

Forum:[Hack Forums](#)

Thread: [How to hide a process from the task manager](#)

PDGLOBAL - May 24, 2017, 2:37 pm

Hello friends, I am looking into a way to hide a process from task manager in windows 10, as well as all other versions of windows.

I have some code I've been playing around with, but in windows 10 the task manager list view doesn't have a control ID, so the code I have is not working. Please help!

Code:

```
Code Func HideFromTaskMannager($sProcessName)
    If $sProcessName <> 1 Then
        $iProcessIndex = ControlListView("Windows Task Manager", "", 1009, "FindItem",
$sProcessName)
        If $iProcessIndex = -1 Then
            Sleep(2)
        Else
            $h_listview = ControlGetHandle("Windows Task Manager", "", 1009)
            DllCall("user32.dll", "int", "SendMessage", "hwnd", $h_listview, "int", $LVM_DELETEITEM,
"int", $iProcessIndex, "int", 0)
        EndIf
    EndIf
EndFunc
```

PDGLOBAL - June 27, 2017, 10:49 am

Still looking for an alternative to this, any ideas?

Kjetil - June 27, 2017, 12:56 pm

posting here because i am searching too

AmericanExpress - June 27, 2017, 1:07 pm

Hello.

In reality the solution you are looking for would be a rootkit component; a rootkit is designed to conceal/protect other malicious software on the system, however the name comes from Linux where you can gain "root" privileges. Obviously a rootkit component will desire as many privileges as it can gain (root access on Linux and ring 0 access on Windows) however there are also user-mode based rootkits.

The easiest solution for you would be via API hooking. You need to inject code into the TaskMgr.exe process and then have that injected code detour specific functions (targeting the Native API would be the smart idea although you can target Win32 API functions if you'd prefer) used by Task Manager for scanning running processes. You can check the imports of TaskMgr.exe and monitor the API calls to understand which ones are used for scanning processes... Then you hook the one/s it uses to control execution flow and filter out your specific process, then make it skip to the next one so yours never gets added to the processes list.

Alternatively, you could use a launcher to perform dynamic forking (which is where you start

another program up in memory in a suspended state - the main thread of the newly started process will be suspended) and then you replace the PE in memory within that process with your own program, forcing your program to be executing within another process' address space - this will mean that when someone checks a process monitor tool like Task Manager or Process Hacker they will not know that your own program is being executed in memory, it won't be displayed as its own process, but the genuine one you had initially targeted would instead.

That being said, you could work with thread hijacking instead of concealing a process through API hooking/dynamic forking instead.

I've made quite a few threads on the Windows Internals, and have also posted a thread on API hooking for both 32-bit and 64-bit processes, therefore you might like to check those out since API hooking will almost certainly assist you on accomplishing your desired goal. Although, my threads are based in C/C++... You can still learn from the threads though and re-create the code in AutoIt.

Good luck. ?

CortexM4 - June 27, 2017, 1:14 pm

Whatever he said^

Anyway tl;dr you would need to somehow inject the autoit interpreter(And make it run your script) in to task manager and then hook the api's that it uses to to query all of the processes in the system and filter out your process or just simply inject in to an already running process which would hide your autoit interpreter process.

Le Pickle Rick - August 18, 2017, 8:59 am

This is from a very old post on a forum from a while back. And also an old post on here.

This only hides it from the ListView.

To actually hide it you need to have it consistently remove it from the systems List 32 file or whatever it's called

BrBi - September 23, 2017, 10:01 am

Or you can inject your process inside another process... that's the only way I think

Unk0wn - December 1, 2017, 6:20 pm

(May 24, 2017 02:37 PM)litzinger Wrote: [\[CITE\]](#) Hello friends, I am looking into a way to hide a process from task manager in windows 10, as well as all other versions of windows.

Hey instead of hiding the processes i suggest that you make prestistor components aka files that watch the file so if one goes down it'll just start a new instance of it usually 3 of them are enough hi hope i was to help :)

PolymorphHF - December 2, 2017, 10:03 am

Keypatch/kernel modification or RunPE injection

Lucifer211 - December 12, 2017, 11:59 am

Thanks guys. this was quite helpful