Vol.2015-CSEC-70 No.19
Vol.2015-SPT-14 No.19
2015/7/2

匿名通信システム Tor におけるウルフウェブサイトの提案

中田 謙二郎[1,a)]　松浦 幹太[1,b)]

概要：匿名通信システム Tor は送信者と受信者のつながりの匿名性を保証する．しかしながら，その匿名性を破る攻撃も発見されつつあり，中でも指紋攻撃は攻撃に必要な資源が少なく現実的な脅威となりうるものとして注目されている．そこで我々は，指紋攻撃に対する防御の糸口としてウルフウェブサイトを提案する．ここでウルフウェブサイトとは，トラフィック上他のウェブサイトになりすましやすいウェブサイトと定義する．それぞれのトラフィックをウルフウェブサイトに擬態させることで，通信量のオーバーヘッドを最小限に抑えたまま指紋攻撃の攻撃成功率を大きく下げることができると考える．本稿では，本提案に向けた基礎実験について記述する．

1. はじめに

近年インターネットは急速に普及しており，様々な用途で利用されている．しかしながら，その急速な拡大に伴い，ユーザのプライバシが十分に保護されていないという問題に対する懸念が大きくなっている．これに対し，現在に至るまでユーザのプライバシを保護するための様々な技術が開発されてきた．その技術の一つに暗号通信がある．しかしながら，暗号通信は通信の内容を秘匿することができるが，送受信者の関係性を秘匿することはできない．そこで，送受信者の関係性を秘匿することを目的として考案されたのが匿名通信システムである．匿名通信システムは 1981 年に Chaum[4] によって初めて提案され，以降様々な匿名通信システムが考案された．この匿名通信システムの中で最も有名かつ普及しているのは Tor[6] である．Tor は，第二世代オニオンルーティングの実装にあたり，TCP プロトコル上の匿名通信を可能にし，各種アプリケーションにプライバシ保護を施す基盤技術として機能する．

一方で，Tor の匿名性を低下させる有効な攻撃手段も発見されている．指紋攻撃はその中の一つであり，Tor の入り口ノードの監視という弱い仮定で攻撃が可能であるため，現実的な脅威となりうるものとして注目されている．

多くの指紋攻撃が提案されるにつれ [3], [9], [13], [19], [20]，防御手法も複数提案されてきた [8], [12], [14], [21]．しかしいずれの防御手法も，防御のためのコストが大きすぎる，または攻撃成功率をほとんど下げられない，といった問題を持つ．例えば，すべてのパケットが MTU の大きさとなるようにパディングを行うことで，トラフィックの特徴を大きく失わせ攻撃を困難にすることはできるが，必要な通信量が非常に大きくなり現実的な解決法とはならない．

本研究では，指紋攻撃への効果的な防御の糸口として，ウルフウェブサイトを提案する．ウルフとは，多くの登録テンプレートに対し誤って受け入れられる入力情報のことである。我々は，ウルフウェブサイトを「指紋攻撃の妨げとなるほど、多くのウェブサイトと似た指紋になるウェブサイト」と定義する．このとき，実世界にウルフウェブサイトは存在するのか，あるウェブサイトをウルフウェブサイトに偽装させる方法にどのような方法があるか，といったことを探るのが研究の目的である．ウルフウェブサイトの発見及び偽装の研究を進めることにより，通信量のオーバーヘッドを最小限に抑えたまま指紋攻撃の成功率を大きく低下させることができると考える．

本稿では，第 2 章で前提となる Tor や指紋攻撃について説明し，第 3 章で先行研究について，第 4 章で我々が提案するウルフウェブサイトについて述べ，第 5 章で基礎実験について記述する．最後に第 6 章で今後の課題について述べる．

2. 匿名通信システム Tor と指紋攻撃

2.1 匿名通信システム

インターネットは我々に様々な恩恵を与えているが，同時に利用者のプライバシを脅かしている．今日，インターネットの急速な拡大に伴い，個人のプライバシを保護したまま情報の交換を行いたいという需要が高まっている．

[1] 東京大学 生産技術研究所
IIS, Meguro, Tokyo 153–8505, Japan
[a)] nakatak@iis.u-tokyo.ac.jp
[b)] kanta@iis.u-tokyo.ac.jp

SSL 通信に代表される暗号化通信は，通信の内容を第三者に対して隠蔽し，一定のプライバシを確保する．しかしながら自分が通信をしたという事実そのものを隠蔽したい場合も存在し，このような時に匿名通信システムが必要となる．

社会的にデリケートな問題を扱うとき，匿名通信システムは有用である [5]．例えば，WikiLeaks 等の内部告発サービスでは，ユーザは自分が告発したという事実を隠蔽したまま告発を行いたいはずである．また，国境なき記者団などは，告発やインターネット上での情報調査に匿名通信システムを用いることにより情報提供者のプライバシと安全を守っている．この様な告発者の匿名性を確保することは，匿名通信システムの有力な用途である．また，インターネット検閲が厳しい一部地域では，インターネットを通じた言論の自由が得られない．例えば中国では金盾と呼ばれるネット検閲システムを導入しており，インターネットユーザが政府に対し不利な情報を発信すると，金盾によって情報提供者の特定がなされる．しかし，このような検閲が行われているインターネット上でも匿名通信システムを導入することにより，自由な議論が行えるようになる．

このような用途が考えられる匿名通信システムであるが，その基本概念は 1986 年に Pfitzmann と Waidner によって提唱された [15]．その中で，彼らは匿名通信において最も重要な要素は以下の三つであることを示した．

- 受信者匿名性 recipient anonymity
  メッセージ M が受信者を持たないとき，受信者匿名性を考える必要はない．一方で，特定の受信者 R のみに M を送るとき，「M の受信者が R であること」を第三者に対して秘匿できるかどうか，を受信者匿名性と呼ぶ．
- 送信者匿名性 sender anonymity
  メッセージ M の送信者 S が，「M の送信者が S であること」を第三者に対して秘匿できるかどうか，を送信者匿名性と呼ぶ．
- 送信者と受信者のつながりの匿名性 unlinkability of sender and recipient
  送信者 S が受信者 R にメッセージ M を送るとき，M について「S が R に送信したメッセージであること」を第三者に対して秘匿できるかどうか，を送信者と受信者のつながりの匿名性と呼ぶ．

匿名通信は，実用上の観点では送信者と受信者のつながりの匿名性が満たされれば十分であることが多く，従ってほとんどの匿名通信システムもこの匿名性のみを保証している．

2.2 Tor

現在最も普及している匿名通信システムは，第二世代のオニオンルーティングにあたる Tor[6] である．Tor は，有志が提供する約 6,500[2] のノードとそれらの情報を管理するディレクトリサーバによってオーバーレイネットワークを構築する，低レイテンシの通信システムである．Tor は一日当たり約 2,000,000 のユーザに用いられており，毎秒約 6,000MB のデータ通信を行っている．本節では，指紋攻撃に関連する部分を中心に Tor を紹介する．

2.2.1 オニオンルーティング

オニオンルーティングとは，インターネット上で匿名通信を実現させるための技術である．これは，Michael G. Reed，Paul F. Syverson，David M. Goldschlag らによって発明され，アメリカ海軍によって米国特許 No.6266704 が取得されている [17]．Tor はオニオンルーティングを採用している最も普及した実装であり，ここで Tor がオニオンルーティングをどのように実装しているかの概略を説明する．

Tor では，三つの Tor ノード (OR1,OR2,OR3 とする) を用いてオニオンルーティングを実行する．最初に，クライアントは Diffie-Hellman 鍵交換を用いて，OR1，OR2，OR3 それぞれとセッション鍵を共有する．次に，クライアントはその三つの鍵を用いて，図 1 の様にメッセージを多重に暗号化し，三つのノードを中継してメッセージを送信する．このとき，メッセージが各ノードを通って段階的に復号されていく様が，玉ねぎの皮をむいていく様子になぞらえられてオニオンルーティングと呼ばれている．

オニオンルーティングを行うと，中継に参加した各ノードは自身の直前直後に関するつながりしか知ることはできない．従って，各ノードや途中の通信を盗聴した第三者が送信者と受信者の真のつながりを知ることはできない．オニオンルーティングの利点に，すべてのノードを信頼する必要がないということが挙げられる．仮にある一つのノードが悪意のある第三者に占拠された場合であっても，上記の理由により匿名通信の匿名性は破られない．

2.2.2 Tor のデザイン

Tor は，クライアントがその通信先と結び付けられるのを防止することを目的としている．すなわち，クライアントを監視する第三者が，クライアントがどのサーバにアクセスしているかを特定することができないようにし，またサーバ側からも，Tor を使用しているクライアントを一意に特定することができないようにする．

クライアントはフリーソフトウェアをダウンロード，インストールすることによって Tor を利用することができる．簡単には Tor ブラウザを介してインターネットにアクセスすればよく，これは Tor の利便性を高めている．

ディレクトリサーバは各 Tor ノードの可用性や帯域幅などを観測しており，定期的に既知の Tor ノードの状態リストを作成している．クライアントが Tor を利用する際には，まずディレクトリサーバに接続しこのリスト (consensus file と呼ばれる) をダウンロードする．その後クライアン

Vol.2015-CSEC-70 No.19
Vol.2015-SPT-14 No.19
2015/7/2

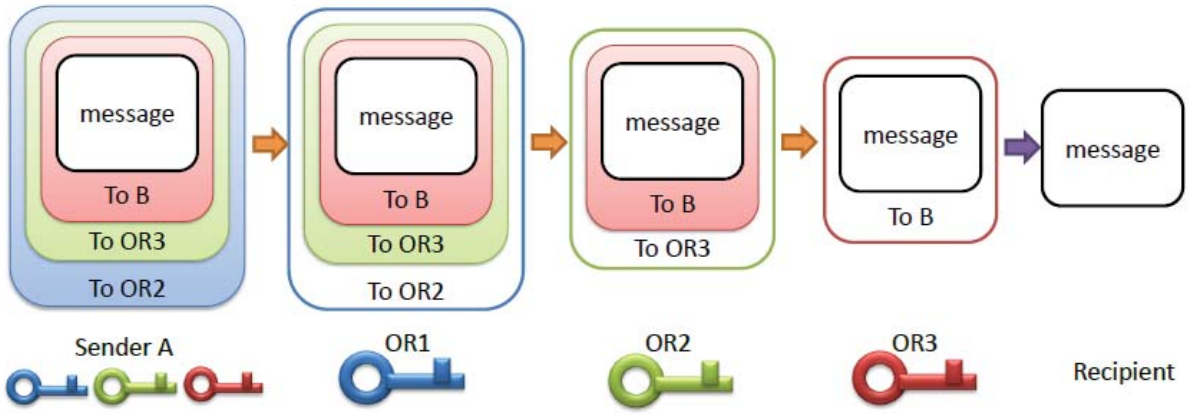

図 1　オニオンルーティング

トはリスト中のノードから三つのノード (OR1,OR2,OR3) を選択し，最初の中継ノード OR1 との暗号化通信路を形成する．この通信路は Diffie-Hellman 鍵交換を用いたセッション鍵の共有によって暗号化される．そして，この暗号化通信路を用いて OR1 と OR2 の間に関しても同様の暗号化通信路を形成し，また OR2 と OR3 についても同様の操作を行う．各暗号化通信路では TLS 通信が行われる．このようにして，クライアントは三番目のノード OR3 との接続を保持しているが，OR3 は OR1 やクライアントについての情報を知ることはない．同様に，OR1 もクライアントが OR3 にどのノードを選択したかを知ることはない．

クライアントの IP アドレスを知る OR1 によって匿名性が破られる確率を低くするために，Entry guard と呼ばれる仕組みが採用されている．まず，Tor ノードのうち十分な帯域幅を持ったノードのみが Guard フラグを得る．クライアントはフラグを持ったノードの中からいくつか (デフォルトで三つ) ランダムに選択し，それらを Guard リストとして保持する．このリストを作成する際，それぞれの guard について満了時間を 30 日から 60 日の間でランダムに設定する．この満了時間が過ぎると，その guard はリストから外れ，フラグを持った別のノードがリストに組み込まれる，といったようにローテーションさせている．実際に使われる OR1 は，Tor のパスが新しく作られるごとにリストの中からランダムに選ばれる．このように Entry guard として実際に使われるノードを限定することで，Tor では攻撃者の占拠，監視対象となる確率を下げている．

Tor の通信は，512bytes 固定サイズの，ヘッダとペイロードから構成されるセルにより行われる．ヘッダにはサーキット ID(circID) とコマンド (CMD) が含まれる．コマンドによりセルは制御セルかリレーセルに分けられ，リレーセルにはペイロードの前に付加的なヘッダが含まれる．

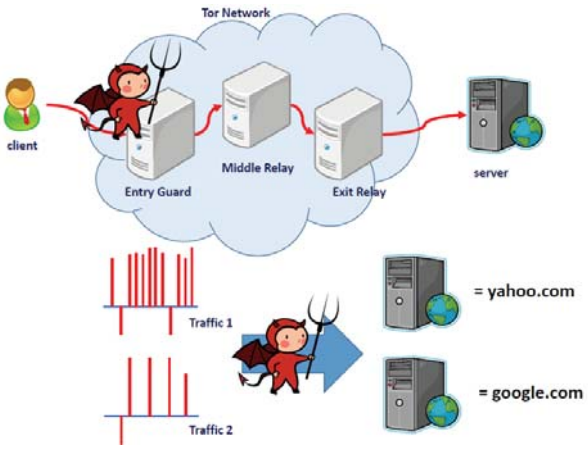

図 2　指紋攻撃

2.3　指紋攻撃

匿名通信システムに対する攻撃手法の中で有効な手法

に，指紋攻撃がある [18]．Tor に対する指紋攻撃の概観を図 2 に示す．Tor への指紋攻撃では，あるインターネットユーザが匿名通信システム Tor を用いてあるウェブサイトにアクセスしており，そのユーザが利用する Entry guard を攻撃者が観察している状態を想定する．

一般的なウェブページは，画像ファイルや javascript のソースファイルなど，多くの関連ファイルを読み込んだうえで表示される．すなわち，ウェブサイトにアクセスする際は，ウェブページファイルだけでなくそれらのファイルに対してもリクエストを行う．ウェブページファイル自体のサイズや関連ファイルの総数，及び個々のサイズはウェブサイト毎に異なるため，ウェブサイトにアクセスした際に生じる通信の流れもウェブサイト毎に異なったものとなる．この通信の流れ (ウェブトラフィックと呼ぶ) の中に生じるウェブサイト独自の特徴 (これを指紋と呼ぶ) をとらえ，ユーザがどのウェブサイトにアクセスしているかを特定する，ということが指紋攻撃の基本的な方針である．

指紋攻撃が他の攻撃と大きく異なるのは，観察すべきノードが一か所でよいという点である．Tor においては，OR1 もしくはクライアントから OR1 に至る経路上でパケットを盗み見ることができればよい．匿名通信システムに対する攻撃手法には，結託攻撃 (sybil attack)[7] や先行点攻撃 (predecessor attack)[22]，タイミング攻撃 (timing attack)[10], [11] や反射攻撃 (replay attack)[16] といった手法が考案されているが，いずれの手法も二つ以上のノードの観察が必要であり，実現可能性が低い．これに対し，指紋攻撃は必要な仮定が他の手法と比べて非常に弱く，攻撃の実現性が高いため現実的な脅威となりうる．

一般的な指紋攻撃の手順は以下である．まず，攻撃者は監視したいウェブページ群にアクセスし，生じるパケット列を収集する．次に攻撃者は，被攻撃者がウェブページにアクセスすることで生じるパケット列を観察する．そして教師有学習された分類器で被攻撃者がどのウェブページにアクセスしたかを特定する．

指紋攻撃を行う攻撃者は基本的に以下三つの仮定を置く

(1) 攻撃者は観測するパケットについてある一つのページロードの始点と終点を知る．
(2) 被攻撃者は一度に一つのページをロードし，ページローディングとファイルダウンロードを同時に行うといった行為は行わない．
(3) 攻撃者は被攻撃者と同じ条件で分類器を学習させることができる．つまり，クライアントの OS やネットワーク接続，Tor ブラウザのバージョンなどを同様にしたうえで攻撃を行うことができる．

これらの仮定は問題を簡単にするために置かれており，攻撃者優位な仮定である．

指紋攻撃を Tor で成功させるのは，SSH や VPN tunneling 上で成功させることよりも難しい [9]．これは，Tor はセルを用いて固定長のデータ単位で通信を行っているためである．これに加え，Tor では回線構築に用いられる制御セルなどがパケットとして流れており，指紋攻撃を行う際にノイズとなり攻撃の性能を低下させる．

3. 関連研究

本章では，指紋攻撃に関するいくつかの手法について，防御手法と攻撃手法に分けて記述する．なお，ここで紹介する防御手法は必ずしも Tor のために考案されたものではない．

3.1 防御

すでに Tor で実装されている防御手法に，pipeline randomization がある [14]．これは Tor の開発者たちにより提唱されたもので，HTTP パイプラインを使用可能にし，パイプラインサイズ及びリクエストの順番をランダムに定める．実装コストが小さく，オーバーヘッドが発生しないため現在でも用いられている．しかしながら，この手法によって指紋攻撃の成功率を大きく下げたという報告はない．

他のランダム性を防御に用いる手法に，traffic morphing[21] や Panchenko らの background noise[13] がある．traffic morphing では，あるウェブサイトにアクセスした際に生じたトラフィック (以下インスタンスと呼ぶ) のパケットサイズ分布を，別のインスタンスのパケットサイズ分布に確率的に近づける．しかし小さくないオーバーヘッドが生じること，静的に適用されること，Tor 上で適用したときにあまり効果が見られなかったという報告があることから [3]，解決法とはなりにくい．Background noise では，クライアントがウェブページにアクセスするのと同時に，ランダムに選ばれた別のページにもアクセスする．これにより攻撃成功率は大きく減少するが，大きなオーバーヘッドを必要とする．

決定論的な防御手法に HTTPOS[12] や BuFLO[8] がある．Tor が固定長セルを用いて通信を行っているのも決定論的手法にあたる．HTTPOS はクライアントのブラウザ上で動作し，MSS やウィンドウサイズのパラメータを調節することでパケットの大きさを不明瞭にする．オーバーヘッドは小さいが，この防御手法は効果が小さいという主張がある [3]．BuFLO では，送受信両方向に，通信が終わるまで，一定間隔でデータを送り続ける．攻撃成功率を大きく下げるが，オーバーヘッドが非常に大きい．

3.2 攻撃

Tor に対する指紋攻撃の初期の研究においてもっともよく知られているのは，2009 年の Herrmann らの研究である [9]．彼らは様々なプライバシ保護技術に対し単純ベイズ分類器を用いた指紋攻撃を行ったが，Tor に対する攻撃成功率は非常に小さかった．2011 年，Panchenko らは Tor

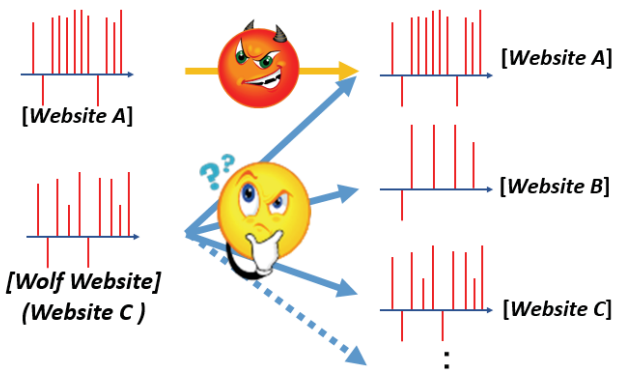

図 3　ウルフウェブサイトのイメージ

上でウェブサイトにアクセスするクライアントに対し攻撃を成功させた [13]．彼らは分類器に SVM を用い，HTTP ドキュメントの大きさや送受信パケットの割合などを特徴量として用いた．2012 年，Cai らは Tor における指紋攻撃成功率を大きく向上させた [3]．そこでは，パケット列を比較するために編集距離が用いられた．2013 年，Wang 及び Goldberg は編集距離を計算するアルゴリズムを変更することで攻撃精度をさらに向上させた [20]．2014 年，Wang らは現在最高の攻撃成功率をもつ手法を提案した [19]．先行研究で用いられた特徴量をほとんどすべて抽出し，それらに自動的に重みづけを行うことで最適なクラス分類を目指した．分類には k 近傍法を用いている．多くの特徴量を用い自動的に重みづけを行うことで，各防御手法が守ることができない部分を分類に使うことができるとし，この手法の頑健性を主張している．

4. ウルフウェブサイト

本研究では，指紋攻撃への効果的な防御の糸口として，ウルフウェブサイトを提案する．ウルフとは，多くの登録テンプレートに対し誤って受け入れられる入力情報のことである。我々は，ウルフウェブサイトを「指紋攻撃の妨げとなるほど、多くのウェブサイトと似た指紋になるウェブサイト」と定義する．

このイメージを図 3 に示す．登録情報に複数のウェブサイト A,B,C, ... が存在したとし，このうちウェブサイト C がウルフウェブサイトであったと仮定する．このとき，トラフィックを観察した攻撃者は，ウェブサイト A のトラフィックについては高確率でそのトラフィックが A のものであると判別することができるが，ウェブサイト C というウルフウェブサイトのトラフィックについてはそのトラフィックがどのサイトのものであるか判別するのが難しい，ということになる．

実世界にウルフウェブサイトは存在するのか，ウルフウェブサイトが見つかったとき，あるウェブサイトをウルフウェブサイトに偽装させる方法にどのような方法があるか，といったことを探るのが研究の目的である．ウルフウェブサイトの発見及び偽装の研究を進めることにより，通信量のオーバーヘッドを最小限に抑えたまま指紋攻撃の成功率を大きく低下させることができると考える．

5. 基礎実験

本章では，ウルフウェブサイトの提案に向けて行った基礎実験について記述する．主に述べるのは静的な実験である．つまり，通信の途中でノイズを加えるなどするのではなく，あらかじめ収集したパケット列に対しノイズを加え評価を行う．ただし，このシナリオは Tor に適用するには不適切である．Tor は低レイテンシを特徴とする実環境システムであり，遅延の原因となるようなパケットバッファリングは行わない．したがって Tor 上で働く防御手法を考える際には，その手法は動的に動作させる必要がある．しかしながら，実験の初期段階として，本稿では動的な手法を提案する足掛かりとして静的な手法を考える．

本実験システムの概観を図 4 に示す．実験ではますウェブトラフィックを収集し，加工を加える．その後各パケット列について防御を施したパケット列を用意する．そして特徴量を抽出し，それを多クラス分類器にかけることで，攻撃成功率を出す．ここで，攻撃成功率 (Accuracy) は以下の式であらわされる．

$$Accuracy = Success/All \tag{1}$$

ここで All は，攻撃者が指紋攻撃を行った総数である．一度の指紋攻撃では，攻撃者は 1 つのウェブトラフィックに対して 1 つの URL を推定する．この推定結果が正しかった指紋攻撃の総数が $Success$ である．

5.1 評価に用いる指紋攻撃

提案する防御手法を評価するための指紋攻撃として用いるのは，2014 年の Wang らの指紋攻撃である [19]．Tor に対する指紋攻撃の中で現在最高の攻撃成功率を誇るこの攻撃は，過去の指紋攻撃研究で用いられた特徴量を非常に多く採用している．当手法では，各トラフィックから 3736 個の特徴量を抽出し，それらに自動的に重みづけをすることで最適なクラス分類を行う．各防御手法に対し守られていない部分に自動的に集中するため，知られているすべての防御手法に有効であるとして手法の頑健性を筆者らは主張している．

本研究では，攻撃の頑健性に注目し，この指紋攻撃によって防御手法の評価を行う．

5.2 データセット

実験に用いるデータセットは，Wang らが指紋攻撃を評価する際に用いられたデータセットである [19]．このデータセットは実験の再現性のために指紋攻撃のコードとともに著者らによって公開されており，100 のウェブサイトへ

図 4　実験の概観

表 1　実験 1 の結果

	送信セル挿入	受信セル挿入	オーバーヘッド
挿入なし	0.9053	0.9053	0
5 セル毎に挿入	0.9003	0.9033	20%
3 セル毎に挿入	0.8937	0.9041	33%
1 セル毎に挿入	0.8708	0.9047	100%

の複数回アクセスをキャプチャし，加工したものである．100 のウェブサイトとして選ばれたのは中国や英国，サウジアラビアでブロックされているサイト群で，ここではアダルトコンテンツやトレント，宗教的，政治的話題といったものが取り扱われている．それぞれのサイトについて 90 回のアクセスをキャプチャし，セルを取り出す加工を行いデータセットとする．加工の方法は以下である．まず各 TCP/IP パケットについて送信パケットを 1，受信パケットを-1 と定める．そしてペイロードの長さを 512 で割り，その商をセルの数として 1 もしくは-1 と加工する．各サイトの 90 のインスタンスは 60 個が k 近傍法の学習に用いられ，残り 30 個が *Accuracy* の計算に用いられる．

5.3　実験方法

ウルフウェブサイト発見に向けて基本的な実験を主に 2 つ行った．ここでは，攻撃者は防御手法を知っていると仮定する，つまり攻撃者はセルが挿入された後のデータを学習データとして用いる．実験結果として示す値はすべて *Accuracy* である．

実験 1 は，定期的なセルの挿入である．キャプチャした送信，受信セルを足し合わせた数が 5，3，1 の倍数，となったときに送信セルを挿入した．受信セルの挿入も同様に行った．

実験 2 は，送受信セルの並びや頻度を消す挿入である．受信セルの直後に送信セルを挿入し，送信セルの直後に受信セルを挿入するという操作や，受信セルの直前に送信セルを挿入し，送信セルの直後に受信セルを挿入するという操作を行った．また，セルの順序を完全に消すため，受信セルの直前に送信セルを挿入し，送信セルの直後に受信セルを挿入するという操作も行った．

表 2　実験 2 の結果

挿入なし	0.9053
直後に逆向きセル挿入	0.6301
直前に逆向きセル挿入	0.6301
送信と受信が交互になるよう挿入	0.3630

5.4 実験結果

実験 1，実験 2 の結果をそれぞれ表 1，2 に示す．この値は，各場合毎に 5 回指紋攻撃を行い，その *Accuracy* の平均をとったものである．

実験 1 では定期的なセルの挿入を行った．受信セルを挿入した場合，挿入しなかった場合と比べ指紋攻撃に対しほとんど効果を発揮していないことがわかる．送信セルを挿入した場合，受信セル挿入に比べ指紋攻撃に対し効果を発揮している．しかし挿入によって通信量を二倍にしたときでも約 87%の攻撃成功率となっており，定期的なセル挿入の効果の低さが現れている．これは，指紋攻撃で用いる寄与の大きい特徴量のうち，通信量やセルの順序といった特徴があまり変化していないことに依ると考えられる．

実験 2 では，実験 1 の結果を踏まえ，セルの順序を大きく乱すように挿入を加えた．このとき実験のどの場合でも，1 つのセルに対し 1 つのセルを挿入しているので，通信量のオーバーヘッドは 100%である．キャプチャした全てのセルに対し，それぞれの直前または直後に，方向が逆となるセルを挿入した．特に，3 つめの場合では，受信セルの直後に送信セルを，送信セルの直前に受信セルを挿入することにより，送信セルと受信セルが完全に交互に並ぶようにした．実験 1 の結果と比べることで，同じオーバーヘッドが 100%となる挿入でも，実験 2 で行った挿入のほうが効果的であることがわかる．また，実験 1 の結果や，1 つ目，2 つ目の場合の結果，完全にセルの順序情報を消した 3 つ目の場合の結果を比べると，セルの順序が非常に大きな特徴となっていることがわかる．さらに 3 つ目の結果から，総通信量と時間情報のみから，約 36%の精度で攻撃が成功することがわかる．

6. 今後の課題

第 5 章で述べた基礎実験を踏まえ，ウルフウェブサイトの性質を探るため今後以下の点が課題として挙げられる．

6.1 データ収集

本稿で用いたデータセットは Wang らが過去に収集し加工を行ったものである．今後ウルフの研究を進めるにあたり，特徴の偏りを発見するためにより多くのウェブサイトを対象とする必要性が考えられる．しかし Tor のバージョンやネットワーク環境の違いから，公開されているデータに直接新たなトラフィックを増やすことは難しい．そこで独自にデータセットを取る必要があると考える．このときデータセットは以下のように作成する．対象とするサイト群は，Alexa Top sites[1] とする．これには先行研究でしばしば用いられており，比較検討がしやすいという利点がある．また，サイト内容の大きな変化を防ぐため，それぞれのサイトのトラフィック収集は 12 時間以上あけておこなうことがないようにする．さらに，学習データとテストデータで同じ Tor サーキットが用いられないようする．

6.2 特徴量の分布

ウルフは登録テンプレートの特徴量分布が偏ることによって生じる．そのため今後特徴量の分布を調べることになるが，5 章の実験からわかるように，ある程度似た総通信量をもつデータ間で分布を集める必要がある．総通信量という特徴を似せようとすると，大きなオーバーヘッドが必要となり研究の目的から外れるためである．ここで，特徴量としての総通信量をどう区分けるかというのは大きな課題だと考える．

6.3 ウルフウェブサイトの評価方法考案

本稿では，基礎実験の結果を表すために *Accuracy* を用いたが，これを直接ウルフウェブサイトの評価に用いることはできない．また，生体認証の分野においてウルフの評価に良く用いられている指標に WAP (Wolf Attack Probability) があるが，これもウルフウェブサイトの評価に用いることはできない．というのも，生体認証においてウルフの評価をする際その認証は一対一認証であるのに対し，ウルフウェブサイトの評価をする際その認証は一対多認証であり異なる認証がされているためである．従ってウルフウェブサイトの評価に向けて新たな評価基準を考案する必要があり，これは今後の課題である．

参考文献

[1] Alexa. http://www.alexa.com/.
[2] Tor metrics. https://metrics.torproject.org/.
[3] Xiang Cai, Xin Cheng Zhang, Brijesh Joshi, and Rob Johnson. Touching from a distance: Website fingerprinting attacks and defenses. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 605–616. ACM, 2012.
[4] David L Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.
[5] Roger Dingledine. Tor and circumvention: Lessons learned. In *Advances in Cryptology–CRYPTO 2011*, pages 485–486. Springer, 2011.
[6] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *In Proceedings of the 13th USENIX Security Symposium*, pages 303–320. USENIX Association, Aug. 2004.
[7] John R Douceur. The sybil attack. In *Peer-to-peer Systems*, pages 251–260. Springer, 2002.
[8] Kevin P Dyer, Scott E Coull, Thomas Ristenpart, and Thomas Shrimpton. Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 332–346. IEEE, 2012.
[9] Dominik Herrmann, Rolf Wendolsky, and Hannes Federrath. Website fingerprinting: attacking popular privacy enhancing technologies with the multinomial naïve-bayes classifier. In *Proceedings of the 2009*

ACM workshop on Cloud computing security, pages 31–42. ACM, 2009.

[10] Nicholas Hopper, Eugene Y Vasserman, and Eric Chan-Tin. How much anonymity does network latency leak? *ACM Transactions on Information and System Security (TISSEC)*, 13(2):13, 2010.

[11] Brian N Levine, Michael K Reiter, Chenxi Wang, and Matthew Wright. Timing attacks in low-latency mix systems. In *Financial Cryptography*, pages 251–265. Springer, 2004.

[12] Xiapu Luo, Peng Zhou, Edmond WW Chan, Wenke Lee, Rocky KC Chang, and Roberto Perdisci. Httpos: Sealing information leaks with browser-side obfuscation of encrypted flows. In *NDSS*, 2011.

[13] Andriy Panchenko, Lukas Niessen, Andreas Zinnen, and Thomas Engel. Website fingerprinting in onion routing based anonymization networks. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, pages 103–114. ACM, 2011.

[14] M. Perry. Experimental defense for website traffic fingerprinting. https://blog.torproject.org/blog/experimental-defensewebsitetrafficfingerprinting.

[15] Andreas Pfitzmann and Michael Waidner. Networks without user observability – design options. In *In Proceedings of a workshop on the theory and application of cryptographic techniques on Advances in cryptology-EUROCRYPT*, pages 245–253. Springer-Verlag New York, Inc, 1986.

[16] Ryan Pries, Wei Yu, Xinwen Fu, and Wei Zhao. A new replay attack against anonymous communication networks. In *Communications, 2008. ICC'08. IEEE International Conference on*, pages 1578–1582. IEEE, 2008.

[17] M.G. Reed, P.F. Syverson, and D.M. Goldschlag. Onion routing network for securely moving data through communication networks, July 24 2001. US Patent 6,266,704.

[18] Yi Shi and Kanta Matsuura. Fingerprinting attack on the tor anonymity system. In *Information and Communications Security*, pages 425–438. Springer, 2009.

[19] Tao Wang, Xiang Cai, Rishab Nithyanand, Rob Johnson, and Ian Goldberg. Effective attacks and provable defenses for website fingerprinting. In *Proceedings of 23rd USENIX Security Symposium (USENIX Security 14)*, San Diego, CA, August 2014. USENIX Association.

[20] Tao Wang and Ian Goldberg. Improved website fingerprinting on tor. In *Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society*, pages 201–212. ACM, 2013.

[21] Charles V Wright, Scott E Coull, and Fabian Monrose. Traffic morphing: An efficient defense against statistical traffic analysis. In *Proceedings of the 16th Network and Distributed Security Symposium*, pages 237–250, 2009.

[22] Matthew K Wright, Micah Adler, Brian Neil Levine, and Clay Shields. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Transactions on Information and System Security (TISSEC)*, 7(4):489–522, 2004.