Vol.2018-CSEC-81 No.19
Vol.2018-IOT-41 No.19
2018/5/18

MITB攻撃によるコンテンツ改ざん検知手法の検討

高田 一樹[1,2,a)]　邦本 理夫[2]　吉岡 克成[3]　松本 勉[3]

概要：近年，インターネットバンキング等の金融機関サービス利用者を狙ったサイバー攻撃による不正送金被害が社会問題となっている．インターネットバンキングに関わる不正送金の多くは，金融系マルウェアによる Man In The Browser 攻撃（MITB 攻撃）によるものである．MITB 攻撃は，コンテンツを改ざんし，認証情報の盗取や自動送金等を引き起こす．インターネットバンキングにおける MITB 攻撃への対策は，金融機関の配布する専用対策ソフトの利用やワンタイムパスワードによる決済認証の強化等が一般的である．特に専用対策ソフトは金融系マルウェアに特化した検知機能を有しているもの等があり，有効性が高いと考えられる．しかし，専用対策ソフトを使用するか否かは利用者の判断に委ねられるため全ての利用者には導入されない可能性がある．そのため，専用対策ソフトを使用しない利用者も含む全ての利用者に適用可能な，MITB 攻撃対策が必要と考える．我々は，専用ソフト等を使用しない MITB 攻撃によるコンテンツ改ざんを容易に検知する手法の検討を行った．本稿では，MITB 攻撃によるコンテンツ改ざんが外部サーバと連携して行われると言う点に着目し，改ざんされたコンテンツが外部サーバと行う通信を Web ブラウザ上で捕捉することで，改ざん検知を行う手法について提案する．また，検知ロジックの実現性の検証実験を行った．

Investigation of Detecting Web Content Tempering caused by MITB Attack

Kazuki Takada[1,2,a)]　Michio Kunimoto[2]　Katsunari Yoshioka[3]　Tsutomu Matsumoto[3]

1. はじめに

近年，インターネットバンキング等の金融機関サービス利用者を狙ったサイバー攻撃による不正送金被害が社会問題となっている [1]．不正送金を引き起こす攻撃方法は多数存在しているが，その 1 つに金融系マルウェアによる，Man In The Browser 攻撃（以下，MITB 攻撃）が存在している．MITB 攻撃は，金融系マルウェアが感染 PC の Web ブラウザに対し，メモリインジェクション等の方法で入り込み，通信の監視および改ざんを行う攻撃手法である．MITB 攻撃により，インターネットバンキングの正規コンテンツが改ざんされ認証情報の盗取や自動不正送金等が引き起こされる．

金融系マルウェアによる MITB 攻撃の一般的な対策として専用対策ソフトの導入やワンタイムパスワードによる決済時の認証強化等が挙げられる．専用対策ソフトは，金融系マルウェアに特化した検知機能を持つものがあり，代表的なものとして SaAT Netizen[2]，Trusteer Rapport[3]，PhishWall プレミアム [4] 等が挙げられる．これらの専用対策ソフトは，保護対象のサイトに接続した際に自動的にブラウザの保護や金融系マルウェアによる攻撃の検知等を行うものである．また，ソフトによっては，保護対象のサイト接続時以外も対策機能が有効なものが存在する．これらの専用対策ソフトは金融系マルウェアによる MITB 攻撃に限らず有効な対策手法である．しかし，専用対策ソフト

[1] 横浜国立大学大学院環境情報学府
Graduate School of Environment and Information Sciences, Yokohama National University, Yokohama, Kanagawa 240-8501, Japan
[2] 株式会社セキュアブレイン
SecureBrain Corporation, Chiyoda, Tokyo 102-0094, Japan
[3] 横浜国立大学大学院環境情報研究院/先端科学高等研究院
Graduate School of Environment and Information Sciences, Yokohama National University / Institute of Advanced Sciences, Yokohama National University, Yokohama, Kanagawa 240-8501, Japan
[a)] takada-kazuki-hw@ynu.jp

を使用するか否かに関しては，利用者が判断することとなるため全ての利用者に導入を徹底することは困難である．一方，ワンタイムパスワードによる決済時の認証強化は，MITB 攻撃により，利用者を騙してワンタイムパスワードの入力を促し，リアルタイムに送金まで完了させる攻撃の存在が確認されている．このため，MITB 攻撃を検知し，利用者またはシステムの運用者に金融系マルウェアへの感染を通知する必要がある．そこで，専用ソフト等を使用しない利用者も含む全ての利用者に適用可能な MITB 攻撃の対策手法が必要と考える．

MITB 攻撃により，コンテンツが改ざんされた際，改ざんされたコンテンツが外部サーバと通信を行うことで，不正送金を成立させることが分かっている．このコンテンツ改ざんによって発生する外部サーバとの通信を捕捉することで MITB 攻撃による改ざんを検知することが可能となる．また，改ざんによる通信の検知を専用のクライアントソフトを用いることなく Web ブラウザ上で行う方法について検討した．

専用のクライアントソフトを用いずに MITB 攻撃による改ざんを検知する製品として，PhishWall クライアントレス [4] がある．PhishWall クライアントレスは，Web ブラウザに読み込まれたコンテンツを同時に読み込ませた検査ロジックによりチェックすることで改ざんの兆候を判断し，利用者に通知する．本稿では PhishWall クライアントレスの検査ロジックを対象コンテンツに含めて読み込ませる手法を利用して改ざんによる通信の検知を行うシステムを提案する．また，提案システムの用いる検知ロジックの実現性の検証実験を行った．

本稿の構成は，以下の通りである．まず，2 章で関連研究について記述する．3 章で MITB 攻撃について記述する．4 章で改ざん検知の提案手法について記述する．5 章で検証実験の方法について記述する．6 章で検証実験の結果について記述する．7 章で実験結果の考察を行った結果を記述する．最後に 8 章でまとめと今後の課題について記述する．

2. 関連研究

インターネットバンキングのセキュリティに関しては，井澤らの研究 [5] や中村らの研究 [6] から金融業界において MITB 攻撃の対策研究が要望されている．

MITB 攻撃の対策手法の研究には，土屋らの提案する認証方法 [7] がある．[7] は，金融系マルウェア感染下でもセキュアにインターネットバンキング等の通信を行うことを可能とするものである．[7] では，金融系マルウェアの感染自体を検知することは行わない．また，対象とする MITB 攻撃がコンテンツの改ざんを伴わないものである．このため本研究とは目的が異なっている．

MITB 攻撃によるコンテンツ改ざんに着目した研究としては，Andrea らの Prometheus[8] や瀬川らの動的解析手法 [9] がある．Prometheus では，動的解析により金融系マルウェアによる改ざんの特徴を収集し，検知に用いることを目的としている．[9] では，MITB 攻撃の攻撃手法を調査することを目的としている．いずれも，MITB 攻撃の実態把握には有益である．しかし，本研究では特定の攻撃手法を検知するのではなく，改ざんコンテンツの通信に着目することで攻撃手法によらず様々な MITB 攻撃を検知可能とすることを目的としている．

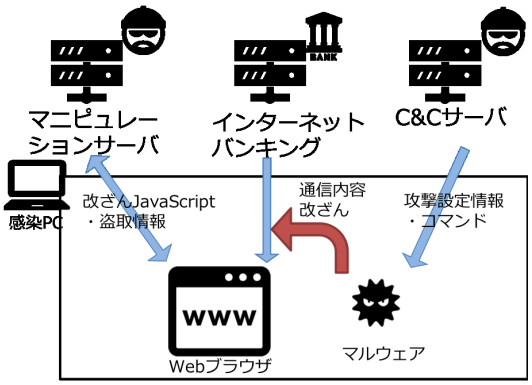

図 1　MITB 攻撃発生時の概要

3. MITB 攻撃

MITB 攻撃について述べる．MITB 攻撃は，金融系マルウェアにより感染 PC の Web ブラウザに対しメモリインジェクション等の方法で入り込み，通信内容の監視・改ざん等を行う攻撃方法である．参考文献 [10] によると MITB 攻撃には，認証情報の盗取を目的とした ID 盗取型 MITB 攻撃と利用者が実行した送金処理の内容をリアルタイムで改ざんする取引内容改ざん型 MITB 攻撃の 2 種類が存在する．本研究では，ID 盗取型 MITB 攻撃を対象とする．本稿における MITB 攻撃とは，ID 盗取型 MITB 攻撃を指す．

MITB 攻撃は，対象コンテンツを改ざんし認証情報を盗取する際に外部サーバと連携することが西田らの行った調査 [11] によって明らかにされている．MITB 攻撃の発生状況の概要を図 **1** に示す．金融系マルウェアは，コマンドアンドコントロールサーバ（以下，C&C サーバ）との通信により攻撃設定情報を受信する．攻撃設定情報には，攻撃対象および改ざん方法が設定されている．金融系マルウェアは，攻撃設定情報に従い Web ブラウザの通信を監視する．Web ブラウザが攻撃対象と通信を行った際に，攻撃設定情報に従い通信内容を改ざんする．その際，多くの MITB 攻撃では，偽画面等の表示を行う不正な JavaScript をダウンロードするためのコード片をコンテンツ内に挿入する．これにより，攻撃者のマニピュレーションサーバから不正な JavaScript をダウンロードする．不正な JavaScript はマニピュレーションサーバと連携して，偽画面の表示や盗取情報のアップロードが行われる．

Vol.2018-CSEC-81 No.19
Vol.2018-IOT-41 No.19
2018/5/18

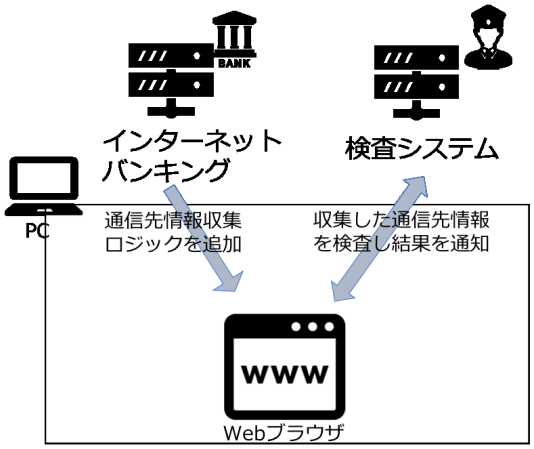

図 2　検知システムの概要

4. 提案手法

提案手法について述べる．MITB 攻撃による改ざんでは，図 1 に示すようにマニピュレーションサーバとの通信が発生する．この通信は，不正な JavaScript のダウンロードおよび不正な JavaScript がマニピュレーションサーバと連携する際に発生する．通常，これらの通信は，“XMLHttpRequest” を用いて行われると考えられる．そこで，保護対象とするコンテンツ（以下，保護対象コンテンツ）における “XMLHttpRequest” の利用状況を監視することで MITB 攻撃による改ざんで発生する通信の検知を行う．また，我々の提案手法では，専用ソフト等を導入することなく検知を可能とすることを目的とする．そこで，改ざん検知システムは，PhishWall クライアントレスのフレームワークを用いる．PhishWall クライアントレスは，保護対象コンテンツに検査スクリプトを含めて読み込ませることで検査を実施する．検査結果の判定は，検査スクリプトから検査サーバに対して判定に必要なデータをアップロードすることで実施する．このフレームワークを利用し，検知を行う．

提案システムのイメージを**図 2** に示す．検査スクリプトでは，保護対象コンテンツが読み込まれる際に予め “XMLHttpRequest” をオーバーライドして呼び出し履歴を収集し，検査サーバにアップロードして検査を実施する．保護コンテンツ内で利用された “XMLHttpRequest” の履歴に MITB 攻撃による改ざんで発生した通信が含まれるかを確認することで検知を可能とする．本稿では，この提案システムで用いる検知ロジックが実現可能であるか検証実験を実施した．

5. 検証実験

提案システムにおける検知ロジックの実現性に関する検証実験について述べる．

5.1 実験方法

実験方法について述べる．検証実験の方法は，金融系マルウェアに感染した PC（仮想マシン）と非感染 PC の 2 種類の環境から同一の Web ブラウザで，金融系マルウェアの攻撃対象サイトに接続した際の “XMLHttpRequest” を用いた通信先情報を収集して比較を行う．“XMLHttpRequest” を用いた通信の監視は，自作の Chrome Extension を用いて “XMLHttpRequest.open” を予めオーバーライドすることにより実現する．これは，JavaScript のみで “XMLHttpRequest” の監視を行うことが可能かを確認するためである．自作の Chrome Extension の詳細は，5.2 に記述する．実験に用いたマルウェアを**表 1** に示す．表中の攻撃パターンは，我々の独自調査によるものである．攻撃パターンは，攻撃設定情報の違いによる分類である．検知対象は，攻撃パターン毎に 1 検体ずつ使用する．ただし，攻撃パターン 3 は，2015 年頃に用いられた比較的古いものであり，C&C サーバおよびマニピュレーションサーバが有効なマルウェアがなかったため擬似感染再現環境を用いる．擬似感染再現環境の詳細は，5.2 に記述する．

金融系マルウェアを用いる場合の実験手順を以下に示す．

(1) 検知対象 1，2 の攻撃設定情報を調査し，攻撃手法毎に分類する
(2) 攻撃手法毎に攻撃対象を 1 つ選定し，感染 PC の Web ブラウザで攻撃対象サイトに接続する
(3) 非感染 PC の Web ブラウザで攻撃対象サイトに接続する
(4) 感染 PC および非感染 PC で収集した XMLHttpRequest の通信先情報を比較する

攻撃手法は，MITB 攻撃発生時に挿入されるコード片およびマニピュレーションサーバからダウンロードされる不正な JavaScript の共通性で分類を行った．

擬似感染再現環境を用いる場合の実験手順を以下に示す．

(1) 擬似感染再現環境を感染状態に設定する
(2) Web ブラウザで擬似感染再現環境に接続する
(3) 擬似感染再現環境を非感染状態に設定する
(4) Web ブラウザで擬似感染再現環境に接続する
(5) 感染状態，非感染状態で収集した XMLHttpRequest の通信先情報を比較する

金融系マルウェアを用いた実験手順を手順 A とする．また，擬似感染環境を用いた実験手順を手順 B とする．

5.2 実験環境

実験環境について述べる．実験環境を**表 2** に示す．表 2 の環境を用いて，5.1 に示した各実験手順を実施する．

擬似感染再現環境の概要を**図 3** に示す．擬似感染再現環境は，下記の要素で構成される．

ダミーコンテンツ

攻撃対象サイトのコンテンツ情報を実際の Web サイ

表 1　検知対象

検知対象名	HASH 値（SHA-1）	備考
検知対象 1	c5d19f5ef423c8cb7067f91e5d3f03447aac45bc	Ursnif（DreamBot），攻撃パターン 1
検知対象 2	219e82f8222298b8e6f97cfe99d6fe1b4a419576	Ursnif（DreamBot），攻撃パターン 2
検知対象 3	なし（擬似感染再現環境を使用）	攻撃パターン 3

表 2　実験環境

仮想環境	VMware Fusion 8.5.10
ホスト OS	macOS High Sierra
仮想マシン	Windows7 Professional 32bit
Web ブラウザ	Chrome 63.03239.132

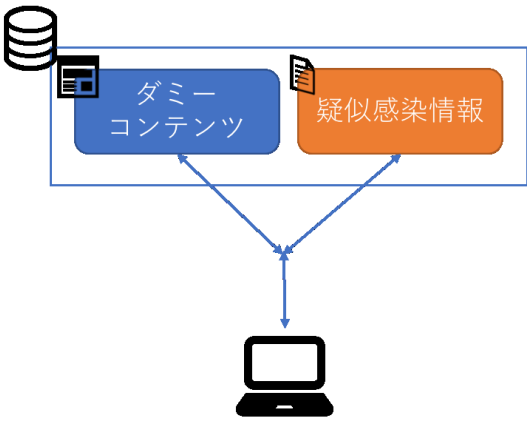

図 3　擬似感染再現環境の概要

トから収集して構築したダミーコンテンツ

擬似感染情報

金融系マルウェアを解析することにより収集した攻撃設定情報およびマニピュレーションサーバの応答情報を用いて構築した擬似感染情報

本来，MITB 攻撃は Web ブラウザ内で攻撃対象のコンテンツに改ざんを行う手法である．擬似感染再現環境では，サーバ側で Web ブラウザが取得するダミーコンテンツに対し，擬似感染情報に基づいた改ざんを予め行う．また，改ざんによって発生する通信にマニピュレーションサーバに代わってダミー情報で応答することによって金融系マルウェアを用いずに MITB 攻撃の再現を行うことを可能とするシステムである．また，擬似感染情報による改ざんやダミー情報による応答を停止することで非感染環境を再現することも可能である．

“XMLHttpRequest” を利用した通信先情報を収集するための Chrome Extension は，“XMLHttpRequest.open” を予めオーバーライドする機能を実装したものである．“XMLHttpRequest.open” をオーバーライドし，通信先情報を記録する実装を図 4 に示す．

6. 実験結果

検証実験の結果について述べる．

```
function initHookXHRopen() {
    var oldOpen = XMLHttpRequest.prototype.open;
    XMLHttpRequest.prototype.open = function (method, url, async, user, pass) {
        console.log("==start==")
        console.log("url :"+url+"\n method: "+method);
        console.log("==end==")
        oldOpen.apply(this,arguments);
    };
}
```

図 4　XMLHttpRequest のオーバーライド実装

表 3　検知対象 1 の攻撃設定情報サマリー

	攻撃対象概要	URL 数
攻撃手法 1	インターネットバンキング 仮想通貨取引所	51
攻撃手法 2	インターネットバンキング EC サイト フリーメール	8
攻撃手法 3	カード会社	12

```
/imgc/?c=script&r=bpw0010-pers&b=b578ffe33e0383183d78776ae9257e56
/imgc/?c=gate&d=%7B%22message%22%3A%7B%22type%22%3A%22warning ...(後略)
/imgc/?c=gate&d=%7B%22message%22%3A%7B%22type%22%3A%22note%... (後略)
```

図 5　感染 PC から銀行 A 接続時の XMLHttpRequest 通信先情報

6.1 検知対象 1 の攻撃手法分類

検知対象 1 のマルウェアから攻撃設定情報を抽出した結果を表 3 に示す．検知対象 1 のマルウェアでは，攻撃設定情報に 3 種類の攻撃手法が含まれていた．各攻撃手法から 1 つずつ攻撃対象を選定し，手順 A で検証実験を行った．

6.2 手順 A-検知対象 1-攻撃手法 1 の結果

手順 A-検知対象 1-攻撃手法 1 では，銀行 A のインターネットバンキングログイン画面に接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC では，“XMLHttpRequest” を用いた通信が発生しており，非感染 PC では，“XMLHttpRequest” を用いた通信は発生していなかった．感染 PC から接続した際の通信先情報を図 5 に示す．この結果から，感染 PC では MITB 攻撃の改ざんによって “XMLHttpRequest” を用いた通信が発生することが分かる．また，この時の通信先ドメインは，攻撃対象のドメインと同一であった．これは，Web ブラウザの Same Origin Policy （以下，SOP）のために “XMLHttpRequest” を用いた通信は，同一ドメインとしか通信が行うことができない．SOP を回避するために攻撃対象のドメインと通信を行う様に改ざんしている．その後，改ざんされた箇所が “XMLHttpRequest” を用いて通信を行うとマルウェアが通信先に含まれる特徴文字列を検知し，通信先ドメインをマニピュレーションサーバに変

表 4　検知対象 2 の攻撃設定情報サマリー

	攻撃対象概要	URL 数
攻撃手法 1	EC サイト	1
攻撃手法 2	インターネットバンキング	7

更することで，別ドメインとの通信を可能としている．このことから，銀行 A に対する攻撃では，感染 PC において改ざんによって攻撃対象ドメインに対する不正な通信を確認することができた．

6.3　手順 A-検知対象 1-攻撃手法 2 の結果

手順 A-検知対象 1-攻撃手法 2 では，EC サイト A に接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC および非感染 PC の双方で，“XMLHttpRequest” を用いた通信を確認した．通信先情報を比較した結果，感染 PC においてのみ発生する攻撃対象ドメインへの通信を確認することができた．この通信先情報には，通信先をマニピュレーションサーバに変更するための特徴文字列が含まれていることを確認した．このことから，EC サイト A に対する攻撃においても，感染 PC において改ざんによって攻撃対象ドメインに対する不正な通信を確認することができた．

6.4　手順 A-検知対象 1-攻撃手法 3 の結果

手順 A-検知対象 1-攻撃手法 3 では，カード会社 A の Web サービスに接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC および非感染 PC の双方で，“XMLHttpRequest” を用いた通信を確認した．また，通信先情報は同一であった．このことから，カード会社 A に対する攻撃では，改ざんによる不正な通信は確認されなかった．

6.5　検知対象 2 の攻撃手法分類

検知対象 2 のマルウェアから攻撃設定情報を抽出した結果を**表 4** に示す．検知対象 2 のマルウェアでは，攻撃設定情報に 2 種類の攻撃手法が含まれていた．各攻撃手法から 1 つずつ攻撃対象を選定し，手順 A で検証実験を行った．

6.6　手順 A-検知対象 2-攻撃手法 1 の結果

手順 A-検知対象 2-攻撃手法 1 では，EC サイト A に接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC および非感染 PC の双方で，“XMLHttpRequest” を用いた通信を確認した．また，通信先情報は同一であった．このことから，EC サイト A に対する攻撃では，改ざんによる不正な通信は確認されなかった．

6.7　手順 A-検知対象 2-攻撃手法 2 の結果

手順 A-検知対象 2-攻撃手法 2 では，銀行 B のインターネットバンキングログイン画面に接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC では，“XMLHttpRequest” を用いた通信が発生しており，非感染 PC では，“XMLHttpRequest” を用いた通信は発生していなかった．この結果から，感染 PC では MITB 攻撃の改ざんによって “XMLHttpRequest” を用いた通信が発生することが分かる．また，この時の通信先ドメインは，攻撃対象のドメインと同一であり，通信先ドメインをマニピュレーションサーバに変更する特徴文字列を含んでいることが分かった．このことから，銀行 B に対する攻撃では，感染 PC において改ざんによって攻撃対象ドメインに対する不正な通信を確認することができた．

6.8　手順 B-検知対象 3 の結果

検知対象 3 は，銀行のみを攻撃対象とする 1 種類の攻撃手法を含む Ursnif の攻撃設定情報から作成した銀行 C のインターネットバンキングログイン画面の擬似感染環境を用いた手順 B で検証実験を行った．

銀行 C のインターネットバンキングログイン画面に接続した際の “XMLHttpRequest” を用いた通信先情報の比較を行った．その結果，感染 PC および非感染 PC の双方で，“XMLHttpRequest” を用いた通信を確認した．通信先情報を比較した結果，感染 PC においてのみ発生する通信を確認した．この通信先は，マニピュレーションサーバのドメインに対する通信であった．この結果から，検知対象 3 では，検知対象 1，2 における検知結果とは異なり，マニピュレーションサーバドメインに対する不正な通信を確認することができた．

7.　考察

検証実験の結果について考察する．各検証実験の結果を**表 5** に示す．この結果から，6 つの実験のうち 4 つで検知が可能であった．検知可能な項目のうち 6.2，6.3，6.7 では，SOP を回避するために攻撃対象ドメインに対して，通常は発生しない “XMLHttpRequest” を用いた通信が発生していた．攻撃対象の正規ドメインに対して通常は発生しない通信が確認されることは正常な環境では起こり得ないと考えられる．このため，6.2，6.3，6.7 の様に攻撃対象ドメインに通信が発生する場合は，MITB 攻撃による改ざんが発生していると見なすことが出来る．これに対して 6.8 では，マニピュレーションサーバのドメインと “XMLHttpRequest” を用いた通信が発生している．この場合は，Ajax+JSONP 等の SOP を回避するための正規の手法を用いていると考えられる．このため，マルウェアではなくブラウザプラグイン等の通信でも同様に検知する可能性がある．よって，6.8 の様な通信が発生する場合は，そ

表 5　各検証実験の結果

節	実験	攻撃対象	結果
6.2	手順 A-検知対象 1-攻撃手法 1	銀行 A	検知可
6.3	手順 A-検知対象 1-攻撃手法 2	EC サイト A	検知可
6.4	手順 A-検知対象 1-攻撃手法 3	カード会社 A	検知不可
6.6	手順 A-検知対象 2-攻撃手法 1	EC サイト A	検知不可
6.7	手順 A-検知対象 2-攻撃手法 2	銀行 B	検知可
6.8	手順 B-検知対象 3	銀行 C	検知可

れのみで MITB 攻撃による改ざんが発生していると判断することが難しい．この様な通信を検知した場合は，感染の可能性を示唆するリスクとして利用することや，マニピュレーションサーバドメインの BlackList やドメイン名および通信パラメータ等の特徴等による悪性判定など，別の判断指標と組み合わせる必要がある．

検知が行えなかった 6.4，6.6 では，Script タグをコンテンツ内に挿入する等の方法で，“XMLHttpRequest” を用いることなくマニピュレーションサーバと通信を行っていた．このため，6.4，6.6 の様な場合は，別の検知手法を用いる必要がある．例えば，タグを追加する際に用いられる “ApendChild” 等を予め改ざんすることで，MITB 攻撃による外部と通信するための Script タグの追加を検知する等の方法が考えられる．

また，提案手法に対して，攻撃者は感染端末内のマルウェアを利用することで，以下のような検知回避を行うことが考えられる．

(1) 検査スクリプトより前に “XMLHttpRequest” のバックアップや改ざんを行う
(2) 検査スクリプトの通信の改ざんまたは妨害

これらの対策として，(1) に対しては，検査スクリプトより前に不正スクリプトの読み込みが行われていないかの検査手法の検討が必要である．また，(2) に対しては，通信内容の秘匿による改ざんの防止および検査スクリプトによる検査が正しく行われたかの確認について検討が必要である．

8. まとめと今後の課題

まとめと今後の課題について述べる．本稿では，MITB 攻撃による改ざんで発生する通信を Web ブラウザ上で検知する手法について提案を行った．また，提案手法で用いる検知ロジックの実現性について検証実験を行った．検証実験の結果から既存の金融系マルウェアの MITB 攻撃による改ざんで発生する “XMLHttpRequest” を用いた通信を JavaScript のみで検知可能な場合があることを確認した．また，通信先が攻撃対象ドメイン自体である場合は，通信が確認された時点で MITB 攻撃による改ざんが発生していると判断出来ると考えられる．この状態に当てはまる場合は，マルウェアの攻撃設定情報等を調査することなく容易に検知が可能になると考えられる．一方で，通信による検知のみでは MITB 攻撃と断定が困難な場合や通信自体を検知することができない場合が存在していた．

今後の課題として，本稿では提案手法の概要報告と検知ロジックの検証のみに留まっているため，提案システム全体の有効性を検証する．また，本稿の提案手法で検知することが困難な場合に関しては，タグを追加する際に用いられる “ApendChild” 等を予め改ざんすることによる検知等の補完手法について検討および追加検証を実施する．更に攻撃者による検知回避に対する対策を継続して検討する必要があると考える．

参考文献

[1] 情報処理推進機構：情報セキュリティ 10 大脅威 2017，（オンライン），入手先 ⟨https://www.ipa.go.jp/security/vuln/10threats2017.html⟩（参照 2017-09-04）．
[2] ネットムーブ株式会社：SaAT Netizen, (online), available from ⟨https://www.saat.jp/netizen/⟩ (accessed 2018-02-05).
[3] 日本アイ・ビー・エム株式会社：Trusteer Rapport, (online), available from ⟨https://www-01.ibm.com/software/jp/info/trusteer/⟩ (accessed 2018-04-07).
[4] 株式会社セキュアブレイン：PhishWall プレミアム，（オンライン），入手先 ⟨http://www.securebrain.co.jp/products/phishwall/index.html⟩（参照 2018-02-05）．
[5] 井澤秀益：金融業界において注目されている情報セキュリティ上の研究課題について，コンピュータセキュリティシンポジウム 2015 論文集，No. 3, pp. 336–339 (2015).
[6] 中村啓佑，宇根正志：金融業界において注目されている情報セキュリティ上の研究課題：認証技術に焦点を当てて，技術報告 15 (2016).
[7] 土屋貴史，神農泰圭，藤田真浩，高橋健太，尾形わかは，西垣正勝：Man In The Browser 攻撃対策を実現する人間・銀行サーバ間のセキュア通信プロトコル（その 2），技術報告 6 (2017).
[8] Continella, A., Carminati, M., Polino, M., Lanzi, A., Zanero, S. and Maggi, F.: Prometheus: Analyzing WebInject-based information stealers, *Journal of Computer Security*, Vol. 25, No. 2, pp. 117–137 (2017).
[9] 瀬川達也，神薗雅紀，星澤裕二，吉岡克成，松本　勉：Man-in-the-Browser 攻撃を行うマルウェアの安全な動的解析手法，技術報告 8 (2013).
[10] 鈴木雅貴，中山靖司，古原和邦：インターネット・バンキングに対する Man-in-the Browser 攻撃への対策「取引認証」の安全性評価，Vol. 32, No. 3, pp. 51–76 (2013).
[11] 西田雅太，太刀川剛，岩本一樹，遠藤　基，奥村吉生，星澤裕二：静的解析と挙動観測による金融系マルウェアの攻撃手法の調査，コンピュータセキュリティシンポジウム 2014 論文集，Vol. 2014, No. 2, pp. 859–866 (2014).