であるため, 基盤となる通信プロトコルに TCP 通信を使用

**表 3** 攻撃開始時刻と異常検知時刻の差

**Table 3** Difference Between Attack Start Time and Anomaly Detection Time.

| 攻撃 | 実験 1 | 実験 2 |
|---|---|---|
| FTP-Patator | - | - |
| SSH-Patator | 1 分 | 1 分 |
| slow loris | - | - |
| SlowHTTPTest | 11 分 | 11 分 |
| Hulk | 2 分 | 3 分 |
| GoldenEye | - | - |
| Brute Force | 5 分 | 5 分 |
| XSS | - | - |
| Port Scan | 1 分 | 1 分 |
| DDoS | 2 分 | 4 分 |

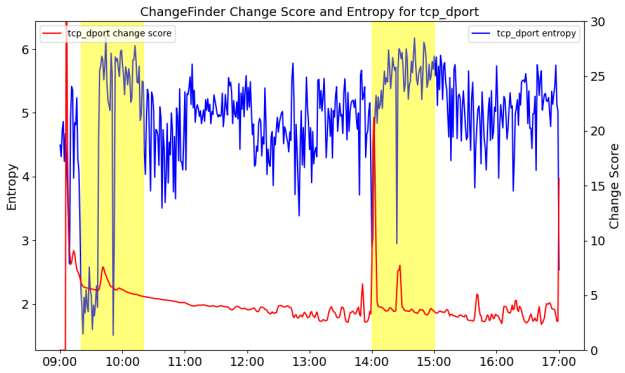



**図 6** 変化点スコア算出結果 (火曜日)

**Fig. 6** Results of Change Point Score (Tuesday).

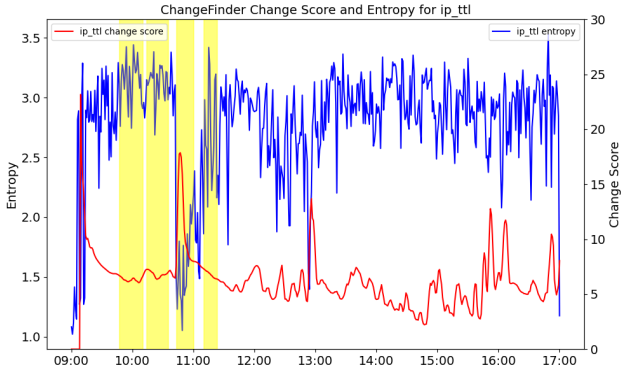



**図 7** 変化点スコア算出結果 (水曜日)

**Fig. 7** Results of Change Point Score (Wednesday).

するためだと考えられる. SlowHTTPTest 攻撃では, TCP のオプションフィールドにかかわる特徴量が有効であることを確認した. これは攻撃者が TCP オプションフィールドを利用したためであると考える. また, Port Scan 攻撃では TCP プロトコルの宛先, 送信元ポート番号が特に有効な特徴量であることを確認した. これは, Port Scan 攻撃が攻撃先の開いているポートを探索する攻撃であるため, これらの特徴量のエントロピーが急激に変化したと考えら

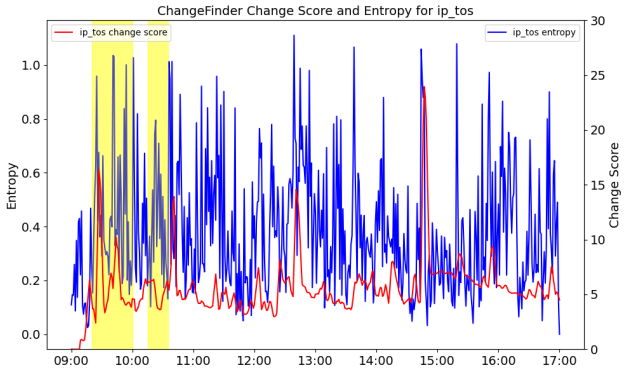



**図 8** 変化点スコア算出結果 (木曜日)
**Fig. 8** Results of Change Point Score (Thursday).

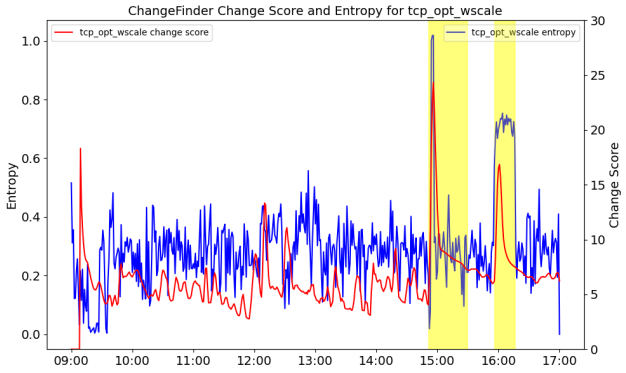



**図 9** 変化点スコア算出結果 (金曜日)
**Fig. 9** Results of Change Point Score (Friday).

**表 4** 攻撃検知に有効な特徴量
**Table 4** Effective Features for Attack Detection.

| 攻撃 | 有効な特徴量 |
|---|---|
| SSH-Patator | ip_ttl, tcp_sport, tcp_dport, tcp_flags, tcp_opt_MSS, tcp_opt_wscale, tcp_opt_sackok |
| SlowHTTPTest | tcp_opt_MSS, tcp_opt_wscale, tcp_opt_sackok |
| Hulk | ip_flags, ip_ttl, tcp_sport, tcp_dport, tcp_do, tcp_winsize |
| Brute Force | ip_totallen, ip_ttl, tcp_sport, tcp_dport, tcp_do, tcp_flags |
| Port Scan | ip_flags, ip_totallen, tcp_sport, tcp_dport, tcp_winsize, tcp_opt_MSS, tcp_opt_wscale, tcp_opt_sackok |
| DDoS | ip_flags, ip_totallen, ip_ttl, tcp_do, tcp_winsize, tcp_opt_wscale |

れる．今後，今回の実験では検出できなかった slow loris, GoldenEye, XSS 攻撃で有効な特徴量を検討し，追加したいと考えている．

## 5. おわりに

本稿では，パケットから抽出した特徴量の出現割合を基にエントロピーを算出し，エントロピーの変化点に注目してサイバー攻撃を検知する手法を提案した．実験では，CICIDS2017 データセットを用いて本手法の有効性を確認した．実験の結果，複数の攻撃において異常の始まりをリアルタイムで捉えることができた．一方，抽出した特徴量の種類によって異常の発見時間が異なり，攻撃の種類ごとに有効な特徴量が存在することを確認した．今後の課題として，新たな特徴量の追加，特徴量ごとの適切な閾値の設定方法の検討などが挙げられる．

## 参考文献

[1] Snort, https://www.snort.org/(参照 2024-08-09).
[2] Suricata, http://suricata-ids.org/(参照 2024-08-09).
[3] 平松尚利, 和泉勇治, 角田裕：複数の通常状態を用いたネットワーク異常検出, 信学技報, CS2006-32, pp.61-66 (2006)
[4] M. J. De Lucia, D. E. Krych, S. Raio and J. E. Ellis：An Empirical Investigation of Packet Header-Only Network Traffic Anomaly Detection and Classification, DEVCOM Army Reseach Laboratory, 2023
[5] 小島俊輔, 中嶋卓雄, 末吉敏則：エントロピーベースのマハラノビス距離による高速な異常検知手法, 情報処理学会論文誌, Vol.52, No.2, pp.656-668, 2011
[6] 二瓶凌輔, 青木茂樹, 宮本貴朗：自己組織化マップによるネットワークトラフィックの逐次学習と異常検出, Computer Security Symposium 2019, 4D1-2, pp.1414-1421, 2019
[7] 柏木宏務, 青木茂樹, 宮本貴朗：ネットワークトラフィックのエントロピーに注目した異常検知, 第 22 回情報科学技術フォーラム, 2023
[8] 山村翔, 熊谷充敏, 神谷和憲, 倉上弘：変化点検知を用いた新種スキャンの早期発見手法の検討, Computer Security Symposium 2017, 2B4-3, 2017
[9] J. Takeuchi and K. Yamanishi：A Unifying Framework for Detecting Outliers and Change Points from Time Series. IEEE Transactions on Knowledge and Data Engineering, Vol.18, No.4, pp.482–492 (2006)
[10] I. Sharafaldin, A. Habibi Lashkari and A. A. Ghorbani：Toward generating a new intrusion detection dataset and intrusion traffic characterization, Proc. of 4th ICISSP, pp. 108-116, 2018