CVE-2010-3962 分析

相关代码参考 win2k: win2k\private\inet\mshtml\src\site\download\

问题代码为:<table style=position:absolute;clip:rect(0)>

html 解析器(CHtmParse::ParseEof)在解析到 html 尾部(最后一个括号)时发现当前结点(table 结点)还需要一个隐式的_etagTextSubcontainer 对象，这里由 Table 的 HtmParseClass 决定_etagTextSubcontainer 到底是什么，具体见下。

```
// TABLE

static CHtmlParseClass s_hpcTable =
{
    SCOPE_NESTED,                   // _scope
    TEXTTYPE_ALWAYS,                // _texttype                    ; textlike; force BODY etc
    s_atagTable,                    // _atagEndContainers
    s_atagTableCellCaption,         // _atagBeginContainers         ; allow nesting inside TD, CAPTION, TABLE
    NULL,                           // _atagMaskingContainers
    s_atagTableCloses,              // _atagProhibitedContainers    ; close prevous TABLE, TC, SELECT, OPTION
    NULL,                           // _atagRequiredContainers
    ETAG_NULL,                      // _etagDefaultContainer
    FALSE,                          // _fQueueForRequired
    TEXTSCOPE_EXCLUDE,              // _textscope                   ; exclude text
    ETAG_TC,                        // _etagTextSubcontainer        ; wrap contained text in a TC
    NULL,                           // _atagMatch
    ETAG_NULL,                      // _etagUnmatchedSubstitute
    NULL,                           // _pfnHpxCreator
    FALSE,                          // _fMerge
    ETAG_NULL,                      // _etagImplicitChild
    FALSE,                          // _fCloseImplicitChild
};
```

所需 etagTextSubcontainer 对象名为 CTableCaption、类型为 ETAG_TC（标题类型）、值为 0x62，随后 CTableCaption 对象被创建之.

CTableLayout 管理器会负责管理要显示的 table，并把上面创建的 CTableCaption 对象加到标题数组中去。

html 解析器随后会通知各 HTML 元素刷新,table 标签中的 CTableCaption 对象也会刷新

显示。

CTableLayout 检查到要显示的结点中含有标题对象时，会检查该标题对象是否有显示结点容器 CDispContainer，当不存在时需为该标题对象创建 CDispContainer 类型的显示容器对象，同时由于 table 标签具有 style=position:absolute;clip:rect(0)，因而需要对显示的结点进行裁剪，这些信息也会保存到 CDispContainer 对象中，以供显示之用，不巧的是 CDispContainer 是一个变体大小的对象，最小有 0x48 字节，附加大小是由一张静态表控制：Sizeof(CDispContainer) = _extraSizeTable[index]*4+0x48，_extraSizeTable[index]*4 大小的数据用来存放裁剪信息，CDispContainer 的基本结构为：

CliprectInfo|VT|CliprectInfo_size|.....

带标题属性而没有 CDispContainer 结点的结果导致动态生成 index 的值为 0,这样新生成的对象会只有 0x48 大小而 CliprectInfo 为空,那么在设置 CliprectInfo 时导致 VT 表直接被修改.

```
.text:7E2C5D9D                                                  ;
.text:7E2C5D9D                  push    eax
.text:7E2C5D9E                  lea     eax, [ebx+0Ch]
.text:7E2C5DA1                  push    eax
.text:7E2C5DA2                  call    CDispContainer::New(CDispClient *,ulong)

*******************************************************
.text:7E2B5D2A  CDispContainer::New
.............................................................
.text:7E291CF0      movzx   esi, ds:uchar const * const CDispNode::_extraSizeTable[ebx]
.text:7E291CF7              shl     esi, 2
.text:7E291CFA              push    edi
.text:7E291CFB              add     eax, esi
.text:7E291CFD              push    eax                 ; dwBytes
.text:7E291CFE              call    _MemAllocClear(x)
.text:7E291D03              mov     edi, eax
.text:7E291D05              test    edi, edi
.text:7E291D07              jz      short loc_7E291D17
.text:7E291D09              add     edi, esi
.text:7E291D0B              test    bl, 40h
.text:7E291D0E            mov       [edi+4], ebx   <-------这里保存 CliprectInfo_size
的大小

*******************************************************
.text:7E36B4C4  CDispNode::SetUserClip
.................................
.text:7E36B54D              mov     eax, [edi+4]   <-------当这里为 0 时,虚函数表就
被悲剧了.
.text:7E36B550              and     eax, esi
```

```
.text:7E36B552          movzx   ecx, ds:uchar const * const CDispNode::_extraSizeTable[eax]
.text:7E36B559                  mov     eax, edi
.text:7E36B55B                  shl     ecx, 2
.text:7E36B55E                  sub     eax, ecx
.text:7E36B560                  or      dword ptr [eax], 1      <----------虚函数表被改了
```

<table style=position:absolute;clip:rect(0)>改为

<table style=position:absolute;clip:rect(0)><td>后将导致 CTableCaption 对象不会生成,因为 td 的 CHtmlParseClass 根本没有_etagTextSubcontainer，这样不会导致代码出问题,到底是 s_hpcTable 出了问题还是 SetUserClip 检查不严格?还是两者都有问题?

函数调用序列：

CHtmParse::ParseEof

CHtmParse::ParseText

|

```
CHtmlParseClass *phpc;

phpc = HpcFromEtag(etag);
if (phpc->_atagProhibitedContainers)
{
    hr=THR(CloseAllContainers(phpc->_atagProhibitedContainers,
                phpc->_atagBeginContainers));
    if (hr)
        goto Cleanup;
}

hr = THR(OpenContainer(etag));
```

CHtmParse::OpenContainer

|

此时 etagItem=0x62 表示 ETAG_TC,这时将创建一个 CTableCaption 对象

hr = THR(CreateElement(etagItem, &pel, _pDoc, _pMarkup, TRUE, &_fDie));

CreateElement 函数根据 etagItem=0x62 去索引 hash 表 g_atagdesc,查表得具体的对象的 CreateElement 函数接着 CTableCaption::CreateElement 函数将被调用

CHtmPos::Exec

CHtmPos::Notify

CTableCell:Notify(CTableCaption 继承该类)

CTableCell::EnterTree
|

hr = pTableLayout->AddCaption(pCaption);导致　pTableLayout 对象中存在标题

CTableLayoutBlock::EnsureTableDispNode(CTableLayoutBlock 即上面的 pTableLayout)
CDispNode::SetUserClip
|

```
movzx      ecx, ds:uchar const * const CDispNode::_extraSizeTable[eax]
mov         eax, edi
shl         ecx, 2
sub         eax, ecx
or          dword ptr [eax], 1
```

当对 CliprectInfo 操作时虚函数表在这里活生生成被修改了一位，后续对对象的调用都会导致访问异常。