3 - Dynamic Link Libraries

DLL Hijacking : https://attack.mitre.org/techniques/T1574/001/

DLL Sile-Loading : https://attack.mitre.org/techniques/T1574/002/

DLL Injection : https://attack.mitre.org/techniques/T1055/001/

Task 5 Dynamic Link Libraries

The Microsoft docs describe a DLL as "a library that contains code and data that can be used by more than one program at the same time."

DLLs are used as one of the core functionalities behind application execution in Windows. From the Windows documentation, "The use of DLLs helps promote modularization of code, code reuse, efficient memory usage, and reduced disk space. So, the operating system and the programs load faster, run faster, and take less disk space on the computer."

When a DLL is loaded as a function in a program, the DLL is assigned as a dependency. Since a program is dependent on a DLL, attackers can target the DLLs rather than the applications to control some aspect of execution or functionality.

- DLL Hijacking (T1574.001)
- DLL Side-Loading (T1574.002)
- DLL Injection (T1055.001)

DLLs are created no different than any other project/application; they only require slight syntax modification to work. Below is an example of a DLL from the *Visual C++ Win32 Dynamic-Link Library project*.

```
#include "stdafx.h"
#define EXPORTING_DLL
#include "sampleDLL.h"
BOOL APIENTRY DllMain( HANDLE hModule, DWORD ul_reason_for_call, LPVOID lpReserved
)
{
    return TRUE;
}

void HelloWorld()
{
    MessageBox( NULL, TEXT("Hello World"), TEXT("In a DLL"), MB_OK);
}
```

Below is the header file for the DLL; it will define what functions are imported and exported. We will discuss the header file's importance (or lack of) in the next section of this task.

```
#ifndef INDLL_H
    #define INDLL_H
    #ifdef EXPORTING_DLL
        extern __declspec(dllexport) void HelloWorld();
```

```
#ifndef INDLL_H
    #define INDLL_H
    #ifdef EXPORTING_DLL
        extern __declspec(dllexport) void HelloWorld();
    #else
        extern __declspec(dllimport) void HelloWorld();
    #endif

#endif
```

The DLL has been created, but that still leaves the question of how are they used in an application?

DLLs can be loaded in a program using *load-time dynamic linking* or *run-time dynamic linking*.

When loaded using *load-time dynamic linking*, explicit calls to the DLL functions are made from the application. You can only achieve this type of linking by providing a header (*.h*) and import library (*.lib*) file. Below is an example of calling an exported DLL function from an application.

```
#include "stdafx.h"
#include "sampleDLL.h"
int APIENTRY WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nCmdShow)
{
    HelloWorld();
    return 0;
}
```

When loaded using *run-time dynamic linking*, a separate function (`LoadLibrary` or `LoadLibraryEx`) is used to load the DLL at run time. Once loaded, you need to use `GetProcAddress` to identify the exported DLL function to call. Below is an example of loading and importing a DLL function in an application.

```
...
typedef VOID (*DLLPROC) (LPTSTR);
...
HINSTANCE hinstDLL;
DLLPROC HelloWorld;
BOOL fFreeDLL;

hinstDLL = LoadLibrary("sampleDLL.dll");
if (hinstDLL != NULL)
{
    HelloWorld = (DLLPROC) GetProcAddress(hinstDLL, "HelloWorld");
    if (HelloWorld != NULL)
        (HelloWorld);
```

```
hinstDLL = LoadLibrary("sampleDLL.dll");
if (hinstDLL != NULL)
{
    HelloWorld = (DLLPROC) GetProcAddress(hinstDLL, "HelloWorld");
    if (HelloWorld != NULL)
        (HelloWorld);
    fFreeDLL = FreeLibrary(hinstDLL);
}
...
```

In malicious code, threat actors will often use run-time dynamic linking more than load-time dynamic linking. This is because a malicious program may need to transfer files between memory regions, and transferring a single DLL is more manageable than importing using other file requirements.

Answer the questions below

Open the provided file: "Logfile.PML" in Procmon and answer the questions below.

No answer needed — ✓ Correct Answer

What is the base address of "ntdll.dll" loaded from "notepad.exe"?

0x7ffd0be20000 — ✓ Correct Answer — Hint

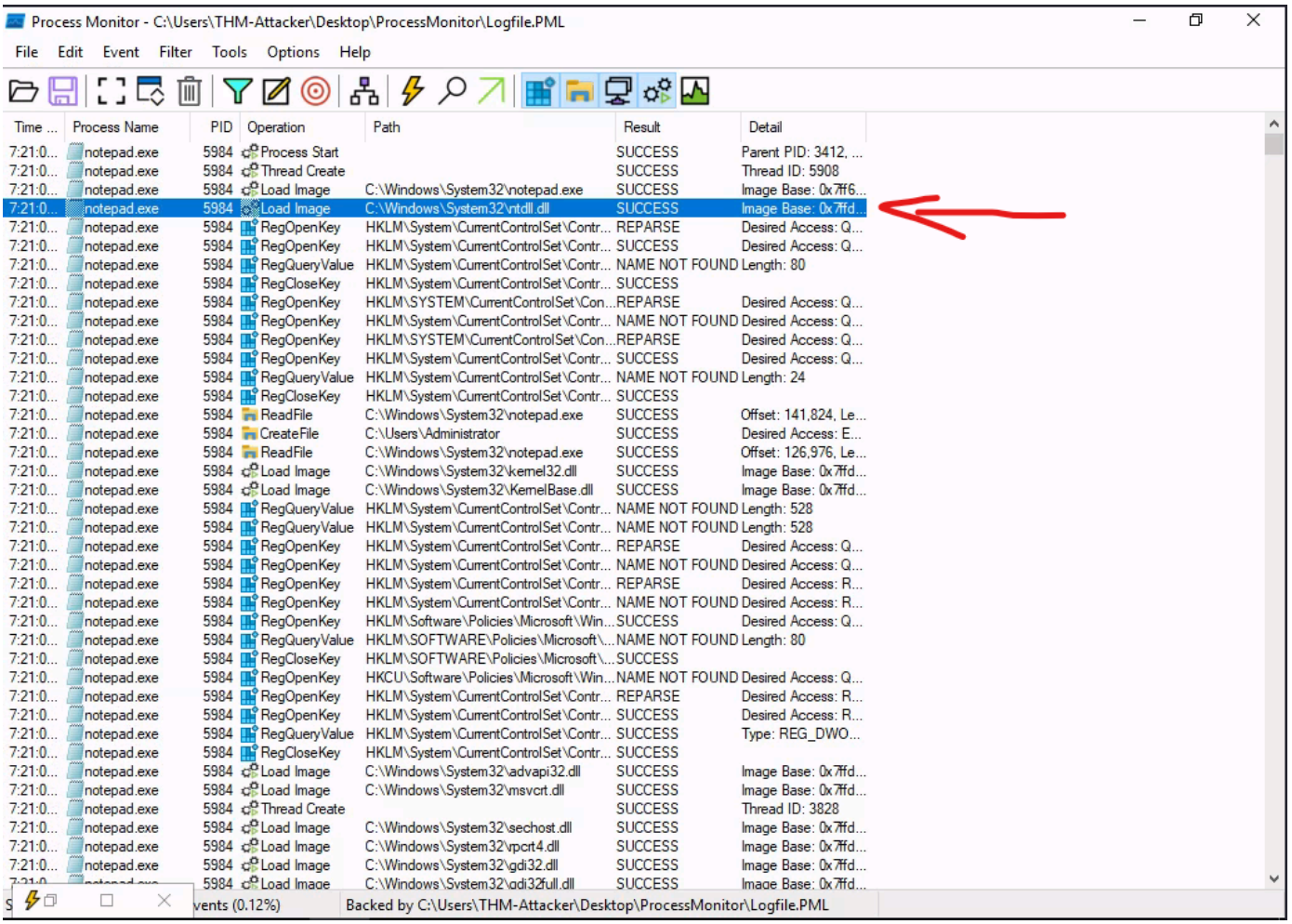
What is the size of "ntdll.dll" loaded from "notepad.exe"?

0x1ec000 — ✓ Correct Answer — Hint

How many DLLs were loaded by "notepad.exe"?

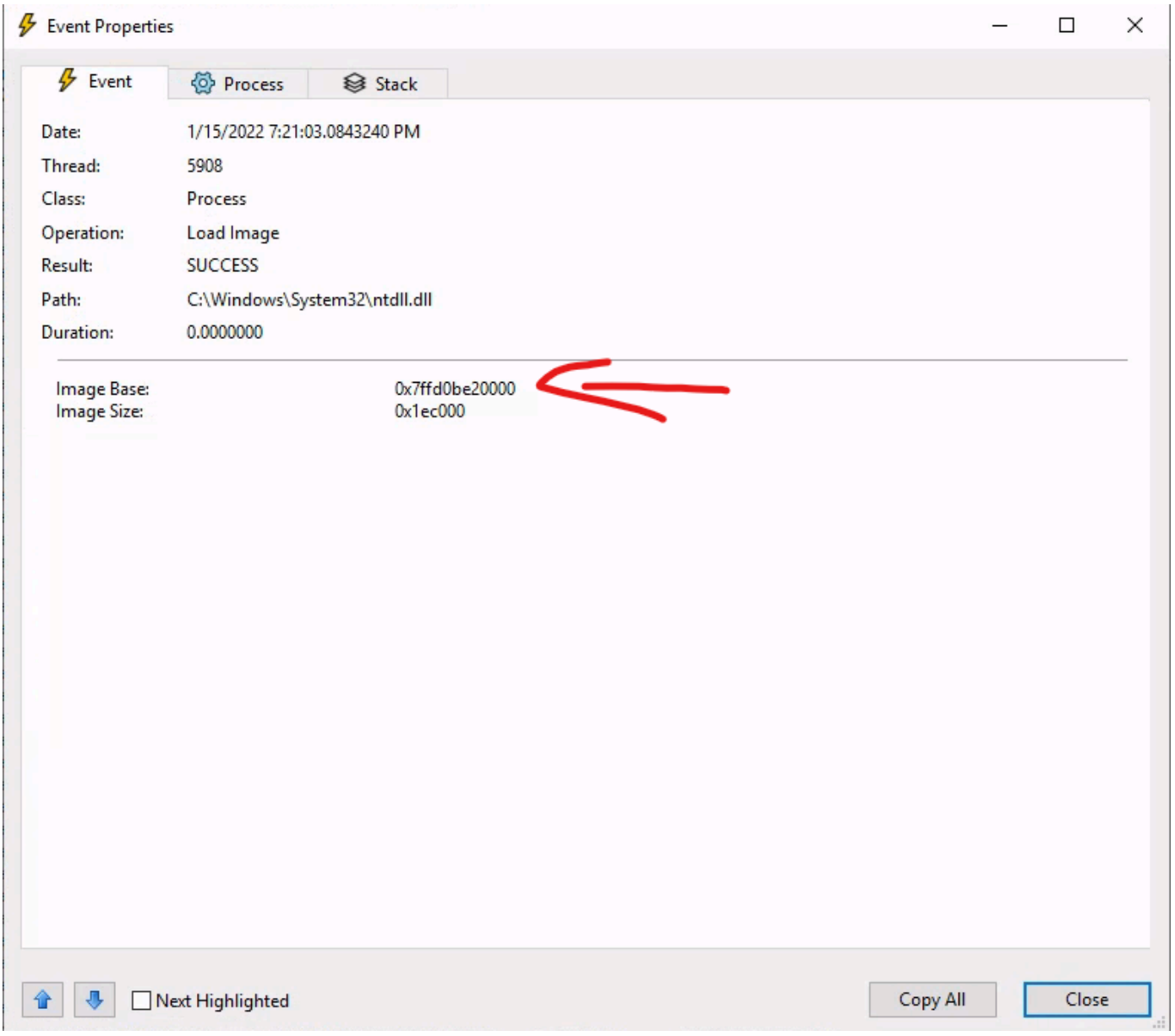
51 — ✓ Correct Answer — Hint

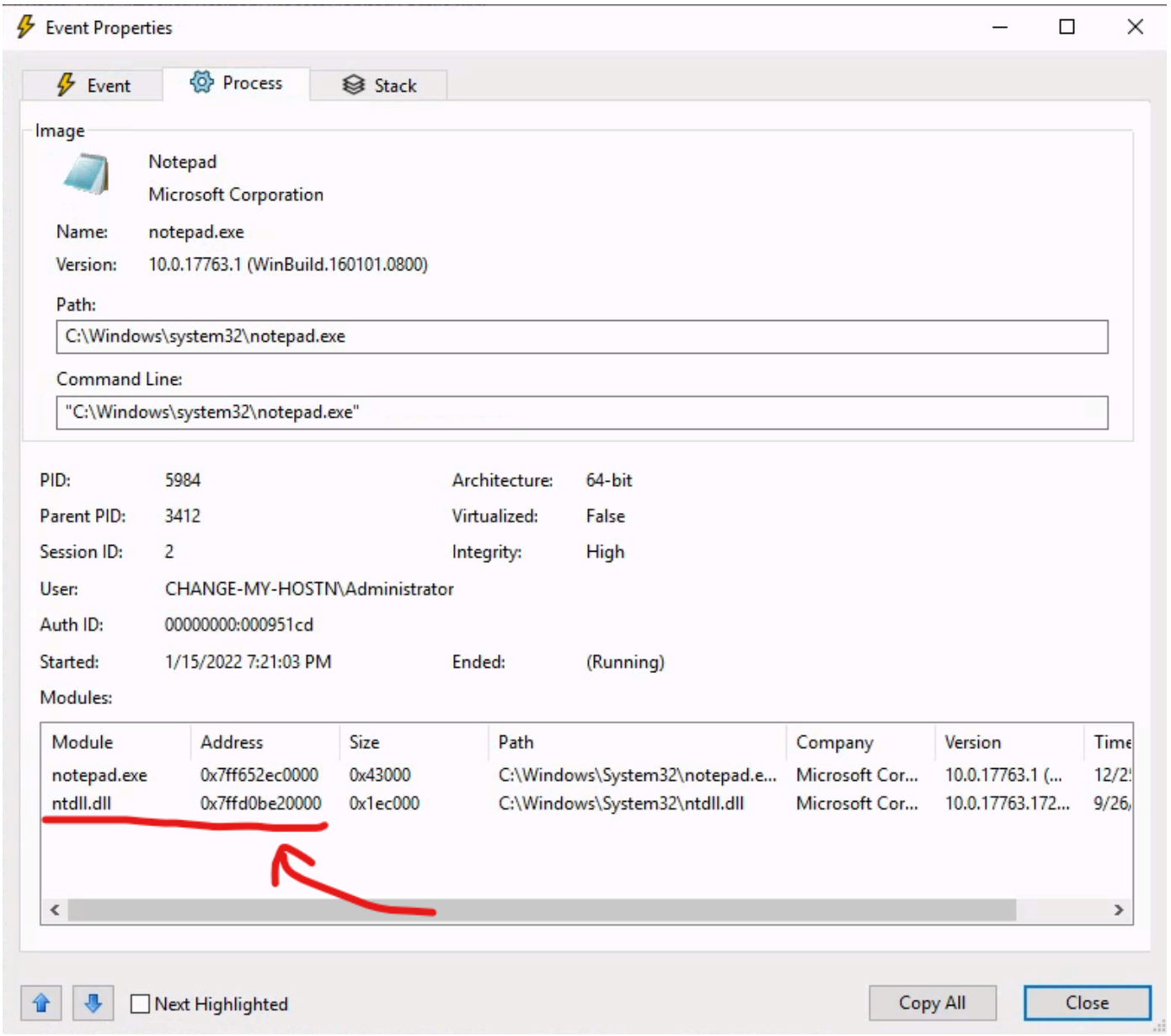
The "ntdll.dll" is loaded after the process is assigned a base address, which should be the very next threat.



Time ...	Process Name	PID	Operation	Path	Result	Detail
7:21:0...	notepad.exe	5984	Process Start		SUCCESS	Parent PID: 3412, ...
7:21:0...	notepad.exe	5984	Thread Create		SUCCESS	Thread ID: 5908
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\notepad.exe	SUCCESS	Image Base: 0x7ff6...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 80
7:21:0...	notepad.exe	5984	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 24
7:21:0...	notepad.exe	5984	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:21:0...	notepad.exe	5984	ReadFile	C:\Windows\System32\notepad.exe	SUCCESS	Offset: 141,824, Le...
7:21:0...	notepad.exe	5984	CreateFile	C:\Users\Administrator	SUCCESS	Desired Access: E...
7:21:0...	notepad.exe	5984	ReadFile	C:\Windows\System32\notepad.exe	SUCCESS	Offset: 126,976, Le...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\kernel32.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\KernelBase.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 528
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 528
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: R...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Desired Access: R...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\Software\Policies\Microsoft\Win...	SUCCESS	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\...	NAME NOT FOUND	Length: 80
7:21:0...	notepad.exe	5984	RegCloseKey	HKLM\SOFTWARE\Policies\Microsoft\...	SUCCESS	
7:21:0...	notepad.exe	5984	RegOpenKey	HKCU\Software\Policies\Microsoft\Win...	NAME NOT FOUND	Desired Access: Q...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: R...
7:21:0...	notepad.exe	5984	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: R...
7:21:0...	notepad.exe	5984	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Type: REG_DWO...
7:21:0...	notepad.exe	5984	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\advapi32.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\msvcrt.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Thread Create		SUCCESS	Thread ID: 3828
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\sechost.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\vpcrt4.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\gdi32.dll	SUCCESS	Image Base: 0x7ffd...
		5984	Load Image	C:\Windows\System32\gdi32full.dll	SUCCESS	Image Base: 0x7ffd...

And again, we can find the answer in both the Event and Process Tab.


Event Properties
Event
Process
Stack
Date: 1/15/2022 7:21:03.0843240 PM
Thread: 5908
Class: Process
Operation: Load Image
Result: SUCCESS
Path: C:\Windows\System32\ntdll.dll
Duration: 0.0000000
Image Base: 0x7ffd0be20000
Image Size: 0x1ec000
Next Highlighted
Copy All
Close



The size is in the same page.

To answer last questions, we need to make a couple filters. Thanks to cyb3rm3 in his walkthrough I was able to figure out what I was doing wrong. (https://www.cyb3rm3.com/w1nd0w51nt3rnal5)

We need 3 filters in order to get to the correct answer: process name "notepad.exe", operations "load image", and path ends with ".dll".

If you try to filter only by process name, and path, we get to a close number, but not quite right.

Process Monitor - C:\Users\THM-Attacker\Desktop\ProcessMonitor\Logfile.PML

File Edit Event Filter Tools Options Help

Process Monitor Filter

Display entries matching these conditions:

Architecture | is | | then | Include

Reset | Add | Remove

Column	Relation	Value	Action
Process N...	is	notepad.exe	Include
Operation	is	Load Image	Include
Path	ends with	.dll	Include
Process N...	is	Procmon.exe	Exclude
Process N...	is	Procexp.exe	Exclude
Process N...	is	Autoruns.exe	Exclude
Process N...	is	Procmon64.exe	Exclude
Process N...	is	Procexp64.exe	Exclude

OK | Cancel | Apply

Time ...	Process Name	PID	Operation	Path	Result	Detail
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\WinSxS\amd64_microsoft.windows....	SUCCESS	Image Base: 0x7ffc...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\propsys.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\oleaut32.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\bcrypt.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\urlmon.dll	SUCCESS	Image Base: 0x7ffc...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\IPHLPAPI.DLL	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\iertutil.dll	SUCCESS	Image Base: 0x7ffc...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\cryptbase.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\imm32.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\uxtheme.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\clbcatq.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\MrmCoreR.dll	SUCCESS	Image Base: 0x7ffc...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\msctf.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\dwmapi.dll	SUCCESS	Image Base: 0x7ffd...
7:21:0...	notepad.exe	5984	Load Image	C:\Windows\System32\cryptsp.dll	SUCCESS	Image Base: 0x7ffd...

Showing 51 of 1,176,740 events (0.0043%) | Backed by C:\Users\THM-Attacker\Desktop\ProcessMonitor\Logfile.PML