Vol.2018-CSEC-81 No.11
Vol.2018-IOT-41 No.11
2018/5/17

匿名システムTorにおける指紋攻撃・遅延攻撃に対する分散レンジリクエストによる対策

安井 賢也[†1,a)]　真鍋 義文[†1,b)]

概要：インターネットで匿名性を得る手段として Tor プロキシが存在する。Tor ではオニオンルーター (OR) と呼ばれる複数のプロキシで通信データを多重に暗号化して送信を行う。多重に暗号化が行われているため、第三者が OR を流れているデータの内容および送受信者を直接把握することは不可能である。しかし、データの内容および送受信者を特定する手法として指紋攻撃、遅延攻撃およびそれらの組み合わせが考えられている。指紋攻撃とはデータ量やトラフィックの特徴などを統計解析にかけアクセスページを特定するものであり、遅延攻撃とはトラフィックが流れている途中などのルーターやスイッチなどで特定の遅延を発生させ複数の OR 間での通信を結びつけるものである。本稿では、利用ユーザーのアクセス先の末端の OR において、WEB ページの複数の静的コンテンツをまとめたデータに対するレンジアクセスを可能にすることと複数の経路を同時に使うことで、OR 間のネットワーク通信量を増大させることなく指紋攻撃、遅延攻撃を困難にする手法を提案する。

キーワード：Onion Router，Tor，匿名化通信，指紋攻撃，遅延攻撃

Distributed range request scheme against website fingerprinting attack and Timing attack on the Tor anonymous system

Kenya Yasui[†1,a)]　Yoshifumi Manabe[†1,b)]

Abstract: Tor proxy is used for anonymous transmission on the Internet. Tor transmits encrypted data via multiple proxies called onion router (OR). It is impossible for a third party to directly grasp the transmitter and the contents of the data flowing among the ORs since multiple encryption is performed. However, fingerprint attacks, timing attacks, and their combination are considered as methods for specifying data contents, senders, and receivers. Fingerprint attacks can identify the access page by statistically analyzing the amount of data and characteristics of traffic. Timing attacks are to link transmission among multiple ORs by generating delays in routers and switches. In this paper, we propose a range access to a cache file which summarizes contents on a web page to one data at the Exit Relay(OR). Furthermore, we propose a method to make fingerprint attacks and attacks difficult without increasing the amount of network traffic between ORs by using multiple routes at the same time.

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

1. はじめに

インターネットが普及し日常的に情報を取得し共有する時代になった。人々は盗聴されたり情報を盗まれる可能性があるため、https や ssh などが推奨されるようになり point to point での暗号化が進んでいる。しかし point to point の暗号化ではどのようなサイトを閲覧しているのか第三者に漏れる可能性がありプライバシーの侵害を受けることもある。そこで匿名化システムとして Tor プロキシが

†1 現在，工学院大学大学院情報学専攻
Presently with Guraduate school of Infomatics, Kogakuin University
a) em17019@ns.kogakuin.ac.jp
b) manabe@cc.kogakuin.ac.jp

存在する [1]。Tor ではオニオンルーター (OR) と呼ばれる複数のプロキシで通信データを多重に暗号化して送信を行う。多重に暗号化が行われているため、第三者が OR を流れているデータの内容および送受信者を直接把握することは不可能である。しかしこの Tor でさえ盗聴によってプライバシーが損なわれる可能性がある。攻撃者は Tor の利用者が訪れたウェブサイトを推測するため、通信のデータ部を直接分析するのではなくウェブサイトの指紋攻撃 (フィンガープリンティング) を行うことができる。フィンガープリンティングは、暗号化されたトラフィックを盗聴し、パケットサイズ、パケット間隔、転送されたデータ量などトラフィックの特定の特徴に基づいて統計的にウェブサイトを特定する手法である [2]。また遅延攻撃という特定のパケットを遅延させ、OR 間の通信の特定を行う攻撃手法なども存在する [3]。更に、パケットの遅延攻撃を指紋攻撃に利用し、より高い精度でウェブサイトを特定するアクティブな指紋攻撃も存在する [4]。これらはネットワークレベルでパケットを監視するだけでなく、それらを操作できるアクティブな攻撃者、ISP、クラウド事業者、データセンター事業者などの通信事業者が攻撃者となることを想定している。そういった攻撃者は、Tor 利用者のリクエストを数百ミリ秒間遅延させてから、匿名化ノードに中継して、要求されたウェブページに含まれる後続の通信パケットを入手することができる。アクティブな攻撃手法は、単にパッシブな情報収集だけをする攻撃手法 [2] と比較して、検出率を 48.5 %から 65.0 %に向上させ、誤検知率は 0.35 %と低い。こういった背景より、既存の対策手法としてパケットに余分なデータを付加する方法などがあげられている [5]。しかし既存手法では OR 等に余計な負荷をかけてしまったり、ウェブサーバー側の実装に依存していたりするため、本稿では負荷増大や依存などのない Tor の機能として 1 経路あたりのトラフィック量を操作可能な分散レンジリクエスト手法を提案する。本稿では、第 2 節で Tor について、第 3 節で提案手法、第 4 節でシミュレーションについて、最後に結論を述べる。

2. Tor

2.1 Tor について

Tor は米海軍調査研究所により開発され、現在では有志によって 6000 以上のノードが動作している。Tor は最も利用されている匿名通信システムの一つで複数のプロキシを経由させる仮想回線接続を行い、オニオンのように多重に暗号化をかけることで高い匿名性を実現されている。検閲が厳しい国や企業内などからの不正告発に利用されたり、Tor でしかアクセス出来ず、匿名でサイトが運営出来る.onion に利用されている。匿名でやりとりが出来る掲示板などが存在する。

2.2 Tor に対する攻撃

Tor は匿名通信を提供しているが、Tor の匿名性を下げる攻撃が存在する。例えば、ウェブサイトと直接通信を行う出口 OR では通信内容を Tor で暗号化できないため通信内容を盗聴することが可能である。この攻撃にはコンテンツを提供しているサーバー自身を https などで暗号化する必要があり、個人でも簡単に証明書を入手出来る Let's encrypt 等のサービスが広がり、世界的に暗号化が推進されている。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ウェブサイトを特定するような手法が提案されており、根本的な対策はまだ確立されていない。以降で Tor に対する攻撃として指紋攻撃と遅延攻撃について記述する。

2.3 指紋攻撃

2.3.1 攻撃手法

指紋攻撃は、通信インフラや特定の OR ノードを掌握することで可能となる。Tor 以外のシングルホッププロキシに対しても検証されており、ベイズ推定を利用した検出率 97%以上の結果が出ている [4]。また、Tor に対しても SVM(Support Vector Machine) を使うことで 54%の検出率を出している。ここでは、既存手法を 1 つ紹介する [2]。

まず、指紋情報の収集として照合するサイトに定期的にアクセスしトラフィックの盗聴を行い、表 1 に示す指紋情報をデータベースに保存する。

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	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	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	チャンク (パケット一つあたり) の平均サイズ
チャンク分散	チャンクの分散

そして利用者がアクセスしたサイトの指紋情報をデータベースの情報とコサイン類似度を用いて照合する。「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%であり、指紋攻撃が有効なサイトが 93%あることが示されている。

2.3.2 既存の対策手法

既存の対策手法としては、いくつかの種類が提案されている [6], [7], [8]。

(1) 効果的にパケットに余分なデータを付加する
(2) `http` レンジリクエストが可能なコンテンツは分割してアクセスする
(3) `MSS` などパケットのデータ部の大きさを任意の大きさに変えられる通信経路を通る

(4) html ファイルと静的コンテンツファイル (css,javascript,image 等) を別の経路で取得する

しかし、いずれも Tor ネットワークに余計な負荷をかけてしまったり、サイト側の実装に依存していたり、静的コンテンツの特徴量が出てしまっていたりする。

2.4 遅延攻撃

遅延攻撃は TCP のセッションが切断されないように故意にパケットを遅延させることであり、この効果は様々な攻撃に利用されている。DoS 攻撃でも利用されており、サーバーのセッションを故意に開かせたままにして負荷をあげたり、Tor に対しては特定 OR 間のパケットを遅延させ、別の OR 間のセッションと結びつける攻撃、指紋攻撃と組み合わせてより特徴量の高いトラフィックの取得などに利用されている。Tor に対する攻撃を実施するには、通信インフラまたは特定の OR を掌握しておく必要があるが、より効果的な攻撃が出来ることが示されている [4]。遅延をさせないで攻撃した結果が 54%だったのに対して、匿名化システムを利用しているユーザーに気づかれないようなアクティブな遅延を起こさせることで 65%まで検出率を増大させている。本稿では、こういった指紋攻撃や遅延攻撃の対策手法として次節で、分散レンジリクエストを提案する。

3. 提案手法 分散レンジリクエスト

提案手法が解決する Tor の問題点は以下の 2 点である。

(1) パケットの特徴量からサイトを特定する指紋攻撃への根本的対策
(2) 特定の OR 間でパケットを遅延させた時に他の OR 間のセッションと結びつけようとする攻撃に対する耐性をつける

上記の特性を満たすために提案手法では、以下のような機能を追加する。

提案手法

(1) コンテンツサーバーに一番近い OR で html ファイルや静的コンテンツファイルを取得した際、一つのファイルとして処理出来るようにし、レンジリクエストを可能とする
(2) OR 経路の作成時に分岐していくようにし、複数の経路で一つのセッションのように処理する

レンジリクエストは、http レンジリクエストと同様に 1 つのファイルに対して、範囲を決めてその部分だけリクエストを送れるようにする機能である。これらの機能を追加するとコンテンツにアクセスするトポロジーは、以下の図 1 のようになる。

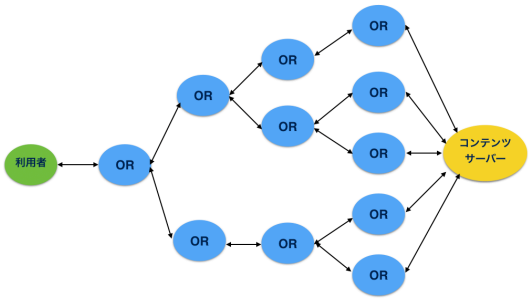

図 1 分散レンジリクエストのトポロジー

利用者から OR にリクエストがされた後、各 OR は以下の処理を行う。

提案手法

(1) 入口 OR・中継 OR：リクエストが通った際に分岐するかどうかを確認し、http のリクエスト内容はそのままでレンジリクエストのレンジを分岐ごとにわけて、次の OR に繋いでいく。
(2) 出口 OR：コンテンツサーバーにリクエストを送り、コンテンツを一つのファイルにまとめた後、リクエストデータレンジの部分をレスポンスとして返す。

OR 間のプロトコルは、http 情報をラップしメタデータとして各 OR で分岐するかのフラグとレンジリクエストに使用するレンジを出口 OR に送れるようにする。提案手法の分散レンジリクエストの処理順序は、以下のようになる。

分散レンジリクエスト

(1) 最初に Tor クライアントでホップ数から使用する OR を決める際、分岐数から分岐する OR も決定し、各出口 OR でリクエストするレンジを決めておく。
(2) Tor ネットワークに通信を開始し、各 OR をプロキシに設定していく際には、分岐するかどうかの情報も追加で送る。
(3) 出口 OR に http リクエストを送る際にはリクエストデータレンジの情報も追加で送る。
(4) 出口 OR は、http リクエストを送ったレスポンスの html、css、Javascript、image 等を一つのファイルにまとめ、レンジリクエストのレンジデータだけをレスポンスとして返す。
(5) レスポンスはデータとそのレンジを返す。
(6) 全てのレンジデータが Tor のクライアントに返ってきたら、ファイルを再構成し、html、CSS、Javascript、image 等を読み出す。

上記のようにコンテンツを取得すれば、1 つの出口 OR から返ってくるパケットを任意のパケット総量にすること

が可能になる。出口 OR でのパケット取得量は増えるが、Tor ネットワーク内の流通トラフィック量は、余計なデータを付加しないので負荷が増大しない。また、ある経路で TCP セッションが切断されない程度の遅延が発生したとしても別経路からもパケットの返答がされており、遅延させている OR 以外の OR 間で遅延させたデータを発見することを困難にさせる。

次節にて、実際にシミュレーションした手法と結果を示す。

4. シミュレーション

4.1 実験・環境

実験環境・ツールに関しては、Linux、wireshark[9]、tshark、tcpdump、iproute2、gRPC[10]、nginx を主に利用した。OR 間のプロトコルには、gRPC という http2 の特性を持った RPC フレームワークを用いて通信を行い、http リクエストとリクエストレンジ情報はオニオンのように暗号化し、分岐フラグについては各 OR の公開鍵で暗号化しその OR でのみ参照出来るようにした。シミュレーションする際に注意した点は、内部ネットワークを作成する際に MTU を 1500、MSS を 1454 にし、日本のフレッツ回線に合わせて行った。そのためチャンクの平均は、基本的に殆ど 1500 近いものになった。ファイルの大きさを先にとれないためレンジは、1 から 1000 として、出口 OR でとったファイルの大きさを 1000 としてレンジを出口 OR の数に合わせて区切った。トポロジーとしては、先に説明した図 1 のものを作成した。コンテンツサーバーは、nginx サーバを立て、html、css、javascript、image 等のファイルの大きさを変えながら配信を行った。

4.2 結果

html、CSS、Javascript、image 等をリクエストした際に分岐以降の任意の OR でパケットを観測した結果、分岐数とリクエストレンジのレンジ幅を任意の値にすることでトラフィック量を高い自由度で操作することが出来た。指紋攻撃に対しては最初の OR で分岐するだけで Tor ネットワーク内で流れる 1 経路あたりのトラフィック量は、分岐せずにアクセスした時と異なるものになり、ウェブサイトの特徴量から外すことが出来た。遅延攻撃に関しては遅延させられた OR 間から分岐した場所以降は他の経路のパケットと混ざり、分岐する場所ではレンジリクエストのデータ部を分ける処理等が入るためパケットの大きさが変わり特徴も大きく変わるため遅延を起こしたパケットを見つけるのは困難になった。分岐していない場所で特徴的なパケットを遅延させると OR 間のリンクを行いやすく、分岐している場所が偏っていると多くの場所でリンクすることが出来た。そのため遅延させられた場所の入口・出口側の両サイドの近い所で分岐するほど OR がリンクされるリスクは低いため OR は分散させてトポロジーを作る必要があった。遅延攻撃に対しては、分岐を交互にするようなトポロジーだとリンクするのが困難になった。

5. まとめ

トラフィック量を操作できるようにしたため、OR の分岐以降で指紋攻撃をしてもサイトを特定することは困難になった。遅延攻撃に対しても分岐を超えた OR 間の通信を紐付けることが困難に出来た。今後としては、提案手法では動画や websocket などセッションを張り続けるものに対応するのは難しいので、そういったもので特徴量がとれるのかどうかを調査した上で改善を考えいく。また WebRTC など多くのプロトコルが生まれているので、そういったものの匿名化についても考慮する余地がある。

参考文献

[1] Tor Project: tor, (online), available from, 入手先 ⟨https://www.torproject.org/⟩ (参照 2018-04-09)
[2] A. Panchenko, L. Niessen, A. Zinnen, and T. Engel : *Website fingerprinting in onion routing-based anonymization networks*, in Proc. of the ACM CCS Workshop on Privacy in the Electronic Society, Chicago, Illinois, USA, 2011, pp. 103-114.
[3] 青木 太一, 青木 卓矢, 佐藤 直, 島田 要, 山口 信幸, 高橋 正樹 : 匿名通信システムの攻撃手法に関する調査 コンピュータセキュリティシンポジウム (2015-10)
[4] Gaofeng He, Ming Yang, Xiaodan Gu, Junzhou Luo, Yuanyuan Ma : *A novel active website fingerprinting attack against Tor anonymous system* Computer Supported Cooperative Work in Design (CSCWD), Proceedings of the 2014 IEEE 18th International Conference (2014-05)
[5] S. Yu, G. Zhao, W. Dou and S. James *Predicted Packet Padding for Anonymous Web Browsing Against Traffic Analysis Attacks* in IEEE Transactions on Information Forensics and Security, vol. 7, no. 4, pp. 1381-1393, Aug. 2012.
[6] Xiapu Luo , Peng Zhou , Edmond W. W. Chan , Wenke Lee , Rocky K. C. Chang , Roberto Perdisci : *HTTPOS: Sealing Information Leaks with Browser-side Obfuscation of Encrypted Flows* Network and Distributed System Security Symposium (2011)
[7] 横山 絵美里, 宗 裕文, 川端 良樹, 久保田 真一郎, 岡崎 直宣 : 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 マルチメディア、分散協調とモバイルシンポジウム 2014 論集,2014,498-505 (2014-07-02)
[8] Tao Wang, Xiang Cai, Rishab Nithyanand, Rob Johnson, Ian Goldberg : *Effective Attacks and Provable Defenses for Website Fingerprinting* Proceedings of the 23rd USENIX Security Symposium. (2014-08)
[9] Wireshark: Wireshark, (online), available from, 入手先 ⟨http://www.wireshark.org/⟩ (参照 2018-04-09)
[10] gRPC: gRPC, (online), available from, 入手先 ⟨https://grpc.io/⟩ (参照 2018-04-09)