

# CVE-2012-1856 分析报告

启明星辰安全研究团队

## 漏洞概要:

该漏洞是由于 MSCOMCTL.OCX 中的通用控件 TabStrip ActiveX 控件存在漏洞，允许攻击者构建特制的文档或 WEB 页面，诱使用户解析，可破坏内存，可以应用程序上下文执行任意代码。

## 漏洞分析:

漏洞样本来自于互联网，在此表示感谢。以下是该漏洞的详细分析：

1. 找到 DOC 文档中 OLE 结构，并查找“1EFB6596-857C-11D1-B16A-00C0F0283628”的 clsid。
2. 找到之后查找结构中的 Contents 部分。正是在解析该部分内容的时候出现了错误。
3. 动态调试，打开 OD，附加上 Microsoft Word 软件。并使用 Word 打开漏洞 POC。我们从解析 Contents 内容开始分析。

打开 Contents 内容为一个 Stream 流。

276008DF	8D55 0C	lea	edx, dword ptr [ebp+C]	
276008E2	52	push	edx	
276008E3	6A 00	push	0	
276008E5	8B 08	mov	ecx, dword ptr [eax]	
276008E7	6A 10	push	10	
276008E9	6A 00	push	0	
276008EB	68 A0565E27	push	275E56A0	Unicode "Contents"
276008F0	50	push	eax	
276008F1	FF51 10	call	dword ptr [ecx+10]	ole32.CExposedDocFile::OpenStream
276008F4	85C0	test	eax, eax	
276008F6	7C 1E	j1	short 27600916	
276008F8	8B45 08	mov	eax, dword ptr [ebp+8]	
276008FB	56	push	esi	
276008FC	FF75 0C	push	dword ptr [ebp+C]	

读取内容，读取的内容放到栈中，第三个参数为读取的字节数，此处读取 4 字节，然后比较头部是否为 0x12344321

275C4DA2	8D4D EC	lea	ecx, dword ptr [ebp-14]	ReadBuf
275C4DA5	8B07	mov	eax, dword ptr [edi]	
275C4DA7	53	push	ebx	
275C4DA8	6A 04	push	4	
275C4DA9	51	push	ecx	
275C4DAB	57	push	edi	
275C4DAC	895D F0	mov	dword ptr [ebp-10], ebx	
275C4DAF	FF50 0C	call	dword ptr [eax+C]	ole32.CExposedStream::Read
275C4DB2	3BC3	cmp	eax, ebx	
275C4DB4	7C 2B	j1	short 275C4DE1	
275C4DB6	817D EC 214334	cmp	dword ptr [ebp-14], 12344321	比较头部是否为 0x12344321
275C4DBD	0F85 F7A0000	jnz	275CC8BA	
275C4DC3	8B17	mov	edx, dword ptr [edi]	
275C4DC5	6A FC	push	-4	
275C4DC7	58	pop	eax	

重新定位到 Contents 头。

275C40C3	8B 17	mov	eax, dword ptr [edi]	
275C40C5	6A FC	push	-4	
275C40C7	58	pop	eax	
275C40C8	83C9 FF	or	ecx, 0FFFFFFF	
275C40CB	53	push	ebx	
275C40CC	6A 01	push	1	
275C40CE	51	push	ecx	
275C40CF	50	push	eax	
275C40D0	57	push	edi	
275C40D1	FF52 14	call	dword ptr [edx+14]	ole32.CExposedStream::Seek
275C40D4	3BC3	cmp	eax, ebx	
275C40D6	7C 09	j1	short 275C4DE1	
275C40D8	57	push	edi	
275C40D9	FF75 08	push	dword ptr [ebp+8]	
275C40DC	E8 E660FCFF	call	<HandleContent>	
275C4DE1	5F	pop	edi	
275C4DE2	5E	pop	esi	
275C4DE3	5B	pop	ebx	
275C4DE4	C9	leave		

读取 8 字节内容，并校验头部。

2758AEEC	55	push	ebp	
2758AEEF	8BEC	mov	ebp, esp	
2758AEF2	83EC 10	sub	esp, 10	
2758AEF3	56	push	esi	
2758AEF6	8B75 08	mov	esi, dword ptr [ebp+8]	
2758AEF7	57	push	edi	
2758AEF9	8BF9	mov	edi, ecx	
2758AEFB	8B06	mov	eax, dword ptr [esi]	
2758AEFD	8D4D F8	lea	ecx, dword ptr [ebp-8]	
2758AF00	6A 08	push	8	读取8个字节
2758AF02	51	push	ecx	
2758AF03	56	push	esi	
2758AF04	FF50 0C	call	dword ptr [eax+C]	
2758AF07	85C0	test	eax, eax	
2758AF09	7C 3D	j1	short 2758AF48	
2758AF0B	817D F8 214334	cmp	dword ptr [ebp-8], 12344321	校验头部
2758AF12	0F85 C47F0400	jnz	275D2EDC	
2758AF18	837D FC 08	cmp	dword ptr [ebp-4], 8	
2758AF1C	0F85 BA7F0400	jnz	275D2EDC	
2758AF22	8B06	mov	eax, dword ptr [esi]	
2758AF24	6A 00	push	0	
2758AF26	8D4D F0	lea	ecx, dword ptr [ebp-10]	
2758AF29	6A 08	push	8	
2758AF2B	54	push	eax	

再次读取 8 字节内容，并对齐进行一些检查，与漏洞无关不细讲

2758AED3	8BCE	mov	ecx, esi	
2758AED5	E8 12000000	call	2758AEEC	
2758AED9	85C0	test	eax, eax	
2758AEDC	7C 0A	j1	short 2758AEE8	
2758AEDF	FF7424 0C	push	dword ptr [esp+C]	
2758AEE2	8B06	mov	eax, dword ptr [esi]	
2758AEE4	56	push	esi	
2758AEE5	FF50 38	call	dword ptr [eax+38]	之后的主要处理函数
2758AEE8	5E	pop	esi	
2758AEE9	C2 0800	ret	8	
2758AEEF	CC	push	0	

上一个函数跳出后，进入另一个主要处理函数

2758AED3	8BCE	mov	ecx, esi	
2758AED5	E8 12000000	call	2758AEEC	
2758AED9	85C0	test	eax, eax	
2758AEDC	7C 0A	j1	short 2758AEE8	
2758AEDF	FF7424 0C	push	dword ptr [esp+C]	
2758AEE2	8B06	mov	eax, dword ptr [esi]	
2758AEE4	56	push	esi	
2758AEE5	FF50 38	call	dword ptr [eax+38]	之后的主要处理函数
2758AEE8	5E	pop	esi	
2758AEE9	C2 0800	ret	8	
2758AEEF	CC	push	0	

跟进该函数，首先读取 0xC 个字节

275C4DFC	8B07	mov	eax, dword ptr [edi]	
275C4DFE	53	push	ebx	
275C4DFF	6A 0C	push	0C	读取0xC个字节
275C4E01	51	push	ecx	
275C4E02	57	push	edi	
275C4E03	895D FC	mov	dword ptr [ebp-4], ebx	
275C4E06	895D D0	mov	dword ptr [ebp-30], ebx	
275C4E09	895D F8	mov	dword ptr [ebp-8], ebx	
275C4E0C	FF50 0C	call	dword ptr [eax+C]	ole32.CExposedStrea
275C4E0F	8BF0	mov	esi, eax	
275C4E11	3BF3	cmp	esi, ebx	
275C4E13	0F8C FE7E0000	j1	275CCD17	
275C4E19	817D D4 C17A2A	cmp	dword ptr [ebp-2C], D12A7AC1	比较标志是否是0xD12
275C4E20	0F85 0D7F0000	jnz	275CCD33	
275C4E26	8B4D D8	mov	ecx, dword ptr [ebp-28]	
275C4E29	B8 00000000	mov	eax, 00000000	
275C4E2E	3BC8	cmp	ecx, eax	

比较头部标志，并比较 dwVersion，这里比较重要的是 dwVersion 之后的 cbBytes 字节数，用该字节数加上 0x14 为接下来要读取的字节数，由文件可知，cbBytes=0x4c，再加上最终等于 0x60

275C4E0F	8BF0	mov	esi, eax		
275C4E11	3BF3	cmp	esi, ebx		
275C4E13	0F8C FE7E0000	j1	275CCD17		
275C4E19	817D D4 C17A2A	cmp	dword ptr [ebp-2C], 012A7AC1	比较标志是否是0xD12A7AC1	
275C4E20	0F85 0D7F0000	jnz	275CCD33		
275C4E26	8B4D D8	mov	ecx, dword ptr [ebp-28]	dwVersion	
275C4E29	B8 00000000	mov	eax, 00000000		
275C4E2E	3BC8	cmp	ecx, eax		
275C4E30	0F85 367F0000	jnz	275CCD6C		
275C4E36	33D2	xor	edx, edx		
275C4E38	3BC8	cmp	ecx, eax		
275C4E39	0F95C2	setne	dl		
275C4E3D	8B45 DC	mov	eax, dword ptr [ebp-24]	cbBytesWritten	
275C4E40	4A	dec	edx		
275C4E41	83E2 0C	and	edx, 0C		
275C4E44	83C2 08	add	edx, 8		
275C4E47	8BF2	mov	esi, edx		
275C4E49	83C6	add	eax, esi	cbBytes+0xC+0x8结果作为下次读取的字节数	
275C4E4B	8975 F4	mov	dword ptr [ebp-C], esi		
275C4E4E	3D 00000000	cmp	eax, 00		
275C4E53	0F87 387F0000	ja	275CCD91		
275C4E59	8B45 50FFFFFF	lea	eax, dword ptr [ebp-80]		
275C4E5F	3BC9	cmp	eax, ebx		
275C4E61	8B45 08	mov	eax, dword ptr [ebp-8]		

后面读取 0x60 字节到栈内存，如图为读取到的内容

0012BD04	0073006F
0012BD08	00000000
0012BD0C	00720045
0012BD10	000003E8
0012BD14	00730068
0012BD18	ABCDEF01
0012BD1C	00050000
0012BD20	07853CA0
0012BD24	00650016
0012BD28	FFFFFFFF
0012BD2C	FFFFFFFF
0012BD30	00000000
0012BD34	00000000
0012BD38	00000010
0012BD3C	00000000
0012BD40	00000000
0012BD44	00000000
0012BD48	00000000
0012BD4C	00000000
0012BD50	00000000
0012BD54	00000000
0012BD58	00000000
0012BD5C	00000000
0012BD60	00000000
0012BD64	00000039

定位到下一个比较关键的结构



275C4E9E	C1E9 02	shr	ecx, 2	
275C4EA1	F3:A5	rep	movs dword ptr es:[edi], dword	
275C4EA3	8BCA	mov	ecx, edx	
275C4EA5	83E1 03	and	ecx, 3	
275C4EA8	F3:A4	rep	movs byte ptr es:[edi], byte ptr	
275C4EAA	8B4D F4	mov	ecx, dword ptr [ebp-C]	
275C4EAD	014D FC	add	dword ptr [ebp-4], ecx	定位到 STOCKPROP_PERSISTHEADER
275C4EB0	817D D8 000000	cmp	dword ptr [ebp-28], 60000	
275C4EB7	0F85 F2E0000	jnz	275CCDAF	
275C4EBD	8D45 FC	lea	eax, dword ptr [ebp-4]	
275C4EC0	8BCB	mov	ecx, ebx	

0012BD18	ABCDEF01			0012BCF8	0CF903B0
0012BD1C	00050000			0012BCFC	08D45D98
0012BD20	07853CA0			0012BD00	00000000
0012BD24	00650016			0012BD04	0073006F
0012BD28	FFFFFFFF			0012BD08	00000000
0012BD2C	FFFFFFFF			0012BD0C	00720045
0012BD30	00000000			0012BD10	000003E8
0012BD34	00000000			0012BD14	00730068
0012BD38	00000010			0012BD18	ABCDEF01
0012BD3C	00000000			0012BD1C	00050000
0012BD40	00000000			0012BD20	07853CA0
0012BD44	00000000			0012BD24	00650016
0012BD48	00000000			0012BD28	FFFFFFFF
0012BD4C	00000000			0012BD2C	FFFFFFFF
0012BD50	00000000			0012BD30	00000000
0012BD54	00000000			0012BD34	00000000
0012BD58	00000000			0012BD38	00000010
0012BD5C	00000000			0012BD3C	00000000
0012BD60	00000000			0012BD40	00000000
0012BD64	00000000			0012BD44	00000000
0012BD68	00000000			0012BD48	00000000
0012BD6C	00000000			0012BD4C	00000000
0012BD70	00000000			0012BD50	00000000
0012BD74	00000000			0012BD54	00000000
0012BD78	00000000			0012BD58	00000000
0012BD7C	00000000			0012BD5C	00000000
0012BD80	00000000			0012BD60	00000000
0012BD84	00000000			0012BD64	00000000
0012BD88	00000000			0012BD68	00000000
0012BD8C	00000000			0012BD6C	00000000
0012BD90	00000000			0012BD70	00000000
0012BD94	00000000			0012BD74	00000000
0012BD98	00000000			0012BD78	00000000
0012BD9C	00000000			0012BD7C	00000000
0012BDA0	00000000			0012BD80	00000000
0012BDA4	00000000			0012BD84	00000000
0012BDA8	00000000			0012BD88	00000000
0012BDAC	00000000			0012BD8C	00000000
0012BDB0	00000000			0012BD90	00000000
0012BDB4	00000000			0012BD94	00000000
0012BDB8	00000000			0012BD98	00000000
0012BDBC	00000000			0012BD9C	00000000
0012BDC0	00000000			0012BDA0	00000000
0012BDC4	00000000			0012BDA4	00000000
0012BDC8	00000000			0012BDA8	00000000
0012BDC	00000000			0012BDAC	00000000

在读取数据的时候，还利用一个栈中的地址存放下一次需要读取的数据指针。

275C4EB0	817D D8 000000	cmp	dword ptr [ebp-28], 60000	
275C4EB7	0F85 F2E0000	jnz	275CCDAF	
275C4EBD	8D45 FC	lea	eax, dword ptr [ebp-4]	此处地址存储的值为下次需要读取的位置
275C4EC0	8BCB	mov	ecx, ebx	
275C4EC2	50	push	eax	
275C4EC3	E8 8660FCFF	call	2758AF4E	
275C4EC8	8BF8	mov	edi, eax	
275C4ECA	85FF	test	edi, edi	
275C4ECC	7C 78	jl	short 275C4F46	
275C4ECE	8B45 FC	mov	eax, dword ptr [ebp-4]	

0012BD00	0012BD18			0012BCF8	0CF903B0
0012BD04	0012BE08			0012BCFC	08D45D98
0012BD08	2758AEE8	返回到	MSCOMCTL.2758AEE8	0012BD00	00000000
0012BD0C	08D45D98			0012BD04	0073006F
0012BD10	0CF903B0			0012BD08	00000000
0012BD14	00000000			0012BD0C	00720045

0012BD18	ABCDEF01			0012BCF8	0CF903B0
0012BD1C	00050000			0012BCFC	08D45D98
0012BD20	07853CA0			0012BD00	00000000
0012BD24	00650016			0012BD04	0073006F
0012BD28	FFFFFFFF			0012BD08	00000000
0012BD2C	FFFFFFFF			0012BD0C	00720045
0012BD30	00000000			0012BD10	000003E8
0012BD34	00000000			0012BD14	00730068
0012BD38	00000010			0012BD18	ABCDEF01
0012BD3C	00000000			0012BD1C	00050000
0012BD40	00000000			0012BD20	07853CA0
0012BD44	00000000			0012BD24	00650016
0012BD48	00000000			0012BD28	FFFFFFFF
0012BD4C	00000000			0012BD2C	FFFFFFFF
0012BD50	00000000			0012BD30	00000000
0012BD54	00000000			0012BD34	00000000
0012BD58	00000000			0012BD38	00000010
0012BD5C	00000000			0012BD3C	00000000
0012BD60	00000000			0012BD40	00000000
0012BD64	00000000			0012BD44	00000000
0012BD68	00000000			0012BD48	00000000
0012BD6C	00000000			0012BD4C	00000000
0012BD70	00000000			0012BD50	00000000
0012BD74	00000000			0012BD54	00000000
0012BD78	00000000			0012BD58	00000000
0012BD7C	00000000			0012BD5C	00000000
0012BD80	00000000			0012BD60	00000000
0012BD84	00000000			0012BD64	00000000
0012BD88	00000000			0012BD68	00000000
0012BD8C	00000000			0012BD6C	00000000
0012BD90	00000000			0012BD70	00000000
0012BD94	00000000			0012BD74	00000000
0012BD98	00000000			0012BD78	00000000
0012BD9C	00000000			0012BD7C	00000000
0012BDA0	00000000			0012BD80	00000000
0012BDA4	00000000			0012BD84	00000000
0012BDA8	00000000			0012BD88	00000000
0012BDAC	00000000			0012BD8C	00000000
0012BDB0	00000000			0012BD90	00000000
0012BDB4	00000000			0012BD94	00000000
0012BDB8	00000000			0012BD98	00000000
0012BDBC	00000000			0012BD9C	00000000
0012BDC0	00000000			0012BDA0	00000000
0012BDC4	00000000			0012BDA4	00000000
0012BDC8	00000000			0012BDA8	00000000
0012BDC	00000000			0012BDAC	00000000

将要处理的数据拷贝出来，并校验头部数据

2758AF63	59	pop	ecx	
2758AF64	8975 FC	mov	dword ptr [ebp-4], esi	
2758AF67	F3:A5	rep	movs dword ptr es:[edi], dword ptr [esi]	
2758AF69	817D E0 01EFC0	cmp	dword ptr [ebp-20], ABCDEF01	要处理的数据拷贝出来 比较头部是否是 0xABCDEF01
2758AF70	0F85 50890400	jnz	275D38C6	
2758AF76	8B45 E4	mov	eax, dword ptr [ebp-1C]	
2758AF79	66:85C0	test	ax, ax	
2758AF7C	0F85 44890400	jnz	275D38C6	
2758AF82	C1E8 10	shr	eax, 10	
2758AF85	66:3D 0500	cmp	ax, 5	
2758AF89	0F85 37890400	jnz	275D38C6	
2758AF8F	8D83 88010000	lea	eax, dword ptr [ebx+188]	
2758AF95	8D75 EC	lea	esi, dword ptr [ebp-14]	
2758AF98	8BF8	mov	edi, eax	
2758AF9A	A5	movs	dword ptr es:[edi], dword ptr [esi]	
2758AF9B	A5	movs	dword ptr es:[edi], dword ptr [esi]	
2758AF9C	A5	movs	dword ptr es:[edi], dword ptr [esi]	

比较 dwVersion, 并处理该结构, 与漏洞无关, 不细分析

2758AF67	F3:A5	rep	movs dword ptr es:[edi], dword ptr [esi]	要处理的数据拷贝出来 比较头部是否是 0xABCDEF01
2758AF69	817D E0 01EFC0	cmp	dword ptr [ebp-20], ABCDEF01	
2758AF70	0F85 50890400	jnz	275D38C6	
2758AF76	8B45 E4	mov	eax, dword ptr [ebp-1C]	dwVersion
2758AF79	66:85C0	test	ax, ax	
2758AF7C	0F85 44890400	jnz	275D38C6	
2758AF82	C1E8 10	shr	eax, 10	
2758AF85	66:3D 0500	cmp	ax, 5	
2758AF89	0F85 37890400	jnz	275D38C6	
2758AF8F	8D83 88010000	lea	eax, dword ptr [ebx+188]	
2758AF95	8D75 EC	lea	esi, dword ptr [ebp-14]	
2758AF98	8BF8	mov	edi, eax	
2758AF9A	A5	movs	dword ptr es:[edi], dword ptr [esi]	
2758AF9B	A5	movs	dword ptr es:[edi], dword ptr [esi]	
2758AF9C	A5	movs	dword ptr es:[edi], dword ptr [esi]	
2758AF9D	8B00	mov	eax, dword ptr [eax]	
2758AF9F	8D83 D4000000	lea	esi, dword ptr [ebx+D4]	
2758AFA5	C1E8 0B	shr	eax, 0B	
2758AFA8	83E0 03	and	eax, 3	
2758AFAB	8BCE	mov	ecx, esi	
2758AFAD	50	push	eax	
2758AFAE	E8 999DFFFF	call	27584D4C	接下来为处理该结构的一些函数
2758AFB3	8B83 88010000	mov	eax, dword ptr [ebx+188]	
2758AFB9	8BCE	mov	ecx, esi	
2758AFBB	C1E8 0D	shr	eax, 0D	
2758AFBE	83E0 03	and	eax, 3	
2758AFC1	50	push	eax	
2758AFC2	E8 C99DFFFF	call	27584D90	
2758AFC7	8B75 FC	mov	esi, dword ptr [ebp-4]	
2758AFCA	83C6 1C	add	esi, 1C	需要处理的数据指针加 0x1C
2758AFCD	F683 24010000	test	byte ptr [ebx+124], 12	

需要处理的数据指向下一个结构, 并保存起来, 接下来为

ImageListName (BSTR 类型)

2758AFBE	83E0 03	and	eax, 3	
2758AFC1	50	push	eax	
2758AFC2	E8 C99DFFFF	call	27584D90	
2758AFC7	8B75 FC	mov	esi, dword ptr [ebp-4]	
2758AFCA	83C6 1C	add	esi, 1C	需要处理的数据指针加 0x1C
2758AFCD	F683 24010000	test	byte ptr [ebx+124], 12	
2758AFD4	0F85 B0880400	jnz	275D388A	
2758AFDA	8B45 08	mov	eax, dword ptr [ebp+8]	
2758AFDD	808B 94010000	or	byte ptr [ebx+194], 6	
2758AFE4	8930	mov	dword ptr [eax], esi	esi为下面要处理的数据指针, 存储起来
2758AFE6	33C0	xor	eax, eax	
2758AFE8	5F	pop	edi	
2758AFE9	5E	pop	esi	
2758AFEA	5B	pop	ebx	
2758AFEB	C9	leave		
2758AFEC	C2 0400	retn	4	

BSTR 前面是一个 cchText 数值, 该数值表明 BSTR 的大小, 如果为 0 则表示该 ImageListName 不存在。当不存在时, 则无需申请内存存放该 BSTR, 跳转

275C4EC8	8BF8	mov	edi, eax	
275C4ECA	85FF	test	edi, edi	
275C4ECC	7C 78	jl	short 275C4F46	
275C4ECE	8B45 FC	mov	eax, dword ptr [ebp-4]	取出要处理的数据指针
275C4ED1	8B30	mov	esi, dword ptr [eax]	cchText
275C4ED3	8345 FC 04	add	dword ptr [ebp-4], 4	
275C4ED7	85F6	test	esi, esi	如果该值为0则表示ImageListName无值 所以也不申请内存存放字符串
275C4ED9	7E 3D	jle	short 275C4F18	
275C4EDB	56	push	esi	
275C4EDC	6A 00	push	0	
275C4EDE	FF15 24155827	call	dword ptr [<OLEAUT32.114>]	OLEAUT32.SysAllocStringLen
275C4EE4	8BF8	mov	edi, eax	
275C4EE6	85FF	test	edi, edi	
275C4EE8	897D F8	mov	dword ptr [ebp-8], edi	
275C4EEB	0F84 DE7E0000	je	275CCDCF	
275C4EF1	8D1436	lea	edx, dword ptr [esi+esi]	
275C4EF4	8B75 FC	mov	esi, dword ptr [ebp-4]	
275C4EF7	8BCA	mov	ecx, edx	
275C4EF9	FF75 F8	push	dword ptr [ebp-8]	
275C4EFC	8BC1	mov	eax, ecx	
275C4EFE	C1E9 02	shr	ecx, 2	
275C4F01	F3:A5	rep	movs dword ptr es:[edi], dword ptr [esi]	

之后开始解析 Ctab 结构数组, 在该数组的前面是 numofTabs, 表示该数组中有多少个 Ctab 结构, 如图该文件有 0x10 个 Ctab 结构





275859C7	53	push	edx	
275859C8	56	push	esi	
275859C9	57	push	edi	
275859CA	FF 75 10	push	dword ptr [ebp+10]	
275859CD	8BF1	mov	esi, ecx	
275859CF	E8 E6010000	call	275858BA	
275859D4	8B7D 08	mov	edi, dword ptr [ebp+8]	
275859D7	59	pop	ecx	
275859D8	85C0	test	eax, eax	
275859DA	75 7A	jnz	short 27585A56	
275859DC	FF75 0C	push	dword ptr [ebp+C]	
275859DF	E8 D6010000	call	275858BA	
275859E4	59	pop	ecx	
275859E5	6A 01	push	1	
275859E7	85C0	test	eax, eax	
275859E9	5B	pop	ebx	
275859EA	0F85 32010000	jnz	27585B22	
275859F0	8B46 28	mov	eax, dword ptr [esi+28]	MainObject偏移 0x28存放的是LatestCTabObject
275859F3	85C0	test	eax, eax	
275859F5	8947 2C	mov	dword ptr [edi+2C], eax	
275859F8	74 03	je	short 275859FD	
275859FA	8978 28	mov	dword ptr [eax+28], edi	
275859FD	8B46 3C	mov	eax, dword ptr [esi+3C]	
27585A00	40	inc	eax	
27585A01	8945 FC	mov	dword ptr [ebp-4], eax	
27585A04	837F 2C 00	cmp	dword ptr [edi+2C], 0	
27585A08	0F84 75010000	je	27585B83	
27585A0E	837F 28 00	cmp	dword ptr [edi+28], 0	
27585A12	75 03	jnz	short 27585A17	
27585A14	897E 28	mov	dword ptr [esi+28], edi	
ds:[08D45F8C]=00000000 eax=00000000				

08D45F64	2759E280	MSCOMCTL.2759E280	0012BCA0	00000000	
08D45F68	08D45F70		0012BCA4	08D45F64	
08D45F6C	08D45FAC		0012BCA8	00000010	
08D45F70	2758FB00	MSCOMCTL.2758FB00	0012BCAC	08D59050	
08D45F74	00000001		0012BCB0	08D58FF8	
08D45F78	00000009		0012BCB4	0012BCEC	
08D45F7C	00000000		0012BCB8	00000000	
08D45F80	2759E268	MSCOMCTL.2759E268	0012BCBC	00000000	
08D45F84	2759E248	MSCOMCTL.2759E248	0012BCC0	00000000	
08D45F88	00000000		0012BCC4	0012BCEC	
08D45F8C	00000000		0012BCC8	275C4FD3	返回到 MSCOMCTL.275C4FD3 来自
08D45F90	00000000		0012BC8C	08D58FF8	
08D45F94	00000000		0012BCC0	00000000	
08D45F98	00000000		0012BCD0	00000000	
08D45F9C	00000000		0012BCD4	00000000	

并将该值赋给新生成对象的 0x2C 处(pLastCTabObj)

275859C7	E8 E6010000	call	esi, ecx	
275859C8	8B7D 08	mov	edi, dword ptr [ebp+8]	
275859D7	59	pop	ecx	
275859D8	85C0	test	eax, eax	
275859DA	75 7A	jnz	short 27585A56	
275859DC	FF75 0C	push	dword ptr [ebp+C]	
275859DF	E8 D6010000	call	275858BA	
275859E4	59	pop	ecx	
275859E5	6A 01	push	1	
275859E7	85C0	test	eax, eax	
275859E9	5B	pop	ebx	
275859EA	0F85 32010000	jnz	27585B22	
275859F0	8B46 28	mov	eax, dword ptr [esi+28]	MainObject偏移 0x28存放的是LatestCTabObject
275859F3	85C0	test	eax, eax	
275859F5	8947 2C	mov	dword ptr [edi+2C], eax	存于新创建的对象偏移 0x2C处,最终形成双向链表
275859F8	74 03	je	short 275859FD	
275859FA	8978 28	mov	dword ptr [eax+28], edi	
275859FD	8B46 3C	mov	eax, dword ptr [esi+3C]	
27585A00	40	inc	eax	
27585A01	8945 FC	mov	dword ptr [ebp-4], eax	
27585A04	837F 2C 00	cmp	dword ptr [edi+2C], 0	
27585A08	0F84 75010000	je	27585B83	
27585A0E	837F 28 00	cmp	dword ptr [edi+28], 0	
27585A12	75 03	jnz	short 27585A17	
27585A14	897E 28	mov	dword ptr [esi+28], edi	
27585A17	85F6	test	esi, esi	
27585A19	8977 48	mov	dword ptr [edi+48], esi	
27585A1C	74 06	je	short 27585A24	
27585A1E	8B46 34	mov	eax, dword ptr [esi+34]	
27585A21	8947 50	mov	dword ptr [edi+50], eax	
27585A24	FF46 3C	inc	dword ptr [esi+3C]	
27585A27	804E 28	mov	ecx, dword ptr [esi+28]	
27585A2A	8B46 3C	mov	eax, dword ptr [esi+3C]	
27585A2D	FF46 40	inc	dword ptr [esi+40]	
27585A30	8941 4C	mov	dword ptr [ecx+4C], eax	
27585A33	8B4E 44	mov	ecx, dword ptr [esi+44]	
27585A36	8B55 FC	mov	edx, dword ptr [ebp-4]	
27585A39	897E 2C	mov	dword ptr [esi+2C], edi	
27585A3C	3BD1	cmp	edx, ecx	
27585A3E	7C 05	je	short 27585A45	
27585A40	83F9 FF	cmp	ecx, -1	
27585A43	75 08	jnz	short 27585A4D	
27585A45	3BD0	cmp	edx, eax	
27585A47	0E85 14010000	je	27585B8E	
跳转未实现 27585B8E=27585B8E				

如果之前已经有 CTabObj, 则将新生成的 CTabObj 对象指针赋给之前的 CTabObj 对象偏移 0x28 处。

275859F5	8947 2C	mov	dword ptr [edi+2C], eax	存于新创建的对象偏移 0x2C处,最终形成双向链表
275859F8	74 03	je	short 275859FD	
275859FA	8978 28	mov	dword ptr [eax+28], edi	将新生成的对象放于之前的CTab对象的偏移 0x28处
275859FD	8B46 3C	mov	eax, dword ptr [esi+3C]	取出MainObject偏移 0x3C的值,该值存储了一共有几个CTab结构
27585A00	40	inc	eax	
27585A01	8945 FC	mov	dword ptr [ebp-4], eax	
27585A04	837F 2C 00	cmp	dword ptr [edi+2C], 0	比较是否是第一个CTab结构
27585A08	0F84 75010000	je	27585B83	如果是则跳转
27585A0E	837F 28 00	cmp	dword ptr [edi+28], 0	
27585A12	75 03	jnz	short 27585A17	
27585A14	897E 28	mov	dword ptr [esi+28], edi	更新MainObj的LatestCTabObj指针
27585A17	85F6	test	esi, esi	
27585A19	8977 48	mov	dword ptr [edi+48], esi	将MainObj指针存储在CTab对象的0x48处
27585A1C	74 06	je	short 27585A24	
27585A1E	8B46 34	mov	eax, dword ptr [esi+34]	
27585A21	8947 50	mov	dword ptr [edi+50], eax	
27585A24	FF46 3C	inc	dword ptr [esi+3C]	CTabCount加1
27585A27	804E 28	mov	ecx, dword ptr [esi+28]	
27585A2A	8B46 3C	mov	eax, dword ptr [esi+3C]	
27585A2D	FF46 40	inc	dword ptr [esi+40]	
27585A30	8941 4C	mov	dword ptr [ecx+4C], eax	
27585A33	8B4E 44	mov	ecx, dword ptr [esi+44]	
27585A36	8B55 FC	mov	edx, dword ptr [ebp-4]	
27585A39	897E 2C	mov	dword ptr [esi+2C], edi	
27585A3C	3BD1	cmp	edx, ecx	
27585A3E	7C 05	je	short 27585A45	
27585A40	83F9 FF	cmp	ecx, -1	
27585A43	75 08	jnz	short 27585A4D	
27585A45	3BD0	cmp	edx, eax	
27585A47	0E85 14010000	je	27585B8E	
跳转未实现 27585B8E=27585B8E				

之后更新 MainObj 的 pLastestCTabObj 指针，并更新计数 (CTabCount)。在处理完 MainObj 之后，对数据进行读取，读取一个 Ctab 结构并将相关信息记录在 CTabObj 对象中

275C501D 53	push	ebx		
275C501E 55	push	ebp		
275C501F 8B09	mov	ebx, ecx		
275C5021 8B28	mov	ebp, dword ptr [eax]	取得CTabObj指针	
275C5023 56	push	esi		
275C5024 57	push	edi		
275C5025 8B45 00	mov	eax, dword ptr [ebp]		
275C5028 8B40 04	mov	ecx, dword ptr [ebp+4]		
275C502B 83C5 08	add	ebp, 8		
275C502E 894C24 18	mov	dword ptr [esp+18], ecx		
275C5032 F6C1 01	test	cl, 1		
275C5035 66:8943 64	mov	word ptr [ebx+64], ax		
275C5039 0F85 4F840000	jnz	275CD48E		
275C503F 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5043 D1E8	shr	eax, 1		
275C5045 A8 01	test	al, 1		
275C5047 0F85 87840000	jnz	275CD4D4		
275C504D 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5051 C1E8 02	shr	eax, 2		
275C5054 A8 01	test	al, 1		
275C5056 0F85 C6840000	jnz	275CD522		
275C505C 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5060 C1E8 03	shr	eax, 3		
275C5063 A8 01	test	al, 1		
275C5065 0F85 05850000	jnz	275CD570		
275C506B 8D73 70	lea	esi, dword ptr [ebx+70]		
275C506E 56	push	esi		
275C506F FF15 48155827	call	dword ptr [<OLEAUT32.09>]	OLEAUT32.VariantClear	
275C5075 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5079 C1E8 04	shr	eax, 4		
275C507E A8 01	test	al, 1		
275C5080 0F85 36850000	jnz	275CD58A		
275C5084 66:C706 0200	mov	word ptr [esi], 2		
275C5089 66:8B45 00	mov	ax, word ptr [ebp]		
275C508D 45	inc	ebp		
275C5090 66:8943 78	mov	word ptr [ebx+78], ax		

第二个 DWORD 是一个按位运算的值，分别表示其后面是否存在一些 BSTR 字符串，其第一位表示是否存在 Caption，第二位表示是否存在 Key，第三位表示 Tag，第四位表示是否存在 ToolTipText，第五位如果存在则为 ImageKey (BSTR 类型)，如果不存在则为一个 SHORT 类型的值

分别按位运算各个位的值，以确定后面的 BSTR 类型是否存在

275C502E 894C24 18	mov	dword ptr [esp+18], ecx		
275C5032 F6C1 01	test	cl, 1		
275C5035 66:8943 64	mov	word ptr [ebx+64], ax		
275C5039 0F85 4F840000	jnz	275CD48E		
275C503F 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5043 D1E8	shr	eax, 1		
275C5045 A8 01	test	al, 1	接下来对pph进行按位运算 比较第一位是否为1	
275C5047 0F85 87840000	jnz	275CD4D4		
275C504D 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5051 C1E8 02	shr	eax, 2		
275C5054 A8 01	test	al, 1	比较第二位是否为1	
275C5056 0F85 C6840000	jnz	275CD522		
275C505C 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5060 C1E8 03	shr	eax, 3		
275C5063 A8 01	test	al, 1	比较第三位是否为1	
275C5065 0F85 05850000	jnz	275CD570		
275C506B 8D73 70	lea	esi, dword ptr [ebx+70]		
275C506E 56	push	esi		
275C506F FF15 48155827	call	dword ptr [<OLEAUT32.09>]	OLEAUT32.VariantClear	
275C5075 8B4424 18	mov	eax, dword ptr [esp+18]		
275C5079 C1E8 04	shr	eax, 4		
275C507E A8 01	test	al, 1		
275C5080 0F85 36850000	jnz	275CD58A		
275C5084 66:C706 0200	mov	word ptr [esi], 2		
275C5089 66:8B45 00	mov	ax, word ptr [ebp]		
275C508D 45	inc	ebp		
275C5090 66:8943 78	mov	word ptr [ebx+78], ax		

最后将指针指向下一个 Ctab 结构，并保存起来



275C05C	8B424 18	mov	eax, dword ptr [esp+18]	
275C060	C1E8 03	shr	eax, 3	
275C063	A8 01	test	al, 1	比较第三位是否为1,是否存在fTag
275C065	0F85 05850000	jnz	275C0570	
275C068	8D73 70	lea	esi, dword ptr [ebx+70]	
275C06E	56	push	esi	
275C06F	FF15 8E155827	call	dword ptr [<OLEAUT32.VariantClear>]	
275C075	8B424 18	mov	eax, dword ptr [esp+18]	
275C079	C1E8 04	shr	eax, 4	比较第四位是否为1,是否存在ToolTipText
275C07C	A8 01	test	al, 1	
275C07E	0F85 36850000	jnz	275C0580	
275C084	66:0706 0200	mov	word ptr [esi], 2	
275C089	66:8845 00	mov	ax, word ptr [ebp]	
275C08D	45	inc	ebp	
275C08E	66:8943 78	mov	word ptr [ebx+78], ax	
275C092	45	inc	ebp	
275C093	8B424 20	mov	eax, dword ptr [esp+20]	
275C097	8928	mov	dword ptr [eax], ebp	下面一个CTab结构
275C099	33C8	xor	eax, eax	
275C09B	5F	pop	edi	
275C09C	5E	pop	esi	
275C09D	5D	pop	ebp	
275C09E	5B	pop	ebx	
275C09F	83C4 0C	add	esp, 0C	
275C0A2	C2 0400	ret	4	
275C0A5	830C24 04 34	sub	dword ptr [esp+4], 34	
275C0A8	E9 F10400FF	jmp	275C25A0	
275C0AB	830C24 04 04	sub	dword ptr [esp+4], 4	
ebp=0012B064 堆栈 05:[0012BCE8]-0012B058				
0012B05C	00000000		0012B080	00000000
0012B060	00000000		0012B084	000A5F64
0012B066	00000000		0012B088	0012B0EC
0012B06B	0012B068		0012B08C	00000010

处理完一个之后，比较返回值是否小于 0，如果小于 0 则代表失败。

275C4FC	E8 EA09FCFF	call	<HandleMainObject>	下面对于该对象进行一些处理
275C4FD3	8BF8	mov	edi, eax	
275C4FD5	85FF	test	edi, edi	
275C4FD7	0F8C 9A890000	j1	275C0977	
275C4FD0	8B4D F8	mov	ecx, dword ptr [ebp-8]	
275C4FE0	8D45 FC	lea	eax, dword ptr [ebp-4]	
275C4FC3	59	push	eax	
275C4FE4	E8 2D000000	call	<HandleCTabObj>	接下来处理CTabObj
275C4FE9	8BF8	mov	edi, eax	检测返回值,是否成功
275C4FEB	85FF	test	edi, edi	
275C4FED	0F8C 84890000	j1	275C0977	小于0则代表失败
275C4FF3	FF45 F4	inc	dword ptr [ebp-C]	计数器,记录已经处理了几个CTab结构
275C4FF6	395D F4	cmp	dword ptr [ebp-C], ebx	比较是否已经处理完毕,是否达到了numofTabs,如果没有处理完继续处理下一个结构
275C4FF9	7C 9C	j1	short 275C4F97	
275C4FFB	8B4D 08	mov	ecx, dword ptr [ebp+8]	
275C4FFE	8B45 FC	mov	eax, dword ptr [ebp-4]	
275C5001	33FF	xor	edi, edi	
275C5003	85FF	test	edi, edi	
275C5005	8901	mov	dword ptr [ecx], eax	
275C5007	0F8C 6A890000	j1	275C0977	
275C500D	8BC7	mov	eax, edi	

接下来比较是否已经处理完了 CTabObj 数组，（本次数组中一共有 16 个 CTabObj 结构）

275CAFE0	8D45 FC	lea	eax, dword ptr [ebp-4]	
275CAFE3	50	push	eax	
275CAFE4	E8 2D000000	call	<HandleCTabObj>	接下来处理CTabObj
275CAFE9	8BF8	mov	edi, eax	
275CAFEB	85FF	test	edi, edi	
275CAFEF	0F8C 84890000	j1	275C0977	
275CAFF3	FF45 F4	inc	dword ptr [ebp-C]	计数器,记录已经处理了几个CTab结构
275CAFF6	395D F4	cmp	dword ptr [ebp-C], ebx	比较是否已经处理完毕,是否达到了numofTabs,如果没有处理完继续处理下一个结构
275CAFF9	7C 9C	j1	short 275C4F97	
275CAFFB	8B4D 08	mov	ecx, dword ptr [ebp+8]	
275CAFFE	8D45 FC	mov	eax, dword ptr [ebp-4]	
275C5001	33FF	xor	edi, edi	
275C5003	85FF	test	edi, edi	
275C5005	8901	mov	dword ptr [ecx], eax	
275C5007	0F8C 6A890000	j1	275C0977	
275C500D	8BC7	mov	eax, edi	
275C500F	5F	pop	edi	
275C5010	5E	pop	esi	
275C5011	5B	pop	ebx	

由于 numofTabs 明显大于 Ctab 大小，因此在读取了 3 个 Ctab 结构之后，读取越界了。如图，明显已经越过了开始读取的 0x60 个字节了。

275C5016	83EC 0C	sub	esp, 0C	
275C5019	8B424 10	mov	eax, dword ptr [esp+10]	
275C501D	53	push	ebx	
275C501E	55	push	ebp	
275C501F	8BD9	mov	ebx, ecx	
275C5021	8B28	mov	ebp, dword ptr [eax]	取得CTabObj指针
275C5023	54	push	esi	ebp处的值已经不是开始读取的数据了,CTab结构处理越界了
275C5024	57	push	edi	
275C5025	8B45 00	mov	eax, dword ptr [ebp]	
275C5028	8B4D 04	mov	ecx, dword ptr [ebp+4]	取出其值,第一个DWORD低0位为TabState,高0位为unused
275C502B	83C5 08	add	ebp, 8	第二个DWORD为pph,它是一个按位表示的值,分别表示后面的一些BSTR数据是否存在
275C502E	89AC24 18	mov	dword ptr [esp+18], ecx	
275C5032	F6C1 01	test	cl, 1	
275C5035	66:8943 64	mov	word ptr [ebx+64], ax	
275C5039	0F85 AF8A0000	jnz	275C0A8E	
275C503F	8B424 18	mov	eax, dword ptr [esp+18]	
275C5043	D1E8	shr	eax, 1	接下来对pph进行按位运算
275C5045	A8 01	test	al, 1	比较第一位是否为1,是否存在Caption
275C5047	0F85 878A0000	jnz	275CD4D4	
275C504D	8B424 18	mov	eax, dword ptr [esp+18]	
275C5051	C1E8 02	shr	eax, 2	
275C5054	A8 01	test	al, 1	比较第二位是否为1,是否存在fKey
275C5056	0F85 C68A0000	jnz	275CD522	
275C505C	8B424 18	mov	eax, dword ptr [esp+18]	
275C5060	C1E8 03	shr	eax, 3	
275C5063	A8 01	test	al, 1	比较第三位是否为1,是否存在fTag
275C5065	0F85 05850000	jnz	275CD570	
275C5068	8D73 70	lea	esi, dword ptr [ebx+70]	
275C506E	56	push	esi	
275C506F	FF15 8E155827	call	dword ptr [<OLEAUT32.VariantClear>]	
esi=000A5F64				
0012B064	00000000		0012B080	0012B0EC
0012B06B	0012B068		0012B08C	00000010

接下来继续把错误的栈内存当作 Ctab 结构进行解析，这里正好假的 pph 第三位为 1，于是跳转，将 pph 下面的 DWORD 值当作 BSTR 的 cchText 了。也就是 BSTR 的大小。

275CD570	8B7D 00	mov	edi, dword ptr [ebp]		
275CD573	8B83 84000000	mov	eax, dword ptr [ebx+84]		
275CD579	8B83 84000000	lea	esi, dword ptr [ebx+84]		
275CD57F	83C5 04	add	ebp, 4		
275CD582	85C0	test	eax, eax		
275CD584	74 07	je	short 275CD58D		
275CD586	50	push	eax		
275CD587	FF15 40155827	call	dword ptr [<OLEAUT32.#6>]	OLEAUT32.SysFreeString	
275CD58D	57	push	edi		
275CD58E	6A 00	push	0		
275CD590	FF15 24155827	call	dword ptr [<OLEAUT32.#4>]	OLEAUT32.SysAllocStringLen	
275CD596	85C0	test	eax, eax		
275CD598	8906	mov	dword ptr [esi], eax		
275CD59A	74 39	je	short 275CD5D5		

而申请这么大的字符串内存肯定会出错。

275CD57F	83C5 04	add	ebp, 4		
275CD582	85C0	test	eax, eax		
275CD584	74 07	je	short 275CD58D		
275CD586	50	push	eax		
275CD587	FF15 40155827	call	dword ptr [<OLEAUT32.#6>]	OLEAUT32.SysFreeString	
275CD58D	57	push	edi	edi=0x2758FA0F, 申请这么大的BSTR字符串肯定会出错	
275CD58E	6A 00	push	0		
275CD590	FF15 24155827	call	dword ptr [<OLEAUT32.#4>]	OLEAUT32.SysAllocStringLen	
275CD596	85C0	test	eax, eax		
275CD598	8906	mov	dword ptr [esi], eax		
275CD59A	74 39	je	short 275CD5D5		
275CD59C	8D143F	lea	edx, dword ptr [edi+edi]		
275CD59F	8BF8	mov	edi, eax		
275CD5A1	8BCA	mov	ecx, edx		
275CD5A3	8BF5	mov	esi, ebp		

于是申请内存出错, 返回 0。

275CD57F	83C5 04	add	ebp, 4		
275CD582	85C0	test	eax, eax		
275CD584	74 07	je	short 275CD58D		
275CD586	50	push	eax		
275CD587	FF15 40155827	call	dword ptr [<OLEAUT32.#6>]	OLEAUT32.SysFreeString	
275CD58D	57	push	edi	edi=0x2758FA0F, 申请这么大的BSTR字符串肯定会出错	
275CD58E	6A 00	push	0		
275CD590	FF15 24155827	call	dword ptr [<OLEAUT32.#4>]	OLEAUT32.SysAllocStringLen	
275CD596	85C0	test	eax, eax		
275CD598	8906	mov	dword ptr [esi], eax	eax返回0, 申请失败	
275CD59A	74 39	je	short 275CD5D5		
275CD59C	8D143F	lea	edx, dword ptr [edi+edi]		
275CD59F	8BF8	mov	edi, eax		
275CD5A1	8BCA	mov	ecx, edx		
275CD5A3	8BF5	mov	esi, ebp		
275CD5A5	8BC1	mov	eax, ecx		
275CD5A7	C1E9 02	shr	ecx, 2		
275CD5A8	F305	rep	movs dword ptr es:[edi], dword ptr [esi]		
275CD5AA	8BC6	mov	ecx, eax		
275CD5AC	83E1 03	and	ecx, 3		
275CD5AE	8BC6	mov	ecx, eax		

最终导致 HandCtabObj 函数返回失败。跳转

275C4FE0	8D45 FC	lea	eax, dword ptr [ebp-4]		
275C4FE3	50	push	eax		
275C4FE4	E8 2D000000	call	<HandleCtabObj>	接下来处理CTabObj	
275C4FE9	8BF8	mov	edi, eax	检测返回值, 是否成功	
275C4FEB	85FF	test	edi, edi		
275C4FED	0F8C 84890000	j1	275CD977	小于0则代表失败	
275C4FF3	FF45 F4	inc	dword ptr [ebp-C]	计数器, 记录已经处理了几个CTab结构	
275C4FF6	395D F4	cmp	dword ptr [ebp-C], ebx	比较是否已经处理完毕, 是否达到了numofTabs, 如果没有处理完继续	
275C4FF9	7C 9C	j1	short 275C4F97		
275C4FFB	8B4D 08	mov	ecx, dword ptr [ebp+8]		
275C4FFE	8B45 FC	mov	eax, dword ptr [ebp-4]		
275C5001	33FF	xor	edi, edi		
275C5003	85FF	test	edi, edi		

出错后, 将出错的 CTabObj 对象释放了。

275CD977	8B45 F8	mov	eax, dword ptr [ebp-8]	
275CD97A	85C0	test	eax, eax	
275CD97C	0F84 8B76FFFF	je	275C500D	
275CD982	8B48 1C	mov	ecx, dword ptr [eax+1C]	
275CD985	83C0 1C	add	eax, 1C	
275CD988	50	push	eax	
275CD989	FF51 08	call	dword ptr [ecx+8]	
275CD98C	E9 7C76FFFF	jmp	275C500D	
275CD991	B8 0E000780	mov	eax, 8007000E	
275CD996	E9 7476FFFF	jmp	275C500F	
275CD998	33DB	xor	ebx, ebx	
275CD99D	E9 DE4EFFFF	jmp	275C2880	
275CD9A2	8B46 B8	mov	eax, dword ptr [esi-48]	
275CD9A5	8D4E B8	lea	ecx, dword ptr [esi-48]	
275CD9A8	6A 07	push	7	
275CD9AA	6A 07	push	7	
275CD9AC	68 07000A80	push	800A0007	
275CD9B1	FF50 30	call	dword ptr [eax+30]	
275CD9B4	E9 724FFFFF	jmp	275C292B	
275CD9B9	33C0	xor	eax, eax	
275CD9BB	E9 5AA8FBFF	jmp	2758821A	
275CD9C0	8320 00	and	dword ptr [eax], 0	
275CD9C3	8B01	mov	eax, dword ptr [ecx]	
275CD9C5	51	push	ecx	
275CD9C6	FF50 08	call	dword ptr [eax+8]	
275CD9C9	E9 4170FCFF	jmp	27594A0F	
275CD9CE	8320 00	and	dword ptr [eax], 0	
275CD9D1	8B01	mov	eax, dword ptr [ecx]	
275CD9D3	51	push	ecx	
275CD9D4	FF50 08	call	dword ptr [eax+8]	
eax=00219228				
00219228	2759E3F8	MSCOMCTL.2759E3F8		001:
0021922C	00219234			001:
00219230	00219280			001:
00219234	2758FB00	MSCOMCTL.2758FB00		001:
00219238	00000001			001:
0021923C	0000000A			001:
00219240	00000000			001:
00219244	2759E3E8	MSCOMCTL.2759E3E8		001:
00219248	2759E3D8	MSCOMCTL.2759E3D8		001:
0021924C	2759E3B8	MSCOMCTL.2759E3B8		001:
00219250	00000000			001:
00219254	00000000			001:



275CD977	8B45 F8	mov	eax, dword ptr [ebp-8]	
275CD97A	85C0	test	eax, eax	
275CD97C	0F84 8B76FFFF	je	275C500D	
275CD982	8B48 1C	mov	ecx, dword ptr [eax+1C]	
275CD985	83C0 1C	add	eax, 1C	
275CD988	50	push	eax	
275CD989	FF51 08	call	dword ptr [ecx+8]	Release
275CD98C	E9 7C76FFFF	jmp	275C500D	
275CD991	B8 0E000780	mov	eax, 8007000E	
275CD996	E9 7476FFFF	jmp	275C500F	
275CD99B	33DB	xor	ebx, ebx	
275CD99D	E9 DE4EFFFF	jmp	275C2880	
275CD9A2	8B46 B8	mov	eax, dword ptr [esi-48]	
275CD9A5	8D4E B8	lea	ecx, dword ptr [esi-48]	
275CD9A8	6A 07	push	7	
275CD9AA	6A 07	push	7	
275CD9AC	68 07000A80	push	800A0007	
275CD9B1	FF50 30	call	dword ptr [eax+30]	
275CD9B4	E9 724FFFFF	jmp	275C292B	
275CD9B9	33C0	xor	eax, eax	
275CD9BB	E9 5AA8FBFF	jmp	2758821A	
275CD9C0	8320 00	and	dword ptr [eax], 0	
275CD9C3	8B01	mov	eax, dword ptr [ecx]	
275CD9C5	51	push	ecx	
275CD9C6	FF50 08	call	dword ptr [eax+8]	
275CD9C9	E9 4170FCFF	jmp	27594A0F	
275CD9CE	8320 00	and	dword ptr [eax], 0	
275CD9D1	8B01	mov	eax, dword ptr [ecx]	
275CD9D3	51	push	ecx	
275CD9D4	FF50 08	call	dword ptr [eax+8]	

00219228	00000000		00128CD4	00000000
0021922C	00219234		00128CD8	00000000
00219230	00219280		00128CDC	00D45D98
00219234	2758FB00	MSCOMCTL.2758FB00	00128CE0	00000004
00219238	00000000		00128CE4	00219228
0021923C	0000000A		00128CE8	0012BD64
00219240	00000000		00128CEC	0012BD84
00219244	27590C30	MSCOMCTL.27590C30	00128CF0	275C4F27
00219248	27590C20	MSCOMCTL.27590C20	00128CF4	0012BDB0
0021924C	27590C00	MSCOMCTL.27590C00	00128CF8	0CF903B0
00219250	00000000		00128CFC	00D45D98
00219254	00D59090		0012BD00	00000000

而在释放该 CTabObj 之后，并未更新双向链表中的节点。函数便直接返回！

275C5005	8901	mov	eax, ecx	
275C5007	0F8C 6A890000	ja	dword ptr [ecx], eax	
275C500D	80C7	mov	eax, edi	
275C500F	5F	pop	edi	
275C5010	5E	pop	esi	
275C5011	5B	pop	ebx	
275C5012	C9	leave		
275C5013	C2 0400	ret	4	

而在释放该CTabObj结构以后并未对双向链表进行处理便直接返回

导致后面在根据该双向链表访问各个 CTabObj 对象的时候必然会出错！

2758FC03	58	pop	ebx	
2758FC04	C3	ret		
2758FC05	57	push	edi	
2758FC06	8078 28	mov	edi, dword ptr [eax+28]	
2758FC09	8048 1C	mov	ecx, dword ptr [eax+1C]	
2758FC0C	8958 48	mov	dword ptr [eax+48], ebx	
2758FC0D	80C0 1C	add	eax, 1C	
2758FC12	50	push	eax	
2758FC13	FF51 08	call	dword ptr [ecx+8]	
2758FC16	30F0	cmp	edi, ebx	
2758FC18	80C7	mov	eax, edi	
2758FC1A	75 EA	jnz	short 2758FC06	
2758FC1C	5F	pop	edi	
2758FC1D	E8 D1	jmp	short 2758FCC0	
2758FCE0	56	push	esi	
2758FCE1	57	push	edi	
2758FCE2	80F9	mov	edi, ecx	
2758FCE3	8077 30	mov	esi, dword ptr [edi+30]	
2758FCE6	85F6	test	esi, esi	
2758FCE8	75 03	jnz	short 2758FCF0	
2758FCFA	5F	pop	edi	
2758FCFB	5E	pop	esi	
2758FCFC	C3	ret		
2758FCFD	FF36	push	dword ptr [esi]	
2758FCFF	80CE	mov	ecx, esi	
2758FD01	E8 545A0000	call	2759575A	
2758FD06	56	push	esi	
2758FD07	E8 FC31FFFF	call	27582F08	
2758FD0C	8367 30 00	and	dword ptr [edi+30], 0	
2758FD10	C3	ret		

后面在通过链表逐个访问CTabObj的时候则会由于最后一个CTabObj已经被释放而出错！

00219228	00000000		0012DFC0	00D45F64
0021922C	00740066		0012DFC4	00D45F64
00219230	00610077		0012DFC8	00000000
00219234	00650072		0012DFCC	27590BDC
00219238	0043005C		0012DFD0	00D45D98
0021923C	0061006C		0012DFD4	00D45D98
00219240	00730073		0012DFD8	2759E77C
00219244	00730065		0012DFDC	00D45F6C
00219248	0049005C		0012DFE0	00D45D94
0021924C	0073006E		0012DFE4	27582561
00219250	00610074		0012DFE8	00CCDE00
00219254	006C006C		0012DFEC	00CCDE8C
00219258	00720065		0012DFE4	275825B9

返回到 MSCOMCTL.27590BDC 来自 MSCOMCTL.2758FCAE

返回到 MSCOMCTL.2759E77C

返回到 MSCOMCTL.27582561

返回到 MSCOMCTL.275825B9

2758FC05	3f	push	eax		EDX
2758FC06	8078 20	mov	edi, dword ptr [eax+28]		EDX
2758FC09	8048 1c	mov	ecx, dword ptr [eax+1c]		EBX
2758FC0C	8058 40	mov	dword ptr [eax+48], ebx		ESP
2758FC0F	83c0 1c	add	eax, 1c		EBX
2758FC22	50	push	eax		ESI
2758FC23	FF51 08	call	dword ptr [ecx+8]	后面在通过链表逐个访问CTabObj的时候则会由于最后一个CTabObj已经被释放而出错！	EDI
2758FC26	3BF8	cmp	edx, ebx		EIP
2758FC2B	80C7	mov	eak edi		C 0
2758FC2E	75 EA	jnz	short 2758FC06		P 1
2758FC2E	5F	pop	edi		A 1
2758FC2D	EB D1	jnp	short 2758FC00		Z 0
2758FC2F	56	push	esi		S 0
2758FC30	57	push	edi		T 0
2758FC31	8BF0	mov	edi, ecx		D 0
2758FC33	8077 30	mov	esi, dword ptr [edi+30]		D 0
2758FC36	85F6	test	esi, esi		0 0
2758FC38	75 03	jnz	short 2758FCF0		EFL
2758FC3A	5F	pop	edi		ST0
2758FC3B	5E	pop	esi		ST1
2758FC3C	C3	ret			ST2
2758FC3D	FF36	push	dword ptr [esi]		ST3
2758FC3F	80CE	mov	ecx, esi		ST4
2758FC41	E8 54500000	call	27595750		ST5
2758FC46	56	push	esi		ST6
2758FD07	E8 FC31FFFF	call	27582F08		ST7
2758FD0C	8367 30 00	and	dword ptr [edi+30], 0		FST
2758FD13	C3	ret			rev
05:[00730000]=??? 05:[00730000]=???					