IE8 以降の「F12 開発者ツール」を使った DOM Based XSS の検査法について

NTT コミュニケーションズ株式会社 ソリューションサービス部 第四エンジニアリング部門 セキュリティオペレーション担当

2011年09月05日

Ver. 1.0

NTT Communications



1.	調査	概要	3
2.	F12	開発者ツール	3
2.	.1.	F12 開発者ツール	3
3.	F12	開発者ツールを使って DOM BASED XSSを確認する	5
3.	.1.	サンプルHTML	5
3.	.2.	HTMLロード直後のスクリプト	6
3.	.3.	ユーザイベント駆動型のスクリプト1	2
4.	まと	ø1	9
5.	検証	作業者1	9
6.	参考		9
7.	履歴		9
8.	最新	版の公開URL	0
9.	本レ	ポートに関する問合せ先20	0



1. 調査概要

Microsoft Internet Explorer(以下 IE)の version8 以降には、Web サイトの開発者向けに「開発者ツール」または「F12 開発者ツール」という機能が同梱されている。本文書では、この機能を使い、DOM Based XSS を検査する初歩的方法について、記述する。

2. F12 開発者ツール

2.1. F12 開発者ツール

「F12 開発ツール」とは、IEに対して「F12」を押す事で起動するデバグ・ツールである(図 2.1-1~ 図 2.1-3)。

このような開発ツールによって Web アプリケーションのクライアント側スクリプトや Web ページのデ ザインのデバッグ環境は、大幅に改善されるだろう。

さて、DOM Based XSS は、Web ブラウザ上の DOM に対しての XSS(Cross-Site Scripting) 攻撃である。一般的な Web アプリケーションの診断手順とは、異なる箇所で発現するため、DOM Based XSS の診断作業には、このような Web ブラウザ上のデバグ・ツールは欠かせない。

InPrivate ブラウズ(D) 最終閲覧セッションを再度開く(S)	Ctrl+Shift+P
InPrivate フィルター InPrivate フィルター設定(S)	Ctrl+Shift+F
ポップアップ ブロック(P) SmartScreen フィルター機能(T) アドオンの管理(A)	
互換表示(<u>V</u>) 互換表示設定(B)	
このフィードの購読(E) フィード探索(E) Windows Update(<u>U</u>)	
開発者ツール(L)	F12
Windows Messenger 接続の問題の診断…	
インターネット オブション(0)	

図 2.1-1: 「F12」キー、または「ツール」→「開発者ツール」で起動できる



ファイル(E) 検索(M) 無効化(G) 表示(M) アウトライン(Q) イメージ(D) キャッシュ(D) ツール(E) 検証(A) ブラウザー モード: IE8(B) ドキュメント モード: Quirks(M) HTML CSS スクリフト ママイラー HTML の検索 ママイラー HTML の検索 ママイラー HTML の検索 ママイラー HTML の検索 ママイラー アクトライン(P) 原性 ママイク ママイク ママイク ママイク ママイク ママイク ママイク オーロンの ママイク ママイク ママイク オーロンの ママイク ママイク ママイク オーロンの ママイク ママイク ママイク オーロンの ママイク オーレーススタイル レイアウト 原 マクロースの オーレーススタイル ママクロースの マーレース マクロースの マーレース マクロースの マーレース マ	🕼 DOM Based XSS Test - 開発者ツール	
ブラウザーモード: IE8(B) ドキュメントモード: Quirks(M) ■ HTML CSS スカリフト プロファイラー HTML の検索 R R P R P P P P P P P P P P P P P P P P	ファイル(E) 検索(N) 無効化(S) 表示(V) アウトライン(Q) イメージ(D) キャッシュ(C) ツール(T) 検証(A)	
HTML CSS スカリプト プロファイラー HTML の検索 スタイル トレース スタイル レイアウト 属性 C-Socipt language="JavaScript" type="tex - ズ字列 - Hello - <hr/> - E <6a href="test.html"> - 文字列 - 空のテキスト / E <form action="test.html" m<br="" name="myFrm">- - <hr width="center"/> E - <script javascript"="" language="JavaScript" lmntag"="" type="tex:</p> - 次字列 - Hello -<hr/>-id="></script> E- -文字列 - 空のテキスト / E-<form action="test.html" m<="" name="myFrm" p="">id="abcTag"/> -<hr width="center"/> E-id="xyzTag"> -<script language="JavaScript" type="tex-</p></form></form>	□ ● □ ● □ □ □ □ □ □ □ □ □ □ □ □ □ □	
	□- <body> -<script language="JavaScript" type="tex:
</body>	

図 2.1-2: IE8の「F12 開発者ツール」。

「CTRL」+「P」キーで、図 2.1-3のような画面になる

	🖉 DOM Based XSS Test - Windows Internet Explorer		
ファイル(全) 編集(全) 表示(公) はまにしり(金) ツール(ク) ヘルブ(金)	😋 💿 🗢 😰 http://www.example.com/test.html		💌 🗟 😽 🗙
	ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)		
Reload send Display Referer 2011/9/1 14:32:36 アイル(D) 検索(M) 無効化(S) 表示(M) アウトライン(M) イメージの キャッシュ(G) ツール(D) 検証(M) フラウサーモード IE8(G) ドキュメントモード Ourks(M) = ロ × HTML CSS スカリフト プロフィブラー Li SS スカリフト プロフィブラー MTML CSS スカリフト プロフィブラー Li SS スカリフト プロフィブラー MTML CSS スカリフト プロフィブラー Li SA ログロ ア C Script Language="JavaScript" type="text/javascript" cype="text/javascript" cype	🍲 お気に入り 🏾 🏉 DOM Based XSS Test		ツール(<u>0</u>) ∗ @ ∙ »
Display Referer 2011/9/1 14:32:36 ファイルビ 検索(W) 無効比(S) 表示(W) アウトライン(Q) イメージ(P) 本ャッシュ(Q) ツール(D) 検証(A) ブラヴサーモード IE8(B) ドキュメントモード Quirks(M) = ロ × HTML CSS スクリプト プロファイラー マ ロ マ ロ マ ロ マ ロ マ ロ マ ロ マ ロ マ ロ マ ロ マ ロ	Reload		*
Image: Send Display Referer 2011/9/1 14:32:36 ファイル(E) 検索(W) 無効化(S) 表示(W) アウトライン(W) イメージ(P) キャッシュ(W) ツール(D) 検証(A) ブラガザーモード: IE8(B) ドキュメントモード: Ouirks(M) = 白 × HTML 05S スカリフト ブロファイラー HTML 05S スカリフト ブロファイラー HTML 05(素) ブラアイラー Image: Type="text/javascr: スタイル トレース スタイル レイアウト Image: Type="text/javascr: スタイル マキア - 空のデキスト/ Image: Type="text/javascr: マキャッション Image: Type="text/javascr: マキャッシ Image: Type="text/javascr: マキャッシ Image: Type="text/javascr: マキャッシ Image: Type="text/javascr: マキャッシ Image: Type="text/javascr: Image: Type=Type="text/ja			
Display Heterer 2011/9/1 14:32:36 ファイル(E) 検索(W) 無効比(S) 表示(V) アウトライン(Q) イメージ(P) 検証(A) ブラウザーモード IE8(B) ドキュメントモード: Ourks(M) = ロ × HTML OSS 2007ト プロファイラー マ このまままままままままままままままままままままままままままままままままままま	s	send	
2011/9/1 14:32:36 アrイル④ 検索(Ψ) 無効化⑤ 表示(Ψ) アウトライン(Ψ) イメージΦ キャッシュ(Δ) ツール(T) 検証(Φ) ブラウザーモード IE8(®) ドキュメントモード Quirks(Ψ) = □ × HTML CSS 2/0/7ト プロファイラー HTML の検索 スタイル トレース スタイル レイアウト 属性 Control Contr	Display Referer		
2011/9/1 1432:36 7r1ルE) 検索(型) 無効化(S) 表示(型) アウトライン(型) イメージ(型) キャッシュ(型) ツール(型) 検証(型) 75ウザーモード: IE8(B) ドキュメントモード: Ouirks(M) = □ × HTML OSS スクリフト プロファイラー マスタイル トレーススタイル レイアウト 属性 日 <hrad 日 <hrad 日</hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad </hrad 			
2011/9/1 14:32:36 277(ル④ 検索(Ψ) 無効(L⑤) 表示(Ψ) アウトライン(Ψ) イメージ(Ψ) キャッシュ(ῶ) ツール(Ψ) 検証(Δ) 7ラウザーモード IE8(Ψ) ドキュメントモード Quirks(Ψ) = □ × HTML OSS 2097ト プロファイラー N TML O(½索 ////////////////////////////////////	0014 /0 /4 4400 00		
ファイル(E) 検索(W) 無効化(G) 表示(W) アウトライン(Q) イメージ(Q) キャッシュ(Q) ツール(D) 検証(A) ブラウザーモード: IE8(B) ドキュメントモード: Quirks(M) = □ × HTML CSS スカリフト フロファイラー HTML の検索 W (C) アウトライン(D) アウトライン(D) イメージ(Q) キャッシュ(Q) ツール(D) 検証(A) HTML CSS スカリフト フロファイラー HTML の検索 Image: State of the add of the	2011/9/1 14:32:36		-
ブラウザーモード: IE8(B) ドキュメントモード: Quirks(M) = □ × HTML CSS スカリブト プロファイラー HTML CSS スカリブト プロファイラー マーク マーク マーク マーク i="lanflag"/> マー	' ファイル(F) 検索(N) 無効化(S) 表示(V) アウトライン(O) イメージの キャッジ	シュ(C) ツール(T) 検証(A)	
HTML CSS ZØJJ ト JU77-J- HTML CSS ZØJJ ト JU77-J- Image: Strategy of the strategy			
HIML CSS 2/9/3ト プロファイラー CSS 2/9/3ト プロファイラー CALL ONSA CALL ON			
スタイル トレース スタイル レイアウト 腐性 スタイル トレース スタイル レイアウト 腐性 マhtml> マhtml> マhtml> マhtml> マオン レイアウト 腐性 マオン レース スタイル レイアウト 腐性 マオン レイアウト 肉 マオン レイアウト 肉 マオント ロー ロー マオン レイアウト 肉 マオン マオン レイアウト 肉 マオン ロン マオン マオン ロン ロン ロン マオン ロン		IT ME ORA	<u> </u>
<pre>C Cheal> C < Ch</pre>		スタイル トレース スタイル レイアウト 属性	
C <body> <script language="JavaScript" test.html"="" type="text/javascr: 文字列 - Hello <hr/> <pre>-<hr/> <pre>-<hr/> <pre>-</pre> </script></body>			

「CTRL」+「P」キーで、図 2.1-2のような画面になる



3. F12 開発者ツールを使って DOM Based XSSを確認する

3.1. サンプルHTML

今回は、非常にシンプルな「http://www.example.com/test.html」(図 3.1-1)を対象にして、 DOM Based XSS の確認していく。

シンプルとはいえ、setTimeout()関数を使い、スクリプトをバックグラウンドで実行している部分も確認できる。

```
<html>
<head><title>DOM Based XSS Test</title></head>
<body>
  <script language="JavaScript" type="text/javascript">
  <!--
   document.write("Hello");
  // -->
  </script>
  <hr>>
  <A HREF="test.html">Reload</A>
  <form name="myFrm" METHOD="get" ACTION="test.html">
  <input type="text" name="myText" size="72"><input type="submit" name="send" value="send"><br>
  <input type="button" value="Display Referer" onClick="test()">
  </form>
  <hr width="center">
  <script language="JavaScript" type="text/javascript">
  var Obj1 = document.getElementById("ImnTag");
  Obj1. innerHTML = document. referrer;
  function test() {
   var Obj = document.getElementById("abcTag");
   Obj.innerHTML = document.referrer;
  }
  function myNow() {
   var myObj = document.getElementById("xyzTag");
   var myDate = new Date();
   var str = "" + myDate.getYear() + "/" + (myDate.getMonth()+1) + "/" + myDate.getDate();
   str += " " + myDate.getHours() + ":" + myDate.getMinutes() + ":" + myDate.getSeconds();
   myObj.innerHTML = str;
  }
  function myTimer() {
   myNow();
   setTimeout('myTimer()', 1000);
  }
  setTimeout('myTimer()', 1000);
  </script>
</body>
</html>
```







<html>
<head><title>Post Page</title>
</head>
</body>
<form action="test.html" method="get">
<input type="text" name="myText" size="72">
<input type="submit" name="send" value="send">
</form>
</body>
</html>

3.1-2: [http://www.example.com/test2.html]

test.html ヘデータを送信する入力画面

3.2. HTMLロード直後のスクリプト

今回のサンプル HTML は、HTML ロード時に実行されるスクリプトで「Referer」の値を画面に書き出していることが確認できる。

この節では、HTML ロード時に実行されるスクリプト中に含まれる DOM Based XSS について考察する。

手順は、以下の通りとなる。

ひとまず該当ページにアクセスし、「F12 開発者ツール」を起動し、コードのなるべく先頭部分にブレークポイントを設定し、「デバグ開始」を押下する(図 3.2-1)。

一つ前のページへ戻り、再度アクセスする(図 3.2-2→図 3.2-3)。

図 3.2-3で、ブレークポイントのところで処理が停止するので、ステップ実行などでコード解析を行い、XSSが発現する可能性のある部分を探し出す(図 3.2-4)。

図 3.2-4には「innerHTML」というDOM Based XSSを誘発させやすいプロパティ名を確認したので、そこにブレークポイントを設定する(図 3.2-5)。

再度、一つ前の画面に戻り、図 3.2-5ではRefererの値がそのままinnerHTMLの値にコピーされているようなので、この一つ前の画面からRefererにXSSコードを仕込んでみる(図 3.2-6)。 つまり、図 3.2-6では、アドレス欄に直接クエリ文字列を与えて「test2.html」を呼び出した後で、

「test2.html」から「test.html」へ遷移する(「send」ボタン)。

その後は、図 3.2-7~図 3.2-8のようにブレークポイントのところで停止する。「F12 開発者ツール」 では実行中のスクリプトが扱っているDOMの状態を細やかに確認することができる。 最終的には、図 3.2-9のように、XSSを発現させることができた。



ODM Based XSS Test - Windows Internet Explorer	
Coo v Dhttp://www.example.com/test.html	💌 🗟 😽 🗙
ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)	
👷 お気に入り 🏉 DOM Based XSS Test	ツール(<u>©</u>)★ @ + ≫
Hello	^
Reload	
send	
Display Referer	
2011/9/1 14:49:19	•
ファイル(E) 検索(N) 無効化(S) 表示(V) アウトライン(Q) イメージ(Q) キャッシュ(Q) ツール(T) 検証(A)	
ブラウザーモード IE8(B) ドキュメントモード Quirks(M) 😑 🗗 🗙	
HTML CSS スクリプト プロファイラー スクリプトの検索	Q
💫 🔜 🔟 II 👞 🖅 🕼 🧉 🥑 デバッグ開始 🗋 testhtml コンソール ブレークポイント ローカル ウォッチ コ	ール スタック
1 Khtml) 2 Khead/Kittle>DOM Based XSS Test//titlデバッグの開始(F5) 名前 値	種類
3 (body)	
4 <script language="JavaScript" type="text/javascript"> 5 <!</td><td></td></tr><tr><td>6 document.write("Hello"); 7 //></td><td></td></tr><tr><td>8 </script>	
	1
	A + 100% +

図 3.2-1:とりあえず該当ページにアクセスし、「F12 開発者ツール」を起動する。

なるべく先頭部分にブレークポイントを設定する

Fost Page - Window	s anternet E	and the second se						
🕞 🕘 🗢 🕼 http	//www.exar	nple.com/test2.h	tml				- 🖻 🖻	4 ×
ファイル(E) 編集(E)	表示⊙	お気に入り(<u>A</u>)	ツール①	ヘルプ(円)				
🍰 お気に入り 🛛 🌈 P	ost Page			📄 👘 🖓 – 🖾 – 🖃	-ジ(₽) *	セーフティ(<u>S</u>)・	ツール(@) • 🌘) •
[(send)				P
				\bigcirc				

914 前の画面に戻り、再度アク



♠>Post Page - 開発者ツール		_ 🗆 🗵
ファイル(E) 検索(U) 無効化(G) 表示(V) アウトライン(Q) イメージロ	・キャッシュ(<u>C</u>) ツール(T) 検証(<u>A</u>)	
ブラウザー モード: <u>E8(B)</u> ドキュメント モード: Quirks(<u>M</u>) 📃		
HTML CSS スクリプト プロファイラー	スクリプトの検索	9
💫 🛒 🔘 II 🜇 📧 🗊 デバッグの停止 🛛 test.ht	r コンソール ブレークポイント ローカル ウォッチ	
<pre>1 <html> 2 <head><title>DOM Based XSS Test</title></head> 3 <body> 4 <script language=" JavaScript" type=" text/javascript"></script></body></html></pre>		

図 3.2-3: 図 3.2-2後の画面。このように、

ロード直後に実行されるスクリプトにもブレークポイントは設定できる

♠ Post Pa	ee - 開発者ツール
ファイル(E)	検索(1) 無効化(2) 表示(2) アウトライン(2) イメージの キャッシュ(2) ツール(3) 検証(3)
ブラウザー・	ード: IE8(B) ドキュバントモード: Quirks(M) 🔲
HTML C	SS スクリプト プロファイラー スクリプトの検索 の
	○ II IN 201 (3) (3) デバッグの停止 testhtr コンソール ブレークボイント ローカル ウォッチ コール スイト
4	<pre><script <="" language=" JavaScript" pre="" type=" text/javascript"></script></pre>



◇ DOM Based XSS Test - 開発者ツール	
ファイル(E) 検索(N) 無効化(G) 表示(V) アウトライン(Q) イメージ(D) キャッシュ(C) ツール(T) 検証(A)	A)
ブラウザー モード:IE8(B) ドキュメント モード: Quirks(M) 📃	
HTML CSS スクリプト プロファイラー スクリプトの検索	٩
Ref 10 11 10 10 10 10 10 10 10 10 10 10 10	ローカル ウォッチ コール ス・・
7 //> A前	値
9 <hr/>	
10 <pre>// id="ImnTag"> 11 Reload </pre>	
12 <form action="test</td><td></td></tr><tr><td>14 <input type=" button"="" method="get" name="myFrm" onclic<="" td="" value="Display Referer"><td></td></form>	
15 16	
17 <hr width="center"/>	
18 <pre></pre>	
20 var Obj1 = document.getElementById("ImnTag");	
21 Dbj1 immer H I ML = document referrer.	
23 function test0{ 24 var 0bi = document getElementBuld("abcTag");	
25 Obj.innerHTML = document.referrer;	
26 } 27	
28 function myNow0{	
 29 var myObj = document.getElementById("xyzTag"); 30 var myDate = new Date0; 	
図 3.2-5:図 3.2-4からのつながりで、ここにブレークポイン	トを設定する
Post Page - Windows Internet Explorer	
	👻 🐼 😽 👂
ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルブ(H)	
🍃 お気に入り 🏾 🏉 Post Page 👘 🔹 👘 🔹 ペー	ジ(P)・ セーフティ(S)・ ツール(Q)・ 🕡・
> <s>PostData(/s></s>	
につびが表示されました	차ット 🛛 🖓 🔹 🔍 100% 🔹
図 3.2-6:図 3.2-5後、再度、一つ前の画面に戻る	0
図 3.2-5では、Refererの値にXSS攻撃コードを仕込めそうなので、	実際に仕込んでみる

(今回は、診断作業として脆弱性の有無を確認できる程度のコード)



∢N> Post Pa	age - 開発者ツール			
ファイル(E)	検索(1) 無効化(5) 表示(2) アウトライン(2) イメージの	キャッシュ(2) ツール(1) 検証(4	ð	
ブラウザー・	モード: 168(8) ドキュメント モード: Quirks(<u>M</u>) 📃			
HTML C	SS スクリプト プロファイラー	スクリプトの検索		Q
	○ II File File Call デバッグの停止 L test btr	コンソール ブレークポイント	ローカル・ウォッチ	
1		2前		
2	<pre><head><title>DOM Based XSS Test</title></head> </pre>		11	
3	<body></body>	Executed Sile	bed.	
4	<pre><script <="" language="JavaScript" pre="" type="text/javascript"></td><td>Events]</td><td></td><td></td></tr><tr><td>5</td><td><!</td><td></td><td>{}</td><td></td></tr><tr><td>6</td><td>document.write("Hello");</td><td>E clipboardData</td><td>{}</td><td></td></tr><tr><td>7</td><td>$// \rightarrow$</td><td></td><td>false</td><td></td></tr><tr><td>8</td><td></script></pre>	constructor	{}	
9	<hr/>	defaultStatus		
10		Co- document	{}	
11	Reload	+ 🥥 [Methods]		
12	<pre><form action="test</pre></td><td>🕂 🥥 [Events]</td><td></td><td></td></tr><tr><td>13</td><td><pre><input type=" method="get" name="myText" size="72" text"=""><inpu< pre=""></inpu<></form></pre>	🚽 🥥 activeElement	null	
14	<pre><input onclic<="" pre="" type="button" value="Display Referer"/></pre>	alinkColor	"#0000ff"	
15		🛨 🧳 all	{}	
16	<pre></pre>	🕂 🥥 anchors	{}	
17	<hr width="center"/>	🕂 🥥 applets	{_}}	
18	<pre></pre>	🚽 🥥 attributes	null	
19	<pre>Kscript language="JavaScript" type="text/javascript"</pre>	🚽 🥥 bgColor	"#ffffff"	1
21	var Obj1 = document.getEiementById("ImnTag");	庄 🥥 body	{}	1
Q 21	Obj1.innerHTML = document.referrer;	🥥 charset	"shift_jis"	
		庄 🧳 childNodes	{}	
23	function test0{	庄 🧳 compatible	{}	
24	var Obj = document.getElementById("abcTag");	🧼 🧼 compatMode	"BackCompat"	-
25		[4]		•
		 ブレークポイントで処理	が一時停止する	
		ノレー ノホイントで処理	л. нд т .т. у Ф	
IN Post Pa	ige - 開発者ツール			
Dad (L(E)	検索(小) 無効(化(2) 表示(小) プロトライン(の) イメージの きゃい	(1) ((1) 地一山(1) 絵歌(4)		
27-17PAL	ーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーー	NIT OF A NOTA INSTATE		
ブラウザー モ	E-ド IE8(B) ドキュメント モード: Quirks(M) 🛛 🛄			
()		フカロプトの絵表		0
HTML C	SS ×997 707715-	A372100128		~





🖉 DOM Base	ed XSS Test -	Window	s Internet Expl	orer										
00-	🖉 http://w	ww.exan	nple.com/test.h	itml?myText	=%22%27%3E	%3Cs%3EF	°ostData%	3C%2Fs	%3E&send=:	end		-	1 😣 😽	×
」ファイル(E)	編集(三)表	₩	お気に入り(<u>A</u>)	ツール①	ヘルプ(円)									
👍 お気に入り	р 🄏 ром	Based >	KSS Test			🟠 •	<u>a</u> * =	-	ページ(P)	• t-:	7ティ(<u>S</u>) ▼	ツール	© • 🕜	, »
Hello														4
						201								-
http://w	ww.example	.com/	test2.html?i	myText="	′> Test	5								
Delegal					-									
Reload														
	a Statistic Concernante					send								
Display	Referer													
														-
0011 (0 (1														
2011/9/1	1 14:07:04													
														1000

図 3.2-9: 図 3.2-7~図 3.2-8の結果。XSSが発現した。



3.3. ユーザイベント駆動型のスクリプト

この節では、ボタンなどの押下処理時の DOM Based XSS に対して考察する。 まずは、HTMLのソースを確認し、ボタン押下など、どのJavaScript関数が呼び出されるかを確認 する(図 3.3-1)。

図 3.3-1から、test()関数がボタン押下によって、呼び出されることから、test()関数にブレークポイントを設定し、「デバグ開始」ボタンを押下する(図 3.3-2)。

そして、HTML上のボタン(「Display Referer」ボタン)を押下すると、test()関数が実行され、指定したブレークポイントで処理が一旦停止する(図 3.3-3)。

さらに、ステップ実行(「F11」キー)すると、図 3.3-4のようになり、DOM Based XSSとして注目すべき「innerHTML」プロパティが登場した。また、その値に「Referer」がセットされることが分かった。 さらには、「ローカル」というプロパティで、DOM の状態を確認することも可能だ。

以上より、Refererの値にXSSを引き起こすような情報を埋め込むことができれば、上記のところで、XSSを発生させることが可能ではないかと推定できる。

よって、今度はXSSを引き起こすような情報をWebブラウザが「Referer」として認識させることが可能かという点であるが、「send」ボタンによって、任意のクエリ文字列を与えることが可能であるため、 実際に(XSSを引き起こすような)文字列をセットし、XSSを誘発してみる(図 3.3-5~図 3.3-12)。

最初におこなったことは、そのままのクエリ文字列を使ってみることである(図 3.3-5~図 3.3-8)。 つまり、テキストボックスにXSS試験用文字列を埋め込み(図 3.3-5)、サーバへリクエストを送りペー ジ書き換えを実施し、WebブラウザにRefererを記憶させる(図 3.3-6~図 3.3-7)。 その上で「Display Referer」を押下してみたが、結果はうまく行かなかった(図 3.3-8)。 クエリ文字列が URL エンコードされてしまうことが原因である。

次は、URLエンコードされないようにアドレス欄に直接入力してみる(図 3.3-9~図 3.3-12)。 まず、アドレス欄に直接クエリ文字列を記述し、ページ書き換えを行う(図 3.3-9)。 図 3.3-9の状態ではアドレス欄のURLは、まだRefererの値とはなっていないので、「send」ボタン を押下し、もう一度ページ書き換えを行い、WebブラウザにRefererとして認識させる(図 3.3-10)。 この状態で、「Display Referer」を押下げると、「F12 開発者ツール」のブレークポイントで処理が 停止する(図 3.3-11)。さらに処理を進めた結果が、図 3.3-12であり、XSSを発現させることに成功し た。





図 3.3-1:「HTML ソース」を確認する。



COM Base	d XSS Test	- Window	s Internet Expl	orer										×
GO -	🔊 http://	/www.exar	nple.com/test.	ntml?myText	=&send=sen	d						-	+	×
ファイル(E)	編集(<u>E</u>)	表示⊙	お気に入り(<u>A</u>)	ッール(①	ヘルプ(円)									
👍 お気に入り	0 🏀 DO	M Based)	KSS Test				6) - 🖃 🖶	 ページ(<u>り</u> ・セ	-7ティ(<u>S</u>)、	• ツール(<u>O</u>) •	? •	»
Hello														4
http://w	ww.examo	le com/	test2.html											
<u>Reload</u>														
						send								
Display	Referer	D												
	-													
2011/9/1	15:39:54													w.
ファイル(E) 桁	_{食索(N)} 無効	助化⑤ ま	長示(⊻) アウト:	ライン(2) イン	人一ジ① キャ	アツシュ(<u>C</u>)	ツール(丁)	検証(A)						
ブラウザー モー	- K: IE8(B)	ドキュメン	ト∓ード: Quirk	s(M) _	d⊒ x									
	スクリプト	70774	(=_]					スクリ	ブトの検索					0
		Ma Cal	F= 7/1/2	グ開始	test.html?m	/Text: 그)	バール	ブレークポイン	/h [n-;	hil d	し ォッチ :	コール スタック		
19 <	script langu	age=" Java	aScript"	*text/javasci	ript">	▲ 名	前		(值					
20	var Obj1 = d Obj1 innerH1	locument.g FML = doci	etElementById umentreferrer	("ImnTag");						-				
22			uniona orientia,											
23 1	tunction tes var Obj = d	t01 ocument.g	etElementById	("abcTag");										
25	Obj.innerHT	ML = docu	ument.referrer;											
20 5	i					-								
21						and the second se								
28														▶

Copyright 2011 NTT Communications Corporation



(№ DOM Based XSS Test - 開発者ツール									
ファイルビ 検索(型) 無効化(の) 表示(型) アウトライン(型) イメージの キャッシュ(型) ツール(型) 検証(血)									
ブラウザー モード、IE8(B) ドキュメント モード、Quirks(団) 📃									
HTML	css スクリプト フロファイラー スクリプトの検索 スクリプトの検索								
	「」 「 「 「 「 」 「 」 「 」 「 」 「 」 「 」 「								
21 22 24 26 26 27 28 29 30 31 33 34 35 33 34 35 36 37 38 39 40 40	<pre>var Obj1 = document.getElementById("ImnTag"); Obj1.innerHTM(<u>2797 (2) (F11)</u>)rer; unction fest0{ var obj = document.getElementById("abcTag"); ObjinnerHTML = document.getElementById("xyzTag"); var myObj = document.getElementById("xyzTag"); var myDate = new Date0; var str = "" + myDate.getYear0 + "/" + (myDate.getMonth str += " " + myDate.getHours0 + ":" + myDate.getMinutes' myObjinnerHTML = str; } function myTimer0{ myNow0; setTimeout(myTimer0',1000); } setTimeout(myTimer0',1000);</pre>								
42									
43	 								

図 3.3-3: 図 3.3-2で設定したブレークポイントで停止する

DO	M Based XSS Test - 開発者ツール			
ファイル	(E) 検索(II) 無効化(G) 表示(V) アウトライン(Q) イメージの キャッジ	ノュ(Q) ツール(T) 検証(A)		
ブラウサ	ーモード: IE8(因) ドキュメントモード: Quirks(<u>M</u>) 📃			
HTML	OSS スクリプト プロファイラー	スクリプトの検索		Q
	〇川 「 「 「 「 」 「 」 デバッグの停止 testhtml?myT	コンソール ブレークポイン	ローカル ウォッチ	コール スタック
2	var Obi1 = document.getElementBvId("ImnTag");	名前	値	種類
2	1 Obj1.innerHTML = document.referrer;			Disr A
2	2	Methods]	L1	Disp -
2	3 function test0{	Events]	-	
0 2	var Obi = document.getElementById("abcTag")	accessKev		Strin
0 2	Obj.innerHTML = document.referrer;	- align		Strin
2	8]	all	{}	Disp
2		ariaActivedescenda.		Strin
2	function myNow0{	🛶 🧳 ariaBusy		Strin
2	var myObj = document.getElementById("xyzTag");	🧳 ariaChecked		Strin
3	var myDate = new Date0;	🛶 🧳 ariaControls		Strin
3	var str = "" + myDate.getYear0 + "/" + (myDate.getMonth	🛶 🧳 ariaDescribedby		Strin
3	2 str += " " + myDate.getHours0 + ":" + myDate.getMinutes	🥏 ariaDisabled		Strin
3	3 myObj.innerHTML = str;	ariaExpanded		Strin
3	4 }	🧼 🧳 ariaFlowto		Strin
3		🛶 🧳 ariaHaspopup		Strin
3	function myTimer0{	🛶 🧳 ariaHidden	""	Strin
3	/ myNow0;	🛶 🧳 ariaInvalid		Strin
3	setTimeout("myTimer0",1000);	🛶 🧳 ariaLabelledby	""	Strin
3	}	🛶 🧳 ariaLevel	0	Num
4		🛶 🧳 ariaLive		Strin
4	setTimeout("myTimer0',1000);	🛶 🧳 ariaMultiselectable		Strin
4	2	🛶 🧳 ariaOwns		Strin
4	≤	🚽 🧳 ariaPosinset	0	Num
4		4) b

図 3.3-4: 図 3.3-3後、さらに進める(ステップ実行は「F11」)すると、

注目の「innerHTML」と外部から書き換え可能なプロパティ「document.referer」が出現した



CDOM Based >	(SS Test - Wind	ows Internet Explo	rer								- 🗆 ×
00 - 1) http://www.e	xample.com/test.ht	ml							- 🖻	* 7 ×
」 ファイル(E) 編	課(E) 表示()) お気に入り(<u>A</u>)	ツール(丁)	ヘルプ(円)							
🖕 お気に入り	🏉 DOM Base	ed XSS Test				• 🗟 • 🖻	•	ページ(<u>P</u>) ▼	セーフティ(<u>S</u>)・	ツール(<u>0</u>) ▼	0• »
Hello											4
11.57932574											-
Reload											
Display Re	terer				sena						
2011/9/1 1	5:44:11										
							_				*
ページが表示されま	した			,			1	ンターネット		∕ ≙ ▼ * 1	00% 👻 //.

図 3.3-5: 最初はテキストボックスに XSS 試験用文字列を与え、

「send」ボタンによって画面書き換えを実施する



図 3.3-5で入力した情報をRefererに記憶させる

Copyright 2011 NTT Communications Corporation



🖉 DOM Based	XSS Test - Windows Internet Explorer	- D ×
	http://www.example.com/test.html?myText=&send=send	🔹 🗟 🍫 🗙
ファイル(E)	編集(E) 表示(V) お気に入り(A) ツール(T) ヘルブ(H)	
🏠 お気に入り	🏉 DOM Based XSS Test 🛛 👘 🔹 📾 🔹 ページ(P)・ セーフティ(S)・	ッール@ ・ @・
11.11.		
Hello		
1.0.77		
http://www	v.example.com/test.html/mylext=%22%2/%3E%3Cs%3EPostData%3C%2Fs%3E&send=send	
<u>Reload</u>		
	cend	
Display R	eferer	
2011/9/1	15:46:3	
ページが表示され	alut	
	図 3.3-7: 図 3 3-6の結果。次は「Display Referen」ボタンで、	, - ,
	Refererを表示してXSS発現のはずだが・・・	
C DOM Base	1 XSS Test - Windows Internet Explorer	- D ×
00-	http://www.example.com/test.html?myText=&send=send	• 🗟 +9 🗙
771N(E)	・ 編集(E) 表示(V) お気(こ入り(A) ツール(D) ヘルブ(E)	
👍 お気に入り	Ø DOM Based XSS Test ⑥・⑤・□ 歳・ページ(P)・セーフティ⑥・	ツール@ • @ • »
		4
Hello		
2000 222		
http://ww	w.example.com/test.html?myText=%22%27%3E%3Cs%3EPostData%3C%2Fs%3E&send=send	
<u>Reload</u>		
	and l	
Display	Referer	
Summered		
http://ww	w.example.com/test.html?myText=%22%27%3E%3Cs%3EPostData%3C%2Fs%3E&send=send	
2011/0/1	15/46/30	
2011/9/1	13.40.50	
	[] [] [] [] [] [] [] [] [10 × (+ 100% × 1
		1
	図 3.3-8: 凶 3.3-70 結末。XSSか 光切していないのはクエリズ子列	ι, n'

URLエンコードされていることが原因だ

Copyright 2011 NTT Communications Corporation



DOM Based XSS Test - Windows Inter	et Explorer
🕒 🕤 🗢 🙋 http://www.example.co	v/test.html?myText="`Xs>PostData
ファイル(E) 編集(E) 表示(V) お気(入り(金) ツール(① ヘルブ(出)
- お気に入り 🏾 🏉 DOM Based XSS Te	t 🔰 👘 + 🗟 + 🖻 🚔 + ページ(P) + セーフティ(S) + ツール(Q) + 🙋 +
لاماله	
Reload	
Reload	
Reload	send
<u>Reload</u> Display Referer	send
<u>Reload</u> Display Referer	send
Display Referer	send
Display Referer	send

その上で「send」ボタンで画面書き換えを行い、現在の URL を Web ブラウザが Referer として認識させる

🖉 DOM Based XSS Test - Windows Internet Explorer	
🚱 🕤 🔻 🙍 http://www.example.com/test.html?myText=&send=send 💽 🔯 😏	×
」ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルブ(H)	
🍲 お気に入り 🍘 DOM Based XSS Test	»
Hello	4
	-
http://www.example.com/test.html?myText= [~] > PostData	
Reload	
send	
Display Referer	
	-
2011/9/1 15:51:11	
	Y
	• //.

図 3.3-10: 図 3.3-9の結果。いよいよ「Display Referer」をクリックする



I DOM (Based XSS Test - 開発者ツール							
ファイル(E)	検索(1) 無効化(3) 表示(2) アウトライン(2) イメージの キャッシ	シュ(Q) ツール(T) 検証(<u>A</u>)						
7505-	モード: E8(個) ドキュメントモード: Clunks(M) 🛄	フレビレの分表						
HTML	CSS スクリプト プロファイラー	スクリノトの1度系	٧					
	🔘 🗉 🐻 🖆 🗊 デバッグの停止 🛛 test.html?myT	コンソール ブレークポイン 🚺	ーカル ウォッチ コール スタック					
21	Obj1.innerHTML = document.reterrer;	名前	値 種類					
22	function text0[😑 🥥 Obj 🔰 🕴	.) Disp 🔺					
24	var Obj = document.getElementById(abcTae ^{(*});	Methods]						
Q 25	Obj.innerHTML = document.referrer;		" Strir					
25		align "	" Strir					
27	timotion multiple	⊕ 🧳 all [.) Disp					
29	var mvObi = document.getElementBvId("xvzTag");	🧳 ariaActivedescenda "	"Strin					
30	var myDate = new Date0;	ariaBusy "	"Strin"					
31	var str = "" + myDate.getYear0 + "/" + (myDate.getMonth	🔗 ariaUhecked	" Strip					
32	str += " " + myDate.getHours0 + ":" + myDate.getMinutes	ariaDescribedhy "	" Strin					
33	myObj.innerHTML = str;	ariaDisabled "	" Strin					
34	1		" Strir					
36	function myTimerQ[🧳 ariaFlowto 🥤	"Strin					
37	mvNow0;		"Strir					
38	setTimeout(myTimer0'1000);	ariaHidden	" Strif					
39	1	arial abelledby	" Strin					
40		ariaLevel 0	Num					
41	setTimeout("myTimer0",1000);	ariaLive "	" Strir					
43	(/script) Z Asedu)		" Strir					
44		— 🧳 ariaOwns 🥤	"Strir					
45		🚽 🧳 ariaPosinset 👘 🛛	Num					
C DOM I	図 3.3-11:図 3.3-10の結果。ブレ Based XSS Test - Windows Internet Explorer	 ークポイントで処理か	「停止した 					
00	+ Ittp://www.example.com/test.html?myText=&send=send		▼ 🗟 49 ×					
	(F) 毎年(F) まこれの わた(-3 約(4) パール(F) ムルプ(ル)							
1 79170			· - · · · · · · · · · · · · · · · · · ·					
🙀 お気に	그건이 🌈 DOM Based XSS Test	<u>}</u> ・	セーフティシマ ツールシマ 🕢・					
			<u> </u>					
http:/	http://www.evample.com/test.html?muTevt="">PostPate							
nttp://www.cxample.com/test.ntminingText= /Post/ata								
Reload	Reload							
send								
Display Referer								
http://www.example.com/test.html?myText=">PostData								

図 3.3-12: 図 3.3-10~図 3.3-11の結果。XSSが発現した

🕢 👘 🔛 🔛 😜 ብンターネット

🔺 🔹 100%



4. まとめ

本文書の例題は、非常にシンプルな DOM Based XSS であったが、より実際的、より現実的で複 雑なものであっても、同様の方法で確認することは可能であろう。

実際のところ、多くの Web アプリケーションは、未だに「WEB2.0」のレベルではなく、AJAX も使われていない Web アプリケーションが多いと思われるが、いざ AJAX をふんだんに駆使した Web アプリケーションに対しても、既存の Web アプリケーション診断の方法に加えて、このようなデバグ機能を駆使することで、より精度の高いセキュリティ診断を行うことができる。

5. 検証作業者

NTT コミュニケーションズ株式会社

ソリューションサービス部 第四エンジニアリング部門 セキュリティオペレーション担当 佐名木 智貴

6. 参考

- 1. Windows Internet Explorer 8 開発者ツールを使用してサイトを修正する http://msdn.microsoft.com/ja-jp/library/cc817576.aspx
- 2. Internet Explorer 8の開発者ツールの概要 http://msdn.microsoft.com/ja-jp/library/cc848894(v=vs.85).aspx
- 3. 使ってみよう! [F12] IE9 開発者ツール HTMLと JavaScript のデバッグ編- monoe's blog

http://blogs.msdn.com/b/osamum/archive/2011/08/04/f12-ie9-html-javascript.aspx

- 4. 使ってみよう! [F12] IE9 開発者ツール さまざまな情報の表示 monoe's blog http://blogs.msdn.com/b/osamum/archive/2011/08/09/f12-ie9.aspx
- F12 開発者ツールで Web ページをデバッグする方法 Web ページのデバッグ Web ページ のトラブルシューティング F12 開発者ツール開発者ツール (Windows) http://msdn.microsoft.com/ja-jp/library/gg589507(v=vs.85).aspx

7. 履歴

● 2011 年 09 月 05 日: ver1.0 最初の公開



8. 最新版の公開URL

 $http://www.ntt.com/icto/security/data/soc.html \# security_report$

9. 本レポートに関する問合せ先

NTT コミュニケーションズ株式会社 ソリューションサービス部 第四エンジニアリング部門 セキュリティオペレーション担当

e-mail: scan@ntt.com

以上