匿名通信システム Tor における指紋攻撃とその対策の検討

横山 絵美里[1]　宗 裕文[1]　川端 良樹[1]　久保田 真一郎[1]　岡崎 直宣[1]

概要：近年，利用者のアクセスした Web サイトが特定されることを防ぐ匿名通信システムが注目されている．その中で最も普及しているシステムの一つとして Tor があげられる．Tor は多段階プロキシを利用することで安全な通信を提供しているが，これを脅かすような攻撃が考案されつつある．その中でも流れるトラフィックの特徴から利用者のアクセスする Web サイトを特定する「指紋攻撃」が脅威になっている．そこで本論文では，Tor に対する指紋攻撃に耐性を持たせるための手法を検討する．本手法では利用者が本来受信するはずの Web サイトのトラフィックを特徴の少ない情報のみに抑えることで指紋攻撃の効果を低減することを試みる．そして，提案手法について実験により評価を行い，その有効性を示す．

キーワード：匿名化通信，The Onion Router，Tor，指紋攻撃

An examination on countermeasure toward fingerprinting attack on the Tor anonymity system

Abstract: Tor is the most famous anonymity system supporting the anonymous transport of TCP stream over the Internet. Tor provides the foundation for applications to communicate over public network without compromising their privacy. However, it cannot completely hide the size of the contents and the timing of transmission of packets. Fingerprinting attack uses these features to infer the website being accessed by the specific client, and it has become a serious threat against Tor. In this article, we propose a new countermeasure toward the fingerprinting attacks and examine the feasibility and effectiveness of the proposed method.

Keywords: Anonymity System, The Onion Router, Tor, Fingerprinting Attack

1. はじめに

近年，インターネットの急速な普及により私たちはネットワークを介して様々な情報をやり取りできるようになっている．しかし，これに伴いインターネットを利用する際にパケットの通信を盗聴し，利用者がアクセスする Web サイトを特定する行為が問題となっている．

現在，この問題を解決するために匿名通信システム [1] が注目されている．これは自身を特定するような情報を通信相手に知られることなく通信を行うことができるシステムである．その中で最も普及しているシステムの一つが The Onion Router(Tor)[2], [3] である．今日この技術は，一般人，ジャーナリスト，活動家など様々な人々に自身に迫る脅威から身を守る手段として利用されている．

[1] 宮崎大学
University of Miyazaki

その一方で，Tor 利用者の匿名性を脅かすような攻撃も考案されている．その中でも流れるトラフィックの特徴から Web サイトごとにユニークな特徴 (指紋) を作成し，利用者のアクセスする Web サイトを特定する「指紋攻撃」[4] が脅威になっている．本論文では指紋となるトラフィックの特徴を減らすために，利用者が受信するコンテンツを制限するような手法を提案する．そしてこの提案を実験により評価し，その有効性を示す．

2. The Onion Router

Tor は米海軍調査研究所により開発された技術で，一日あたりおよそ 50 万人の人々に利用されており，最も利用されている匿名通信システムの一つである．Tor は複数のプロキシを経由させる仮想回線接続を行うことで，高い匿名性をもつ通信を実現している．

Tor の主な利用目的は自分自身のプライバシーを守りつ

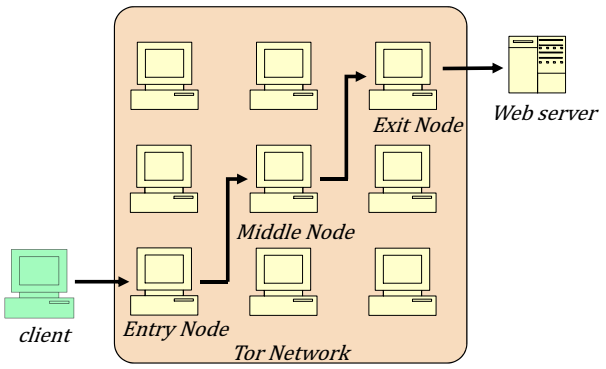

図 1　Tor の概略図
Fig. 1　Basic Components in Tor Network.

つ情報をやり取りすることである [2]．具体例としては，虐待や深刻な病気など同じ境遇を持つ人々間での情報共有や内部告発などがあげられる．このような情報をやり取りをする上で通信の匿名化は重要なことであり，Tor はインターネットを利用する上で必要な技術であるといえる。

Tor の通信は多段階プロキシを利用して行われる．Tor を利用する際，利用者は図 1 のようにオニオンルータ (以下，OR) と呼ばれる中継プロキシを三つ選択し，それぞれの OR と鍵交換を行う．このとき，利用者に近いほうから順に，入口 OR，中間 OR，出口 OR と呼ぶこととする．Web サイトへアクセスする際には，この三つの OR がパケットをそれぞれ順に暗号化を行うことで匿名化通信を実現させる．これにより，図 1 のように経路上のどの OR も利用者がアクセスした Web サイトを特定することができない．

Tor は上記のような処理を行うことにより安全な通信を提供しているが，様々な手段を利用して Tor の匿名性を下げようとする攻撃が存在する．例えば，Web サイトと直接通信を行う出口 OR では通信内容を暗号化できないことを悪用して通信内容を傍受する攻撃が存在する．この攻撃に対しては現在，送信するデータを HTTPS を利用して内容を暗号化することで対処することができる．しかし通信内容を傍受しなくてもトラフィックの特徴などから Web サイトを特定するような手法も考えられており，これに対する根本的な対策はまだ確立されていない．本論文ではこのような攻撃に対する対策手法について考える．

3. 既存研究

現在 Tor ネットワーク内には約 4000 ノードの OR が存在し，全てボランティアにより構成されている．そのため，攻撃者がノード群に攻撃を行う OR を含ませることは容易である（以下，攻撃者により占拠された OR を汚染 OR と呼ぶこととする）．しかし，一つの汚染 OR が特定の利用者（以下，ターゲット）の中継プロキシとして選ばれる可能性は非常に低い．

以下に，Tor の匿名性を低下させるような攻撃手法と代表的な攻撃手法である指紋攻撃への対策を示す．

3.1 偽造 Web ページインジェクション攻撃

[5] は偽造 Web ページを利用した攻撃である．この攻撃ではまず，攻撃者は偽の Web ページと汚染入口 OR，汚染出口 OR を用意する．汚染出口 OR は利用者からの Web ページの要求を検知すると，偽造 Web ページをパケットに混入させる．この Web ページにはリンク先のないイメージタグが挿入されている．利用者がパケットを受信すると，Web ブラウザは画像を取得しようとする．汚染入口 OR は画像の要求時のトラフィックパターンと自身を流れるトラフィックパターンを照合することで利用者を特定することができる．この攻撃は二つの OR が利用者の経路となる必要があり，強力な攻撃だが実現性が低い．

3.2 指紋攻撃

指紋攻撃は，攻撃者が Tor ネットワーク上を流れるトラフィックを観測することで利用者がアクセスする Web サイトを特定する手法である [4]．Web サイトは様々な画像ファイルやスクリプトファイルから構成されているので，Web サイトごとにファイル数やサイズ，トラフィックの流れなどユニークな特徴 (以下，指紋) が表れる．攻撃者は入口 OR としてトラフィックから指紋情報を収集し，その指紋から Web サイトを特定する．

指紋攻撃は前述した攻撃と大きく異なり，攻撃者が用意する OR が入口 OR の一つで済む．これにより二つの汚染 OR が同一経路に選択されるときに比べ，汚染 OR が一つ選択されるときの確率の方が高くなるので実現可能性も高くなる．

[4] では，指紋情報を分類するのに Support Vector Machine(SVM) を使用しており，54%の確率で Web サイトを特定できる．この手法の指紋情報にはパケットの総数，HTML のファイルサイズなど Web サイトから抽出できるような情報を用いている．

また [6] では，指紋攻撃に対する対策を行われた場合でも指紋攻撃を可能にする手法について提案している．この手法は，指紋攻撃への対策のためにトラフィックに何らかの処理がなされた場合でも，その処理を打ち消す逆処理を行うことでその対策を無効化するものである．

このように，指紋攻撃は Tor に対して非常に大きな脅威であるといえる．本論文の目的は指紋攻撃に対する対策を考案することであるが，効率的な対策を考えるためには指紋攻撃に対する理解が必要である．そこで，4. では [4] の手法を参考に指紋攻撃の実装を行いその脅威を検証する．

3.3 指紋攻撃への対策

前述のように指紋攻撃は Tor に対して脅威となる攻撃で

ある．この攻撃に対する対策についていくつか記述する．

[7], [8] では，ダミー情報をパディングすることで指紋攻撃への防御策を提案している．[7] の手法はパケット到着時間上の確率分布に従い，中間ノードでダミー情報をパディングする．これにより，Web サイトの特徴を観測されにくくなる．しかし，この手法はダミー情報を通信路に流すため，Tor に対する通信負荷が発生する．さらに，Web サイトごとの特徴が大きい場合 Web サイトを特定される可能性がある．

[4] では，Web サイトのトラフィックを利用した対策について提案されている．この手法では利用者が Web サイトへアクセスする際に目的の Web サイトとは別に，ランダムで選択された Web サイトを同時に要求するというものである．これにより，二つの Web サイトのトラフィックが混在して Tor ネットワーク内を通るので，攻撃者は利用者がアクセスする Web サイトを特定することが困難になる．しかし，この手法はトラフィック量も通常の倍に増加するため，Tor ネットワークへの通信負荷も大きくなる．

4. 指紋攻撃

本論文で想定する指紋攻撃は，既存のものと同様に攻撃者が入口 OR を汚染することで行うものとする．攻撃者は収集した指紋情報をもとに利用者のアクセスした Web サイトを特定する．

ここでは想定する指紋攻撃について示し，実験によりその脅威の度合いを確かめる．

4.1 指紋情報

ここでは想定する指紋攻撃における指紋情報について定義する．指紋情報とは攻撃者が収集するトラフィックから Web サイトの特徴になり得るものを抽出したものであり，トラフィックの量やパケット数などの要素からなるベクトル量で表される．本論文の指紋攻撃で設定した指紋情報は 6 項目であり，そこにトラフィックの入出力を含めた全 12 要素とする．これについて表 1 にまとめる．ここで，チャンクはパケットの向き (入力，出力) が前回向きの変わったときから次に向きが変わる直前までを一つの塊としてみたものである．その塊の合計サイズをチャンクと定義する．単位は byte である．

4.2 処理手順

本論文で想定する指紋攻撃は指紋情報収集フェーズ・Web サイト特定フェーズの二つに分かれる．

(1) 指紋情報収集フェーズ
攻撃者は指紋情報データベースを作成するために利用者として Tor を使用し，定期的に各 Web サイトへアクセスする．そして，そのトラフィックを収集し指紋情報のみを抽出する．その後，指紋情報をデータベースに格納する．このように，定期的に Web サイトへアクセスすることでショッピングサイトのような時間とともに変化するサイトにも対応することができる．

(2) Web サイト特定フェーズ
攻撃者は入口 OR としてターゲットが接続してくるのを待つ．ターゲットの接続を確認すると，(1) と同様にしてターゲットのトラフィックを収集する．指紋情報の抽出が終わると指紋情報データベースと比較を行う．このとき類似度が最も高かった Web サイトをターゲットがアクセスしたサイトとする．本論文では収集したターゲットの指紋情報と指紋情報データベースを比較するのにコサイン類似度を利用する．

表 1 指紋情報の要素
Table 1 Fingerprinting information.

指紋情報の要素	説明
トラフィック総量（byte）	パケットサイズの総量
パケット総数	パケットの総数
トラフィック平均（byte）	パケットの平均サイズ
トラフィック分散	パケットの分散
チャンク平均（byte）	チャンクの平均サイズ
チャンク分散	チャンクの分散

表 2 PC の仕様
Table 2 Spec of PC.

OS	Windows 7 Professional
CPU	Core2 Duo E8400 3.00GHz
Browser	Mozilla Firefox 25.0.1
Tor	v0.2.3.25

4.3 実験

ここでは想定する指紋攻撃の実現可能性について調査を行い，その脅威について示す．

攻撃者がデータベースを収集するのに使用する PC とターゲットからトラフィックを収集する PC は同一のものを用いる．このときの仕様を表 2 に示す．本実験で使用する Web サイトは全て実在するものとし，Alexa [9] のアクセスランキングの上位サイトから順に 100 サイト選択したものである．このときランキングには国別トップレベルドメインが異なるだけの同一サイトも含まれているため，これを除く Web サイトを選択することで重複のないようにした．本実験では通信トラフィックをパケットキャプチャすることで指紋情報を収集する．パケットキャプチャには Wireshark [10] を用いる．

本実験では Alexa [9] から選択した Web サイトから攻撃者用の指紋情報データベースを作成し，同様にしてターゲットの指紋情報も収集して比較を行う．全体の特定率 R，および各 Web サイトの特定率 r を，それぞれ $R = \frac{S}{N \cdot M} \times 100$，$r = \frac{s}{M} \times 100$ と定義する．ここで N，M，S，s は，それぞ

表 3　評価指標
Table 3　Evaluation indicators.

Web サイト特定率 (%)	指紋攻撃に対する耐性
$0 \leq r \leq 10$	指紋攻撃に耐性あり
$10 < r < 90$	指紋攻撃に耐性なし
$90 \leq r \leq 100$	指紋攻撃に脆弱

れ訪問する Web サイト数とその回数，全体の特定成功数，各 Web サイトの特定成功数であり，本実験では $N = 100$，$M = 10$ とした．

また特定率 r に対する指紋攻撃耐性の判定について表 3 のように定義した．同表で「指紋攻撃耐性あり」は Web サイトが特定される可能性が低いことを表し，「指紋攻撃耐性なし」は Web サイトを特定される可能性があることを示している．そして，「指紋攻撃に脆弱」は指紋攻撃を行われると高確率で特定される Web サイトのことを表している．

本実験では簡略化のために，ターゲットが Web サイトへアクセスする際に閲覧するページはトップページのみとする．また，閲覧時間は 2 分間とした．閲覧時間は Tor を利用して Web サイトを閲覧する際，通常より接続にかかる時間が多く必要となるため余裕を持たせた値を設定する必要がある．あらかじめ代表的ないくつかの Web サイトについて実験を行ったところ，2 分間の閲覧時間をとることでどの Web サイトでも全てのコンテンツを受信することができた．

4.4　実験結果・考察

本実験で Web サイト全体の特定率 56.8%という結果を得た．以下に本実験で得られた結果をグラフや表を用いて示す．図 2 は特定率 r に対する Web サイト数を示している．そして図 3 は各 Web サイトをコンテンツ数により 4 つに分類し，それに対する特定率の平均を表したものである．

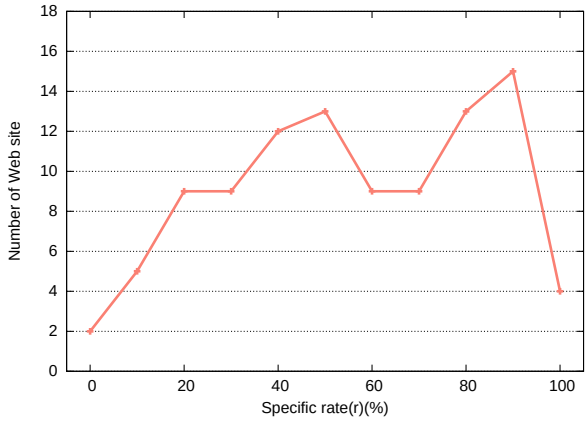

図 2　特定率 r に対する Web サイト数
Fig. 2　Number of Web sites with specific rate.

図 2 より「脆弱な Web サイト」が 19%，「指紋攻撃耐性なし」が 74%，「指紋攻撃耐性あり」が 7%を占めている．

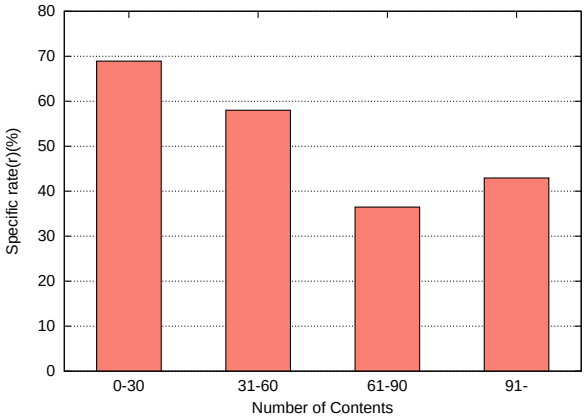

図 3　Web サイトのコンテンツ数ごとの特定率
Fig. 3　Specific rate by the number of contents.

つまり，指紋攻撃の危険にさらされている Web サイトが全体の 93%を占めていることがわかる．この結果から，現在の Tor ネットワークでは指紋攻撃は大きな脅威であり対策が必要であるといえる．

図 3 ではコンテンツ 30 以下の Web サイトの特定率が最も高くなっている．この理由として，コンテンツ数が少ないとその分トラフィックにノイズが入る余地がなくなり，Web サイトごとの特徴が少なくなることが原因だと考えられる．しかし図 3 ではコンテンツ数 91 以上の Web サイトの特定率がコンテンツ数が 31-60 の範囲の Web サイトの特定率より高くなっている．これは，コンテンツ数が多くなりすぎるとそのこと自体が指紋情報になるためだと考えられる．このように，Web サイトの特定率は Web サイトから読み込まれる画像や動画コンテンツの量に関係していることがわかる．もし，コンテンツ数の調整のみで Web サイトに指紋攻撃耐性を持たせようとした場合，コンテンツ数を 61-90 の範囲に収めることで特定率を 30%程度に抑えることが可能である．しかし，Web サイトのコンテンツ数に制限をもたせることは本来望ましくない．そこで，本研究では Web サイトのコンテンツに依存しない指紋攻撃対策を提案する．

5.　提案手法

5.1　目的

4.4 の考察から指紋攻撃に Web サイトから読み込まれる画像や動画コンテンツの量が関係していることがわかった．そこで本提案手法では通常読み込まれるはずの情報を読み込ませないようにすることで指紋情報に差がつきにくくなり，指紋攻撃の耐性があがるのではないかと考えた．以上から本論文では Web サイトアクセス時に HTML ファイルのみを読み込む手法を提案する．この提案により，通常の Web サイトのアクセス時に比べ，読み込むコンテンツ量が減少するので過通信が発生しにくくなる．さらに，Tor 側ではなくクライアント側での実装が可能なので導入

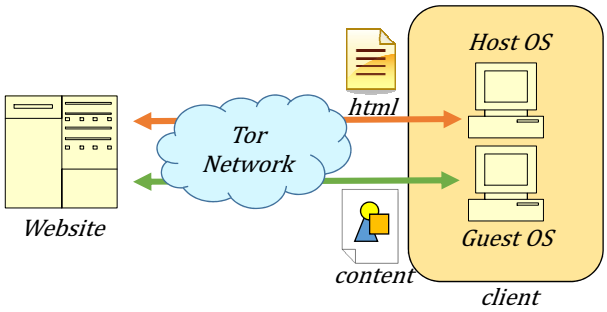

図 4　提案手法の概略図
Fig. 4　Basic components of the proposed method.

が容易である．しかし，本提案によりショッピングサイトや動画サイトなど HTML ファイルのみでは十分なサービスを提供できない Web サイトも存在する．これに関しては HTML ファイルとコンテンツを取得する際にそれぞれ別の経路を用意することで対処する．

5.2　概要

本提案手法はまず一方の経路で HTML ファイルのみを取得する．そして，利用者が Web サイトアクセス時に，必要なコンテンツがあればもう一方の経路で当該コンテンツのみを取得する．しかし，現在の Tor の仕組みでは一つのホスト内で起動することが出来る Tor Browser は一つのみである．そのため，通信経路を分割しようとすると通常 2 台の PC が必要となる．一方で，提案手法の導入を容易にするためには同一 PC 内で同時に Tor Browser を起動させることが望ましい．そこで，本論文では PC 内に仮想 OS を設け，一つの PC 上に二つのホストが存在するようにした．この条件下で同時に Tor を起動させる実験を行ったところ Tor Browser を同時に起動できることを確認した．

以下に本提案手法の前提条件と動作手順について述べる．

5.3　前提条件

指紋攻撃は入口 OR を汚染する必要があるが，本提案手法では経路を 2 本用意するため，攻撃者が指紋情報を収集するには二つの入口 OR を汚染する必要がある．しかし，同時に二つの入口 OR を汚染できる可能性は非常に低い．そのため本論文では二つの入口 OR が同時に汚染される場合を考えず，HTML ファイルを取得する側の経路が占拠された場合について考える．

5.4　処理手順

本提案手法ではホスト OS 側とゲスト OS 側それぞれで Tor を起動し，経路を 2 本用意する．ホスト OS 側では HTML ファイルのみを読み込み，ゲスト OS 側では画像ファイルなどのコンテンツを読み込む．このときの様子について図 4 に表す．

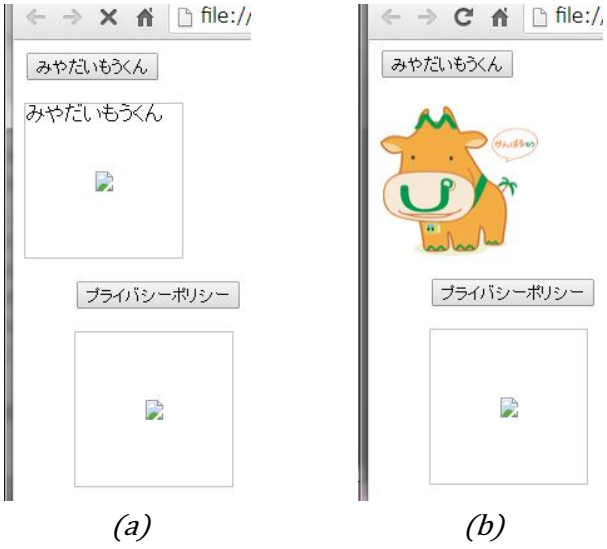

図 5　提案手法適用によるブラウザ
Fig. 5　The effect of the proposed method.

もし利用者が Tor を利用して Web サイトへアクセスする場合，まずホスト OS 側の経路で HTML ファイルのみを読み込み，ブラウザに表示する．このとき HTML ファイルのみを読み込んでいるため，ブラウザには画像が表示されない．そのときのブラウザの状態を図 5 の (a) に示す．しかし，画像がなければ理解できないサイトも存在する．そこで本提案では，利用者が求めるコンテンツのみを取得できるようにブラウザ上に作成したボタンを表示することで対応した．そして，利用者が読み込みたいコンテンツのボタンを押すとゲスト OS 側の Tor Browser が当該コンテンツのみを読み込み，ブラウザに表示する．そのときのブラウザの状態が図 5 の (b) である．

6.　評価

4. で示した指紋攻撃に対して本提案手法がどの程度の耐性をもつのか評価する．比較対象は提案手法と 4. で示した既存手法とする．

6.1　実験

(1) 実験環境

実験環境はゲスト OS 側も含め 4. と同じである．また，実験で使用した Web サイトも 4. で使用したものと同じである．

(2) 実装

本実験における提案システムの動作手順を図 6 に示す．本提案システムは，サブシステム A およびサブシステム B の 2 つ no サブシステムから構成される．同図において，サブシステム A は利用者が Web サイトにアクセスする際，ホスト OS 側から Tor を利用して目的の Web サイトへ HTML のみの要求を行う (a)．次

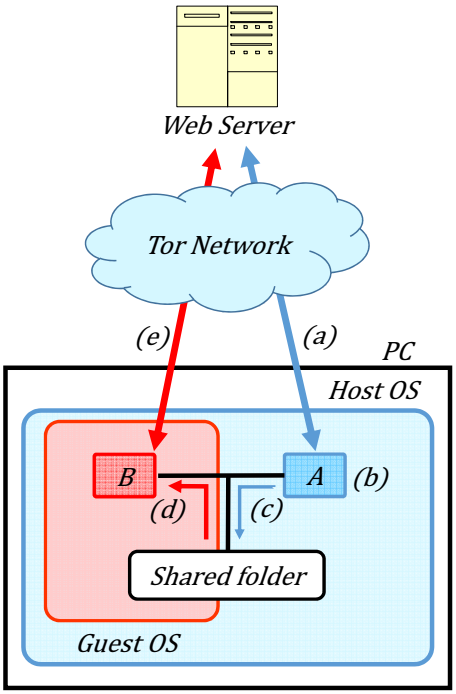

図 6　提案手法の実装図
Fig. 6　Implementation figure of the proposed method.

に (a) で読み込んだ HTML に 5.4 で述べたボタン作成の HTML 文を追加する (b)．ユーザがそのボタンを押してコンテンツを要求した場合，共有フォルダに要求したコンテンツの URL をテキストファイルに書き込み，共有フォルダに保存する (c)．サブシステム B は，共有フォルダから要求されたコンテンツの情報を受け取る (d)．受け取った情報をもとにゲスト OS 側からコンテンツの読み込みを行う (e)．本実験では HTML ファイルのみの評価を行うため (a) の実装を行った．

(3) 実験方法

本実験は 4．での実験と同様の 100 サイトを使用する．また，トラフィックをパケットキャプチャする方法も 4．の実験と同様に Wireshark [10] を用いる．

攻撃者の指紋情報データベースの作成は提案手法を適用し，4．と同様に 100 サイトへ 10 回ずつ訪問して HTML ファイルのみを取得することで行う．同様にしてターゲットの指紋情報も収集する．

6.2　実験結果・考察

本実験で Web サイト全体の特定率 34.7%という結果を得ることができた．以下に，本実験で得られた結果をグラフや表に示す．表 4 は提案手法と既存手法での特定率，表 5 は提案手法による指紋攻撃への耐性の変化について示している．図 7 は特定率 r に対する Web サイト数を示している．図 8 は各 Web サイトをコンテンツ数で分類したときに，それに対する特定率の平均を表したものである．また，図 7 と図 8 は本実験結果に加え，4．の実験結果も含めて示す．図 9 は実験により得られた結果の中でも脆弱な Web サイトのみに注目した特定率の効果について表している．

表 4　Web サイト全体の特定率
Table 4　Specific rate of Web sites.

	既存手法	提案手法
Web サイト特定率	56.8 %	34.7 %

表 5　指紋攻撃に対する耐性
Table 5　Tolerance for the fingerprint attack.

	既存手法	提案手法
指紋攻撃に耐性あり	7%	24%
指紋攻撃に耐性なし	74%	73%
指紋攻撃に脆弱	19%	3%

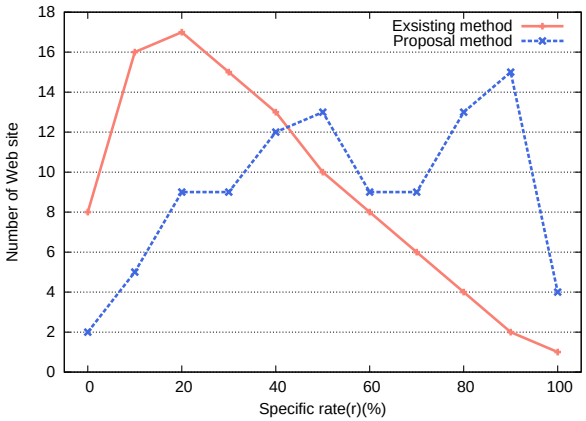

図 7　特定率ごとの Web サイト数
Fig. 7　Number of Web sites with specific rate .

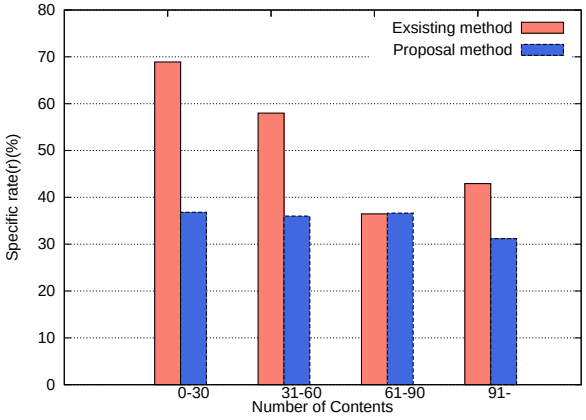

図 8　Web サイトのコンテンツ数ごとの特定率
Fig. 8　Specific rate by the number of contents.

表 4 と図 7 から本提案手法を適用することで多くの Web サイトに，指紋攻撃に対する耐性を持たせることができたといえる．

図 8 は本提案手法の適用範囲を示している．本実験で提案手法を適用したことで全てのコンテンツ数の範囲で特定率を低下させることができた．ここから本提案手法の適用

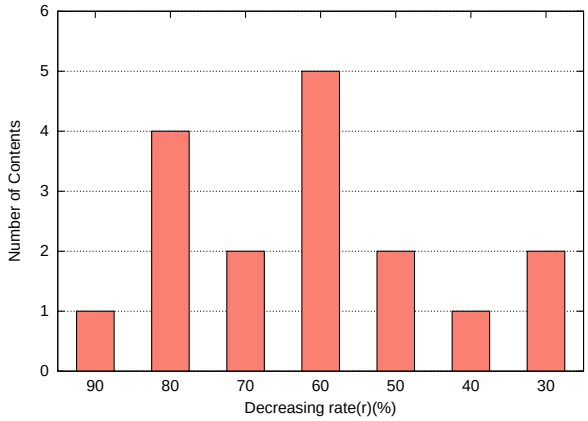

図 9　脆弱な Web サイトに対する提案手法の適用
Fig. 9　Effect of vulnerable Web sites.

範囲がどのコンテンツ数でも有効であること，そしてコンテンツ数 30 未満の Web サイトに最も効果があることがわかった．今後はコンテンツ数からだけでなく，コンテンツサイズなど他の項目からも本提案手法の適用範囲について調査していきたい．

図 9 と表 5 から本提案手法が最も効果を表す Web サイトが脆弱な Web サイトに対して高い耐性を提供していることがわかる．図 9 から，脆弱な Web サイトの特定率を最大で 90%，平均すると 61.76%低下させることに成功したことがわかる．全ての脆弱な Web サイトの特定率については 0%から 60%の範囲になった．そして，表 5 と図 7 より，提案手法を適用させることで「指紋攻撃耐性あり」を 24%にまで増加させ，「指紋攻撃に脆弱」を 3%まで大幅に減らすことができたことがわかる．しかし，「指紋攻撃に耐性なし」についてはあまり変化がみられなかった．

Web サイトの特定率を個別に見ていくと提案手法が全ての Web サイトに耐性を持たせているのではなく，逆に提案手法を適用することで匿名性を低下させている場合があることがわかった．特定率の上がった Web サイトに共通することは全ての Web サイトで，使用されている画像ファイルの更新が頻繁に行われていることであった．Web サイトの画像ファイルが頻繁に変化すると，そのときどきで取得される指紋情報に変化が生じ，指紋情報データベースと収集した指紋情報が一致しなくなる．しかし，本提案手法では HTML ファイルのみを取得するため，頻繁に画像ファイルが変化したとしても指紋情報は変化しない．これにより提案手法を適用した場合の方が特定率が高くなったと考えられる．

7. まとめ

本論文では Tor ユーザに対する匿名性を低下させる指紋攻撃に対して耐性をもたせるために，ユーザに取得させる情報を最小限に抑えるような防御手法を提案した．そして，この防御手法の実現可能性と効果を示すために実験を行うことで検証を行った．その結果，全体の特定率を約半分に抑えることができ，脆弱な Web サイトに対しては平均 60 %，最大で 90 %特定率を下げることができた．このことから，本提案手法は特に指紋攻撃に脆弱な Web サイトに対して最も効果を発揮できることがわかった．

本提案手法の目的は「指紋攻撃耐性あり」の Web サイトを増やすことであったが，実験により 3%から 24%にまで増加させることができた．今後はさらに「指紋攻撃耐性あり」の Web サイトを増やすための手法を考案していきたい．

さらに，本提案手法は [4] のように負荷をかけず，[7] のように Tor 側や Web サイト側で実装を行う必要がなく，クライアント側で即時に導入ができるため実現性の高い防御手法であるといえる．

今後は，提案手法を適用することで起こる特定率上昇の改善策の考案や，追加で取得される画像や動画ファイルによる特定率の違いについて調査を行っていきたい．

参考文献

[1] David Chaum, Communications Of The Acm, R. Rivest, David L. Chaum: Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM, Vol.24, pp.84-88, (1981).
[2] Tor Project: Anonymity online, (online), available from (`https://www.torproject.org/`), (2014.01.29).
[3] Roger Dingledine, Nick Mathewson, and Paul Syverson: Tor: The Second-Generation Onion Router, In Proceedings of the 13th USENIX Security Symposium Volume13, pp.303-320, (2004).
[4] Panchenko, A, Niessen, L, Zinnen, A, Engel, T: Website Fingerprinting in Onion Routing Based Anonymization Networks, In Proceedings of the 10th annual ACM workshop on Privacy in the electronic society, pp.103-113, (2011).
[5] X. Wang, J. Luo, M. Yang and Z. Ling: A potential HTTP-based application-level attack against Tor, Future Generation Computer Systems, Elsevier Science Publishers, vol.27, issue 1, pp.67-77, (2011).
[6] 横手健一, 松浦幹太: 匿名通信システム Tor の安全性を低下させるトラフィック逆加工, Computer Security Symposium 2012, Vol.3, pp.624-631, (2012).
[7] Vitaly Shmatikov and Ming-Hsui Wang: Timing analysis in low-latency mix networks: Attacks and defenses, Computer Security ESORICS 2006, 11th European Symposium on Research in Computer Security, pp.18-33, (2006).
[8] Andrew Hintz: Fingerprinting Websites Using Traffic Analysis, Privacy Enhancing Technologies, Lecture Notes in Computer Science Vol.2482, pp 171-178, (2003)
[9] Alexa: Alexa, The top 500 sites on the web, (online), available from , (`http://www.alexa.com/topsites`), (2014.01.29)
[10] Wireshark: Wireshark, (online), available from , (`http://www.wireshark.org/`), (2014.01.29)