Vol.2022-DPS-190 No.5
Vol.2022-CSEC-96 No.5
2022/3/10

機械学習を用いた悪性TLS通信の検知と通信特徴の推移に関する考察

藤原 魁成[1,a)]　小澤 誠一[1]　春木 博行[2]　Park Chanho[2]

概要：インターネット通信の暗号化が急速に普及し，ほとんどの通信はTLSなどによって暗号化されている．一方，TLSを用いて暗号化された通信を利用してC&Cサーバとの通信や機密データの流出などを行うサイバー攻撃も同様に増加している．通信の悪性度判定のために復号化を行うことは，ネットワークパフォーマンス低下や機密性の観点から望ましくなく，復号化することなく検知する手法が求められる．本稿では，フローメタデータなど復号化なしで得られる特徴に着目し，機械学習を用いて悪性通信を検知することを目指す．いくつかの機械学習モデルの比較の結果，XGBoostを用いたもので適合率0.99，再現率0.89の精度を実現した．また，時期が異なる通信データに対する検知率の評価および通信トラフィック特徴の時間推移について考察する．

キーワード：機械学習，TLS，マルウェア

Detecting Malicious TLS Communications Using Machine Learning and Considerations on the Transition of Communication Characteristics

Kaisei Fujiwara[1,a)]　Seiichi Ozawa[1]　Hiroyuki Haruki[2]　Park Chanho[2]

Abstract: Encryption of Internet communications is rapidly spreading, and most communications are encrypted using TLS and other methods. On the other hand, cyber-attacks that use TLS-encrypted communications to communicate with C&C servers and leak confidential data are increasing as well. Decryption to determine the maliciousness of the communication is undesirable from the perspective of network performance degradation and confidentiality. Therefore, it is necessary to develop a detection method that does not require decryption. In this paper, we focus on the features that can be obtained without decryption, such as flow metadata. In this paper, we focus on features that can be obtained without decryption, such as flow metadata, and aim to detect malicious communication using machine learning. As a result of comparing several machine learning models, we achieved a goodness-of-fit rate of 0.99 and a recall rate of 0.89 using XGBoost. In addition, we evaluate the detection rate for communication data from different time periods and discuss the time evolution of communication traffic characteristics.

Keywords: Machine Learning, TLS, Malware

1. はじめに

インターネットは私たちの生活に欠かせないほど普及した．コミュニケーションやショッピングなど様々な活動において日常的にやり取りを行う上で，扱う情報の機密性が保証されることが重要な問題となってきた．そのような問題に対して，SSL(Secure Socket Layer)プロトコル，さらにその次世代規格としてTLS(Transport Layer Security)プロトコルが開発された．最も普段私たちが目にする機会が多いものとしてHTTPS通信が挙げられる．これは，

[1] 神戸大学
Kobe University
[2] LINE 株式会社
LINE Corporation
[a)] 1814359t@stu.kobe-u.ac.jp

HTML などのハイパーメディア文書を転送するためのアプリケーション層プロトコルである HTTP(Hyper Text Transfer Protocol)[1] に TLS を用いて暗号化を施したものであり，HTTP か HTTPS かはアドレスバーで確認ができる．ブラウザによっては HTTPS であるものは保護されていることを示すような表記を行っている．企業，政府，個人すべてが暗号化によってプライバシー保護の恩恵を得ており，これからも TLS の使用はより増えていくと考えられる．しかし，暗号化による恩恵を得るのはサイバー犯罪者も同じである．彼らは TLS を使って C2 サーバと通信を行い，コマンド指示を受け取ったり，感染したマシンから機密データなどを抜き出して送信したりする活動を隠そうとしていることが報告されている [2]．TLS が普及したことで，TLS 通信を行うための認証を安価で取得できるようになったきたことを考えると，このような攻撃の傾向はこれからさらに増加していくと予想される．

このような攻撃に対処する一つの方法として復号化を行い検査をする方法があるが，これは復号化のために鍵を追跡・傍受する必要があり，復号化して TLS パケットのペイロードを検査して再び暗号化をするという過程から，ネットワークパフォーマンスが著しく低下してしまう．また，これは本来の通信内容を保護するという観点にも反することとなる．これらの理由から，パケットを復号化することなくマルウェアなどによる悪性な通信かどうかを検知することが求められている．

本研究では，Olivier らの研究 [3] から着想を得て，パケットのメタデータに着目した機械学習モデルによる悪性通信検知を目的とした．復号化することによって発生する問題のため，本研究でも暗号化された状態で取得できる特徴のみに着目する．また，pcap ファイルの解析手法によって用いることのできない特徴量も多く用いられているため，悪性データを検知するのに有効であると考える新たな特徴量も取得する必要がある．そこで本研究では，先行研究で用いられているフローメタデータや TLS メタデータ特徴量に加え，証明書の追加情報やサードパーティから得られる付加情報，伊藤ら研究 [4] を参考にした証明書のコストに関する特徴などを加え，正常なネットワークトラフィックから悪意ある TLS フローを検出するモデルを構築し，それに対して精度の評価，各特徴量の有効性について考察を目的とする研究を行う．作成した特徴量を用いて，先行研究で用いられた機械学習手法の SVM, Random Forest, Logistic Regression と別の分類手法として，勾配ブースティング法モデルの一種である XGBoost，脳の神経回路の一部を模した数理モデルである Neural Net の 5 つを用いてモデル比較を行う．さらに，機械学習モデルへの解釈性付与のため，協力ゲーム理論のシャープレイ値を応用した SHAP 値 [5][6] を用いて特徴量の寄与度に関しても可視化を行う．最新のマルウェアに対しても有効であるのか，これらの特徴量が攻撃者にとって変えやすいのかなどを確かめるために，訓練データに用いた 2018 年から，本研究を行った 2021 年までの悪性データを用意して，訓練済みモデルで検知精度を確認する．また，その結果から，取得時期の異なるマルウェアにおける通信特徴の違いなどについて考察する．

2. 関連研究

Olivier ら [3] は，TLS で暗号化された通信を復号化せずに得られる特徴として，Flow Metadata，Distribution，TLS Metadata の大きく 3 つの特徴を定義している．これらを特徴量とした，機械学習を用いて悪性の TLS 通信を復号化せずに検知するシステムを提案している．

- Flow Metadata：NetFlow で観測可能な TLS に関係のない特徴として，バイト数やパケット数などを取得する．
- Distributions：フローのパケットに対する頻度分析の特徴．パケットの長さについて，ビンの大きさを決めて，A[i, j] にビン i と j の間の遷移数をカウントする，行列 A を構築する．行に関して正規化され，特徴として使用する．パケットの到着時間間隔についても同様に出来る．
- TLS Metadata：ClientHello や ServerHello, Certificate などのパケットから抽出できる特徴として，CipherSuite や TLS Extensions, EllipticCurve などや，証明書の有効期間やサーバ証明書が自己署名であるかなどを取得している．

これらを用いて，RandomForest で悪意ある TLS フローを検出する分類モデルを提案した．accuracy, precision, recall ともに 0.999 以上を得た．これは，マルウェアやユーザの習慣など，時間的バイアスによるデータの性質にも起因しているとしている．また，未知データに対して 96.13%の精度を達成した．偽陰性が増えた理由として，マルウェアの TLS 設定が，強力なデフォルトを設定した標準的な TLS ライブラリを使用するなどすれば，正常に見えるからとしている．検知能力が向上しマルウェアがもたらす脅威への認識が広まる一方で，攻撃者もツールや技術を進化させ，対応していると予想され，検知能力の低下が考えられるため，有用性の維持に対する改善が必要としている．

Olivier らの用いた特徴について，ツールによってはセッション中のパケットの到着時間間隔など特徴を得るのが難しいほか，時間的な検知能力の低下へ対策する手掛かりのため，マルウェアのメタデータに関する特徴などにおける特徴推移について検証が必要であると考える．

3. 提案手法

3.1 解析手法

pcap ファイルの解析に用いた zeek[7] は，Bro IDS というオープンソースのネットワークトラフィックアナライザ

で，主にセキュリティ監視システムだが，pcap ファイルからの様々なトラフィック分析もサポートしている．各接続に関するログ情報は，情報によって各ログファイルに分散しており，相互接続する id が割り振られている．我々はそれを活用することで各接続単位での特徴量抽出が可能になる．以下に，作成される代表的なファイルとその説明をそれぞれ示す，

- conn.log：TCP/UDP/ICMP の接続に関するログ．
- ssl.log：SSL/TLS セッションに関するログ．
- http.log：HTTP リクエストとその応答に関するログ．
- x509.log：X.509 証明書に関する情報．
- smtp.log：SMTP の活動をまとめたログ．
- ftp.log：FTP セッションに関するログ．
- dns.log：DNS クエリとその応答に関するログ．

これらのうち，本研究では conn.log, ssl.log, x509.log の 3 つのみを用いた．TLS をプロトコルを使用するものに限定するので，conn.log にある接続のうち，ssl.log にログがある接続のみを抽出し，それに結び付けられる TLS 証明書の情報を x509.log から得た．

(1) conn.log：2 つのエンドポイント間ごとの接続情報が保存される．ここでは双方の IP アドレス，パケットの数，プロトコルや接続の状態などの情報が得られる．また，ここでの uid が ssl.log の uid に結び付く．

(2) ssl.log：SSL/TLS のハンドシェイクと暗号化のプロセスを記述している．TLS のバージョン，使用した暗号方式や拡張機能，サーバ名，サーバ証明書からルート証明書までを含むそれぞれの証明書の ID 等の情報が得られる．ここでは用いた暗号アルゴリズムや，証明書のパスの情報が得られる．次に示す x509.log と結びつけることで証明書の情報が取得可能である．

(3) x509.log：X509 証明書に関する情報で，各行は発行者やサブジェクト名，証明書の有効期間などのログ，CN やシグネチャアルゴリズムなどを含む．ここで割り振られている ID が，ssl.log の証明書の ID に紐づいている．ssl.log での cert_chain での値が x509.log での fingerprint に一致する証明書の情報を取得することができる．

これらのログファイルから，フローメタデータ，TLS メタデータを特徴量化した．また，zeek にパッケージを追加すれば，選択された暗号方式のみでなく，JA3 ハッシュ値や TLS Extension なども取得が可能となる．

3.2 特徴量

2 章で示した Olivier らの研究から，今回再現可能である特徴量を**表 1** に示す．ここで，表 1 のポート番号について，sorce port は通常 OS によってある範囲内でランダムに選ばれる．このポート番号が ephemeral port[8] で定義された範囲内かそうでないかを特徴量とする．destination port は [9] のシステムポート番号の IANA に登録されたもののうち TLS に関する 10 個のポート番号に一致するかそうでないかを特徴量としている．また，selected cipher suite は，145 個の暗号方式と Hex の対応表に基づいており，これらに一致しないものは cs_ffff として 1 つにまとめている．

表 1: ログデータから再現可能な特徴量

特徴量	型
sorce port	Boolean
destination port	Boolean
number of inbound bytes	Integer
number of outbound bytes	Integer
number of inbound packets	Integer
number of outbound packets	Integer
duration of flow	Integer
selected cipher suite	Binary Vector (146)
number of SAN	Integer
validity (in days)	Integer
certificate self-signed or not	Boolean

表 2: 本研究で追加した特徴量

特徴量名	特徴量の説明	型
TLSver	TLS のバージョンが TLS1.3,1.2 かそれ以外か	Boolean
cert_lev	EV,OV 証明書かそれ以外か	Boolean
val_m, val_s	サーバ証明書と中間証明書の有効期間の平均，分散	Integer
cheap	使用している証明書が格安証明書に該当するか	Boolean
wild	CN または SAN で*(ワイルドカード) が用いられているか	Boolean
cert_num	証明書の階層数	Integer
hist_certs	VT から得られる過去の証明書の数	Integer
hist_san	過去最大 5 個の証明書の SAN の数の平均	Integer
hist_valid	過去最大 5 個の証明書の有効期間の平均	Integer
ref_files, ref_mal	ファイル数, そのうち VT で悪性判定がある数	Integer
com_files, com_mal	ファイル数, そのうち VT で悪性判定がある数	Integer
httphost	ホスト名が取得できるか	Boolean
ja3	JA3 のブラックリストに一致するか	Boolean

ツールによって取得できていない特徴があるほか，TLS Extension などの特徴量は攻撃者の設定によってコストや手間をかければ変えられる特徴量であると考え，今回は用いていない．ただ表 1 のみでは十分な検知性能は期待できない．そこで本研究では，**表 2** に示すような特徴量を追加して悪性通信の検知に有効な特徴量の検証を行った．表 2 の VT というのは VirusTotal[10] を示している．VirusTotal はファイルなどのマルウェア検査を行うサービスであり，ユーザが投稿した検体の解析結果を得ることができ，その解析結果はデータベースに蓄積される．VirusTotal の検索インターフェースで，ドメインや IP アドレスを入力すれば，過去に投稿され解析された検体の結果から，検索したその入力データと関連がある解析結果を取得することができる．本研究では，VirusTotal の検索機能を利用し，実行のある時点での入力 IP アドレスへのあらゆる種類のトラフィックを提示するすべてのファイルの数 (com_files) と VT における悪性判定数 (com_mal)，文字列に指定された IP アドレスを含むファイルの数 (ref_files) と VT における悪性判定数 (ref_mal)，ある時点で関連付けられている SSL 証明書の数 (hist_certs)，そこから取得できる直近最大 10 個のサーバ証明書から，SAN の数 (hist_san) と有効期間 (hist_valid) について平均を取った値を取得して特徴量とし

て追加した．また，JA3 のブラックリストについては [11] で公開されているものを使用した．JA3 は，SSL ハンドシェイクの Client Hello パケットを参照して，クラインと側でサポートしている，SSLVersion, Cipher, SSLExtension, EllipticCurve, EllipticCurvePointFormat の 5 つのフィールドの値をカンマで区切りハッシュ値にしたものである．得られたフィンガープリントを使用して，特定のトラフィックの識別が可能であり，JA3 のブラックリストも公開されている．しかしクライアント側のサポートするフィールドのみによって決まり，[11] にも記述があるように，これのみでは，多くの誤検知が発生してしまう可能性もあるため，今回特徴量という形で用いた．格安証明書に関しては，伊藤らの構築コストに基づく悪性ウェブサイト検知の研究を参考に，無料，格安で証明書を発行する認証局からの証明書 [12][13] を対象に，17 つの Issuer のコモンネームに一致するかを特徴量として扱った．

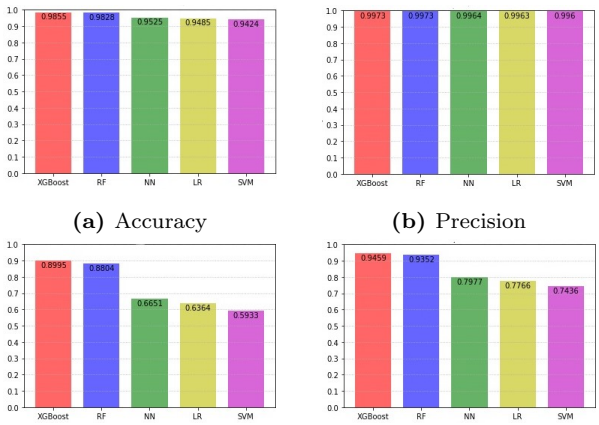
4. 評価実験

4.1 データセット

- 良性データセット
  - LINE 株式会社の業務用ネットワークの通常トラフィックログデータベースから選定し抽出されたデータを用いる．2021 年 10 月から 12 月の期間に取得した計 196,376 件のログデータである．ここでは SSL/TLS を用いた通信に限定したログを使用している．
  - CTU-13 データセット [14]：チェコ工科大学で取得されたものあり，実際のネットワーク環境で実行された，異なる 13 種類のマルウェアのキャプチャデータを確認できる．公開されるキャプチャデータには，マルウェア，正常，バックグラウンドのトラフィックが含まれている．本研究では先行研究で用いられた TLS Extension などの特徴に関しても，マルウェアの特徴推移を確認したため，比較のためのみにこのデータを良性データとして使用した．正常なデータに関して，仮想マシン上で動作する Windows を用いてキャプチャされたもので，今回は 2017 年のデータである．
- 悪性データセット
  - lastline：[3] にて公開されている，ネットワークセンサが検出した 2016 年から 2019 年のマルウェアの非常に短いキャプチャ．
  - stratosphere：こちらも同様に [3] にて公開されている，TLS を使用して通信する 2016 年から 2018 年のマルウェアの長期的なキャプチャ．
  - MALWARE-TRAFFIC-ANALYSIS.NET：[15] は，悪意あるネットワークトラフィックに関する pcap ファイル，マルウェアサンプルが公開されているサイトであり，このサイトから 2018 年から 2021 年の悪性 pcap ファイルを取得し，今回の実験では未知悪性データに対する検知精度の検証と特徴推移の調査対象として用いた．

表 3: 使用したデータセットとその数

データ元	種類	pcap ファイル数	データ数
LINE	良性	-	193,823
CTU-13	良性	8	25,013
stratosphere	悪性	30	32,028
lastline	悪性	183	418
malware data 2021	悪性	33	3,669
malware data 2020	悪性	29	1,944
malware data 2019	悪性	20	982
malware data 2018	悪性	11	6,129



(a) Accuracy **(b)** Precision

(c) Recall **(d)** F1-score

図 1: 各モデルの精度比較

悪性 pcap ファイルに良性データを含む可能性があるため，Alexa[16], Majestic[17], Cisco Umbrella[18] の 3 つの主要トップ 100 万サイトに一致するデータを取り除く形でフィルタリングを行った．これらから収集した pcap ファイルを 3.1 節で説明した zeek を用いて，今回使用するデータを作成した．**表 3** にフィルタリング後のデータ数をまとめたものを示す．

4.2 モデル比較と精度結果

同じデータセットで XGBoost, Random Forest(RF), Support Vector Machine(SVM), Logistic Regression(LR), Neural Network(NN) を用いて各指標の精度を比較した図を以下の**図 1** に示す．結果として，XGBoost, Random-Forest で各指標ともに良い結果を得た．今回の 5 つの中で，全ての指標において最も良い精度を示した XGBoost を用いて以下の実験を行っていくこととする．訓練用データに対する結果と，評価用データに対する XGBoost の結果を**表 4** に示す．学習した良性データ悪性データに関して分類を行えている他，別の悪性データを含めた評価用データに対しても特に precision において高精度を示した．悪性データの見逃しを示す recall ももちろん重要な指標であ

表 4: XGBoost を用いたときの訓練用・評価用データでの各精度結果

	accuracy	precision	recall	f1-score
訓練用データ	0.9996	0.9969	1.0000	0.9985
評価用データ	0.9855	0.9973	0.8995	0.9459

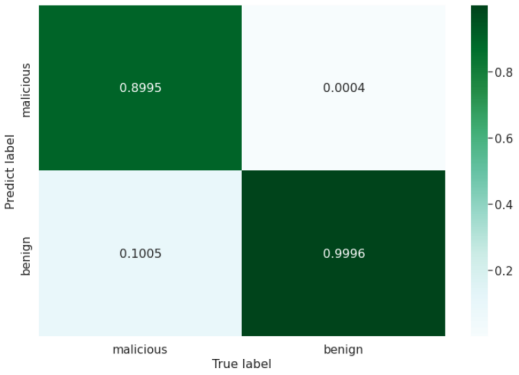


図 2: 分類結果を正規化した混同行列

るが，1 日に大量の通信が発生するためそれに対して 良性データを悪性と判定してしまわないように precision が高いという事も重要となってくると考える．

また，評価用データに対する結果の混同行列を**図 2** に示す．False Positive を十分抑えられている一方，False Negaticve が悪性データの約 10%と，見逃しについて課題が残る結果となった．

ここで，協力ゲーム理論のシャープレイ値を応用した SHAP 値 (SHapley Additive exPlanations) を用いて，XGBoost における，目的変数に対する特徴量の寄与度を**図 3** に示す．赤いプロットはその特徴量の値が大きいことを示し，逆に青いプロットは小さい値を示す． そしてそのプロットが横軸の 0 より右側にあるとここでは，悪性であるという判断に寄与する．TLS のバージョンやバイト数，ja3 などが上位にある他，過去の証明書の情報や，VirusTotal による ref_mal や com_mal なども寄与していることがわかる．

4.3 未知悪性データに対する検知率と特徴推移の考察

4.2 節で使用した訓練用悪性データは 2016 年から 2018 年のものであり，本節では，4.2 節での学習済みの XGBoost を用いて，年度の異なる，未知の悪性データに対してどれだけ検知が可能かを検証した．4.1 節で示した，MALWARE-TRAFFIC-ANALYSIS.NET より，2018 年から 2021 年の悪性 pcap ファイルを取得し，他のデータと同様にツール zeek を用いて特徴量抽出を行った．t-SNE による可視化の図を**図 4** に示す．ここでは，4.2 節で用いた stratosphere, lastline の悪性データも含めて可視化を行った．これらに対して，XGBoost で分類した際の検知精度を**表 5** に示す．今回の XGBoost の学習に用いた悪性データは 2016 年から 2018 年に収集されたものであり，学習に用いられていない

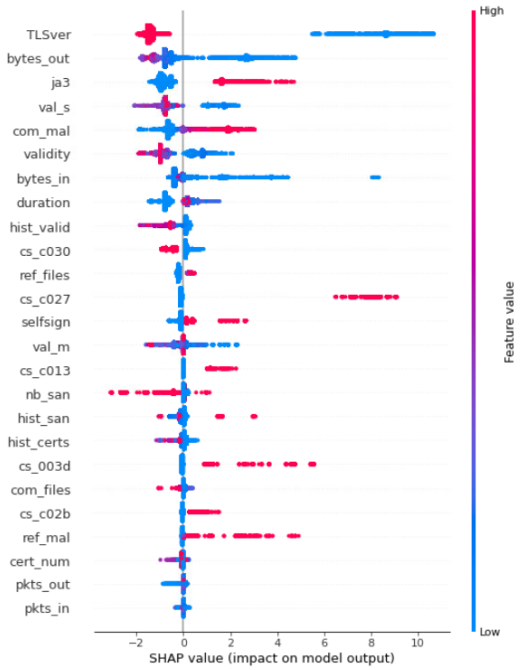


図 3: 目的変数に対する特徴量の寄与度

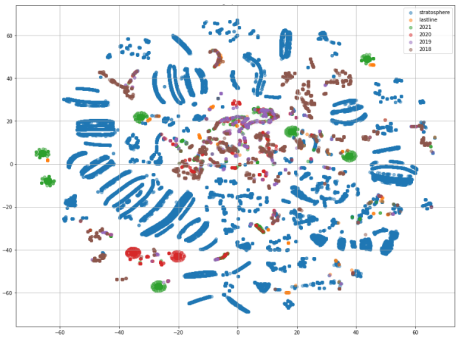


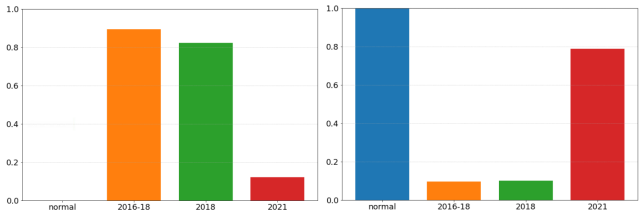
図 4: 2018 年から 2021 年の悪性データの t-SNE による可視化の図

表 5: XGBoost による各年度における検知率

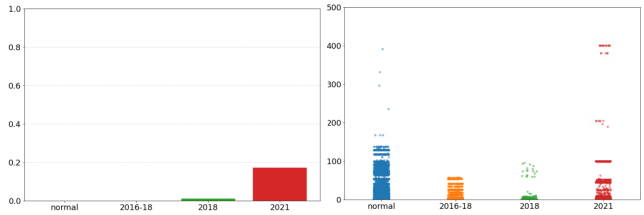
年	データ数 (件)	Recall
2021	3,669	0.6936
2020	1,944	0.9454
2019	982	0.8982
2018	6,129	0.9810

年の悪性データ，特に 2021 年のデータに対して悪性データの見逃しが多くなってしまう結果となった．2021 年の見逃したデータのマルウェアの種類について確認したが，特定のマルウェアの種類に対して検知できていたり，見逃しているという偏りは見られなかった．このことから全体的に通常通信との見分けが難しくなっていると考えられる．**図 5-6** に，良性データ (normal)，訓練時の悪性データ (2016-18), 2018 年と 2021 年の各特徴量の比較図を示す．

図 5 は大きく変化がみられた特徴量の例である．図 5 を見てわかるように，サーバ証明書が自己署名証明書である

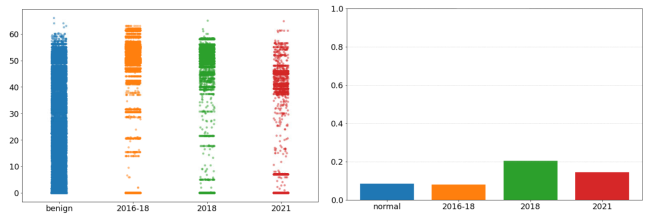


(a) 自己署名証明書を使用している割合　(b) TLS1.2,1.3 を使用している割合

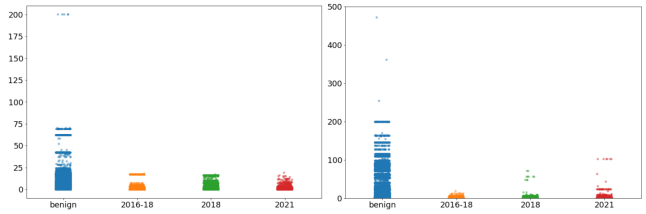


(c) 格安証明書を使用している割合　(d) SAN の数

図 5: 変化が大きく見られた特徴量



(a) トラフィックを提示するファイルの VT による悪性判定数　(b) JA3 がブラックリストに一致する割合



(c) これまで取得した TLS 証明書の数　(d) 過去最大 10 個の証明書の SAN の数

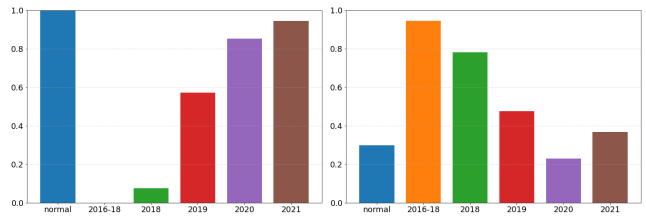
図 6: 変化の小さい特徴量

データ数の割合は，悪性データにおいて 2018 年から 2021 年で大きく減っており，また，**図 5c** を見ると，2021 年の悪性データでは格安証明書を使用した通信の割合が増えていることがわかる．自己署名など明らかな特徴に対しては対策しているものも多く，通常通信に紛れるよう，変更・偽装が行われていると考えられる．TLS のバージョンについても，今回特徴量としたが，**図 5b** を見てわかるように，2021 年の悪性データでは TLS1.2 または 1.3 を約 8 割が使用していた．情報処理推進機構 (IPA) によって 2020 年 7 月に TLS1.0, 1.1 を非推奨とされた [19] ため，この辺りも攻撃者によって通常通信との差がなくなっていくはずである．

逆に，今回あまり違いがみられなかった特徴についても，いくつか以下の図 6 に示す．ja3 についてはクライアント側の Extension などのハッシュ値なので，良性データでも該当しているが，悪性データにおける変化はあまり見られなかった．他にも**図 6c** や**図 6d** など過去の情報は，良性のデータにおいてこれまで蓄積されたものがあり，攻撃者側も意図的に変更するというのは難しいと見られる．

また，本研究では特徴量として用いていないが，先行研究において用いられた，TLS Extension の数，サーバが決定した Cipher Suite がクライアントの提示したリストの何番目であったかを，CTU-13 のデータセットを良性データとして，同様に比較し推移を確認した．それぞれ閾値を設定して，その値以上であれば 1，未満であれば 0 として，1 の値を持つデータの割合をグラフで比較した．これを**図 7** に示す．ここでは，良性データの場合はどのようであるかも同時に比較するために，CTU-13 のデータセットから取得したものを併せて載せている．

図 7 から，示した指標において，より新しいデータほどそ



(a) TLS Extension 数 (閾値：5 個)　(b) 選択された Cipher Suite のクライアント側の順位 (閾値：5 番目)

図 7: TLS メタデータの特徴推移比較 (閾値数を超えるデータの割合)

れぞれの特徴の設定が，通常通信と同傾向になっている悪性通信の割合が増えていることが分かった．OpenSSL などの TLS ライブラリで，Cipher Suite のカスタマイズや，デフォルトで用意されているものを使用することができるので，攻撃者が，これらの特徴が検知に繋がることを認識・意識して，ほとんど使用されていないような暗号方式を避け，通常通信で一般的に使われるような強度の強い暗号方式の選択や，拡張機能や楕円曲線関連の設定などに関しても，知識を持った攻撃者が検知を逃れるために，意図的に変更しているマルウェアが増えていることが予想される．この節の検証には，MALWARE-TRAFFIC-ANALYSIS.NET から取得できた pcap ファイルを使用した．**表 6** に訓練で用いた悪性データと各年度の悪性データについて，マルウェアの内訳を示す．

マルウェアの種類によっても回避のための偽装・変更度合いに違いがあると考えられ，今回もデータの偏りによる特徴の傾向も考慮するべきであると考える．

表 6: 各データセットのマルウェアの種類ごとの数

訓練データ	(2016-18 年)	2021 年		2020 年		2019 年		2018 年	
名前	データ数	名前	データ数	名前	データ数	名前	データ数	名前	データ数
Trickbot	28,268	QakBot	1,107	Trickbot	981	Emotet	575	unclass	5,346
Vawtrak	1,845	Trickbot	669	Emotet	740	Ursnif	143	Ursnif	345
Dridex	1,066	Emotet	586	Dridex	95	QakBot	120	Troldesh	130
miuref	830	unclass	394	QakBot	72	IcedID	89	Emotet	111
Locky	11	Hancitor	379	IcedID	52	Dridex	41	Trickbot	94
Zeus	8	STRRAT	252	GuLoader	4	Trickbot	14	Dridex	68
		IcedID	148						
		Dridex	134						

5. 結論

本研究では，TLS を利用して従来の検知システムを回避する悪性通信に対して，復号化を行わずに得られる情報として，先行研究で用いられたフローメタデータと TLS メタデータに加えて，バージョンや TLS サーバ証明書のコストや証明書のパス，過去の証明書の情報などに着目した特徴を追加し，機械学習の XGBoost を用いて悪性通信の検知を行うことを試みた．また，2018 年から 2021 年の悪性データに対して，本研究で得た学習済みモデルを用いて検知率の評価を行い，さらにマルウェアの通信におけるどのような特徴の変化があるかなどの検証を行った．

結果として，XGBoost において accuracy：0.9855, precision：0.9973, recall：0.8995, f1-score：0.946 を示した．適合率が高い一方見逃し率に対する評価値である再現率は少し劣る結果であった．メタデータのような特徴量のみではなく，通信に振舞いなどに着目した検知モデルを組み合わせ，主に見逃しを減らすことを目的とした検知システムを構築することが必要である．また，未知の悪性データセットに対する検知の実験では，特に 2021 年から取得したデータに対して検知精度の低下が見られた．これには，使用した特徴量の重要度の高いものにおいて，より良性データに近い傾向のデータがあることが原因であると考えられ，今後より通常通信に紛れるために攻撃者が対応してくるところであると考えられるため，，攻撃者の視点から今後深く追及する必要がある．さらに，今回の結果では時期が新しくなるにつれて徐々に検知率が悪くなっているわけではないことから，取得する悪性データ，マルウェアの種類などの偏りなども一因としてあると考えられる．

今後は，攻撃者が実際に攻撃を組み立て実行する手順において，変更や偽装が難しいと考えられる点に着目したシステムを構築した上で検知精度の向上を目指し，未知のマルウェアに対する検知やマルウェアファミリーごとの多値分類なども可能にした，より強固な悪性通信検知を目指したいと考えている．また，同一環境下での良性・悪性データの取得や，マルウェアの種類ごとの特徴の違いなども今後の課題となる．

参考文献

[1] MDN Web Docs.“開発者向けのウェブ技術：HTTP”. https://developer.mozilla.org/ja/docs/Web/HTTP, (参照 2022-01-10).
[2] Blake Anderson.“Detecting Encrypted Malware Traffic(Without Decryption)”. 2017 年 6 月. https://blogs.cisco.com/security/detecting-encrypted-malware-traffic-without-decryption, (参照 2022-01-10).
[3] R. Olivier, et al.,“Detecting Malware in TLS Traffic”, in partial fulfillment of the requirements for the MSc degree in Computing Sc ience Security and Reliability of Imperial College London (2019)
[4] 伊藤大貴, 高田雄太, 神薗雅紀.“安物に悪者が出る: 構築コストに基づく悪性ウェブサイト検知手法”, コンピュータセキュリティシンポジウム (CSS) 2021.
[5] Amirata Ghorbani, James Zou.“Data Shaply:Equitable Valuation of Data for Machine Learning”. arXiv:1904.02868
[6] “Welcome to the SHAP documentation”. https://shap.readthedocs.io/en/latest/index.html, (参照 2022-01-04).
[7] “Zeek Network Monitoring Project”. https://github.com/zeek/zeek, (参照 2021-11-19).
[8] “Ephemeral port”. https://en.wikipedia.org/wiki/Ephemeral_port, (参照 2021-10-21).
[9] “List of TCP and UDP port numbers”. https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers, (参照 2021-10-21).
[10] “VirusTotal”. https://www.virustotal.com/, (参照 2021-11).
[11] SSL blacklist by ABUSE.“JA3 Fingerprint Blacklist”. https://sslbl.abuse.ch/blacklist/, (参照 2022-01-13).
[12] “Let's Encrypt”. https://letsencrypt.org/ja/, (参照 2022-02-04).
[13] “ZeroSSL”. https://zerossl.com/pricing/, (参照 2022-02-04).
[14] Stratosphere Lab.“The CTU-13 Dataset”. 入手先 ⟨https://www.stratosphereips.org/datasets-ctu13⟩, (参照 2021-12-28).
[15] MALWARE-TRAFFIC-ANALYSIS.NET.“a source for pcap files and malware samples”. https://www.malware-traffic-analysis.net/, (参照 2021-12-23).
[16] Alexa.“the top 500 site on the web”. https://www.

alexa.com/topsites, (参照 2021-10).
[17] Majestic. “The Majestic Million”. https://majestic.com/reports/majestic-million, (参照 2021-10).
[18] Cisco Umbrella. “Cisco Umbrella 1 Million”. https://umbrella.cisco.com/blog/cisco-umbrella-1-million, (参照 2021-10).
[19] 情報処理推進機構 (IPA). “TLS 暗号設定ガイドライン”. https://www.ipa.go.jp/security/vuln/ssl_crypt_config.html, (参照 2022-02-07).