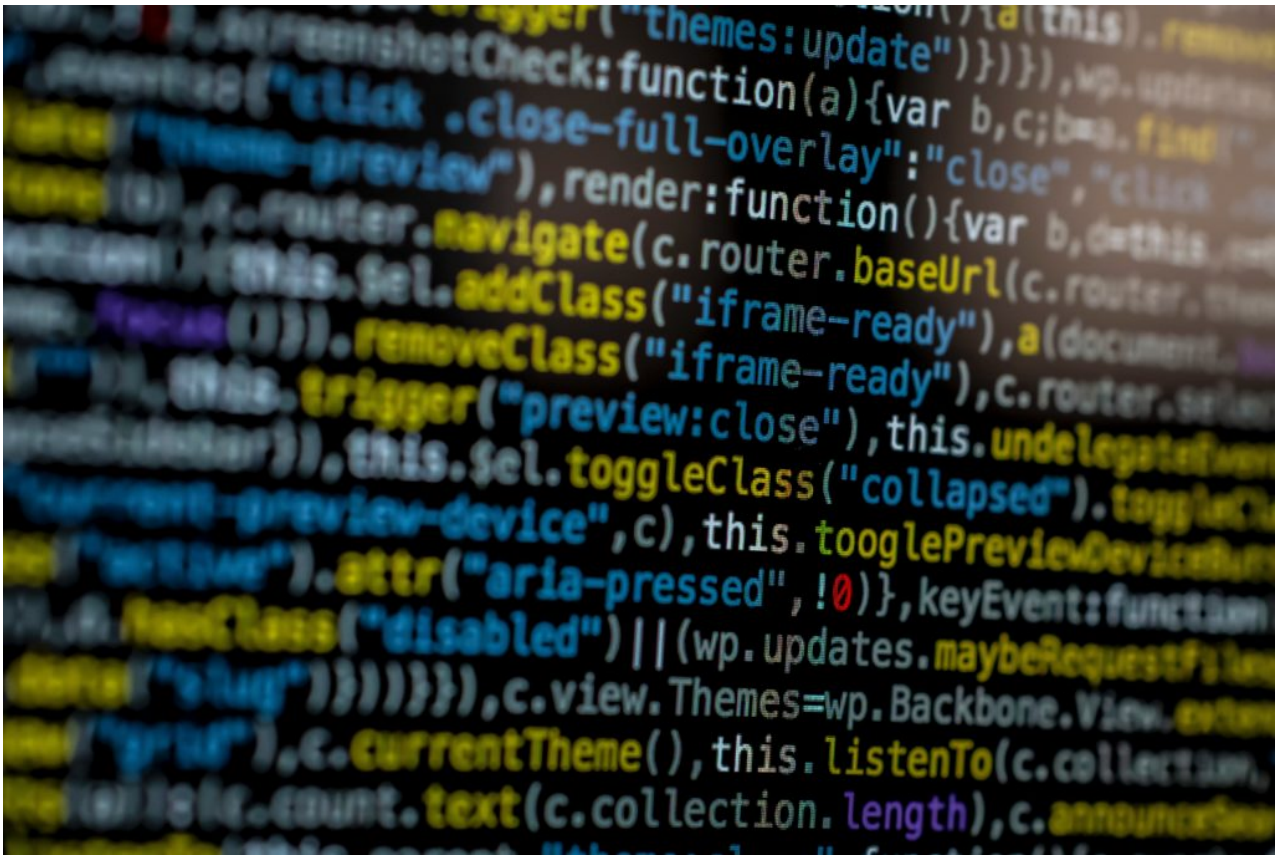
Dropbox Forensics

atropos4n6.com/cloud-artifacts/dropbox-forensics-on-windows-10-part-1/

September 4, 2020



Artifacts of Dropbox Usage on Windows 10 (Part 1)

Published Date : September 4, 2020 , atropos4n6

CAUTION!: This post does not refer to the latest updates of Dropbox app in Windows 10. One colleague of mine, throw me a notice that filecache.dbx is no longer a part of Dropbox installation. So be aware that those results will be valid for older cases/ installations of Dropbox. Thank you.

As I said in me previous posts, I had some research on cloud forensics, about a year ago. I didn't have a blog then, so that research is published with a kinda of delay (only 1 year later!). Today we will see what artifacts remain on a Windows 10 machine, that determine certain type of usage of the Dropbox application. First of all, we will see our setup, scenario and afterwards, we will see our findings.

Setup

My goal was to keep everything as clean as possible. The setup methodology I used was:

- I installed Windows 10 Pro 16299 and Dropbox Client Application 69.4.102 on a brand new VM (Base-VM, using VMware Workstation 14).

- Create a couple of full-clones of the Base-VM.
- I performed a series of actions.
- I acquired the virtual machine's hard drive.
- I examined the images.

Next lets have a look at the super complex scenario of the research.

Scenario

The scenario was pretty basic. I performed each and every of the following actions (wherever applicable):

- Install/Uninstall native Dropbox app.
- Execute the application and synced it with my Dropbox's account.
- Deleted a file.

Lets see our findings.

Findings-Installation

After installing Dropbox app, I found these entries in my C drive:

C:\Program Files (x86)\Dropbox	In this folder you can find the executable file of the application.
C:\ProgramData\Dropbox	In this folder, you can find information about the updates of the application (mostly encoded log files).
C:\Users\ <username>\Dropbox	This folder is being used (by default) by the app for synchronizing user's files with Dropbox cloud service. You can choose from the native app's settings, which directories you wish to be automatically synced with the cloud, but you can also drop any file to this folder and it will automatically be uploaded to the user's cloud.
C:\Users\ <username>\AppData\ Local\Dropbox	In this directory you will find all the native app's files that store information about both the app and the user's data.
C:\Users\ <username>\AppData\ Roaming\Dropbox	Could not safely determine the purpose of the files located here.

Folders and directories created after the installation of the application

If you are trying to determine installation of the application, apart from the registry keys and the above folders, here are the details of a specific **event log** you should look for. Again this is not the most straightforward artifact, but it is an artifact related to the installation of Dropbox application:

Path: ***C:\Windows\System32\winevt\Logs\Application.evtx***

Event ID: ***1040***

Event Description Summary: ***Beginning a Windows Installer transaction.***

Provider Name: ***MsInstaller***

*Event Data: Among others "<EventData><Data>**C:\Program Files (x86)\Dropbox\Update\1.3.189.1\DropboxUpdateHelper.msi5968(NULL)**</Data>"*

Findings – Execution

To determine execution of the application, there is a plethora of artifacts, such as Prefetch and LNK files.

Prefetch

Application Name: **DROPBOX.EXE**

*File Path:**C:\Windows\Prefetch\DROPBOX.EXE-XXXXXXXX.pf***

LNK Files

*Linked Path:***C:\Program Files (x86)\Dropbox\Client\Dropbox.exe**

Findings – Usage

Lets see some of the valuable information that can be stored inside the applications' files and databases.

C:\Users\<username>\AppData\Local\Dropbox**info.json**	This file stores a UID for the specific installation of the application on this machine. This UID is useful in determining which device uploaded a specific file. The UID is stored in "**host:"** variable.
C:\Users\<username>\AppData\Local\Dropbox\instance1**config.dbx**	This database stores information such as user's Dropbox account email.
C:\Users\<username>\AppData\Local\Dropbox\instance1**filecache.dbx**	This database stores information about the files that have been synced with the user's Dropbox account.

IMPORTANT!: Be advised that the above-mentioned databases are encrypted and you cannot view their contents, without decrypting them first. For the decryption of these databases, I used a super useful open-source software called "***Decwindbx***" (Details in creator's blog: https://blog.digital-forensics.it/2017/04/brush-up-on-dropbox-dbx-decryption.html, A big shout out to Francesco Picasso for this great tool!). So after decrypting those dbxs, the difficult part is gone.

You should definitely check the above databases as they are full of valuable information. We will take a glimpse here and will refer to some of their fields, but be advised that there is more information there. If you are interesting in determining the user logged in Dropbox, check these fields:

I was trying to find records determining that a file was deleted, when I stumbled upon a promising table (deleted_fileids) in filecache.dbx. Unfortunately, this table holds exactly what it says. Fileids. What this meant for my research was, that it didn't include only the files that I deleted, but also included files that were just renamed or so. So, I assume that when a file is being renamed, then it change its fileid as well. This also implies that there might be more actions that could possibly change the fileid of a file and therefore populate this table. **So, be careful with these entries. It does not hold deleted files only.**

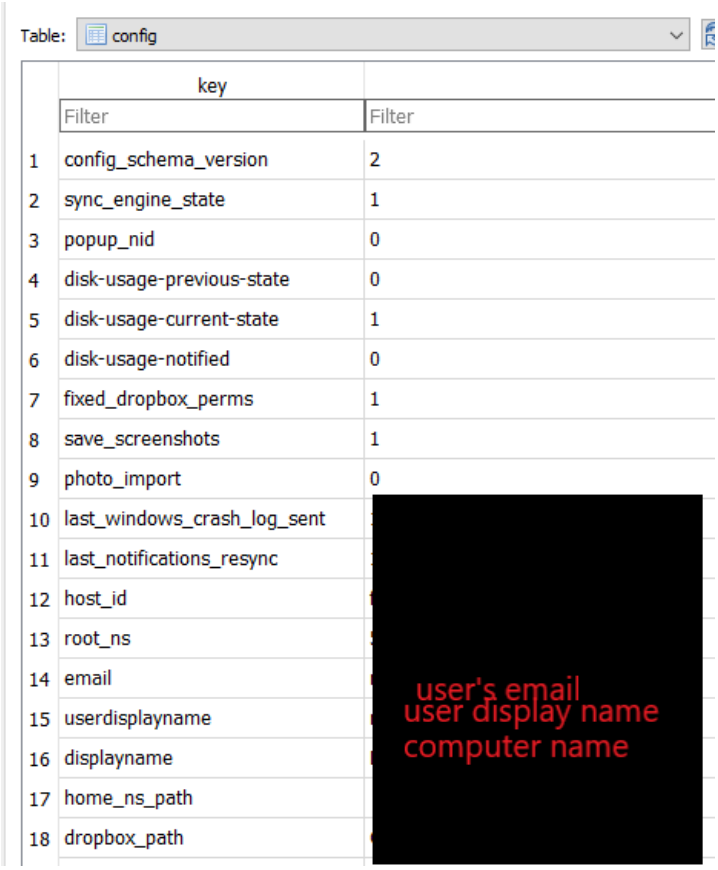


Table: config Database: config.dbx

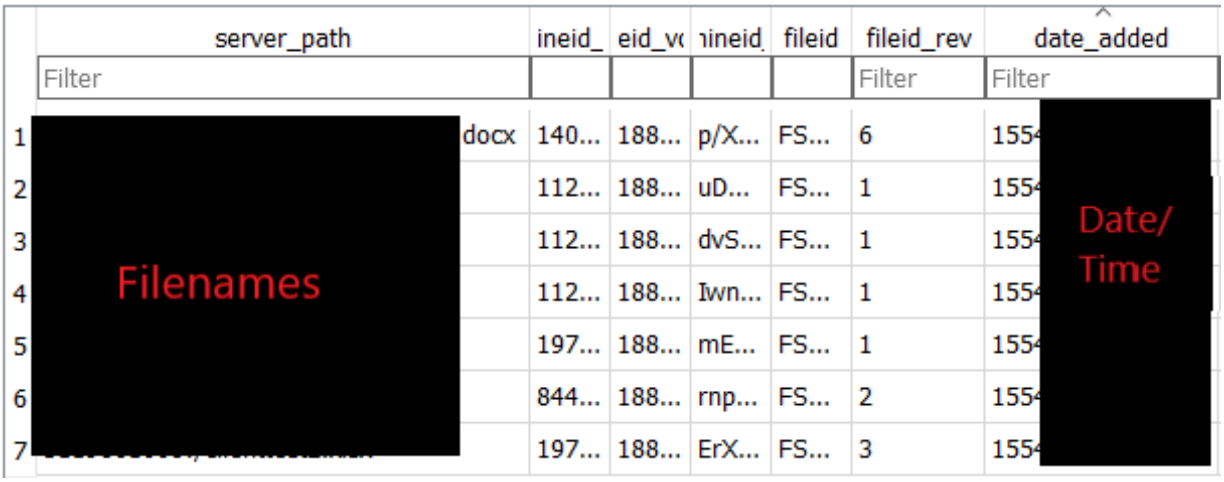
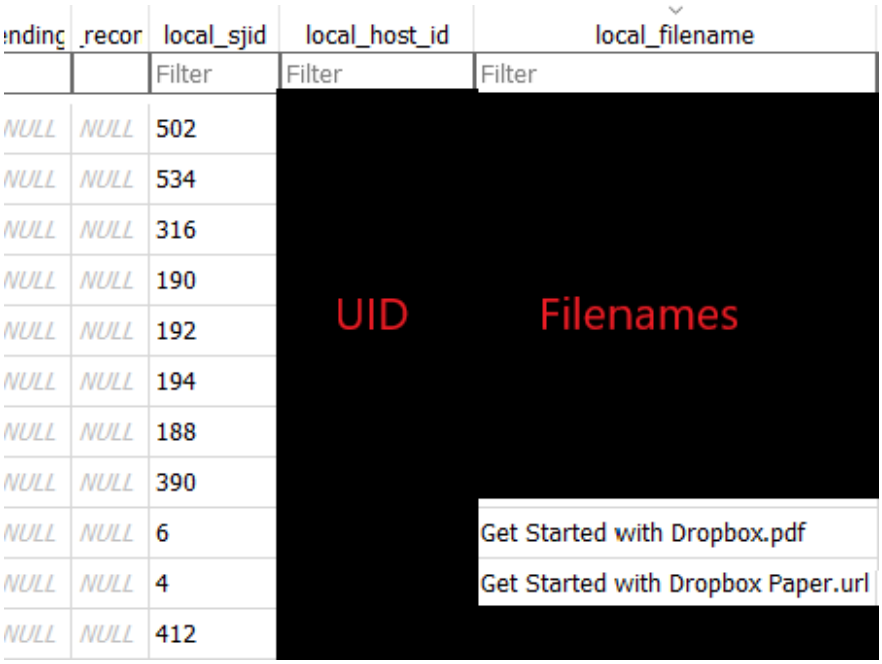


table: deleted_fileids database: filecache.dbx

In the **file_journal** table (holds most file information), I was able to determine which host uploaded each file. This way, you may also find more devices (or more accurately installations) that were linked with the specific account and uploaded files on this account. The UID created with the installation of the application, matches the **local_host_id** field (as seen below):



ending	recor	local_sjid	local_host_id	local_filename
		Filter	Filter	Filter
NULL	NULL	502		
NULL	NULL	534		
NULL	NULL	316		
NULL	NULL	190		
NULL	NULL	192		
NULL	NULL	194		
NULL	NULL	188		
NULL	NULL	390		
NULL	NULL	6		Get Started with Dropbox.pdf
NULL	NULL	4		Get Started with Dropbox Paper.url
NULL	NULL	412		

table:file_journal database:filecache.dbx

Findings – Uninstall

Unfortunately, I did not find any of the above-mentioned important artifacts, after uninstalling the app. I used 2 ways to uninstall it (Windows remove app function and CCleaner) and the results were the same. None of the databases were there. Of course, not everything is bad. When uninstalling the application, it prompts the user to select whether or not he wants to keep the synced files (default option is to keep them). So, the chances are that you may find the synced files at the path mentioned above. What is more, you can prove that the software was uninstalled using the below Event Log:

Path: ***C:\Windows\System32\winevt\Logs\Application.evtx***

Event ID: ***1034***

Event Description Summary: ***Windows Installer removed a product.***

Provider Name: ***MsInstaller***

*Event Data: Among others "<EventData><Data>**Dropbox Update Helper1.3.189.110330Dropbox, Inc.(NULL)**</Data>"*

I know for sure that Dropbox has been updated a lot ever since this research. Dropbox has introduced new features like creating online documents (like Google Docs). This may leave new traces or artifacts behind. I am considering re-researching the topic, but I do not know for sure yet. Tell me in the comments section what you think.

For the last part of these series, we will see Dropbox artifacts that remain on a Windows 10 machine, after using web browsers to access it. Be safe everyone and keep researching!