

Forum:[Hack Forums](#)

Thread: [changing value addresses in dll?](#)

wat iz dat - August 1, 2014, 6:21 pm

Well. I cant figure one thing. For example: i've got an dll injector (injecting dlls from kernel mode - its public but it should do its job) and i want to inject my dll - it'd be a simple hack for game. There is the stair i need to beat. The game is a multi mod for GTA:San Andreas (mta:sa). I've already got the addresses. The value is float. Address is 09476DAC. I know how to change the value but i have no clue how to do it when the DLL is injected (when it stars...)

Can any1 help me? Thanks!

Tidas - August 1, 2014, 6:52 pm

Use [WriteProcessMemory](#).

There are some videos and snippets online if you need more info.

webinject - August 1, 2014, 7:53 pm

Vouch for op. good luck with sales and what not bro.

(Aug 1, 2014 06:52 PM)Tidas Wrote: [\[CITE\]](#) Use [WriteProcessMemory](#).

There are some videos and snippets online if you need more info. Write process memory isn't necessary once hes in the address space of the injectee.

He can easily dereference the pointer or use memcpy.

Dickyb0bSue - August 2, 2014, 5:32 am

Yeh it can be done externally, you dont need to inject a dll

wat iz dat - August 2, 2014, 6:51 am

Well, because of the AC i need to. :x

dberube - August 2, 2014, 7:26 am

A dll contains a dllmain function that is called when the dll is loaded or removed from a process. From this, call a function that deference the value found at your address and change this value. If you need to add some interactivity, add an infinite loop and use getasynkeystate to trigger the changes.

```
Code
BOOL WINAPI DllMain(HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    switch (fdwReason){
        case DLL_PROCESS_ATTACH:
            //Procedure to call here.
    }
    return TRUE;
}
```

If you want an unnecessary complicated example, I created this:

<https://hackforums.net/showthread.php?tid=4356517>

Irritable - August 2, 2014, 5:19 pm

(Aug 1, 2014 06:21 PM)monumento Wrote: [\[CITE\]](#) Well. I cant figure one thing. For example: i've got an dll injector (injecting dlls from kernel mode - its public but it should do its job) and i want to inject my dll - it'd be a simple hack for game. There is the stair i need to beat. The game is a multi mod for GTA:San Andreas (mta:sa). I've already got the addresses. The value is float. Address is 09476DAC. I know how to change the value but i have no clue how to do it when the DLL is injected (when it stars...)

Can any1 help me? Thanks!

In short: WriteMemoryProcess() function.

In long: <http://www.rohitab.com/discuss/topic/35537-cc-reverse-engineering-tutorial-for-newbies/>

Read that tutorial. It'll answer ALL your questions.

Dispersia - August 2, 2014, 5:48 pm

He's already in the memory. You don't need to externally change it internally, that's useless, just simply do something such as `Quote:*(float*)0x09476DAC = 1.0f;` in dllmain, since this isn't an intensive function I doubt you need to start a new thread for it.

0xDEAD10CC - August 2, 2014, 7:44 pm

(Aug 2, 2014 05:48 PM)Dispersia Wrote: [\[CITE\]](#) He's already in the memory. You don't need to externally change it internally, that's useless, just simply do something such as in dllmain, since this isn't an intensive function I doubt you need to start a new thread for it.

No... When you load the DLL that will change the previous address of the float in memory most likely. If he would've just made a call to WriteProcessMemory, he could change memory without injecting any DLL perhaps, however this DLL may be meant to do more than just change a float?

Is this address the one after injection or before then?--is the question

webinject - August 2, 2014, 7:57 pm

what the fuck is injecting dll from kernel mode i'm just curious

Irritable - August 2, 2014, 8:10 pm

(Aug 2, 2014 07:44 PM)0xDEAD10CC Wrote: [\[CITE\]](#) No... When you load the DLL that will change the previous address of the float in memory most likely. If he would've just made a call to WriteProcessMemory, he could change memory without injecting any DLL perhaps, however this DLL may be meant to do more than just change a float?

Is this address the one after injection or before then?--is the question

What is this? I don't even...

Do you know what happens when you call upon LoadLibrary to load a *.dll into a process?

masturbaiter - August 2, 2014, 8:15 pm

(Aug 2, 2014 07:44 PM)0xDEAD10CC Wrote: [\[CITE\]](#) No... When you load the DLL that will change the previous address of the float in memory most likely. If he would've just made a call to WriteProcessMemory, he could change memory without injecting any DLL perhaps, however this DLL may be meant to do more than just change a float?

Is this address the one after injection or before then?--is the question

What are you talking about buddy? You seem to be misinterpreting what's going on here....

Irritable - August 2, 2014, 8:37 pm

(Aug 2, 2014 08:15 PM)masturbaiter Wrote: [\[CITE\]](#) What are you talking about buddy? You seem to be misinterpreting what's going on here....

In many more ways than just one... And "misinterpreting" isn't even the right word in this case. Ignorance is.

dberube - August 2, 2014, 10:20 pm

(Aug 2, 2014 06:51 AM)monumento Wrote: [\[CITE\]](#) Well, because of the AC i need to. :x
ITT, people not reading this thread.

He clearly stated he needs to inject a dll because the game protection detect WriteProcessMemory.

masturbaiter - August 2, 2014, 11:05 pm

made something quick that can help you out

Code typedef struct

```
{  
    void* pFloat;  
    float nFloat;  
}sFloat;
```

```
typedef int (__cdecl* pChangeFloat_t)(void* sFloat_s);  
const unsigned char ChangeFloat[] =  
{  
    0x55,0x8B,0xEC,0x8B,0x45,0x08,0x8B,0x08,0xD9,0x40,  
    0x04,0xD9,0x19,0x5D,0xC3,0xCC  
};
```

Usage

```
Code DWORD ulOld = 0;  
pChangeFloat_t ChangeFloat_o = 0;  
float TMP = 1.0f;  
sFloat szFloat = {0};
```

```
if  
(VirtualProtect((void*)ChangeFloat,sizeof(ChangeFloat),PAGE_EXECUTE_READWRITE,&ulOld))  
{  
    ChangeFloat_o = (pChangeFloat_t)(&ChangeFloat);  
    szFloat.nFloat = 2.0f;  
    szFloat.pFloat = &TMP;  
    ChangeFloat_o(&szFloat);  
}
```

If the game is detecting WPM it might be an inline hook. If so you can apply a patch that could easily remove this hook. Better yet if it detects that the hook is being removed just make the hook go to your callback instead.. if all else fails overwrite their callback

wat iz dat - August 3, 2014, 4:31 am

Thanks masturbaiter. @dberube yeah i wasnt unable to do it but now i got it. Thanks!