

A journey into reversing Internet Explorer in 2006

Task: after loading some MSDN web page the internet explorer crashes with a stack dump.
Find out the reason why it crashed?

By using **WinDBG** (to attach to the internet explorer process and setting a few breakpoints on some key functions (most common, lowest in the dependencies hierarchies), it is possible to learn a lot about the Windows IE internal just by looking at the stacktrace - which show the sequence of calls made by different windows libraries.

These are recorded in 2006 when I was trying to understand Windows IE:

Just clicking on this URL: generated this log
the end is followed by scrolling the screen down.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconManagedExecution.asp>

```
Opened log file 'c:\ie123.log'
0:014> bm /a kernel32!lstr*A "kv L 40;dda esp+4 l 2;g"
breakpoint 2 redefined
    2: 7c80c729 kernel32!lstrcpyA
breakpoint 5 redefined
    5: 7c838fb9 kernel32!lstrcatA
breakpoint 8 redefined
    8: 7c81ee79 kernel32!strcmpA
breakpoint 9 redefined
    9: 7c80c6e0 kernel32!strlenA
breakpoint 12 redefined
   12: 7c80b929 kernel32!strcmpiA
breakpoint 4 redefined
    4: 7c810311 kernel32!lstrcpynA
0:014> g
```

```
ChildEBP RetAddr  Args to Child
0013dbb0 77f706cd 0013dbd8 00000000 001f4b80 kernel32!strlenA (FPO:
[Non-Fpo])
0013dbc8 77f70738 0013dbd8 001f4b80 74666f53
SHLWAPI!PathAddBackslashA+0x18 (FPO: [Non-Fpo])
0013dce0 77f7078a 001f4b80 00000100 0013de50
SHLWAPI!SHRegSubKeyAddBackslashA+0x50 (FPO: [Non-Fpo])
0013dd08 77f6b3ed 0013dd2c 001f4b78 0019b9d8
SHLWAPI!SHRegOpenUSKeyA+0xb0 (FPO: [Non-Fpo])
```

0013de30 77262891 0013e0a8 00020019 0019b9d8
SHLWAPI!SHRegOpenUSKeyW+0x59 (FPO: [Non-Fpo])
0013de54 7726bf3d 0019b9d8 0013e0a8 00020019
urlmon!CRegKey::Open+0x39 (FPO: [Non-Fpo])
0013e2b4 772642c7 0013e31c 0013e860 0013e358
urlmon!CSecurityManager::CheckSiteAndDomainMappings+0xc1 (FPO:
[Non-Fpo])
0013e2e0 77264b26 0013e31c 0013e860 00000000
urlmon!CSecurityManager::_MapComponentsToZone+0xc6 (FPO: [Non-Fpo])
0013e7cc 77264a26 019c6a40 0013e860 00000000
urlmon!CSecurityManager::_MapUrlToZoneDirect+0xc0 (FPO: [Non-Fpo])
0013e804 772630ce 019c6a40 0013e860 00000000
urlmon!CSecurityManager::MapUrlToZoneInternal+0xcf (FPO: [Non-Fpo])
0013e820 7d537cb3 001932c0 019c6a40 0013e860
urlmon!CSecurityManager::MapUrlToZone+0x1a (FPO: [Non-Fpo])
0013e870 7d5a4c0f 00001400 0013e8a4 00000000
mshtml!CMarkup::ProcessURLAction+0x1c0 (FPO: [Non-Fpo])
0013e8ac 7d51cae7 00000000 0013e924 0019b4ec
mshtml!CDoc::CRecalcHost::EngineRecalcAll+0x5c (FPO: [Non-Fpo])
0013e8e4 7d527ea2 00000000 035f5878 0019b338
mshtml!CView::EnsureView+0x117 (FPO: [Non-Fpo])
0013e8fc 7d50c57a 0019b4ec 00000000 00000000
mshtml!CView::EnsureViewCallback+0x41 (FPO: [Non-Fpo])
0013e930 7d508508 0013eacc 7d508423 00000000
mshtml!GlobalWndOnMethodCall+0x66 (FPO: [Non-Fpo])
0013ea64 77d48734 00020230 00008002 00000000
mshtml!GlobalWndProc+0x1e2 (FPO: [Non-Fpo])
0013ea90 77d48816 7d508423 00020230 00008002
USER32!InternalCallWinProc+0x28
0013eaf8 77d489cd 00000000 7d508423 00020230
USER32!UserCallWinProcCheckWow+0x150 (FPO: [Non-Fpo])
0013eb58 77d48a10 0013eb98 00000000 0013eb80
USER32!DispatchMessageWorker+0x306 (FPO: [Non-Fpo])
0013eb68 75fa6885 0013eb98 00000000 00163148
USER32!DispatchMessageW+0xf (FPO: [Non-Fpo])
0013eb80 75faelca 0013eb98 0013ee98 00000000
BROWSEUI!TimedDispatchMessage+0x33 (FPO: [Non-Fpo])
0013ede0 75fae331 00162f38 0013ee98 00162f38
BROWSEUI!BrowserThreadProc+0x32e (FPO: [Non-Fpo])
0013ee6c 75fae5d5 00162f38 00162f38 00000000
BROWSEUI!BrowserProtectedThreadProc+0x44 (FPO: [Non-Fpo])
0013fef0 777e707e 00162f38 00000000 00000000
BROWSEUI!SHOpenFolderWindow+0x22c (FPO: [Non-Fpo])
0013ff10 00402372 001523ba 00000001 00090000 SHDOCVW!IEWinMain+0x129
(FPO: [Non-Fpo])

0013ff60 00402444 00400000 00000000 001523ba iexplore!WinMainT+0x2de
(FPO: [Non-Fpo])
0013ffc0 7c816d4f 00090000 000c8a50 7ffde000
iexplore!_ModuleEntry+0x99 (FPO: [Non-Fpo])
0013fff0 00000000 00402451 00000000 78746341
kernel32!BaseProcessStart+0x23 (FPO: [Non-Fpo])

ChildEBP RetAddr Args to Child

0013e010 77f706cd 0013e038 00000000 001f4b80 kernel32!lstrlenA (FPO:
[Non-Fpo])
0013e028 77f70738 0013e038 001f4b80 74666f53
SHLWAPI!PathAddBackslashA+0x18 (FPO: [Non-Fpo])
0013e140 77f7078a 001f4b80 00000100 0013e2b0
SHLWAPI!SHRegSubKeyAddBackslashA+0x50 (FPO: [Non-Fpo])
0013e168 77f6b3ed 0013e18c 001f4b78 0019b9d8
SHLWAPI!SHRegOpenUSKeyA+0xb0 (FPO: [Non-Fpo])
0013e290 77262891 7726ba60 00020019 0019b9d8
SHLWAPI!SHRegOpenUSKeyW+0x59 (FPO: [Non-Fpo])
0013e2b4 7726baae 0019b9d8 7726ba60 00020019
urlmon!CRegKey::Open+0x39 (FPO: [Non-Fpo])
0013e2e0 77264b26 0013e31c 0013e860 00000000
urlmon!CSecurityManager::_MapComponentsToZone+0x165 (FPO: [Non-Fpo])
0013e7cc 77264a26 019c6a40 0013e860 00000000
urlmon!CSecurityManager::_MapUrlToZoneDirect+0xc0 (FPO: [Non-Fpo])
0013e804 772630ce 019c6a40 0013e860 00000000
urlmon!CSecurityManager::MapUrlToZoneInternal+0xcf (FPO: [Non-Fpo])
0013e820 7d537cb3 001932c0 019c6a40 0013e860
urlmon!CSecurityManager::MapUrlToZone+0x1a (FPO: [Non-Fpo])
0013e870 7d5a4c0f 00001400 0013e8a4 00000000
mshtml!CMarkup::ProcessURLAction+0x1c0 (FPO: [Non-Fpo])
0013e8ac 7d51cae7 00000000 0013e924 0019b4ec
mshtml!CDoc::CRecalcHost::EngineRecalcAll+0x5c (FPO: [Non-Fpo])
0013e8e4 7d527ea2 00000000 035f5878 0019b338
mshtml!CView::EnsureView+0x117 (FPO: [Non-Fpo])
0013e8fc 7d50c57a 0019b4ec 00000000 00000000
mshtml!CView::EnsureViewCallback+0x41 (FPO: [Non-Fpo])
0013e930 7d508508 0013eacc 7d508423 00000000
mshtml!GlobalWndOnMethodCall+0x66 (FPO: [Non-Fpo])
0013ea64 77d48734 00020230 00008002 00000000
mshtml!GlobalWndProc+0x1e2 (FPO: [Non-Fpo])
0013ea90 77d48816 7d508423 00020230 00008002
USER32!InternalCallWinProc+0x28

```

0013eaf8 77d489cd 00000000 7d508423 00020230
USER32!UserCallWinProcCheckWow+0x150 (FPO: [Non-Fpo])
0013eb58 77d48a10 0013eb98 00000000 0013eb80
USER32!DispatchMessageWorker+0x306 (FPO: [Non-Fpo])
0013eb68 75fa6885 0013eb98 00000000 00163148
USER32!DispatchMessageW+0xf (FPO: [Non-Fpo])
0013eb80 75fae1ca 0013eb98 0013ee98 00000000
BROWSEUI!TimedDispatchMessage+0x33 (FPO: [Non-Fpo])
0013ede0 75fae331 00162f38 0013ee98 00162f38
BROWSEUI!BrowserThreadProc+0x32e (FPO: [Non-Fpo])
0013ee6c 75fae5d5 00162f38 00162f38 00000000
BROWSEUI!BrowserProtectedThreadProc+0x44 (FPO: [Non-Fpo])
0013fef0 777e707e 00162f38 00000000 00000000
BROWSEUI!SHOpenFolderWindow+0x22c (FPO: [Non-Fpo])
0013ff10 00402372 001523ba 00000001 00090000 SHDOCVW!IEWinMain+0x129
(FPO: [Non-Fpo])
0013ff60 00402444 00400000 00000000 001523ba iexplore!WinMainT+0x2de
(FPO: [Non-Fpo])
0013ffc0 7c816d4f 00090000 000c8a50 7ffde000
iexplore!_ModuleEntry+0x99 (FPO: [Non-Fpo])
0013fff0 00000000 00402451 00000000 78746341
kernel32!BaseProcessStart+0x23 (FPO: [Non-Fpo])

```

Next is extracting all the functions and sorting it:

```
cat xxxx |cut -d' ' -f6|sort|uniq|grep "\!"
```

Now, identifying the “high level functions” processing the URL or HTTP packets:

```

WININET!HANDLE_OBJECT::Dereference+0x22
WININET!HTTP_HEADERS::AddRequest+0x18
WININET!HTTP_HEADERS::AddRequest+0x1f
WININET!HTTP_HEADERS::AddRequest+0x27
WININET!HTTP_HEADERS::QueryRequestVersion+0x2d
WININET!HttpOpenRequestA+0x272
WININET!HttpOpenRequestA+0x30c
WININET!HttpOpenRequestA+0x368
WININET!HttpQueryInfoA+0x1c8
WININET!HTTP_REQUEST_HANDLE_OBJECT::AddTimeHeader+0xac
WININET!HTTP_REQUEST_HANDLE_OBJECT::AddTimestampsFromCacheToResponseH
eaders+0x2d
WININET!HTTP_REQUEST_HANDLE_OBJECT::AddTimestampsFromCacheToResponseH
eaders+0x42
WININET!HTTP_REQUEST_HANDLE_OBJECT::BuildProxyMessage+0x26
WININET!HTTP_REQUEST_HANDLE_OBJECT::CreateCookieHeaderIfNeeded+0x47

```

WININET!HTTP_REQUEST_HANDLE_OBJECT::FHttpBeginCacheRetrieval+0x25a
WININET!HTTP_REQUEST_HANDLE_OBJECT::FHttpBeginCacheWrite+0x2b9
WININET!HTTP_REQUEST_HANDLE_OBJECT::GetFromCachePostNetIO+0x169
WININET!HTTP_REQUEST_HANDLE_OBJECT::HttpSendRequest_Finish+0x6a
WININET!HTTP_REQUEST_HANDLE_OBJECT::HttpSendRequest_Start+0x189
WININET!HTTP_REQUEST_HANDLE_OBJECT::HttpSendRequest_Start+0x1fc

...

WININET!InternetCreateUrlA+0x18
WININET!_InternetCreateUrlA+0x197
WININET!_InternetCreateUrlA+0x394
WININET!InternetGetCookieExA+0x90
WININET!InternetGetCookieExW+0x18f
WININET!InternetIndicateStatus+0xf5
WININET!InternetLockRequestFile+0xee
WININET!InternetLockRequestFile+0xf9
WININET!InternetQueryDataAvailable+0x18f
WININET!InternetQueryOptionA+0xb93
WININET!InternetUnlockRequestFile+0x77
WININET!InternetUnlockRequestFile+0x82
WININET!InternetUnlockRequestFile+0xe0
WININET!IsDialUpConnection+0x74
WININET!MEMMAP_FILE::AllocateEntry+0x4d
WININET!MEMMAP_FILE::GrowMapFile+0x9c
WININET!MEMMAP_FILE::Init+0x1ca
WININET!MEMMAP_FILE::RemapAddress+0x74
WININET!MEMMAP_FILE::RemapAddress+0x7c
WININET!MEMMAP_FILE::RemapAddress+0xcd
WININET!pHttpGetUrlString+0x71
WININET!pHttpGetUrlString+0xae
WININET!QueryAvailable_Fsm+0x23
WININET!QueryResolverCache+0x79
WININET!RasEnumConnHelp::Enum+0x5c
WININET!ResolverCacheHit+0x20
WININET!RMakeInternetConnectObjectHandle+0x3e
WININET!UnicodeBuf::Convert+0x12
WININET!UrlCacheCreateFile+0x31
WININET!URL_CONTAINER::AddUrl+0x15b
WININET!URL_CONTAINER::AddUrl+0x74
WININET!URL_CONTAINER::CreateUniqueFile+0x45
WININET!URL_CONTAINER::HashFindItem+0x1f3
WININET!URL_CONTAINER::Init+0x1ac

As these functions are processing the input or output, any vulnerabilities is likely exploitable from the HTTP or HTML input. So which are the functions directly relevant to HTTP/HTML input?

Taking “WININET!InternetGetCookieExA” as an example, the stacktrace with this API are shown below:

```
0013b4a4 771c44c3 03613c60 03613c60 77239a98 kernel32!lstrlenA (FPO: [Non-Fpo])
0013b4b8 771c4465 03613c60 00000000 0013b7d0
WININET!UnicodeBuf::Convert+0x12 (FPO: [Non-Fpo])
0013b6d8 771d964e 03613c60 0013b728 771d9582
WININET!GetZoneFromUrl+0x54 (FPO: [Non-Fpo])
0013b6e4 771d9582 03613c60 00000000 0013b7d0
WININET!GetCookieMainSwitch+0xd (FPO: [Non-Fpo])
0013b728 771d94cb 03613c60 00000000 03628c28
WININET!InternetGetCookieExA+0x90 (FPO: [Non-Fpo])
0013b780 7d6044cd 019a43a0 00000000 0013b7d8
WININET!InternetGetCookieExW+0x18f (FPO: [Non-Fpo])
0013b7ac 7d60c073 019a43a0 00000000 0013b7d8
mshtml!CMarkup::GetCookie+0x2d (FPO: [Non-Fpo])
0013d7e0 7d573558 019781e0 029756f8 02a07510
mshtml!CDocument::get_cookie+0x80 (FPO: [Non-Fpo])
0013d804 7d52e243 019781e0 02a07510 019e61c0 mshtml!GS_PropEnum+0x33
(FPO: [Non-Fpo])
0013d884 7d52e0df 019781e0 00000406 7d573525
mshtml!CBase::ContextInvokeEx+0x462 (FPO: [Non-Fpo])
0013d8b0 7d545507 019781e0 00000406 00000409
mshtml!CBase::InvokeEx+0x25 (FPO: [Non-Fpo])
0013d8fc 7d50dff9 019781e0 7d52e0ba 00000000
mshtml!DispatchInvokeCollection+0x158 (FPO: [Non-Fpo])
0013d950 75c56bfb 019781e0 00000406 00000409
mshtml!CDocument::InvokeEx+0xcb (FPO: [Non-Fpo])
0013d988 75c631c9 0003f188 01a0da30 00000406
jscript!IDispatchExInvokeEx+0xa9 (FPO: [Non-Fpo])
0013d9f8 75c576d7 0003f188 01a0da30 00000406
jscript!InvokeDispatchEx+0x78 (FPO: [Non-Fpo])
0013da40 75c61ab3 0003f188 0013dbdc 00000002
jscript!VAR::InvokeByName+0x1c1 (FPO: [Non-Fpo])
0013dc30 75c54d34 0013dfe0 00000000 0013dd4c
jscript!CScriptRuntime::Run+0xb4b (FPO: [Non-Fpo])
0013dcf4 75c57869 0013dfe0 00000000 02975790
jscript!ScrFncObj::Call+0x69 (FPO: [Non-Fpo])
0013dd88 75c577f6 0003f188 00000000 00000001
jscript!NameTbl::InvokeInternal+0x218 (FPO: [Non-Fpo])
```

```

0013ddb4 75c6206c 0003f188 00000000 00000001
jscript!VAR::InvokeByDispID+0xd4 (FPO: [Non-Fpo])
0013ddf8 75c5764d 0003f188 0013de18 00000001
jscript!VAR::InvokeByName+0x164 (FPO: [Non-Fpo])
0013de38 75c56688 0003f188 00000001 0013dfe0
jscript!VAR::InvokeDispName+0x43 (FPO: [Non-Fpo])
0013de5c 75c56e58 0003f188 00000000 00000001
jscript!VAR::InvokeByDispID+0xfb (FPO: [Non-Fpo])
0013e050 75c54d34 0013e400 00000000 0013e16c
jscript!CScriptRuntime::Run+0x18fb (FPO: [Non-Fpo])
0013e114 75c57869 0013e400 00000000 02975800
jscript!ScrFncObj::Call+0x69 (FPO: [Non-Fpo])
0013e1a8 75c577f6 0003f188 00000000 00000001
jscript!NameTbl::InvokeInternal+0x218 (FPO: [Non-Fpo])
0013eld4 75c6206c 0003f188 00000000 00000001
jscript!VAR::InvokeByDispID+0xd4 (FPO: [Non-Fpo])
0013e218 75c5764d 0003f188 0013e238 00000001
jscript!VAR::InvokeByName+0x164 (FPO: [Non-Fpo])
0013e258 75c56688 0003f188 00000001 0013e400
jscript!VAR::InvokeDispName+0x43 (FPO: [Non-Fpo])
0013e27c 75c56e58 0003f188 00000000 00000001
jscript!VAR::InvokeByDispID+0xfb (FPO: [Non-Fpo])
0013e470 75c54d34 0013e55c 75c51b40 0013e55c
jscript!CScriptRuntime::Run+0x18fb (FPO: [Non-Fpo])
0013e534 75c5655f 0013e55c 00000000 00000000
jscript!ScrFncObj::Call+0x69 (FPO: [Non-Fpo])

```

So the function “**mshtml!CDocument::get_cookie**” is the predecessor of the function “**WININET!InternetGetCookieExA**”.

From above we can identify many other functions that is built on the other functions, or DLL building on top other DLL.

So DLL hierarchies can approximated like this:

lexplore.exe => browseui.dll => mshtml.dll => wininet.dll.

The above sequence are derived from below stacktrace:

```

0013e760 7d5d9242 001ef928 019de760 7d595184
WININET!InternetUnlockRequestFile+0x77 (FPO: [Non-Fpo])
0013e76c 7d595184 019de760 0013e7c8 7d51972b
mshtml!CBitsInfo::~~CBitsInfo+0x18 (FPO: [0,0,0])
0013e778 7d51972b 00000001 019de760 019de760
mshtml!CBitsInfo::~`scalar deleting destructor'+0xd (FPO: [Non-Fpo])

```

```

0013e788 7d57f82b 019de8a0 019de5d4 019de7f4
mshtml!CBaseFT::SubRelease+0x22 (FPO: [0,0,0])
0013e7c8 7d52b9e8 00000000 0013e7fc 7d52b992
mshtml!CDwnInfo::Release+0x242 (FPO: [Non-Fpo])
0013e7d4 7d52b992 019de8a0 019de8a0 7d519737
mshtml!CDwnInfo::DelDwnCtx+0x5b (FPO: [Non-Fpo])
0013e7e0 7d519737 019de590 019de8a0 7d5950ed
mshtml!CDwnCtx::Passivate+0x24 (FPO: [0,0,0])
0013e7ec 7d5950ed 019de5c8 019de590 0013e8e0
mshtml!CBaseFT::Release+0x1d (FPO: [0,0,0])
0013e7fc 7d595386 00000000 0013e924 035f583c
mshtml!CScriptElement::SetBitsCtx+0x2b (FPO: [Non-Fpo])
0013e8e0 7d71e3e0 019de8a0 0013e8fc 7d539645
mshtml!CScriptElement::OnDwnChan+0x313 (FPO: [Non-Fpo])
0013e8ec 7d539645 019de8a0 019de590 0013e930
mshtml!CScriptElement::OnDwnChanCallback+0x10 (FPO: [Non-Fpo])
0013e8fc 7d50c57a 019de8a0 00000000 00000000
mshtml!CDwnChan::OnMethodCall+0x19 (FPO: [Non-Fpo])
0013e930 7d508508 0013eacc 7d508423 00000000
mshtml!GlobalWndOnMethodCall+0x66 (FPO: [Non-Fpo])
0013ea64 77d48734 00020230 00008002 00000000
mshtml!GlobalWndProc+0x1e2 (FPO: [Non-Fpo])
0013ea90 77d48816 7d508423 00020230 00008002
USER32!InternalCallWinProc+0x28
0013eaf8 77d489cd 00000000 7d508423 00020230
USER32!UserCallWinProcCheckWow+0x150 (FPO: [Non-Fpo])
0013eb58 77d48a10 0013eb98 00000000 0013eb80
USER32!DispatchMessageWorker+0x306 (FPO: [Non-Fpo])
0013eb68 75fa6885 0013eb98 00000000 00163148
USER32!DispatchMessageW+0xf (FPO: [Non-Fpo])
0013eb80 75faelca 0013eb98 0013ee98 00000000
BROWSEUI!TimedDispatchMessage+0x33 (FPO: [Non-Fpo])
0013ede0 75fae331 00162f38 0013ee98 00162f38
BROWSEUI!BrowserThreadProc+0x32e (FPO: [Non-Fpo])
0013ee6c 75fae5d5 00162f38 00162f38 00000000
BROWSEUI!BrowserProtectedThreadProc+0x44 (FPO: [Non-Fpo])
0013fef0 777e707e 00162f38 00000000 00000000
BROWSEUI!SHOpenFolderWindow+0x22c (FPO: [Non-Fpo])
0013ff10 00402372 001523ba 00000001 00090000 SHDOCVW!IEWinMain+0x129
(FPO: [Non-Fpo])
0013ff60 00402444 00400000 00000000 001523ba iexplore!WinMainT+0x2de
(FPO: [Non-Fpo])

```

Now focusing on the high level DLL "browseui.dll":


```
grep "BROWSEUI" xxxxx | cut -d' ' -f6|sort|uniq
BROWSEUI!BrowserProtectedThreadProc+0x44
BROWSEUI!BrowserThreadProc+0x32e
BROWSEUI!CShellBrowser2::Exec+0x132
BROWSEUI!CShellBrowser2::_GetTempZone+0x126
BROWSEUI!CShellBrowser2::_UpdateZonesPane+0x108
BROWSEUI!SHOpenFolderWindow+0x22c
BROWSEUI!TimedDispatchMessage+0x33
```

We can see that there is not many entrypoint functions in this DLL, and its role is clearly described by the few function names listed above.

And next is “mshtml”:

```
CHtmlLoadmshtml!CHtmlPost::RunNested
mshtml!CAnchorElement::ClickAction
mshtml!CAnchorElement::DoClick
mshtml!CBase::ContextInvokeEx
mshtml!CBase::FireEvent
mshtml!CBase::FirePropertyNotify
mshtml!CBaseFT::Release
mshtml!CBaseFT::SubRelease
mshtml!CBase::InvokeAttachEvents
mshtml!CBase::InvokeDispatchExtraParam
mshtml!CBase::InvokeDispatchWithThis
mshtml!CBase::InvokeEvent
mshtml!CBase::InvokeEx
mshtml!CBase::PrivateRelease
mshtml!CBitsInfo::~~CBitsInfo
mshtml!CBitsInfo::`scalar
mshtml!CBitsLoad::Init
mshtml!CBodyElement::Notify
mshtml!CCssInfo::`vector
mshtml!CDispContainer::DrawChildren
mshtml!CDispContainer::DrawSelf
mshtml!CDispLeafNode::DrawSelf
mshtml!CDispNode::Draw
mshtml!CDispRoot::DrawEntire
mshtml!CDispRoot::DrawRoot
mshtml!CDoc::AttachPeer
mshtml!CDoc::AttachPeersCss
mshtml!CDoc::AttachPeerUrl
mshtml!CDoc::CRecalcHost::EngineRecalcAll
mshtml!CDoc::DoNavigate
mshtml!CDoc::EnsurePeerFactoryUrl
```

mshtml!CDoc::ExecuteFilterTasks
mshtml!CDoc::FilterCallback
mshtml!CDoc::FollowHyperlink
mshtml!CDoc::NotifySelection
mshtml!CDoc::OnMouseMessage
mshtml!CDoc::OnPaint
mshtml!CDoc::OnWindowMessage
mshtml!CDoc::ParseGlobal
mshtml!CDoc::PumpMessage
mshtml!CDoc::SetCurrentElem
mshtml!CDocument::get_cookie
mshtml!CDocument::HostProcessUrlAction
mshtml!CDocument::InvokeEx
mshtml!CDocument::write
mshtml!CDOMChildrenCollection::Release
mshtml!CDwnBindData::Bind
mshtml!CDwnBindData::BufferData
mshtml!CDwnBindData::OnDataAvailable
mshtml!CDwnBindData::OnProgress
mshtml!CDwnBindData::ReadFromBind
mshtml!CDwnBindData::ReportData
mshtml!CDwnBindData::SetCallback
mshtml!CDwnBindData::Signal
mshtml!CDwnBindData::SignalData
mshtml!CDwnBindData::SignalHeaders
mshtml!CDwnBindData::TerminateBind
mshtml!CDwnBindData::TerminateOnApt
mshtml!CDwnChan::OnMethodCall
mshtml!CDwnCtx::Passivate
mshtml!CDwnCtx::SetLoad
mshtml!CDwnDoc::SetSecurityID
mshtml!CDwnInfo::DelDwnCtx
mshtml!CDwnInfo::Release
mshtml!CDwnInfo::SetLoad
mshtml!CDwnLoad::Init
mshtml!CDwnLoad::OnBindCallback
mshtml!CDwnLoad::SetCallback
mshtml!CDwnTaskExec::ThreadExec
mshtml!CElement::AddFilters
mshtml!CElement::BecomeCurrent
mshtml!CElement::BubbleBecomeCurrent
mshtml!CElement::ContextInvokeEx
mshtml!CElement::ContextThunk_InvokeEx
mshtml!CElement::DoClick
mshtml!CElement::EnsureFilterBehavior

```

mshtml!CElement::EnsureIdentityPeer
mshtml!CElement::EnterTree
mshtml!CElement::Inject
mshtml!CElement::Notify
mshtml!CElement::ProcessPeerTask
mshtml!CElement::put_innerHTML

```

This is quite long, so pausing here and searching for documentation related to CElement:

<https://www.geoffchappell.com/studies/windows/ie/mshtml/classes/celement.htm?tx=71,73&ts=0.424>


MSHTML Classes: CElement

Many of the MSHTML classes expose the properties (methods, collections, etc) of the otherwise internal **CElement** class.

Properties that Microsoft does not list in the *HTML and DHTML Reference* are highlighted.

Property	Attribute	Interface	Member
accessKey	same	IHTMLElement2	get_accessKey put_accessKey
addBehavior		IHTMLElement2	addBehavior
addFilter		IHTMLElement2	addFilter
all		IHTMLElement	get_all
applyElement		IHTMLElement2	applyElement
attachEvent		IHTMLElement2	attachEvent
behaviorUrns		IHTMLElement2	get_behaviorUrns
blur		IHTMLElement2	blur
canHaveChildren		IHTMLElement2	get_canHaveChildren
canHaveHTML		IHTMLElement3	get_canHaveHTML
children		IHTMLElement	get_children

And there are many other classes than CElement:

A screenshot of a class list from the MSHTML DLL. The list is displayed in a window with a title bar. The classes are listed in a single column, each preceded by a small icon. The class 'CElement' is highlighted with a blue background. The list includes various HTML-related classes such as CBookmarkCollection, CButton, CCommentElement, CCurrentStyle, CDataTransfer, CDDElement, CDefaults, CDivElement, CDListElement, CDoc, CDocument, CDOMChildrenCollection, CDOMImplementation, CDOMTextNode, CDTElement, CElement, CElementCollection, CEventObj, CFieldSetElement, CFontElement, CFontFace, CFontNameOptions, CFontSizeOptions, CFormElement, CFrameElement, CFrameSetSite, CGenericElement, CHeadElement, CHeaderElement, CHRElement, CHTMLComponentAttach, CHTMLComponentDD, CHTMLComponentDesc, CHTMLComponentEvent, CHTMLComponentProperty, CHTMLDlg, CHtmlElement, CHTMLNamespace, CHTMLNamespaceCollection, CHTMLPopup, CIFrameElement, CImageElementFactory, CImgElement, CInput, CIPrintCollection, CIsIndexElement, CLabelElement, CLegendElement, and CLIElement.

- CBookmarkCollection
- CBRElement
- CButton
- CCommentElement
- CCurrentStyle
- CDataTransfer
- CDDElement
- CDefaults
- CDivElement
- CDListElement
- CDoc
- CDocument
- CDOMChildrenCollection
- CDOMImplementation
- CDOMTextNode
- CDTElement
- CElement**
- CElementCollection
- CEventObj
- CFieldSetElement
- CFontElement
- CFontFace
- CFontNameOptions
- CFontSizeOptions
- CFormElement
- CFrameElement
- CFrameSetSite
- CGenericElement
- CHeadElement
- CHeaderElement
- CHRElement
- CHTMLComponentAttach
- CHTMLComponentDD
- CHTMLComponentDesc
- CHTMLComponentEvent
- CHTMLComponentProperty
- CHTMLDlg
- CHtmlElement
- CHTMLNamespace
- CHTMLNamespaceCollection
- CHTMLPopup
- CIFrameElement
- CImageElementFactory
- CImgElement
- CInput
- CIPrintCollection
- CIsIndexElement
- CLabelElement
- CLegendElement
- CLIElement

Continuing the enumeration of many other functions in MSHTML DLL:

mshtml!CExecFT::StaticThreadProc
mshtml!CExecFT::ThreadProc
mshtml!CFilterBehaviorSite::AddFilterToBehavior
mshtml!CFilterBehaviorSite::CreateFilterBehavior
mshtml!CFilterBehaviorSite::ParseAndAddFilters
mshtml!CFrameElement::Notify
mshtml!CFrameSite::CreateObject
mshtml!CFrameSite::Notify
mshtml!CFrameSite::OnPropertyChange_Src
mshtml!CHtmCtx::SetLoad
mshtml!CHtmCtx::Write
mshtml!CHtmFrameParseCtx::Execute
mshtml!CHtmInfo::Passivate
mshtml!CHtmInfo::UnlockFile
mshtml!CHtmlComponentAttach::fireEvent
mshtml!CHtmlComponentAttach::FireEvent
mshtml!CHtmlComponentBase::InvokeEngines
mshtml!CHtmlComponentConstructor::EnsureFactoryComponent
mshtml!CHtmlComponentConstructor::FindBehavior
mshtml!CHtmlComponentConstructor::LoadMarkup
mshtml!CHtmlComponentConstructor::LoadMarkupAsynchronously
mshtml!CHtmlComponentConstructor::OnMarkupLoaded
mshtml!CHtmlComponentConstructor::Passivate
mshtml!CHtmlComponentConstructor::RequestMarkup
mshtml!CHtmlComponentDD::InvokeEx
mshtml!CHtmlComponent::FireNotification
mshtml!CHtmlComponent::InvokeEx
mshtml!CHtmlComponent::Load
mshtml!CHtmlComponentMethod::InvokeItem
mshtml!CHtmlComponent::OnLoadStatus
mshtml!CHtmlComponent::OnMarkupLoaded
mshtml!CHtmlComponent::Passivate
mshtml!CHTMLNamespace::Release
mshtml!CHtmLoad::Init
mshtml!CHtmLoad::OnBindData
mshtml!CHtmLoad::OnPostRestart
mshtml!CHtmLoad::Write
mshtml!CHtmParse::Commit
mshtml!CHtmParse::Execute
mshtml!CHtmPost::Broadcast
mshtml!CHtmPost::Exec
mshtml!CHtmPost::OnDwnChanCallback
mshtml!CHtmPost::ProcessTokens
mshtml!CHtmPost::Run
mshtml!CHtmPre::AddDwnCtx

mshtml!CHtmPre::DoTokenizeOneTag
mshtml!CHtmPre::Exec
mshtml!CHtmPre::Run
mshtml!CHtmPre::SpecialToken
mshtml!CHtmPre::Tokenize
mshtml!CHtmPre::TokenizeText
mshtml!CHtmRootParseCtx::FlushNotifications
mshtml!CHyperlink::ClickAction
mshtml!CImgLoad::Init
mshtml!CImgLoad::OnBindData
mshtml!CLayout::DrawClientLayers
mshtml!ClearInterfaceFn
mshtml!CLinkElement::EnsureStyleDownload
mshtml!CLinkElement::HandleLinkedObjects
mshtml!CLinkElement::Notify
mshtml!CLinkElement::OnDwnChan
mshtml!CLinkElement::OnDwnChanCallback
mshtml!CLinkElement::SetCssCtx
mshtml!CMarkup::AccessAllowed
mshtml!CMarkup::CanCommitScripts
mshtml!CMarkup::CommitQueuedScripts
mshtml!CMarkup::CommitQueuedScriptsInline
mshtml!CMarkup::CreateWindowHelper
mshtml!CMarkup::GetCookie
mshtml!CMarkup::GetFrameZone
mshtml!CMarkup::GetSecurityID
mshtml!CMarkup::Load
mshtml!CMarkup::LoadFromInfo
mshtml!CMarkup::LoadHistoryInternal
mshtml!CMarkup::Notify
mshtml!CMarkup::NotifyDescendents
mshtml!CMarkup::OnLoadStatus
mshtml!CMarkup::OnLoadStatusDone
mshtml!CMarkup::Passivate
mshtml!CMarkup::PrepareDwnDoc
mshtml!CMarkup::ProcessIdentityPeerTask
mshtml!CMarkup::ProcessPeerTasks
mshtml!CMarkup::ProcessURLAction
mshtml!CMarkup::SendNotification
mshtml!CMarkup::SetReadyState
mshtml!CMarkup::TearDownMarkup
mshtml!CMarkup::UnblockScriptExecution
mshtml!CMarkup::UnblockScriptExecutionHelper
mshtml!CMarkup::UnloadContents
mshtml!CMarkup::UpdateReleaseHtmCtx

mshtml!COMWindowProxy::AccessAllowed
mshtml!COMWindowProxy::CanNavigateToUrlWithZoneCheck
mshtml!COMWindowProxy::GetDispID
mshtml!COMWindowProxy::get_frameElement
mshtml!COMWindowProxy::InvokeEx
mshtml!COMWindowProxy::RestartLoad
mshtml!COMWindowProxy::SecureObject
mshtml!COMWindowProxy::subGetDispID
mshtml!COMWindowProxy::subInvokeEx
mshtml!COMWindowProxy::SwitchMarkup
mshtml!CPeerFactory::AttachPeer
mshtml!CPeerFactoryBuiltin::FindBehavior
mshtml!CPeerFactoryUrl::AttachAllDeferred
mshtml!CPeerFactoryUrl::AttachPeer
mshtml!CPeerFactoryUrl::Create
mshtml!CPeerFactoryUrl::FindBehavior
mshtml!CPeerFactoryUrl::Init
mshtml!CPeerFactoryUrl::LaunchUrlDownload
mshtml!CPeerFactoryUrlMap::~~CPeerFactoryUrlMap
mshtml!CPeerFactoryUrlMap::EnsurePeerFactoryUrl
mshtml!CPeerFactoryUrlMap::~`scalar
mshtml!CPeerFactoryUrl::OnStopBinding
mshtml!CPeerFactoryUrl::Passivate
mshtml!CPeerHolder::CPeerSite::FireEvent
mshtml!CPeerHolder::Create
mshtml!CPeerHolder::Draw
mshtml!CPeerHolder::InvokeExMulti
mshtml!CPeerHolder::InvokeExSingle
mshtml!CProgSink::DoUpdate
mshtml!CProgSink::OnMethodCall
mshtml!CScriptCollection::GetHolderForLanguage
mshtml!CScriptCollection::GetHolderForLanguageHelper
mshtml!CScriptCollection::InvokeEx
mshtml!CScriptCollection::InvokeName
mshtml!CScriptCollection::IsSafeToRunScripts
mshtml!CScriptCollection::ParseScriptText
mshtml!CScriptElement::CommitCode
mshtml!CScriptElement::DownloadScript
mshtml!CScriptElement::EnsureScriptDownloadLeft
mshtml!CScriptElement::Execute
mshtml!CScriptElement::Notify
mshtml!CScriptElement::OnDwnChan
mshtml!CScriptElement::OnDwnChanCallback
mshtml!CScriptElement::SetBitsCtx
mshtml!CScriptHolder::Exec

```
mshtml!CServer::OnWindowMessage
mshtml!CServer::WndProc
mshtml!CTableCell::Notify
mshtml!CTimer::Release
mshtml!CView::EnsureView
mshtml!CView::EnsureViewCallback
mshtml!CView::RenderView
mshtml!CWindow::InvokeEx
mshtml!DispatchInvokeCollection
mshtml!CHTMLEditor::Notify
mshtml!CSelectionManager::Notify
mshtml!CSelectionManager::SetEditContextFromCurrencyChange
mshtml!EdUtil::GetActiveElement
mshtml!EdUtil::GetFrameOrIFrame
mshtml!FindPeer
mshtml!GetMimeInfoFromData
mshtml!GetSIDOfDispatch
mshtml!GlobalWndOnMethodCall
mshtml!GlobalWndProc
mshtml!GS_PropEnum
mshtml!HandleHTMLInjection
mshtml!InlineEvts::Connect
mshtml!Method_void_SAFEARRAYPVPARIANTP
mshtml!Method_void_VARIANT
mshtml!NewDwnBindData
mshtml!NewDwnCtx
mshtml!NotifyElement
mshtml!PostManExecute
mshtml!PostManOnTimer
mshtml!PostManResume
mshtml!ReleaseInterface
```

Finally, what is the use of collecting these stacktrace? These stacktrace are runtime stacktraces generated by IE, and thus showed that exactly how and where the functions can be called through identifying the caller-callee relationship. Except that how the path of execution leading to that specific location is still unknown. This is the initial inputs to the browser that leads to the specific path of execution, ending up the the specific stacktrace.