Vol.2022-CDS-34 No.1
2022/6/9

TLS 暗号通信における接続サービス同定

浅岡諒[1]　相馬悠人[1]　山内啓彰[1]　中尾彰宏[2]　小口正人[3]　山本周[4]　山口実靖[1]　小林亜樹[1]

概要：通信機器にて流れるトラフィックの接続先を同定することは，大規模な災害や輻輳の発生時などにおける優先制御などに活用できる．IP アドレスなどを用いる単純な接続先同定手法より発展した手法として，DPI(Deep Packet Inspection)によりパケットペイロードを解析して同定する手法がある．しかし，近年の通信の多くは，TLS(Transport Layer Security)により暗号化されており，ペイロードの解析は困難となっている．本稿では，TLS1.2 以前を想定し，セッション確立時に送信される平文情報を解析して接続サービスを同定する手法を紹介する．そして性能評価によりその有効性を示す．

キーワード：サービス同定，DPI(Deep Packet Inspection)，SNI(Server Name Indication)，TLS/SSL

1. はじめに

WWW(World Wide Web)は，ウェブページ検索，動画配信，SNS などの様々なサービスを提供している．ネットワーク機器を流れるネットワークフローからそのフローの接続先サービスを同定することは，サービスの QoS(Quality of Service)の管理，セキュリティ対策，特定のサービスへの接続に限り課金対象から除外をするゼロレーティングサービスの提供など，様々な目的において活用でき，重要である．サービス同定の基本的な手法として，IP アドレスとポート番号に基づく手法がある[1]．しかし，これらの基本的な手法では十分な精度でサービス同定ができないことがある．例えば，多くのサービスが一つの IP アドレスによって提供されることや，動的ポート制御[2]が使用されることがある．したがって，IP アドレスとポート番号に基づくサービス同定手法には精度に限界がある．この問題を解決できる先進的な手法として，DPI(Deep Packet Inspection)と呼ばれるパケットペイロードの検査がある．既存の研究[3][4]では，DPI により同定精度が向上することが示されている．しかし，近年のネットワークフローは TLS(Transport Layer Security)によって暗号化されていることが多く，このような暗号化されたパケットのペイロードを解析することはできない．一方で，TLS ヘッダ内の一部のフィールドは暗号化されていないため，解析可能である．例えば，TLS1.2 以下では，SNI(Server Name Indication)と呼ばれるフィールドは暗号化されていない．そして，これらの暗号化されていないフィールドの情報をもとに，サービス同定をすることができると期待できる．このようなサービス同定の実現は，主に 2 つの課題の解決によって実現される．1 つ目の課題は，各 TLS セッションをそれらが属するアクセスにクラスタリングすることである．2 つ目の課題は，1 つ目の課題でクラスタリングされた TLS セッション群からサービスを同定することである．本稿では，2 つ目の課題に焦点を当てて，考察を行う．1 つ目の課題は，アプリケーション同定のための手法[5][6]などの既存の研究により完全にあるいは部分的に解決することができる．これらの手法は，TLS セッションをアプリケーションごとにグループ化するものであり，Web ブラウザが 1 つの Web ページを開くために 1 枚のタブしか使用しない場合や，ブラウザが複数の Web サイトを同時にダウンロードしない場合などに，本アプリケーション同定手法により TLS セッションのクラスタリングが実現できる．

本稿では，TLS1.2 以下で暗号化された TLS セッション群のパケットペイロードを検査することで，それらセッション群の接続先サービスを同定する手法について考察する．サービス同定には，選択肢を与えられその中から選択する同定と，選択肢を与えられずサービスを特定する同定の 2 種類があり，本稿では前者の同定の方法について考察する．すなわち，あらかじめ指定された候補の中から最も可能性の高いサービスを選択する同定手法である．ここで，サービスとはサイトが提供する機能と定義し，ASP（Application Service Provider）のサービスとほぼ同じである．例えば，Gmail，Google Web 検索，YouTube は異なるサービスである．様々なサービスを含んだポータルサイト，例えば yahoo.com は，サービスではなく，接続先のサイトである．通常，ユーザは Web ブラウザに URL を入力し(あるいはリンクをクリックし)，Web ページを開く．本稿ではこのように Web ページを開くことを"1 アクセス"と定義する．多くの場合，1 回のアクセスで複数の TCP コネクションが確立され，HTTPS（Hypertext Transfer Protocol Secure）による接続の場合は，複数の TLS セッションが確立される．

本稿では過去に提案した 2 つの同定手法を紹介する．1 つ目の手法は，1 アクセス中の最初の TLS セッションの SNI

1 工学院大学
Kogakuin University
2 東京大学
University of Tokyo
3 お茶の水女性大学
Ochanomizu University
4

に基づいてサービスを同定する手法[3][23]である．2 つ目の手法は，SNI によって TLS セッションをクラスタリングし，1 回のアクセスに含まれる SNI に基づいてサービスを同定する手法[4]である．TLS1.2 以下の場合，SNI は *ClientHello* のフィールドであり，暗号化されていない．TLS1.3 では，SNI は暗号化される．したがって，これら提案手法は，TLS 1.2 以下用となる．

本稿は以下のように構成されている．2 章では関連する研究を紹介する．3 章では，過去に提案した手法を説明する．4 章では提案した手法を評価する．最後に，5 章で本稿の結論を述べる．

2. 関連研究

2.1 サービスとアプリケーションの同定

本節では，既存のサービスおよびアプリケーションの同定手法を紹介する．多くの場合，サービスの種類とポート番号は密接に関連していて[1], IP アドレスやポート番号に基づくサービス同定は最も基本的で簡単な同定手法である．しかし，この手法は精度が低く，改善の余地がある[5]．また，NAT(Network Address Translation)やリバースプロキシを利用する場合などは，この手法が効果的に機能しない[2]．さらに，一部のサイトでは，動的に変化する 1 つの IP アドレスから複数のサービスを提供していて，その様な場合は IP アドレスからサービスを同定することは困難である．また，Web で提供されるサービスの多くは，80 番や 443 番などのポート番号で提供されており，ポート番号からサービスを同定することは困難である．このように，ポート番号や IP アドレスは, アクセスに用いるプロトコルや接続するサーバを特定するのには有効であるが，サービスを同定するのには活用できない事例も多い．

Velan らは，暗号化されたトラフィックからの情報抽出に関する関連研究を紹介している [7]. その中で, すべてのネットワークプロトコルは暗号化されていない初期化フェーズを経てから暗号化による安全なデータ転送を確保することができることを指摘している．そして，これら初期化フェーズのデータは容易に抽出することができ，ネットワークトラフィックの監視に使用することができる[7]と述べている．Qualys, Inc.は，SSL(Secure Socket Layer)/TLS の初期ハンドシェイクに基づいてクライアントのフィンガープリントを取得する方法[8]を開発している．また，p0f ツール[9]は，トラフィックを監視することにより，OS やユーザエージェントなどのピアのシステムに関する情報を抽出することを実現している．Holz らは，トラフィック中の X.509 証明書を包括的に分析し，証明書の品質を明らかにしている [10]．Husák らは，ネットワーク監視と TLS フィンガープリントに基づくリアルタイムの軽量クライアント同定を提案している[11]．[8]では，SSL におけるハンドシェイクを HTTP クライアントのフィンガープリントとして利用する手法を提案している．Iwai らは，機械学習を用いたアプリケーション同定手法を提案し，82%の精度で同定可能であることを示している[12]. しかし, この手法は, 学習段階を完了するのに 5 日間を要する課題を有している．Hara らは，パケットペイロードを解析し，n-gram データベースを作成することにより，接続サイトの同定を行う手法[13]を提案している．これらの研究では，複数の TLS セッションを考慮していないため，複数の TLS セッションを含むサービスにおいてはうまく同定することができない課題を有している．また，これらの研究は，トラフィックからサービスを同定するのではなく，クライアントやアプリケーションや接続サイトを同定するものであり，サービスの同定に直接貢献するものではない．

Hara らは，与えられたフローからサービスを同定する手法を提案し，Google の 10 個のサービスに対して 100%の精度を達成している[14]．しかし，この手法は，他の Google のサービスや他の ASP サイトのサービスには適用されていない．本稿では，この既存の手法を n-gram 手法と呼ぶ．

2.2 TLS ハンドシェイク

TLS ハンドシェイクは，TLS セッションを確立するための処理である．図 1 は，TLS ハンドシェイクのメッセージシーケンスを示している．クライアントからサーバへの *ClientHello* には，クライアントが使用できる TLS のバージョン，暗号化スイート，圧縮アルゴリズムのリストと，鍵の生成に使用される乱数が記述されている．*ServerHello* には，暗号化で使用される TLS のバージョン，暗号化スイート, 圧縮アルゴリズムが記述されている．*Certificate* では，サーバの証明書と中間証明書が送信される．*ServerKeyExchange* では, 共通鍵の暗号化に使用する公開鍵が送信される．*ServerHelloDone* はサーバからのハンドシェイクのメッセージの送信が完了したことを示すために使用され, その後, クライアントはサーバの証明書を検証する．*ClientKeyExchange* では，サーバから送信された公開鍵を用いて，共通鍵の生成に必要な情報が送信される．*ChangeCipherSpec* と *HandshakeFinished* を用いて，クライアントとサーバは, それぞれ暗号化通信への変更と TLS セッションの確立の完了を確認する．

TLS1.2 以下では，TLS ハンドシェイク中の *ClientHello*, *ServerHello*, *Certificate*, *ServerKeyExchange*, *ServerHelloDone*

は暗号化されていない．そのため，これらのメッセージのペイロードを DPI に基づいて解析することで，サービスの特徴を抽出することができると考えられる．

2.3 N-gram に基づくサービス同定手法

本節では，n-gram 手法[16]について説明する．

n-gram とは，文書中の連続した *n* 個の項目の並びのことであり，一般にテキスト検索分野で用いられ，テキストデータにもバイナリデータにも適用できる，Hara ら[16]は，n-gram に基づくサービス同定手法を提案している．

当該研究では，サービスへの 1 アクセスごとに複数の TLS セッションが確立されることを想定している. 例えば，ユーザが Web ブラウザで「www.google.com」にアクセスすると，約 10 個の TLS セッションが確立され，当該研究および本研究ではこれを「1 アクセス」と見なしている．また, TLS セッションは TCP コネクションごとに 1 つ確立される．n-gram 手法では，これらのセッションをいくつかのグループにクラスタリングする．

共通鍵送信後の送信データは暗号化されているため，パケットペイロードの検査による特徴量の抽出は不可能となる．また，クライアントからサーバへの送信データには，主にクライアントの情報が含まれているため，当該研究ではサーバからクライアントへの送信パケットに着目している．n-gram 手法では，*ServerHello*，*Certificate*，*ServerKeyExchange*, *ServerHelloDone* の各メッセージを解析している．

当該手法は，調査フェーズと同定フェーズの 2 つのフェーズで構成されている．

調査フェーズでは以下の(1)～(4)の処理を，同定フェーズでは(5)の処理を実行する．

(1) 各サービスへアクセスをし，トラフィックを取得する. トラフィックは TLS セッションで構成される．
(2) 各 TLS セッションのハンドシェイクのメッセージ中の非暗号部を解析し，その部分に含まれる全ての n-gram の出現頻度を数え，n-gram 頻度データベースを作成する．このデータベースには，TLS セッションとその中の n-gram の頻度との関係が格納される．
(3) これらの TLS セッションを，TLS セッションの n-gram 頻度同士の相関係数を用いてクラスタリングする．
(4) 各サービスにおいて，各グループのセッション数を数える．この数をグループ頻度と呼ぶ．この調査フェーズで調査したトラフィックとグループ頻度の関係を保存するデータベースを作成する．これはグループ頻度データベースと呼ばれる．
(5) 同定対象のトラフィックのグループ頻度とグループ頻度データベースのトラフィックとの距離を，式(1)のマンハッタン距離を用いて算出する．そして，距離が最も小さいサービスを同定結果として決定する．

$$\text{Manhattan Distance}\,(\mathrm{A},\mathrm{B}) = \sum_{i=0}^{G} d_{org}(a_i, \mathrm{b}_i) \qquad (1)$$

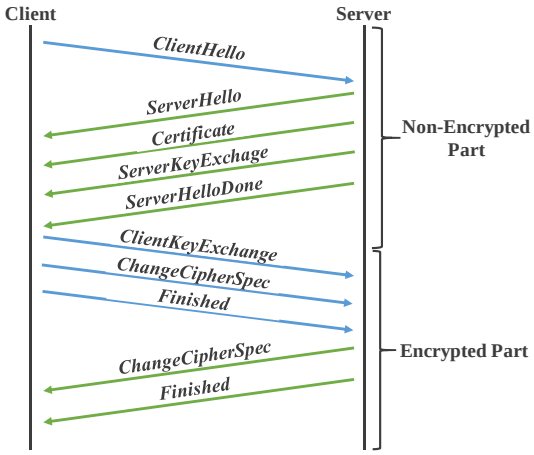

図 1 TLS ハンドシェイク

$$d_{org}(a_i, b_i) = |a_i - b_i|$$

この手法は，2 つのベクトル間の相関係数を多数回計算する．よって，この手法は計算に非常に時間がかかり，論文[16]では, 少ないフローで評価されている．この手法はスケーラブルではなく，短時間で計算できるよう改良する必要があると考える．

既存の手法[16] では，ダウンロードフローの非暗号部に含まれるすべての 2-gram 頻度を数え，すべての 2-gram 頻度からなるベクトルを作成している．この手法では，2 つのセッションのベクトルの間の相関係数を計算し，相関係数が閾値より高いセッションを同一グループに分類している．n-gram 手法では，TLS プロトコルの仕様に従ったバイトストリームの解析は行っておらず，2-gram のランダム値のフィールドも検索している．この動作は，同定精度を低下させると考えられる．

2.4 TLS セッションのクラスタリングと TLS ハンドシェイクのフィールドのクラスタリング能力

本節では，TLS ハンドシェイクにおける非暗号部に基づく TLS セッションのクラスタリングに関する考察を紹介する．

文献[16]の研究では，非暗号部を使用して TLS セッションをクラスタリングできることを示しており，文献[17]の研究では，ダウンロードフローにおける TLS ハンドシェイクの各フィールドのクラスタリング能力を調査し，文献[18]の研究では，*ClientHello* の各フィールドのクラスタリング能力を調査している．そして，*ClientHello* のフィールドである SNI は，TLS セッションを十分な数のグループに分類できることを明らかにしている．しかし，これらの研究は，サービス同定に関する貢献はしていない．

3. SNI 解析に基づくサービス同定手法

本章にて，SNI 解析に基づくサービス同定手法を 2 つ紹介する．1 個目が，最初の TLS セッションの SNI に含まれ

るホスト名による同定[3]であり，2 個目が SNI の出現による同定[4]である．

3.1 最初の TLS セッションの SNI に含まれるホスト名に基づくサービス同定

本節にて, 最初の TLS セッションの SNI に含まれるホスト名に基づくサービス同定[3]を紹介する．

1 アクセスごとの最初の TLS セッションの SNI には，アクセスされたサービスの URL のホスト名が記述されている．「1 アクセス」とは，1 章の第 2 段落で定義されている通りである．本同定手法は，最初の TLS セッションの SNI のホスト名と接続サービスの関係から，サービスを同定する．本手法は，最初の TLS セッションの SNI のホスト名がすべての候補サービスで一意である場合は，サービスを一意に特定する．ホスト名が一意でなく，n 個のサービスで共有されている場合は，候補となるサービスを n 個に絞り込み，同定を完了する．サービスは一意に定まらず，「n 個のサービスのうちの 1 つ」が同定結果となる．

本手法は，n-gram 手法[16]と同様に，調査フェーズと同定フェーズで構成される．調査フェーズでは，候補となる各サービスにアクセスした際のトラフィックを取得し，最初の TLS セッションの SNI フィールドのホスト名とアクセスしたサービスの関係のデータベースを作成する．同定フェーズでは, 同定対象のトラフィックのうち, 最初の TLS セッションの SNI からホスト名を抽出し，このホスト名とデータベースからサービスを同定をする．この手法は最初の TLS セッションのみを分析するため，CPU 時間の消費が少ない．

3.2 SNI の出現に基づくサービス同定

本節にて，SNI の出現の有無に基づくサービス同定[4]を紹介する．

本手法も，調査フェーズと同定フェーズで構成される．調査フェーズでは，1 アクセスで出現する SNI と接続サービスとの関係をデータベース化する．同定フェーズでは，このデータベースに基づいて，ベイズ推定により候補となるサービスの中から最も可能性の高いサービスを選択する．

調査フェーズでは，以下の手順で行われる．

(1) 候補となる各サービスに複数回アクセスし, そのトラフィックを取得する．
(2) 各アクセスにおいて，各 SNI の出現の有無（すべての TLS セッションにその SNI が含まれているか否か）を調査する．SNI の出現の有無は登場 SNI ベクトルで表現する．
(3) 各サービスと SNI の出現の関係をデータベース化する．これを，登場 SNI ベクトル DB と呼ぶ．

例えば，あるサービスにアクセスし，5 つの TLS セッションが確立され，各 TLS セッションに SNI が出現し，3 種類の SNI（sni1.com, sni2.com, sni3.com）が出現したとする．そして，全アクセスに登場する全ての SNI が sni1.com, sni2.com, sni3.com, sni4.com であるとする．このとき，登場 SNI ベクトルは(1, 1, 1, 0)となる．この登場 SNI ベクトルは登場 SNI ベクトル DB に登録される．

同定フェーズは以下の手順で実行される．

(1) 同定の対象となるトラフィックをキャプチャする．
(2) トラフィックに含まれるすべての TLS セッションにおける SNI の出現に基づき，登場 SNI ベクトルを作成する．
(3) (2)式を用いたベイズ推定により，各サービス候補の確率を算出する．最も確率の高いサービスを同定結果とする．

$$P(A|X) = \frac{P(X|A)P(A)}{P(X)} \tag{2}$$

ここで，$P(A|X)$は，登場 SNI ベクトル X が観測された条件下で，サービス A に接続される確率である．$P(A)$は，サービス A に接続される確率である．$P(X)$は，登場 SNI ベクトル DB から計算された，登場 SNI ベクトル X が観測される確率である．$P(X|A)$は，登場 SNI ベクトル DB を用いて算出された，サービス A に接続した際に登場 SNI ベクトル X が出現する確率である．

4. 性能評価

本章にて，n-gram 手法[16]と SNI 解析に基づく手法(3.1 節の手法[3]と 3.2 節の手法[4])のサービス同定精度を評価する．n-gram 手法は著しく時間がかかり，本稿の評価方法を適用することができなかったため，n-gram 手法を軽量化し，その軽量化した n-gram 手法で評価を行った．軽量化した n-gram 手法については 4.1 節で説明する．また，オリジナルの n-gram 手法を用いた場合の評価時間の予想についても，4.1 節で説明する．

Google, Yahoo, MSN のサイトにて提供される複数のサービスにアクセスし，そのときの IP パケットをキャプチャし, このトラフィックを用いて同定精度を評価した. Google および Yahoo のサービスでは，2018 年 9 月と 12 月の 2 シーズンにキャプチャし, MSN のサービスでは 2020 年 10 月と 11 月の 2 シーズンにキャプチャした．

n-gram 手法のパラメータは文献[16]のものと同様であり, 各グループのセッション数は 10, クラスタリング時の相関係数の閾値は 0.95，n-gram は 2-gram とした．

各サービスは 1 シーズンに 100 回アクセスされ，SNI 解析に基づく手法においては，そのうちの 90%と 10%をそれぞれ学習用データとテスト用データとした．サービスへのアクセスには，Mozilla Firefox を使用した．TLS セッションは全て TLS1.2 を用いて確立されたため, SNI は暗号化されていない．n-gram 手法[16]はクロスバリデーションに基づいて評価するに著しく時間がかかるため，クロスバリデーションは行わず性能を評価した．

表 1 Google15 サービスの候補

Name of Service	Hostname in URL
Google Account	myaccount.google.com
Google Calendar	calendar.google.com
Google Document	docs.google.com
Google Drive	drive.google.com
Google Mail	mail.google.com
Google Map	www.google.com
Google Photo	photos.google.com
Google Play	play.google.com
Google Plus	plus.google.com
Google Scholar	scholar.google.com
Google Sheets	docs.google.com
Google Translate	translate.google.com
Google Web Search	ww.google.com
YouTube	www.youtube.com

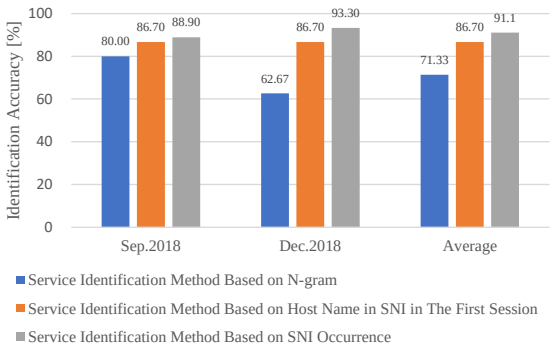

図 2 Google15 サービスにおける同定精度

4.2 節，4.3 節，4.4 節の実験では，各サービスに等確率で接続される($P(A)$は全てのサービスで等しい)と仮定している．これに対して，4.5 節の実験では，サービスへ接続される確率は Zipf の法則に従うと仮定している．

4.1 軽量化した n-gram 手法

本節にて，軽量化していない n-gram 手法において性能評価に要する時間の予測と，軽量化した n-gram 手法について述べる．

2.3 節で示したように，n-gram 手法では，TLS ハンドシェイク内の非暗号部の n-gram の出現頻度を数え，n-gram 頻度データベースを作成する．そして TLS セッションのクラスタリングの際には，TLS セッションの n-gram 頻度間の相関係数を計算する．

ある TLS セッションをクラスタリングするために計算する相関係数の数は，その時点で n-gram 頻度データベースに登録されている TLS セッションの数に比例する．そのため，n-gram 頻度データベースの作成に要する時間は，学習用データ数 n の $o(n^2)$に比例する．また，ある TLS セッションの n-gram 出現頻度のベクトル(2-gram の例にて 65536 次元のベクトル)と，別の TLS セッションの n-gram 出現頻度のベクトルの相関係数の計算も極めて短い時間では終わらない．よって，TLS セッションのクラスタリングには長い時間を要する．具体的には，Google のトラフィックを，サービス数 15，サービスあたりのアクセス数 20 としてクラスタリグを行ったところ，全アクセスに 6948 個の TLS セッションが含まれ，我々の計算機にて完了までに 2 日 3 時間 47 分を要した．完了に要する時間がアクセス数 n の 2

表 2 Yahoo10 サービスの候補

Name of Service	Hostname in URL
Yahoo Account	login.yahoo.com
Yahoo Advertising	avertising.yahoo.com
Yahoo Celebrity	www.yahoo.com
Yahoo Finance	fnance.yahoo.com
Yahoo Mail	mail.yahoo.com
Yahoo News	www.yahoo.com
Yahoo Web Search	search.yahoo.com
Yahoo Smart TV	smarttv.yahoo.com
Yahoo Sports	sports.yahoo.com
Yahoo Lifestyle	www.yahoo.com

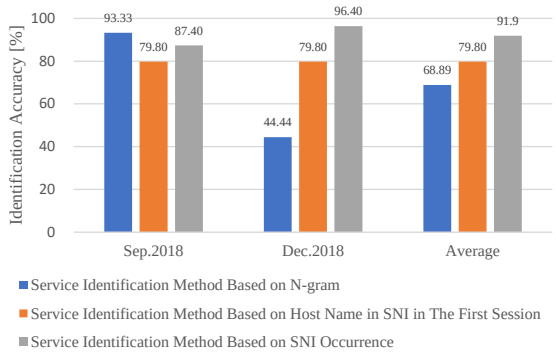

図 3 Yahoo10 サービスにおける同定精度

表 3 MSN6 サービスの候補

Name of Service	Hostname in URL
MSN Account	login.live.com
MSN Mail	outlook.live.com
MSN Map	www.bing.com
MSN News	www.msn.com
MSN Search	www.bing.com
MSN Store	www.microsoft.com

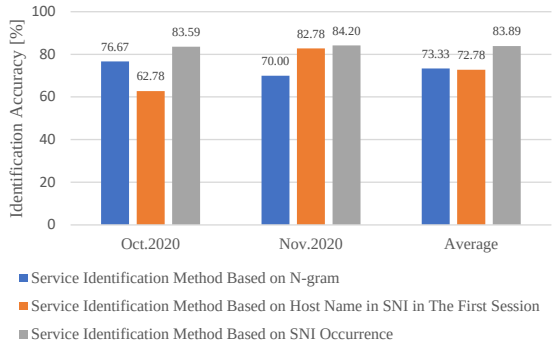

図 4 MSN6 サービスにおける同定精度

表 4 サービスごとのテストデータの数

Google 15 services	Uniform	Zipf
account	100	452
calendar	100	226
document	100	151
drive	100	113
gmail	100	90
map	100	75
news	100	65
photo	100	57
play	100	50
plus	100	45
scholar	100	41
search	100	38
sheets	100	35
translate	100	32
youtube_home	100	30

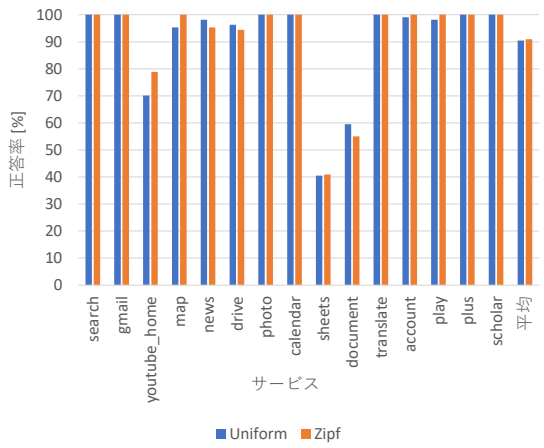

図 5 一様分布と Zipf 分布のサービス発生確率における Google の 15 サービスの同定精度 (SNI の出現に基づくサービス同定)

乗に比例すると仮定すると，データサイズを縮小せずにサービスあたりのアクセス数 90 としてクラスタリングをした場合はクラスタリング処理に 43 日程度要することとなる．我々の環境においてこれは現実的な時間ではなく，現実的な時間で評価を行えるように学習用データのアクセスの数を減らする必要がある．

そこで我々は，4.2 節，4.3 節，4.4 節における n-gram 手法の性能評価では，Google の 15 サービスおよび Yahoo の 10 サービスの学習用データを 1 サービスあたり 10 アクセスとし，テスト用データを 1 サービスあたり 10 アクセスとした．また，MSN の 6 サービスでは学習用データを 1 サービスあたり 5 アクセス，テスト用データを 1 サービスあたり 5 アクセスとした．

4.2 Google サービスによる評価

表 1 に示した Google の 15 サービスを用いた同定精度を図 2 に示す．図では，横軸がパケットをキャプチャしたシーズンを，縦軸が平均の同定精度を示している．同定精度は，正しく同定したときの評価値を 1，誤って同定したときの評価値を 0，サービス候補を n 個に絞りその中に正解が含まれるときの評価値を $1/n$，含まれないときの評価値を 0 として，全同定の評価値の平均である．また，登場 SNI ベクトル DB に登録されていない登場 SNI ベクトルが SNI 解析に基づく手法に与えられた場合は，サービス同定できず，評価値は 1/15 となる．

図 2 の結果から，SNI 解析に基づく手法の同定精度は n-gram 手法の同定精度よりも高いことがわかる．特に，SNI の出現に基づく手法の同定精度は，全てのシーズンで最も高い値となった．

4.3 Yahoo サービスによる評価

表 2 に示した Yahoo の 10 サービスについても評価を行った．図 3 に同定精度を示す．SNI 解析に基づく手法の同定精度が平均にて最も高いことがわかる．2018 年 9 月にキャプチャしたパケットの場合は SNI 解析に基づく手法の同定精度は n-gram 手法の精度にやや劣るが，2018 年 12 月のパケットの場合は SNI 解析に基づく手法の精度が大幅に高くなっている．

4.4 MSN サービスによる評価

表 3 に示した MSN の 6 サービスについても評価を行った．図 4 に同定精度を示す．SNI 解析に基づく手法の同定精度が平均にて最も高いことがわかる．特に，2020 年 11 月のトラフィックにて n-gram 手法と SNI 解析に基づく手法の差が大きいことが分かる．

4.5 Zipf 分布を用いた Google サービスによる評価

本節にて，サービスへ接続される確率が Zips の法則に従うと仮定して，SNI の出現に基づくサービス同定手法を評価する．n-gram 手法は，性能評価に長い時間を要し評価を行うことができなかった．最初の TLS セッションの SNI に含まれるホスト名に基づくサービス同定手法は，動作がサービスの発生確率に依存しないため 4.2 節と同じ精度となる．

表 4 に，本評価で用いたテストデータのアクセス数を示す．サービスの順位は我々の主観で決定し，アクセス数は Zipf の法則に従い決定した．また，比較のために一様分布のアクセス数も Uniform として記載している．

SNI の出現に基づくサービス同定手法の同定精度を図 5 に示す．図より，アクセスが発生する確率が均等でなく偏りがある場合で，SNI 解析に基づくサービス同定手法は高い精度で接続サービスを同定できることが分かる．一様分布と Zipf 分布の平均精度を比較すると，ほぼ同等であり，Zipf 分布の方が僅かに高いことが分かる．

5. おわりに

本稿では, TLS で暗号化された IP フローからそのフローの接続先サービスを同定する手法を紹介した．そして，実アクセストラフィックを用いてこれらの同定精度を評価し，SNI 解析に基づく同定手法が高い精度でサービスを同定できることを示した．

謝辞 本研究は JSPS 科研費 21K11854, 21K11874 の助成を受けたものである．

参考文献

[1] Server Name and Transport Protocol Port Number Registry, http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml, accessed Oct 30, 2020.
[2] A. W. Moore and K. Papagianaki, “Toward the Accurate Identification of Network Applications,” Passive and Active Network Measurement 2005, Lecture Notes in Computer Science, vol.3431, pp.41-54, Springer, Berlin, Heidelberg, 2005.
[3] H. Yamauchi, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “Service Identification Based on SNI Analysis,” IEEE Consumer Communications & Networking Conference (CCNC 2020), Work-in-progress paper, 2020.
[4] H. Yamauchi, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “A Study on Service Identification Based on Server Name Indication Analysis,” 10th International Workshop on Advances in Networking and Computing (WANC), Short paper, 2019.
[5] T. Iwai and A. Nakao, “Identification of Mobile Applications via In-Network Machine Learning Using N-gram for Application-Specific Traffic Control,” IEICE Te ch. Rep., vol.115, no.209, NS2015-78, pp.41-46, September 2015. (in Japanese)
[6] T. Iwai and A. Nakao, “Adaptive mobile application identification through in-network machine learning,” 2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS), Kanazawa, 2016, pp. 1-6, doi: 10.1109/APNOMS.2016.7737226.
[7] P. Velan, M. Čermák, P. Čeleda, and M. Drašar, “A survey of methods for encrypted traffic classification and analysis,” Int. J. Netw. Manag., vol.25, no.5, pp.355-374, September 2015.
[8] Qualys, Inc. “HTTP Client Fingerprinting Using SSL Handshake Analysis,” https://www.ssllabs.com/projects/client-fingerprinting/ , accessed Oct. 30, 2020.
[9] P0f, http://lcamtuf.coredump.cx/p0f3/, accessed Oct. 30, 2020.
[10] R. Holz, L. Braun, N. Kammenhuber, and G. Carle, “The SSL landscape: a thorough analysis of the x.509 PKI using active and passive measurements,” Proc. 2011 ACM SIGCOMM IMC '11, pp. 427-444, 2011.
[11] M. Husák, M. Čermák, T. Jirsík, and P. Čeleda, “HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting,” EURASIP J. Inf. Secur., vol. 2016, no.1, article no.30, December 2016.
[12] T. Iwai and A. Nakao, “Identification of Mobile Applications via In-Network M achine Learning for Application Specific QoS Traffic Control,” IEICE Tech. Rep., vol.114, no.477, NS2014-260, pp.487-492, March 2015. (in Japanese)
[13] M. Hara, S. Nirasawa, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “Fast Application Identification Based on DPI N-gram,” 2016 IEEE 17th Int. Conf. on High Performance Switching and Routing Workshop Prog., June 2016.
[14] M. Hara, S. Nirasawa, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “Service Identification by Packet Inspection based on N-grams in Multiple Connections,” 7th Int. Workshop on Adv. in Netw. Comp., 2016.
[15] E. Kohler, R. Morris, B. Chen, J. Jannoti and M.F. Kaashoek, “The Click Modular Router,” ACM Trans. on Comp. Syst., August 2000.
[16] M. Hara, S. Nirasawa, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “Service Identification by Packet Inspection based on N-grams in Multiple Connections,” 2016 Fourth Int. Symp. on Comp. Netw. (CANDAR), Hiroshima, pp. 686-690, 2016. Doi:10.1109/CANDAR.2016.0123.
[17] H.Yamauchi, A.Nakao, M.Oguchi, S.Yamamoto and S.Yamaguchi, “Clustering TLS Sessions Based on Protocol Fields Analysis,” COMPSAC 2018: 42nd IEEE Comp. Soc. Sign. Conf. on Computers, Software & Applications, FastAbstracts. 2018.
[18] H.Yamauchi, A.Nakao, M.Oguchi, S.Yamamoto and S.Yamaguchi, “A Study on Clustering Sessions of TLS based on Upload Message”, 15th Int. Conf. on IP + Opt. Netw. (iPOP2019), P-9, 2019.
[19] T. Kanaya, H. Yamauchi, S. Nirasawa, A. Nakao, M. Oguchi, S. Yamamoto, and S. Yamaguchi, “Intelligent Application Switch Supporting TCP,” IEEE Int. Conf. on Cloud Netw., October 2018
[20] Y. Soma, A. Nakao, M. Oguchi, S. Yamamoto, S. Yamaguchi and A. Kobayashi, "Ocurring SNIs for Servicve Indetification," 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), Kobe, 2020, pp. 586-587, doi: 10.1109/GCCE50665.2020.9292038.
[21] Yuto Soma, Akihiro Nakao, Shu Yamamoto, Masato Oguchi, Saneyasu Yamaguchi, Aki Kobayashi, "Service Identification of TLS Connections based on SNI Analysis," IEICE 2020 International Conference on Emerging Technologies for Communications, Dec. 2020.
[22] M. Finsterbusch, C. Richter, E. Rocha, J. Muller and K. Hanssgen, "A Survey of Payload-Based Traffic Classification Approaches," in IEEE Communications Surveys & Tutorials, vol. 16, no. 2, pp. 1135-1156, Second Quarter 2014, doi: 10.1109/SURV.2013.100613.00161.
[23] 松崎 涼, 山内啓彰, 中尾彰宏, 小口正人, 山本 周, 山口実靖, "SNI 解析に基づく TLS 暗号化通信のサービス同定", 信学技報, vol. 118, no. 301, NS2018-142, pp. 69-74, 2018 年 11 月.