

Forum:[Hack Forums](#)

Thread: [Hide Process From TaskManager](#)

xeinebiu - April 25, 2015, 3:53 pm

Hello guys ...

In this thread I want to show you a method how to hide your process from TASKMANAGER ...

Open New Project

Import Libraries

Code Imports System.Reflection

Imports System.Runtime.CompilerServices

Form_Load

In Form_Load Event add this code :

Code Dim s As New OpenFileDialog

s.ShowDialog()

Dim filesbytes() As Byte = IO.File.ReadAllBytes(s.FileName)

Dim x As New Threading.Thread(AddressOf RunFromMemory)

x.TrySetApartmentState(Threading.ApartmentState.STA)

x.Start(filesbytes)

me.close

Add this Sub

Code Private Sub RunFromMemory(ByVal bytes As Byte()) 'not sure how it works :(

Dim assembly As Assembly = assembly.Load(bytes)

Dim entryPoint As MethodInfo = [assembly].EntryPoint

Dim objectValue As Object =

RuntimeHelpers.GetObjectValue([assembly].CreateInstance(entryPoint.Name))

entryPoint.Invoke(RuntimeHelpers.GetObjectValue(objectValue), New Object() {New String() {"1"}})

End Sub

This Sub will be called and all bytes you chosed will be run in memory and will be invisible in PROCESS List in TASKMANAGER ...

You can import the bytes to Resources and use them or any method you want ...

REMEBER , just the Program which you load to memory will be hidden ...

suffixx - April 26, 2015, 5:02 am

Very nice :) Next time, I'll also hide my programmes in Taskmanager :)

You also have code to block regedit, cmd and tskmrg?

greetings

Cecilia - April 26, 2015, 5:09 am

This is simple reflection loading.

It is used in most crypters.

Smaller code below

C#:

PHP Code

<?php

```
Assembly.Load(bytes).EntryPoint.Invoke(null, new object[] { null });
```

Only3 - April 26, 2015, 5:46 am

so what the point of it if once u close the loader it will be closed too

xeinebiu - April 26, 2015, 8:17 am

(Apr 26, 2015 05:46 AM)Only3 Wrote: [\[CITE\]](#) so what the point of it if once u close the loader it will be closed too

Mabye you can try to make changes to the code ...

SadJc - April 26, 2015, 2:32 pm

Thank you very useful! I appreciate it.

comeback - April 27, 2015, 3:07 am

if I choose server.exe , then server.exe will run automatic dan hide in task manager ??

xeinebiu - April 27, 2015, 4:24 am

(Apr 27, 2015 03:07 AM)comeback Wrote: [\[CITE\]](#) if I choose server.exe , then server.exe will run automatic dan hide in task manager ??

yea it will be loaded in ram ... but user can close the loader from taskmanager ... this is not good ...

Illusionisten - April 29, 2015, 1:33 pm

IDK how this works but couldnt you just hide yourself instead of having exe 1 hiding your exe 2

Hopsin - April 30, 2015, 3:16 pm

(Apr 25, 2015 03:53 PM)xeinebiu Wrote: [\[CITE\]](#) Hello guys ...

In this thread I want to show you a method how to hide your process from TASKMANAGER ...

Open New Project

Import Libraries

Code Imports System.Reflection

Imports System.Runtime.CompilerServices

Form_Load

In Form_Load Event add this code :

Code Dim s As New OpenFileDialog

s.ShowDialog()

Dim filesbytes() As Byte = IO.File.ReadAllBytes(s.FileName)

Dim x As New Threading.Thread(AddressOf RunFromMemory)

x.TrySetApartmentState(Threading.ApartmentState.STA)

x.Start(filesbytes)

me.close

Add this Sub

Code Private Sub RunFromMemory(ByVal bytes As Byte()) 'not sure how it works :(

Dim assembly As Assembly = assembly.Load(bytes)

```
Dim entryPoint As MethodInfo = [assembly].EntryPoint
Dim objectValue As Object =
RuntimeHelpers.GetObjectValue([assembly].CreateInstance(entryPoint.Name))
entryPoint.Invoke(RuntimeHelpers.GetObjectValue(objectValue), New Object() {New String() {"1"}})
End Sub
```

This Sub will be called and all bytes you chosed will be run in memory and will be invisible in PROCESS List in TASKMANAGER ...

You can import the bytes to Resources and use them or any method you want ...

REMEBER , just the Program which you load to memory will be hidden ...

I think i have seen this before. Nonetheless, many thanks OP for sharing. ?

Moon Rock - April 30, 2015, 3:17 pm

I'm going to try this soon, thanks for sharing :Smile:

Tom2565 - January 13, 2018, 3:40 am

NO WORKING

System.BadImageFormatException

radicalxfrosty - January 13, 2018, 4:11 am

Thanks OP for sharing this method!

nicee

Tom2565 - January 13, 2018, 4:46 am

NO WORK BLYAT There was an error posting your reply: The message is too short. Please enter a message longer than 25 characters.

Vyxn - January 13, 2018, 4:47 am

(Jan 13, 2018 05:46 AM)Tom2565 Wrote: [\[CITE\]](#) NO WORK BLYAT There was an error posting your reply: The message is too short. Please enter a message longer than 25 characters.

You must be a programmer to make it work. :)

Tom2565 - January 13, 2018, 4:49 am

IT NO WORK ON WINDOWS 10 YOU UNDERASTAHND THAT YES

Ninja - January 13, 2018, 5:40 am

(Jan 13, 2018 05:49 AM)Tom2565 Wrote: [\[CITE\]](#) IT NO WORK ON WINDOWS 10 YOU UNDERASTAHND THAT YES

Well this thread is from 2015 so it is pretty dated...

Sn0wX - January 13, 2018, 11:41 am

I can clearly see a thread is dated, yet I cannot help but reply.

ThatCastle - February 10, 2018, 2:53 pm

(Jan 13, 2018 06:40 AM)Ninja Wrote: [\[CITE\]](#) Well this thread is from 2015 so it is pretty dated...

lol, it still works if you do it right and the detection is mh "okey"...

i dont know how to bypass the avire detection but the rest is good

nodistribute.com/result/dElj2Rhs09vFmCAck

gigajew - February 11, 2018, 8:11 am

It works fine on Windows 10 if you load it using the internal nLoadImage method. Besides this won't "protect" anything, the process can still be killed and modules can still be dumped or scanned in memory.

Ninja - February 11, 2018, 9:45 am

(Feb 11, 2018 09:11 AM)gigajew Wrote: [\[CITE\]](#) It works fine on Windows 10 if you load it using the internal nLoadImage method. Besides this won't "protect" anything, the process can still be killed and modules can still be dumped or scanned in memory.

OP only talked about hiding, not protecting.