

Forum:[Hack Forums](#)

Thread: [Process Protection, stop process from being stopped](#)

Razor Sharp - October 6, 2014, 4:11 pm

Hey HF this is a simple method to stop your process from being stopped by any program. It works by finding out the processes .exe then checking if the process is running every two seconds. If the process is not running it starts the process up again. I am also working on Registry and file protection and will post them in the morning. This is by no way a finished product and is still very buggy. Any Ideas on how to improve it would be great ?. Anyway the code is here:

Code Imports System.ComponentModel

Module Module1

Dim t() As Process

Sub Find_exe(ByVal process_name As String)

'finds the location of the executable

For Each p As Process In Process.GetProcessesByName(process_name)

Dim file As String

Try

file = p.Modules(0).FileName

Catch ex As Win32Exception

file = "n/a"

End Try

IsItRunning(file, process_name)

Next

End Sub

Private Sub IsItRunning(ByVal file As String, ByVal process_name As String)

While True

t = Process.GetProcessesByName(process_name)

If t.Count > 0 Then

' Process is running

System.Threading.Thread.Sleep("2000")

Else

Process.Start(file)

' Process is not running

End If

End While

End Sub

End ModuleUsage:

Code Find_exe("your_process_name")If you get "file not found" errors just replace all the find_exe stuff with "Dim file = "your file directory""

Thanks for reading this

regards

Razor Sharp

AEDEAD - October 6, 2014, 4:15 pm

(Oct 6, 2014 04:11 PM)Razor Sharp Wrote: [\[CITE\]](#) Hey HF this is a simple method to stop your process from being

crap, hit enter by accident :/ will post thread in 5 min
Um okay, I'll edit my post in 5 min.

n0\$3ratu\$ - October 6, 2014, 9:25 pm

```
Code If Process.GetProcessesByName("file1").Length = 0 Then
    Process.Start(your path\file1")
End If
```

BlacksmithSec - October 7, 2014, 8:34 am

This is a bad approach - timed checks wont work and are inefficient
Still a nice share but thought i'd throw in my penny worth

Razor Sharp - October 10, 2014, 10:12 am

(Oct 7, 2014 08:34 AM)BlacksmithSec Wrote: [\[CITE\]](#) This is a bad approach - timed checks wont work and are inefficient

Still a nice share but thought i'd throw in my penny worththanks for the reply, just wondering what would you use instead of time checks?

Menalix - October 10, 2014, 10:23 am

I would rather just do a hook on NtOpenProcess, it's like 12 lines of code.

Moon Rock - October 10, 2014, 10:24 am

Nice of you to share the code, however it doesn't seem proficient.

The-One - October 10, 2014, 11:19 am

Although the code isn't reliable, as you can just kill the process using your code and before killing the original process , but the idea is a very good start. I'd say you take this code and try to improve it better (like, maybe injecting it into a system process?).

Good luck!

cwood - October 10, 2014, 1:20 pm

(Oct 10, 2014 10:23 AM)Menalix Wrote: [\[CITE\]](#) I would rather just do a hook on NtOpenProcess, it's like 12 lines of code.

Can you provide more information about this?

Menalix - October 10, 2014, 1:25 pm

(Oct 10, 2014 01:20 PM)cwood Wrote: [\[CITE\]](#) Can you provide more information about this?

Well Alloc some memory in the Remote Process, Write your hookcode (gotta be asm) to it, Get the Function address of NtOpenProcess or NtTerminateProcess on the remote process, write a Call or JMP and the offset to where you wrote the hook code.

Native hook done :) quite simple.

I personally hook NtOpenProcess to prevent anti-cheat from scanning for my cheat (hexui.com cheat)

cwood - October 10, 2014, 2:34 pm

how do I get the asm hookcode for ntopenprocess?

Menalix - October 10, 2014, 3:40 pm

(Oct 10, 2014 02:34 PM)cwood Wrote: [\[CITE\]](#) how do I get the asm hookcode for ntopenprocess? Well it's different from what you need. but you read up about asm.

This would return access denied for any process as an example
Code MOV EAX, C0000022
RETN 8

Zeivu~ - *October 10, 2014, 6:46 pm*

Thanks man this will be good for a crypter one day