



Find out which process is locking a file or folder in Windows

Asked 15 years ago Modified 2 days ago Viewed 1.9m times



How can I find out which process is locking a file or folder in Windows?

1311

For instance, when trying to delete a folder, Windows reports this:



The action can't be completed because the folder is open in another program



It also happens with files.



How do I find out what program or application is currently using it and preventing me from deleting the file or folder?

windows

filesystems

process

community-faq-proposed

Share Improve this question Follow

edited 2 days ago



robertspierre

497 3 6 17

asked Mar 9, 2010 at 14:05



leeand00


23.3k 47 129 197

7 See [this ServerFault question](#). – John Fouhy Jul 17, 2009 at 3:34

2 I learned that Windows 10 (and I guess as early as Windows 7) has a program called Resource Monitor, which lets you see what processes are locking a file or folder: stackoverflow.com/a/24126882/470749 – Ryan Jul 12, 2016 at 22:53

Neither of the answers below helped me to delete an .iso file, and then I looked into Hyper-V manager, and one the virtual machines used the .iso file as source for DVD drive. – Roman O Sep 1, 2019 at 17:35

16 Answers

Sorted by: Highest score (default) 

PowerShell method:

159

```
if ((Test-Path -Path $FileOrFolderPath) -eq $false) {
    Write-Warning "File or directory does not exist."
} else {
    $LockingProcess = CMD /C "openfiles /query /fo table | find /I "$FileOrFolderPath""
    Write-Host $LockingProcess
}
```



The `openfiles` command needs to have support for local files enabled, by running `openfiles /local on` and restarting.

More details [How to find out which process is locking a file or folder in Windows](#) [archive](#)

Share Improve this answer Follow

edited May 30, 2023 at 22:04

answered Apr 26, 2017 at 2:59




Dan Atkinson

593 1 8 13



frank

1,844 1 10 3

- 1 Running this gave me the same Access is Denied error I was getting originally, and then always printed out the does not exist error message. The below answer, however, gave me the information I needed - IIS was locking the directory I needed to change. – [Chris Thompson Jun 20, 2018 at 16:40](#)
- 17 It looks like you could basically achieve it in pure cmd with `openfiles /query /fo table | find /I "<path>" ?` – [mwfearnley Aug 31, 2018 at 11:30](#)
- 14 Instead of `openfiles` you can use a similar tool [Handle from Sysinternals](#): `handle.exe -u -nobanner -accepteula "C:\FileOrFolderForWhichWeWantToFindLockingInformation"` – [PolarBear Jul 20, 2021 at 11:24](#) 
- 2 `openfiles` is not available on all versions of windows or might require a reboot. `Handle.exe` as stated by @PolarBear is an excellent choice and if you don't have it you can get it here: learn.microsoft.com/en-us/sysinternals/downloads/handle – [TekOps Jan 14 at 19:17](#)



You can use the **Resource Monitor** for this which comes **built-in** with Windows 7, 8, 10 and 11!

1624

1. Open **Resource Monitor**, which can be found



- By searching for **Resource Monitor** or **resmon.exe** in the start menu, or
- As a button on the **Performance** tab in your *Task Manager*



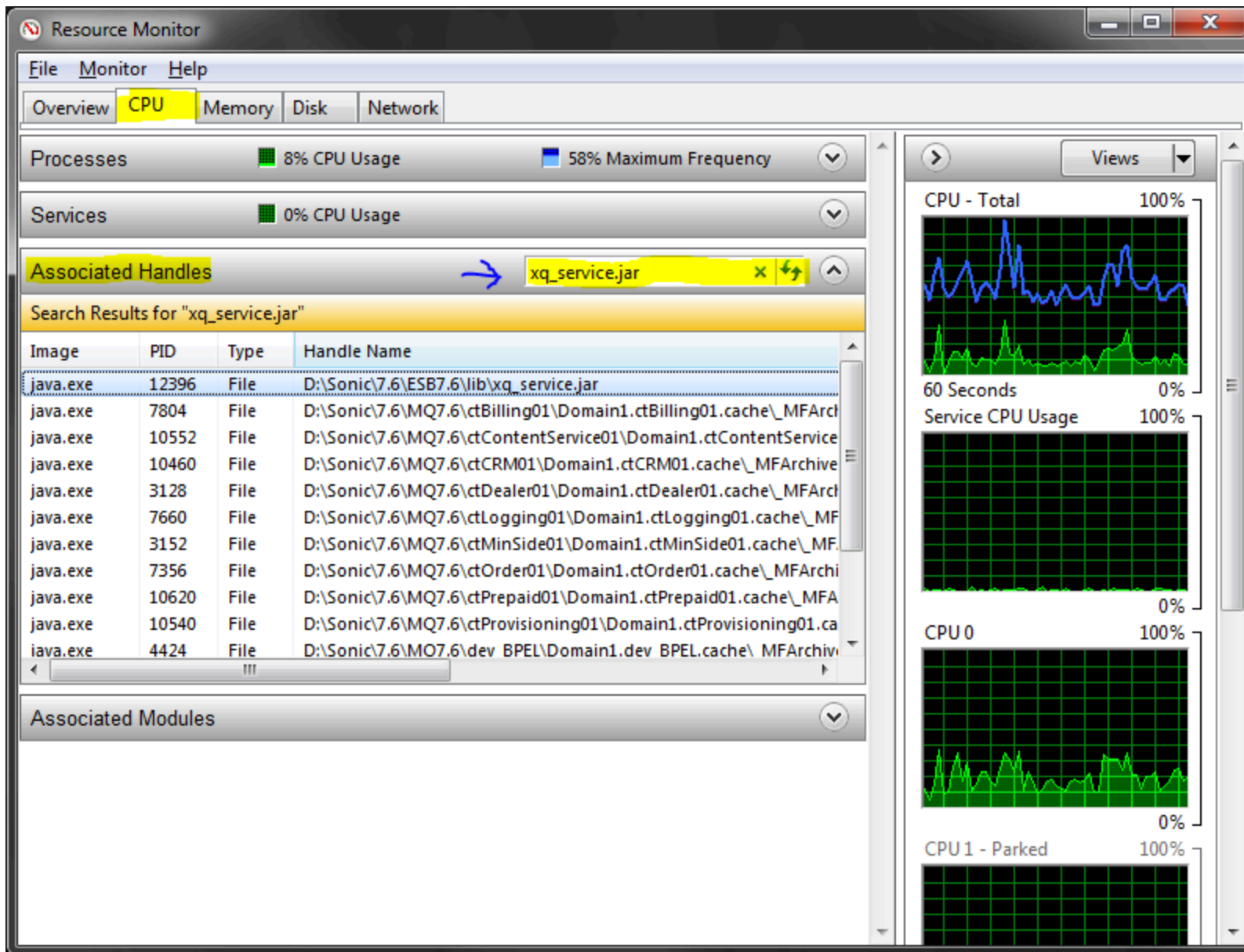
+50



2. Go to the **CPU** tab
3. In the **Processes** section, select all processes by clicking the checkbox next to "Image" in the headers.
4. Use the **search field** in the **Associated Handles** section
 - See blue arrow in screen shot below

When you've found the handle, you can identify the process by looking at the Image and/or PID column.

You can then try to close the application as you normally would, or, if that's not possible, just right-click the handle and kill the process directly from there. Easy peasy!



Share Improve this answer Follow

edited Jun 6, 2023 at 17:47

answered Sep 10, 2013 at 11:43



Kevin Buchs

101 5



Svish

40.9k 62 142 185

41 @ComFreek Well, ProcessExplorer might be more powerful in many cases, but for me Resource Monitor have always found the application locking the file I'm trying to do something with. I also prefer it to Process Explorer simply because it's right there in the OS. No need for an extra download and an extra shortcut somewhere. – [Svish May 6, 2014 at 21:16](#)

I tried to end the process and was told, "When attempting to execute the command, the following system error occurred: Access is denied." I had Resource Monitor running as admin. – [Shaun Luttin Feb 21, 2016 at 23:13](#)

Works for me. Since my error was a python-related error, I had to make sure to get rid of the double backslashes to fit Window's search criteria. – [moondra May 26, 2018 at 16:47](#)

@Svish, what to do if the file is locked by a process running on another machine within the local network? – [Anton Shepelev Sep 4, 2018 at 12:40](#)

Seems to work the same for me. I created a folder, navigated to it using `cmd`, and the folder (with no files in it) showed up in the Associated Handles for `cmd.exe`. The type is listed as `File` though, so I guess `resmon` (or the file system maybe?) doesn't really distinguish between the two. A file handle is a file handle, regardless, maybe. 🙄 – [Svish Apr 24, 2019 at 12:22](#)

worked when given the full path. – [hafiz ali Jun 17, 2021 at 11:31](#)

@srx stackoverflow.com/a/958180/39321 – [Svish Nov 15, 2021 at 9:30](#)

1 Remember to select either specific processes or all processes in the process section before wondering why the handle list is empty – [royalTS May 2, 2023 at 12:10](#)

Cannot rename the folder (reported opened in another program), but Resource Monitor does not find anything about it – [Alexander Sep 12, 2023 at 0:36](#)



A couple of options:

470

[Microsoft/SysInternals Process Explorer](#) - Go to Find > Find Handle or DLL. In the "Handle or DLL substring:" text box, type the path to the file (e.g. "C:\path\to\file.txt") and click "Search". All processes which have an open handle to that file should be listed.



[WhoLockMe](#) - Explorer extension which adds a right-click menu option



N.B. WhoLockMe appears to not work with Win 10 (at least I have been unable to register it with either of the 32- or 64-bit versions of `regsvr32.exe`).



Share Improve this answer Follow

edited Nov 2, 2016 at 15:40

answered Mar 9, 2010 at 14:22



Shevek

16.7k

7

49

76

16 NOTE: Unlocker installs Adware if you are not careful. From their website: "Promotional feature: Fully optional Delta toolbar." – Julian Knight Mar 13, 2014 at 21:50

1 @jpmc26 I have no idea.. I've stopped using it in favour of Unlocker which has a nicer interface IMHO. Unlocker was in my original answer but was edited out because it later had adware added to the installer. – Shevek Sep 24, 2014 at 19:28

6 @jpmc26 No. WhoLockMe does not work on Windows 7. – ceving Nov 28, 2014 at 19:39

The first link looks like it is broken and the second doesn't look like it is working in Windows 10 – Serj Sagan Sep 10, 2015 at 1:44 ✎

Powershell would be quicker, less bloated than all those apps suggestions, see: beamusupscotty.wordpress.com/2012/11/14/... – abourget Nov 5, 2015 at 14:46

2 It should be noted that process explorer has to be run as administrator. Ran it on Windows 10 as normal user and it didn't work. – Devolus Oct 17, 2017 at 7:20

WhoLockMe link is not safe, there is XSS fail !!! – user350726 Apr 4, 2020 at 12:50 ✎

WhoLocksMe looks abandoned and the web site doesn't look trustworthy. I don't recommend to use this tool. There are better alternatives mentioned in this thread like [Lock Hunter](#) and [OpenedFilesView](#) – PolarBear Jul 20, 2021 at 13:00

@Devolus ProcessExplorer has to be run as administrator only if the processes you want to check are themselves run as administrator. – Alex Che Oct 10, 2024 at 15:13



Have a look at [Process Explorer](#) (`procexp.exe`).

275

From its introduction:



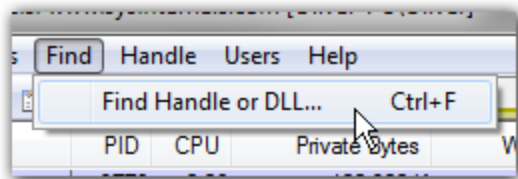
Ever wondered which program has a particular file or directory open? Now you can find out.



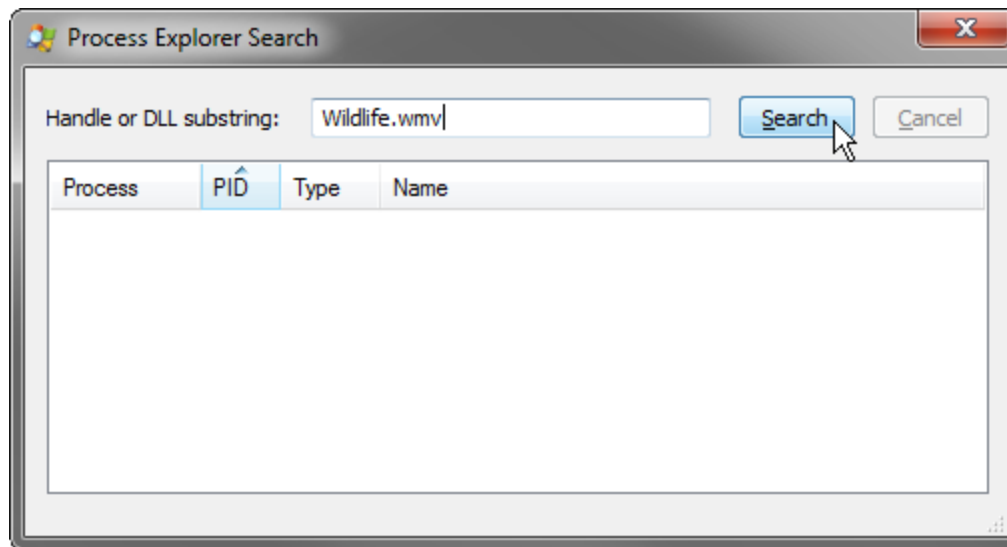
To find out what process is using a specific file follow these steps:



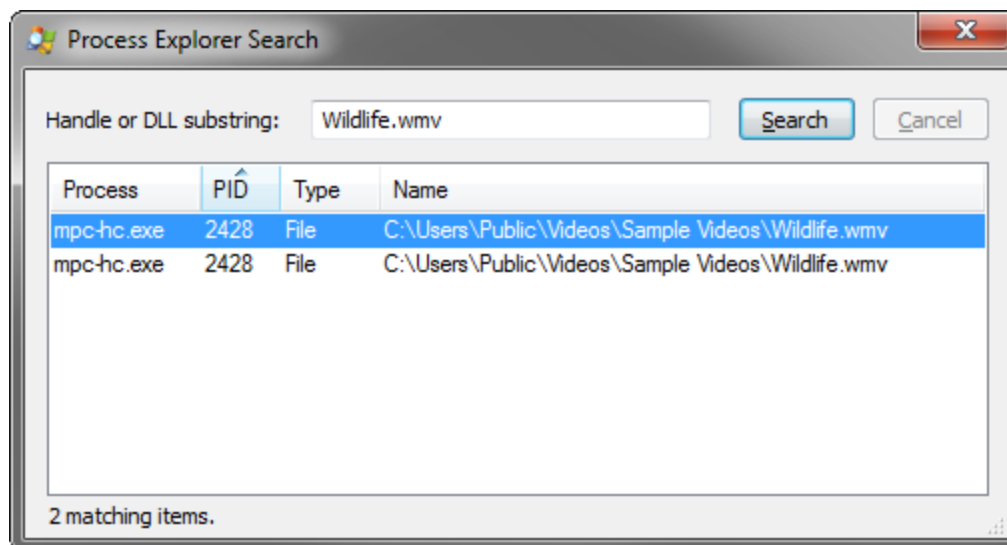
1. Go to **Find, Find Handle or DLL..** or simply press `Ctrl` + `F` .



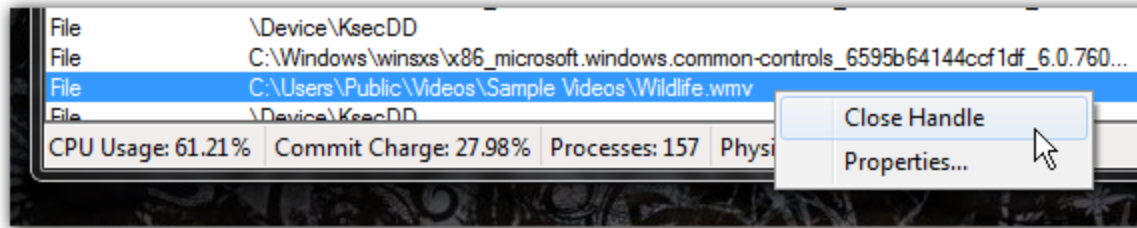
2. Enter the name of the file and press **Search**.



3. Process Explorer will list all processes that have a handle to the file open. Click on an entry to focus the process in the main window.



4. Optionally, you can then even close the handle manually through the lower pane (**Ctrl** + **L**):



Share Improve this answer Follow

edited Nov 28, 2014 at 17:03

answered Mar 12, 2012 at 3:35



Peter Mortensen

12.3k 24 72 90



Eroen

6,521 1 19 25

9 Process Hacker also can do it. – Benoit Mar 12, 2012 at 10:41

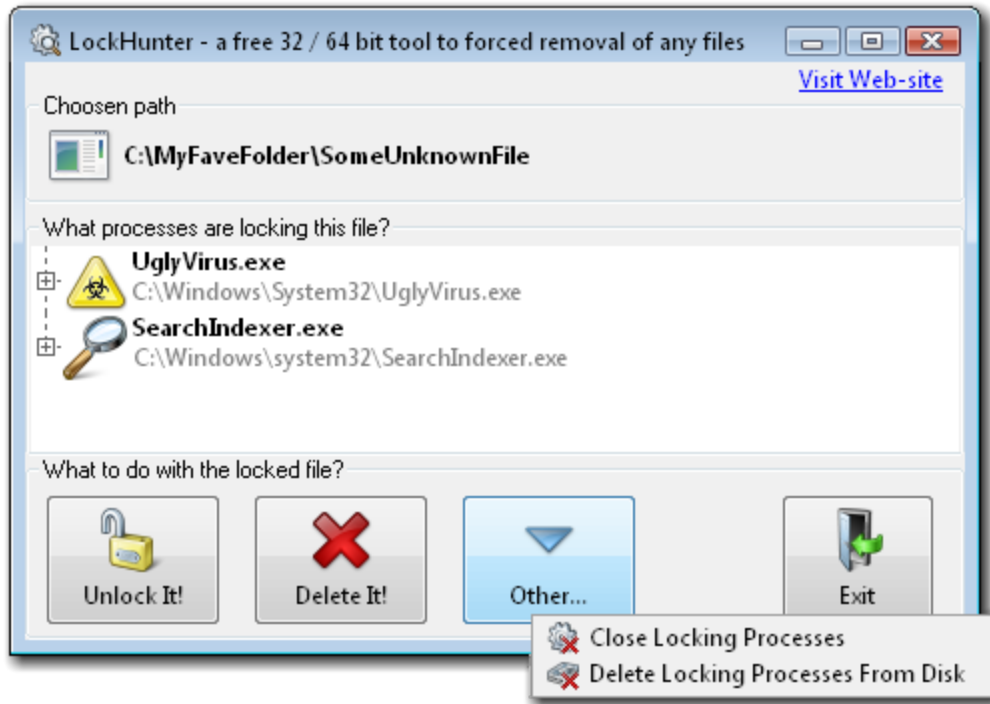
Note to Googlers, current version of PE appears to no longer support **Close Handle** – Deleted Dec 12, 2017 at 8:09

handle didn't work as expected (Win81). Reproduce: open some file in an editor, e.g. nano some.py , then try: handle.exe -nobanner some.py and get: **No matching handles found** . – not2qubit Nov 22, 2018 at 15:47

@Deleted in current version (2023) there is no right-click but the first click on the Handler will focus it in the list of processes where one can review it and Kill process tree if needed. Or at least it shows the PID. – ino Apr 25, 2023 at 11:44

LockHunter can unlock any handlers that may have locked your files or folders. Unlike similar freewares, it supports both 32 and 64-bit Windows.

67



It is a free tool to delete files blocked by something you do not know. LockHunter is useful for fighting against malware, and other programs that are blocking files without a reason. Unlike other similar tools it deletes files into the recycle bin so you may restore them if deleted by mistake.

- Shows processes locking a file or folder
- Allows to unlock, delete, copy or rename a locked file
- Allows to kill locking process
- Allows to remove locking processes from hard drive
- Integrates in to Explorer menu
- It deletes files into the recycle bin, so you may restore them if deleted by mistake
- Supports both 32 and 64bit Windows

Community Bot
1Mehper C. Palavuzlar
55.9k 50 198 254

This is great but the portable version doesn't integrate with the explorer menu – [Phani Rithvij](#) Nov 1, 2021 at 13:28

Yeah this program is from Russia. That's a no. Secure Boot mode fixed the problem for me. – [DSchmidt](#) Dec 3, 2022 at 16:48

- 1 @DSchmidt I'm not sure what problem you were having but secure boot has little to nothing to do with what this tool does (or, indeed, basically any answer/tool on this entire question). Is there a reason you assume all software made by Russian people is inherently unusable? – [TheXenocide](#) Apr 14, 2023 at 20:06 ✎
- 2 @TheXenocide Apparently the process locking my file was not started in secure boot which made any tool unnecessary. On Russia: I'm just avoiding most closed source software made by tiny companies with thin or questionable reputation, history, and source of money. This being Russian is just another drop in the "avoid" bucket, not the sole reason. – [DSchmidt](#) Apr 17, 2023 at 10:51

This tool is not working for me under Windows 10 64-bit when the locked file is on a network drive but locked by a process on my PC. Using the Microsoft Resource Monitor to search for handles did indeed show what was locking my files. – [J Smith](#) Jul 20, 2023 at 18:49

Does not see any locks in my case, even though I cannot rename the folder... – [Alexander](#) Sep 12, 2023 at 0:36



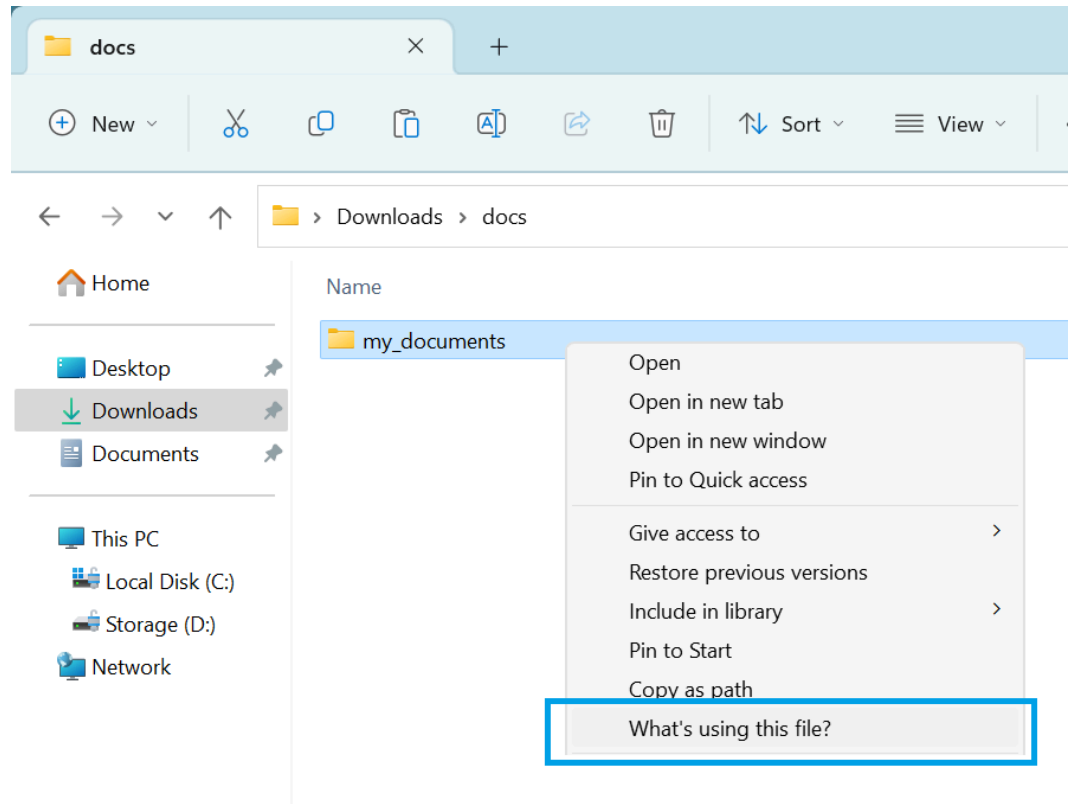
Microsoft PowerToys [File Locksmith](#)

33

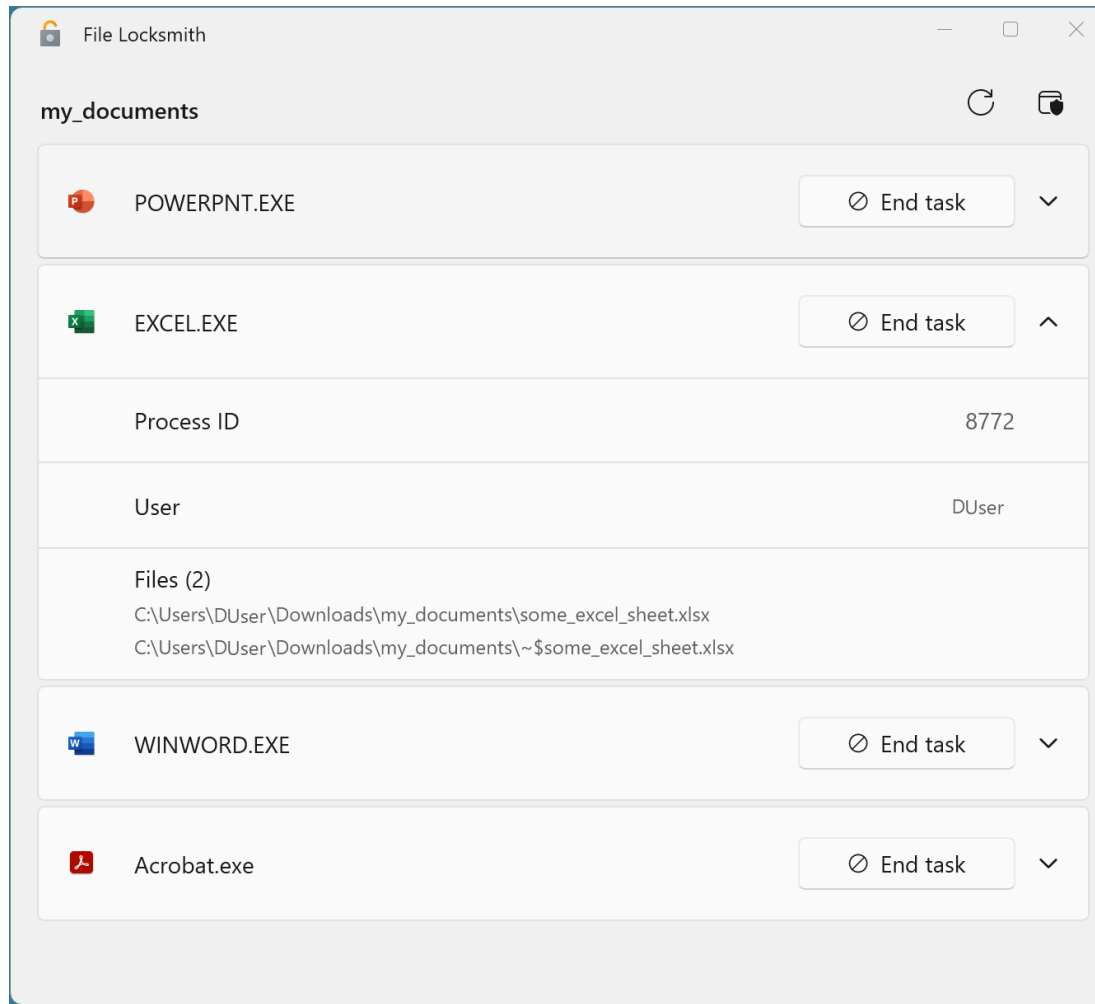
It does exactly what you are asking for.



- Right-click on the file or folder to open a context menu

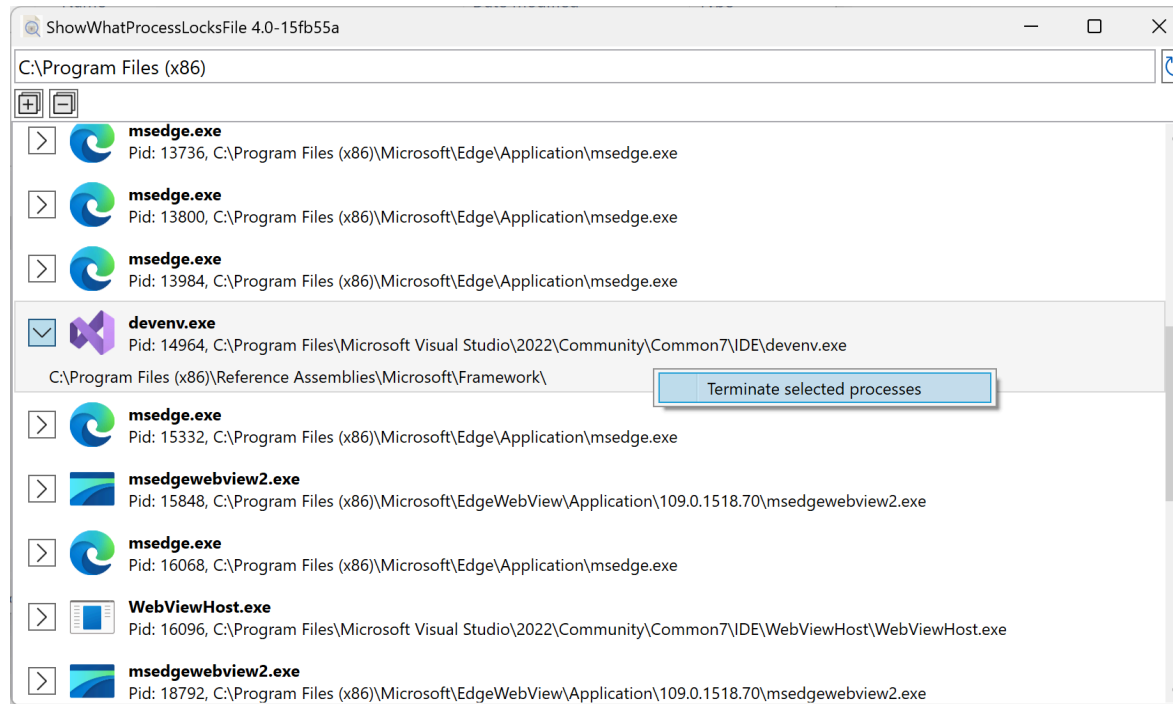


- The utility opens, showing locking processes. There, you can see which files are locked and terminate a particular process



Alternative: ShowWhatProcessLocksFile

[ShowWhatProcessLocksFile](#) is a lightweight and faster clone of [File Locksmith](#) that supports older versions of Windows.



Share Improve this answer Follow

edited Apr 29, 2024 at 3:00

answered Jul 20, 2021 at 20:27



PolarBear

603 6 8

1 Wow, neat an simple. And probably a future built-in in future versions of Windows. – carloswm85 Mar 27, 2023 at 21:34

**14**

[EMCO UnlockIT](#) can identify the process that has locked the file as well as unlock the file so that you may delete/edit/move it. The program is completely free, though the newer version is a bit slower and more bloated than the original (which had a plain, unskinned GUI, but loaded pretty much instantaneously and without an annoying splash screen). Also, the original version used to pop up automatically whenever the error you mentioned is triggered, allowing you to instantly unlock the file and perform the operation you were attempting.

Still, UnlockIT is an incredibly useful program that provides a basic functionality that is critically missing from Windows. It's among the standard toolkit of utilities that I install on all Windows computers I work on.

Share Improve this answer Follow

edited Mar 12, 2012 at 3:42

answered Mar 12, 2012 at 3:37



[Lèse majesté](#)

3,281

1

20

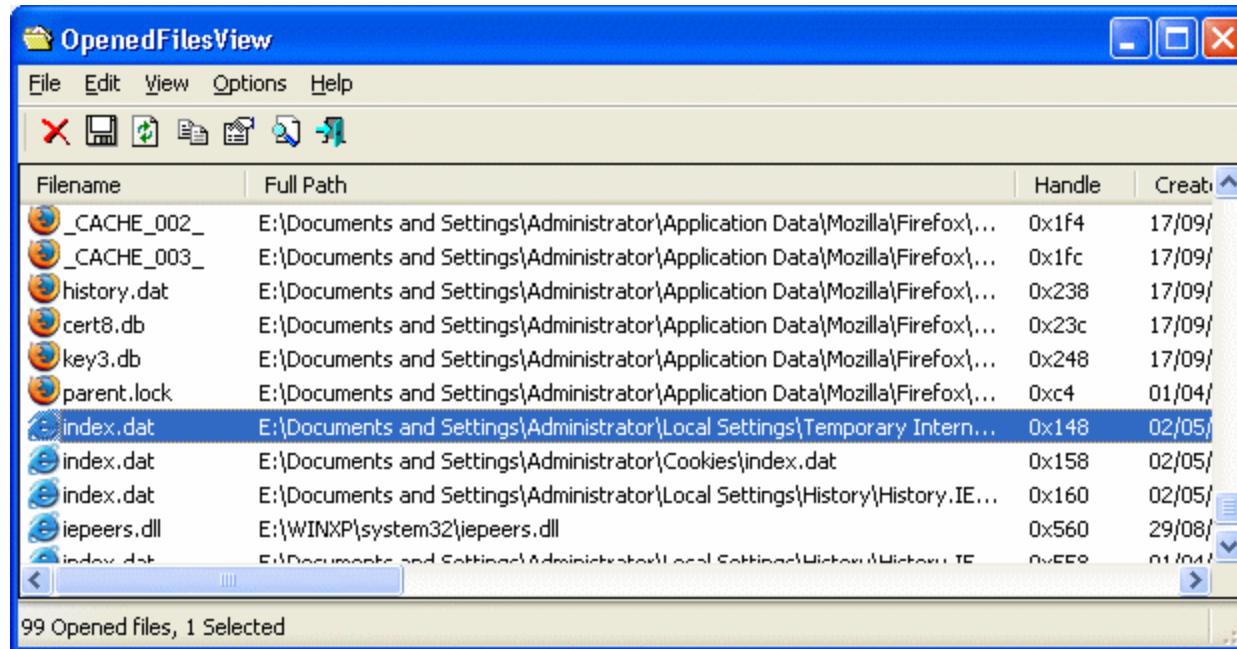
23

@John: Well, it does identify the program that has locked a file. And you can still manually kill the process through task manager. However, it's usually simpler to just unlock the file (especially when it's explorer that has locked the file most of the time) instead of having to kill the process that has locked it (usually due to having accessed the file and left the file handler open due to a program glitch). – [Lèse majesté Mar 12, 2012 at 3:46](#)

As with LockHunter, this tool is not working for me under Windows 10 64-bit when the locked file is on a network drive but locked by a process on my PC. Using the Microsoft Resource Monitor to search for handles did indeed show what was locking my files. – [J Smith Jul 20, 2023 at 18:50](#)

**12**

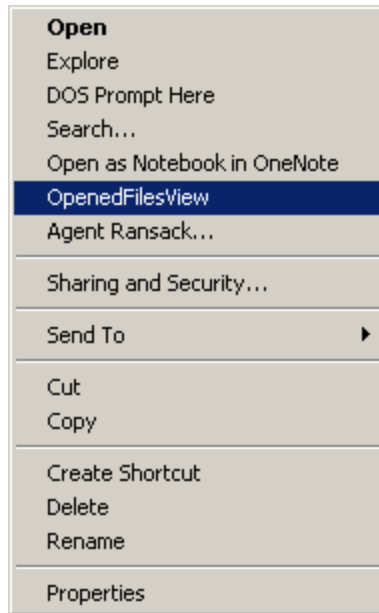
A lot of the programs here are outdated. I finally ended up using nirsoft's [OpenedFilesView](#) which worked really well.



Though the best part is the explorer menu integration, which is easy to enable. As per website

Explorer Context Menu

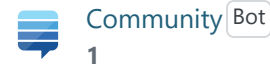
Starting from version 1.10, you can launch OpenedFilesView directly from Windows Explorer, and view only the handles of the file or folder that you want to inspect. In order to enable this feature, check the 'Enable Explorer Context Menu' under the Options menu. After you enable this feature, you can right-click on any file or folder on Windows Explorer, and choose the 'OpenedFilesView' item from the menu. If you run the OpenedFilesView option for a folder, it'll display all opened files inside that folder. If you run the OpenedFilesView option for a file, it'll display all opened handles for that file.



This tool still works reliably in 2019

Share Improve this answer Follow

edited Jun 12, 2020 at 13:48



answered Sep 22, 2019 at 12:39



6

If you do not know the program the file it is using then you can go to My Computer; right click; select Manage. Under System Tools > Shared folders > Open Files, you should be able to see the user who has locked the file. You can close file from here and then you can perform the task of rename or delete the file. Hope this helps



Share Improve this answer Follow



answered Feb 1, 2013 at 6:42



3 This will only work for shared folders. – [sweetfa](#) Aug 14, 2013 at 3:57



[SysInternal's handle utility](#) is designed exactly for this problem for the command line. Just execute the following:

4

`handle -a path`



Here is an example output:



```
→handle -a "C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db"
```



```
Nthandle v5.0 - Handle viewer
Copyright (C) 1997-2022 Mark Russinovich
Sysinternals - www.sysinternals.com
```

```
xplorer2_64.exe pid: 108904 type: File      844:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
xplorer2_64.exe pid: 108904 type: File      1098:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
xplorer2_64.exe pid: 108904 type: File      1B78:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
explorer.exe pid: 75252 type: File      2B68:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
explorer.exe pid: 75252 type: File      4B1C:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
firefox.exe pid: 20884 type: File      15A8:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
firefox.exe pid: 20884 type: File      3BF4:
C:\Users\me\AppData\Local\Microsoft\Windows\Explorer\iconcache_32.db
```

Share Improve this answer Follow

answered Sep 2, 2023 at 15:38



Nathan

1,356 5 18 35



Additional possibility, just to save people the time I just spent:

3

In older versions of Windows, you might get "Access Denied - you might not have rights, or the file might be in use". If you find through Process Explorer that the files are, in fact, not opened by anyone, odds are that there is an issue with security. Using an administrator account, select the files in Explorer, right-click and select Properties, Security, Advanced, Owner. Odds are that the files





are owned by an account that no longer exists or can no longer be verified to exist (because of changing Active Directory trust settings).



Change ownership to Administrators and you are good to go.

Share Improve this answer Follow

answered Sep 26, 2012 at 16:56



Stu

169 3



Here was my discovery & solution.

3

Incidentally, none of the above answers solved my problem.



I even tried using UNLOCKER which proved worthless.



My problem was that of Memeo Autosync Backup



Apparently, this backup process leaves enough of a "ghost like file." This "ghost like file," would show up whenever I would ALT-TAB my computer (Windows Professional XP), i.e. I would see TWO MS Excel Programs running, when I only had ONE visible, on my TASK BAR.

I came across this solution when I thought it might have been the SYMANTEC Endpoint (Anti-Virus) Protection; and disabled the program. However, I kept getting the error message:

cannot delete (LARGE.xls file): It is being used by another person or program. Close any programs that might be using this file and try again.

I subsequently kept seeing the Memeo notice of "syncing" and QUIT the program.

Subsequently, NO ERROR.

For you, it could be ANY of these background saves.

Share Improve this answer Follow

edited Oct 13, 2012 at 10:32

answered Oct 13, 2012 at 9:43



jonsca

4,074 15 36 47



formerly frustrated excel

user

31 1



1



You can also do it programmatically by leveraging on the NTDLL/KERNEL32 Windows API. E.g. have a look at the following code in Python which returns a list of PIDs that can then easily be killed using the Task Manager or similar tools.

```
import ctypes
from ctypes import wintypes

path = r"C:\temp\test.txt"

# -----
# generic strings and constants
# -----

ntdll = ctypes.WinDLL('ntdll')
kernel32 = ctypes.WinDLL('kernel32', use_last_error=True)

NTSTATUS = wintypes.LONG

INVALID_HANDLE_VALUE = wintypes.HANDLE(-1).value
FILE_READ_ATTRIBUTES = 0x80
FILE_SHARE_READ = 1
OPEN_EXISTING = 3
FILE_FLAG_BACKUP_SEMANTICS = 0x02000000

FILE_INFORMATION_CLASS = wintypes.ULONG
FileProcessIdsUsingFileInformation = 47

LPSECURITY_ATTRIBUTES = wintypes.LPVOID
ULONG_PTR = wintypes.WPARAM

# -----
# create handle on concerned file with dwDesiredAccess == FILE_READ_ATTRIBUTES
# -----

kernel32.CreateFileW.restype = wintypes.HANDLE
kernel32.CreateFileW.argtypes = (
    wintypes.LPCWSTR,      # In    lpFileName
    wintypes.DWORD,        # In    dwDesiredAccess
```

```
wintypes.DWORD,          # In      dwShareMode
LPSECURITY_ATTRIBUTES,   # In_opt lpSecurityAttributes
wintypes.DWORD,          # In      dwCreationDisposition
```

Share Improve this answer Follow

answered Nov 10, 2022 at 13:23



Robert

213 1 3 8



Here is a full Powershell solution for that problem:



1

```
cls
remove-variable * -ea 0
$errorActionPreference = 'stop'
```



```
Add-Type -TypeDefinition @"
using System;
using System.IO;
using System.Runtime.InteropServices;
using Microsoft.Win32.SafeHandles;
```

```
public static class ProcessUtils {

    [StructLayout(LayoutKind.Sequential)]
    private struct IO_STATUS_BLOCK {
        public IntPtr Information;
        public IntPtr Status;
    }

    [StructLayout(LayoutKind.Sequential)]
    public struct FILE_PROCESS_IDS_USING_FILE_INFORMATION {
        public ulong NumberOfProcessIdsInList;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 64)]
        public ulong[] ProcessIdList;
    }

    [DllImport("ntdll.dll")]
    private static extern int NtQueryInformationFile(SafeFileHandle FileHandle, ref IO_STATUS_BLOCK IoStatusBlock,
        IntPtr FileInformation, uint Length, int FileInformationClass);

    [DllImport("kernel32.dll", SetLastError = true)]
    private static extern SafeFileHandle CreateFile(string lpFileName, FileAccess dwDesiredAccess,
        FileShare dwShareMode, IntPtr lpSecurityAttributes, FileMode dwCreationDisposition,
        FileAttributes dwFlagsAndAttributes, IntPtr hTemplateFile);
```

```
public static ulong[] GetProcessesUsingFile(string filePath) {  
    var processIds = new ulong[0];  
    var ioStatusBlock = new IO_STATUS_BLOCK();
```

Share Improve this answer Follow

answered Apr 30, 2024 at 8:31



Carsten

311 2 8



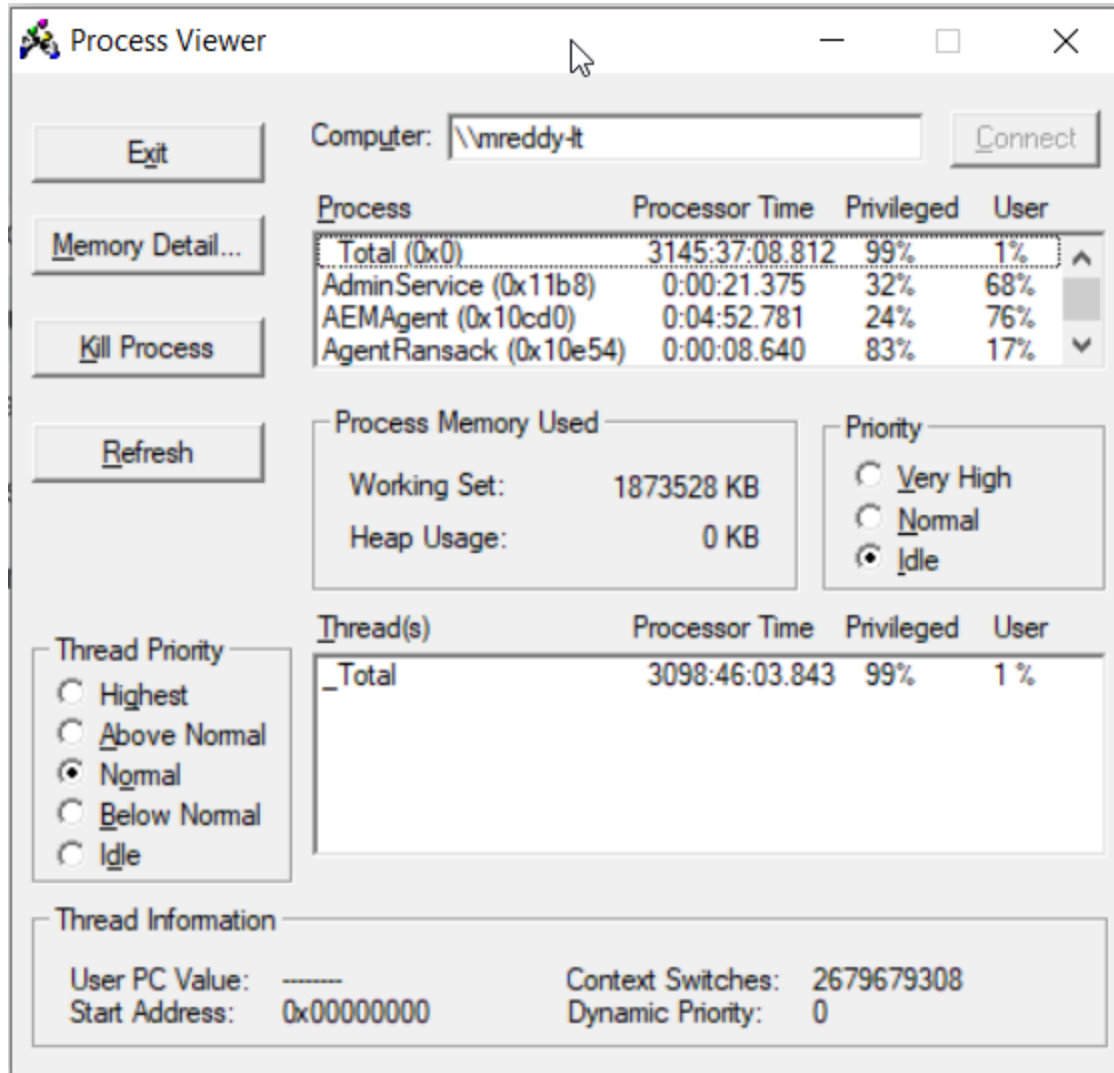
0

I am not sure if anyone used `Process Viewer` (`PVIEW.exe`). I was having trouble finding out the process that locked my `epmd.exe` which i was trying to delete, when I searched for `Process Explorer` app in my Windows 10 box - I found this app. So, I thought of giving it a try and it worked (so, this can be another option):



This app existed at the following location for me - `C:\Program Files\Microsoft Visual Studio\Common\Tools\WinNT\Tools\PVIEW.EXE`





Look for your locked application (file), and **Kill Process** to unlock the file.

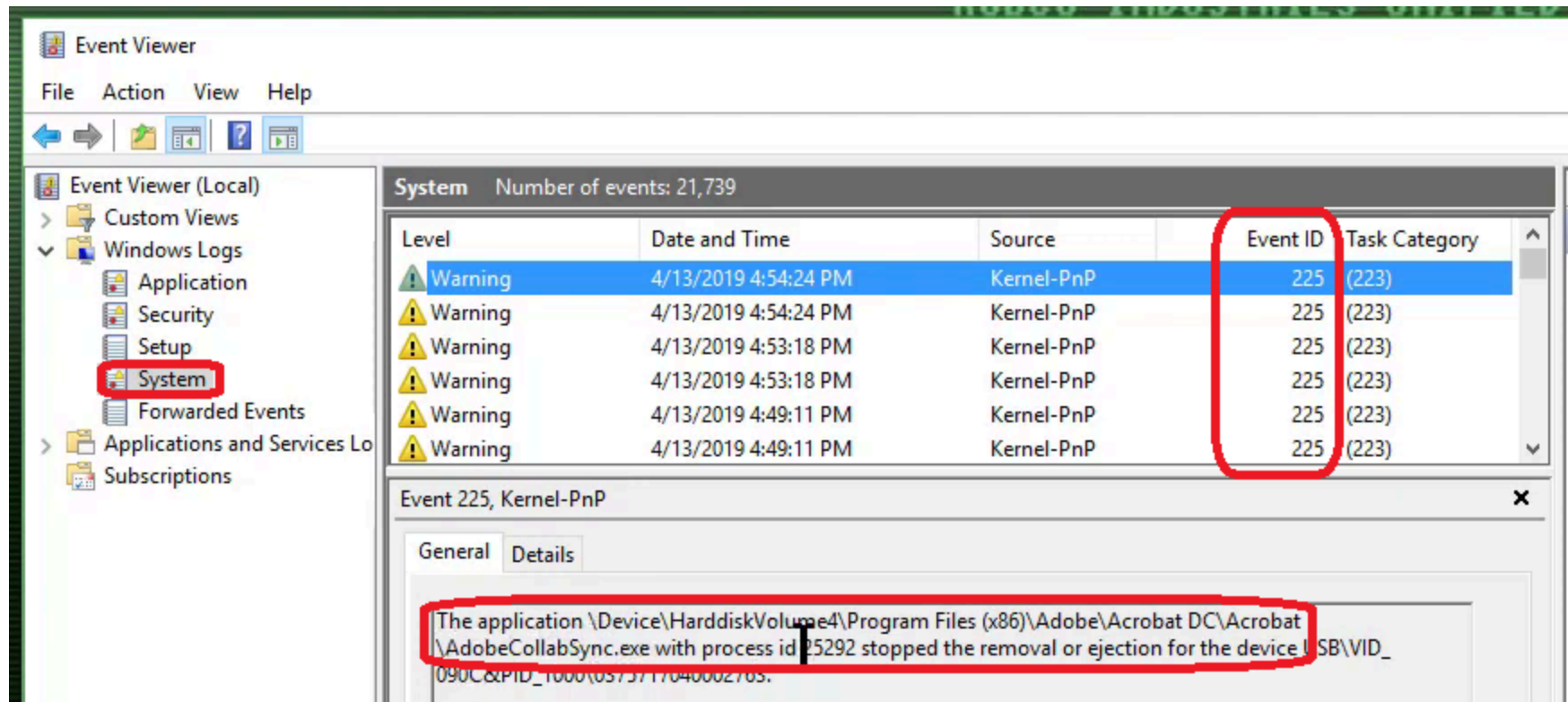
Share Improve this answer Follow

answered May 1, 2020 at 16:54



Sri Reddy
121 2

Event Viewer



- Open *Event Viewer*, for example in any of the following ways:
 - right click on the Start button → Event Viewer
 - or left click on the Start button → start typing "Event Viewer" →
- in the *Event Viewer*, go to Windows Logs → System
- Select recent events with the event ID 225
- Under the list of events, the path and process ID of the culprit will be shown

[source](#)

Share Improve this answer Follow

answered May 3, 2024 at 12:51



root


1,745

2

19

35

That's interesting; I never knew about that. I'm not sure if it answers my question as I was talking about individual files; not so much inability to eject a thumb-drive, but that's still interesting and good to know. – [leeand00](#) May 3, 2024 at 13:33

 **Highly active question.** Earn 10 reputation (not counting the [association bonus](#)) in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

windows

filesystems

process

community-faq-proposed

See similar questions with these tags.