

Skippipe: Skipping the watermark in digital content

Gerard Fuguet (gerard@fuguet.cat)

Abstract

Words written on a physical or digital format, has an origin. The source can come from an individual or group of these who wants to transmit, document something to show it. The owners are free to impose their rules, like the protection of their works, generating a copyright or/and a watermark subject to a subordinate. It sometimes can create extra content; words, images or growing a file (if it's a digital document). The processes that we focus on this white paper is to remove this extra content from a digital document, thus avoiding the obstacles that degrade it, which makes interpretation difficult.

We will demonstrate how to remove easily the student watermark from a *Microsoft MOC* (official courses) that uses *Skillpipe* as platform to read their digital courses. This situation motivated me to write it.

The intention is that owners of their works, takes consciousness protecting well their content but without disturbing it with possible distortions once protected.

Table of Contents

1. Motivation	3
2. Skillpipe: The eReader	5
2.1. Web App: The Past	6
2.1.1. The Cook.....	7
2.2. Web App: Present	8
2.2.1. The Kitchen Assistant.....	9
2.3. Native Application	10
2.3.1. Dusting It	10
2.3.2. Caching It.....	12
2.3.3. BDB2	16
2.3.4. Tracking It.....	18
3. Water-Engine-Mark	24
3.1. Pipe Factory	24
3.2. Types	25
4. Pipe Leak.....	26
4.1. Manual Repair.....	26
4.1.1. Cast Iron (Old).....	26
4.1.2. PVC (New).....	29
4.2. Semi-Auto Repair	31
4.2.1. Ad Blocking	31
5. Conclusions	33
6. References	38

1. Motivation

Since “*minus 0 ages*”, human cannot stop learning. If you think that, in some type of thing cannot be learned more about it, there is something you need to still see, although your eyes cannot perceive it.

We wanted to know, we wanted to learn more about some *Microsoft* products. You can achieve this lot of beautiful information on Internet as *Microsoft* is popular in IT world, but if you look for a some special (be a specialist) you need to join to the rules of academic world. The time to study the official courses started, they are called: *MOC* [1].

The chosen *Microsoft* partner offers many facilities to study. I chose the face-to-face option in an official/authorized center. Despite having an on-site teacher, the rest of the material was online (books and labs) and the documentation was subject to being read with an application; you can only do it with the help of *Skillpipe* [2].

My big problem originated under my offline study needs. The platform offers the print functionality of the material, but some problems bothered me:

- Watermark acts as an image on every page and this generates extra weight in the document (more than 1000% most of the times).

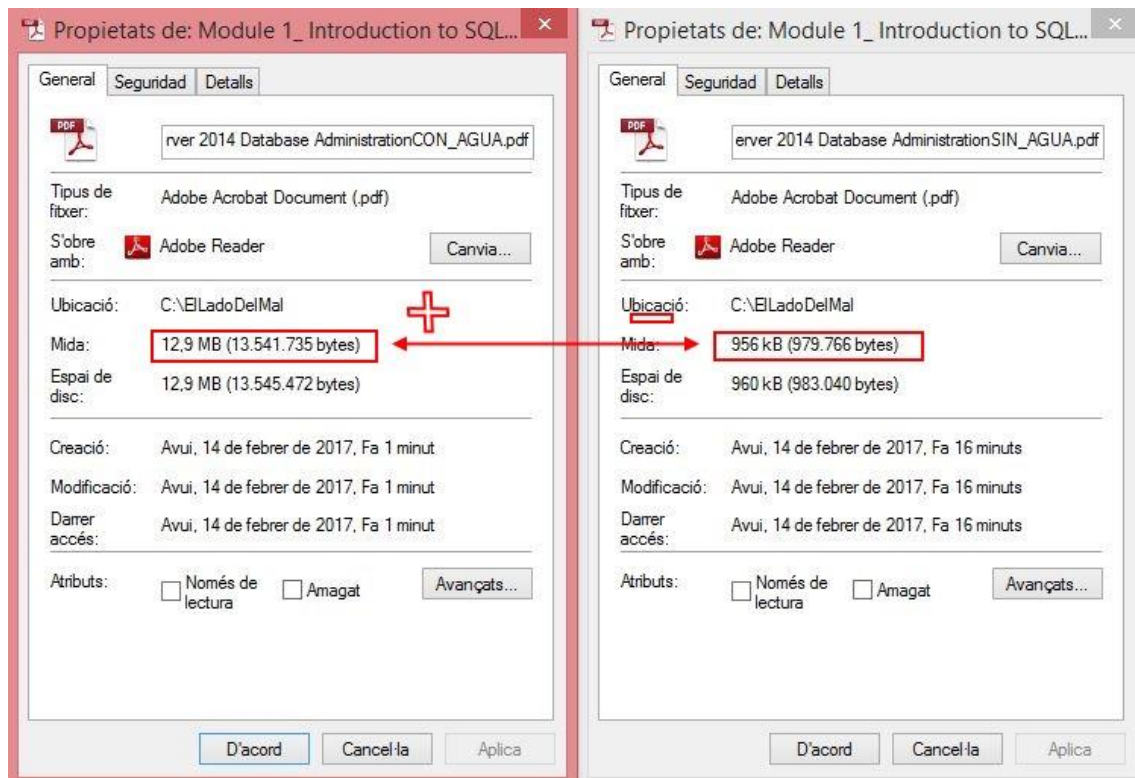


Figure 1: At left, the properties of document with watermark; at right, without watermark

- When you select text around the watermark, part of it can't be catch.

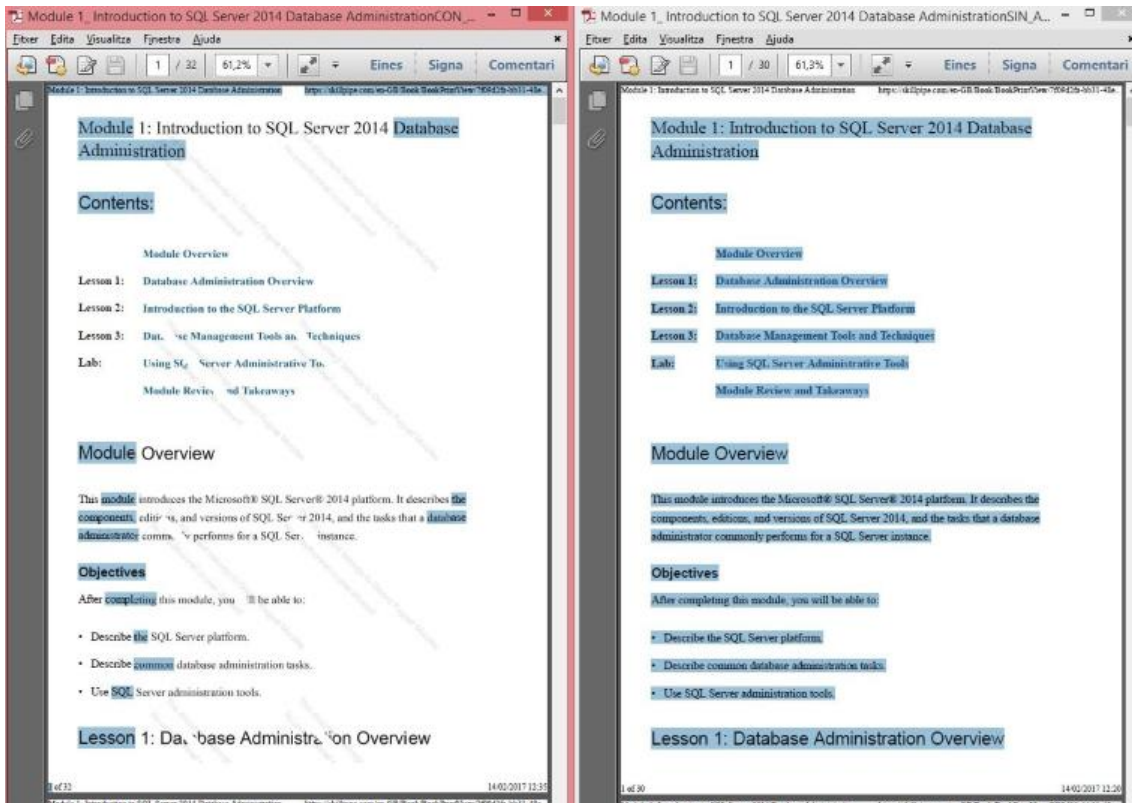


Figure 2: Comparison selecting all the text with and without watermark respectively

- Vertical bars in the corners are generated and produce distortions if some words/letters are in same “road” of the bars.

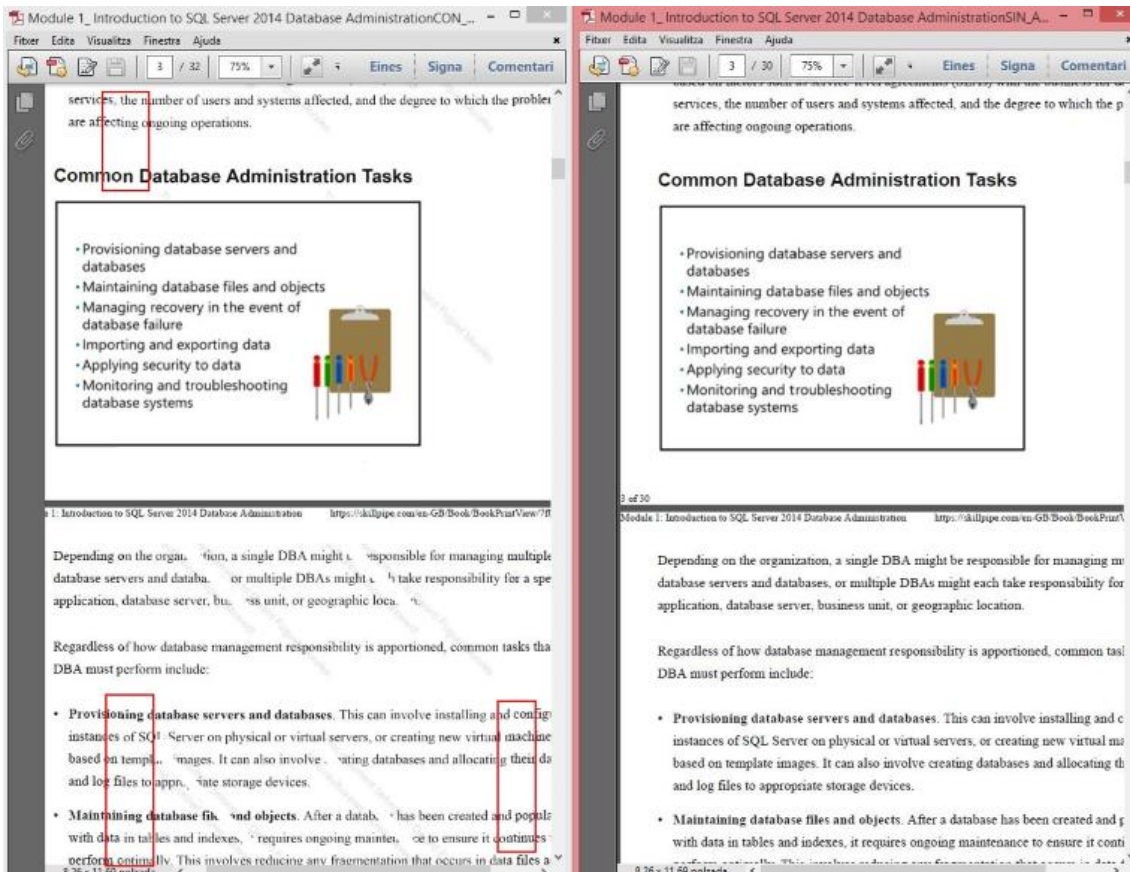


Figure 3: Distortions observed at corners with and without watermark respectively

I think I had enough arguments to delete the watermark from my documents, the intention was never to distribute illegally, of course! :P I wanted to learn comfortably without forcing my eyes, select the text for copy & paste as normal...

I threw some questions to the trainer:

- Is there any solution to select all the text under *PDF* format?
- No
- Well, and... any trick to delete the watermark?
- No, all you can do, is print by modules.

OK, seems nothing to do with it, but didn't convince me. I'm not against the watermarks for protect the rights, but I don't like it to end up as an annoying "leak". Somehow, I needed to "repair" the situation, cleaning or "drying" the watermark with whatever.

2. Skillpipe: The eReader

Is the software, the tool for read the books/documentation via web-based browser, offering only online mode [3]. This isn't developed by *Microsoft*, is made by *Arvato Supply Chain Solutions – Arvato Distribution GmbH* [4]. Delegate the reader part to a third party is not bad, if *Microsoft* take decision to not to develop the reader internally

should be for a variety of reasons that are out of scope here, but externalize some of the things is maybe better if you have a lack of experience (is impossible knowing all when the live is limited!).

The use of the web tool is very easy, maybe this could be one of the reasons to be the perfect candidate for all people. They also have a good user guide explaining all functionalities and the one we are interested on, is in how the web app handle the printing option. We will see two types/manners of behaviors; one is an old methodology used in a previous version (I don't remember the code number), other is in nowadays or *version: 3.0*.

Also, we will take a short walk for the on-premises app that it was intended to be used for offline reading.

2.1. Web App: The Past

This was tested, for about, on the begins of the 2017 year.

The idea is to print all in one “shot”, right? So, after select one assigned course, identify the print button is easy, is located at the left-bottom of the page as shown in figure 4 below:

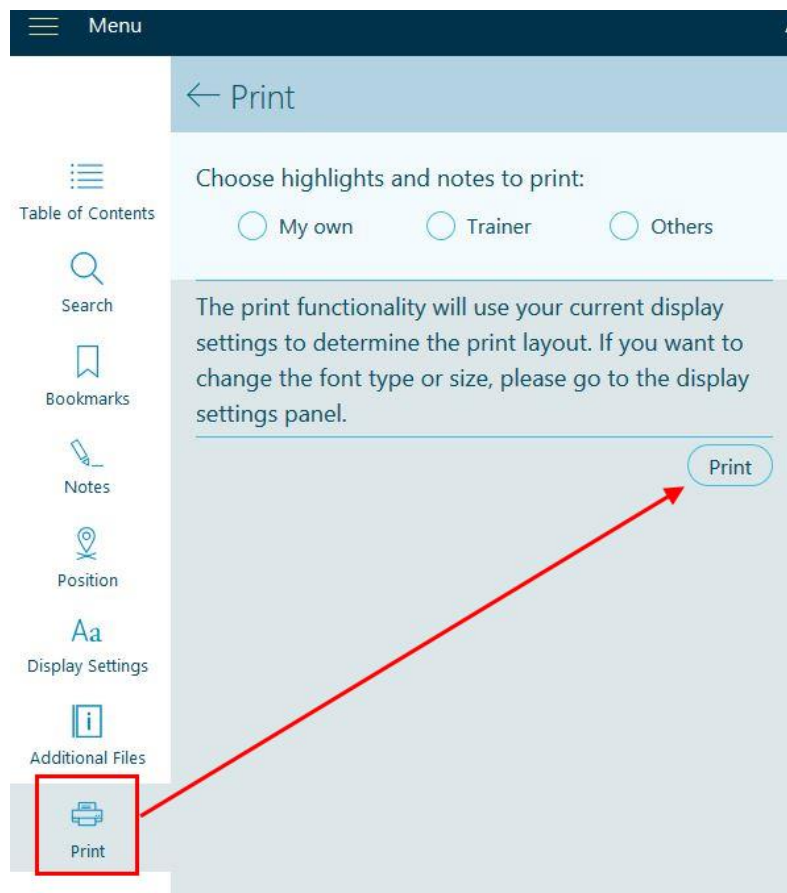


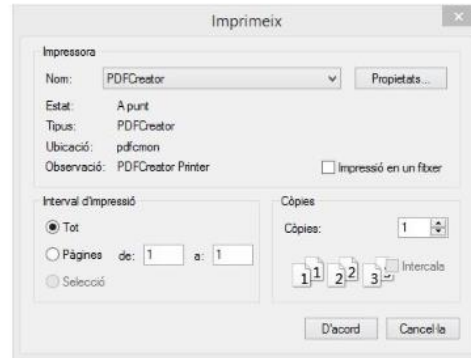
Figure 4: Printing functionality (old version)

Two popups appear: New tab in browser and then the print assistant.

Module 1: Introduction to SQL Server 2014 Database Administration

Contents:

- [Module Overview](#)
- Lesson 1: Database Administration Overview**
- Lesson 2: Introduction to the SQL Server Platform**
- Lesson 3: Database Management Tools and Techniques**
- Lab: Using SQL Server Administrative Tools**
- [Module Review and Takeaways](#)



Module Overview

This module introduces the Microsoft® SQL Server® 2014 platform. It describes the components, editions, and versions of SQL Server 2014, and the tasks that a database administrator can perform.

Figure 5: Printing assistant through new tab

Remember what the trainer said in chapter 1, the book cannot be printed entirely in one time, only the current module prints [5]. You need to choose one module at a time to complete all the book process (*pdf's* in pieces). Then repeat the printing process until complete all modules.

That's all, your *pdf's* will be freshly baked (with hard ingredients).

2.1.1. The Cook

We could “*smell*” in figure 5, the “*fresh*” watermark. This was a web process that is seems to be involved in the moment when the content is printing, the cook put in there the special element and we want to know which the recipe is. The cook is clever if don't want reveal nothing about it (putting it at the end of the process), but don't know we could be “*cook of bits*”.

Using the developer tools of our favorite browser, our magic word to invoke was *watermark*.

We found a node called *print_watermark_wrapper* in the web code:

```
<body class="fontsize_3 font_times textmargin_0 colormode_white">  
  <div class="dtbs_holder">  
    <div class="dtbs_book" style="position: relative;">
```

`<div class="print_watermark_wrapper"></div>`

We discovered where the “*recipe*” was hidden, we just needed get the ingredients and start cooking.

2.2. Web App: Present

Not differs to much from the old version but some differences exist.

The print button is located at same position (left-bottom), the other print button is the responsible to fire up the printing assistant directly, no pop-up new windows like in old web version (section 2.1).

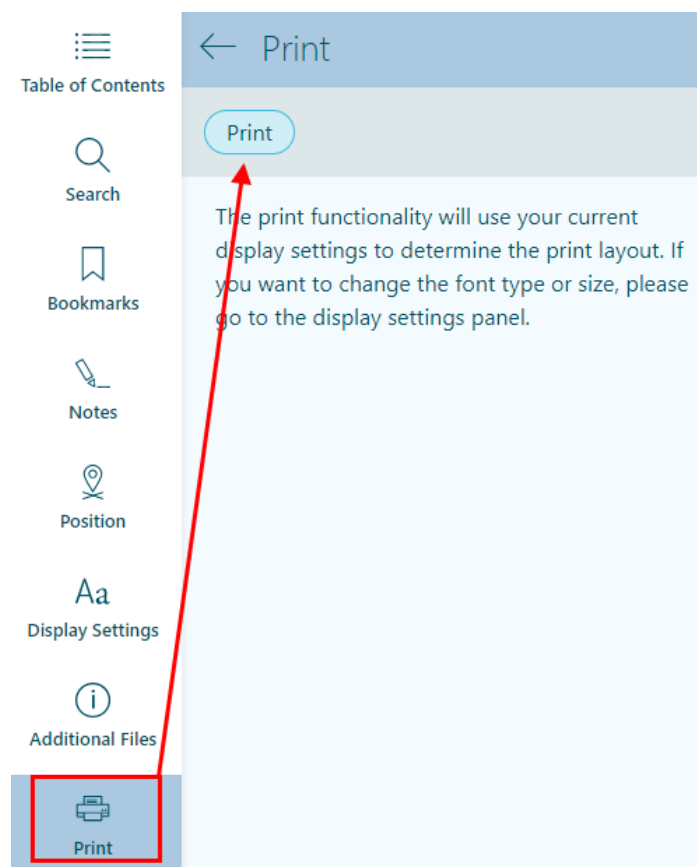


Figure 6: Printing functionality in new version

This second button isn't necessary (has no sense in terms of usability), because there are no other highlights/notes to print.

In next figure 7, we are showing the printing assistant from *Google Chrome* to see how watermark differs.

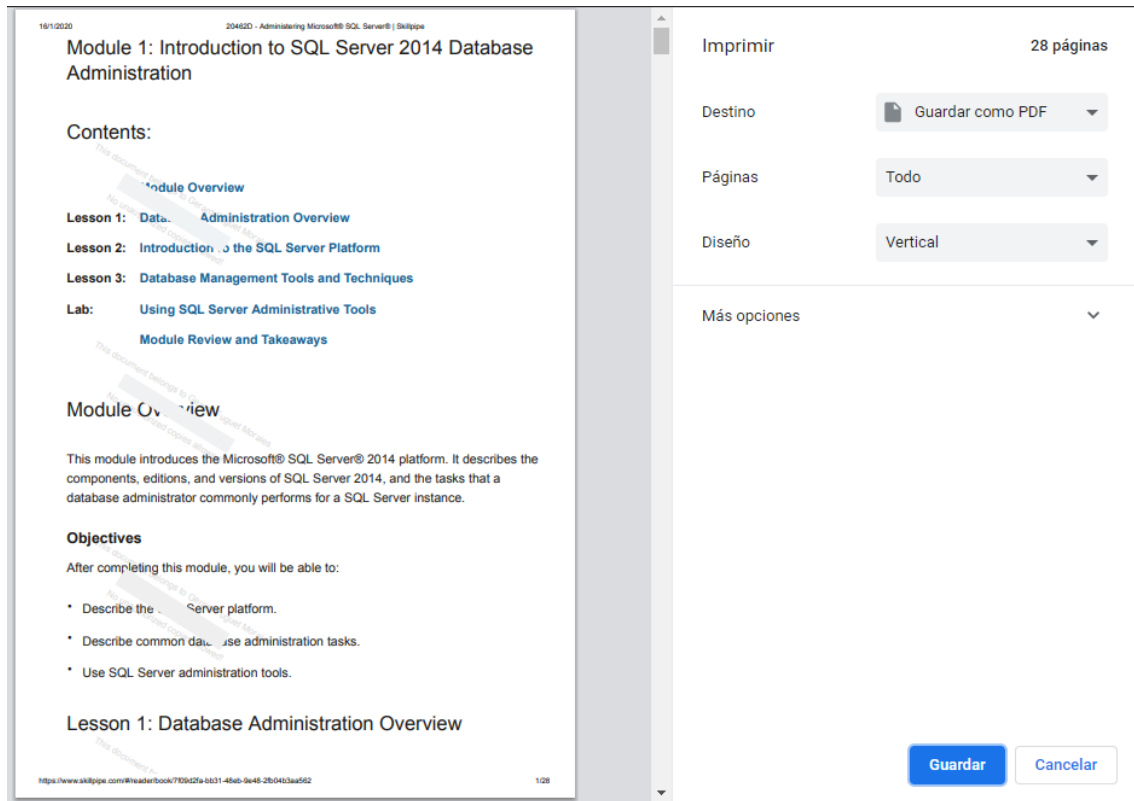


Figure 7: Printing functionality in new version

It's less scattered in document, more vertical than horizontal in comparison as seen in figure 5.

2.2.1. The Kitchen Assistant

Will be nice a little of help to know now where the *“special ingredients”* are.

We used again the developer tools to dissection the *“evolutionistic recipe”* and again, same word to get *“clues”*.

Two matches were discovered, the node it's also called: *watermark*

```

</head>
▼<body style="visibility: visible;" class="at-page emphases-
match-enabled emphases-own-enabled fontSize-3 fontFamily-
undefined textMargin-1">
▼<div id="watermark"> == $0
  ▼<div class="watermark-tile">
    "This document belongs to Gerard Fuguet Morales."
    <br>
    <br>
    "No unauthorized copies allowed!"
  </div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
▶<div class="watermark-tile">...</div>
</div>
▶<div class="dtbs_holder">...</div>
</body>
</html>
</iframe>

```

Figure 8: Watermark node in new version

From the main node, we can see their children, all equal, with the same text repeated several times distinguished as “*watermark-tile*”. So, it was discovered from its hiding place again.

2.3. Native Application

Skillpipe borns with desktop and mobile “*skills*” but they announced the retirement of the offline platforms by end of June 2019, their strategy is, focusing all efforts on one application [6].

The offline apps are no longer available from the official channels, but we still keep the desktop Windows Vista / 7 / 8 application. It will still work? Let's see through “*the dust*”.

2.3.1. Dusting It

We were excited to try it, we wanted to do a quick check and know if some process stops at some point of the circuit. The desktop app is still installed, let's click the *.exe*.

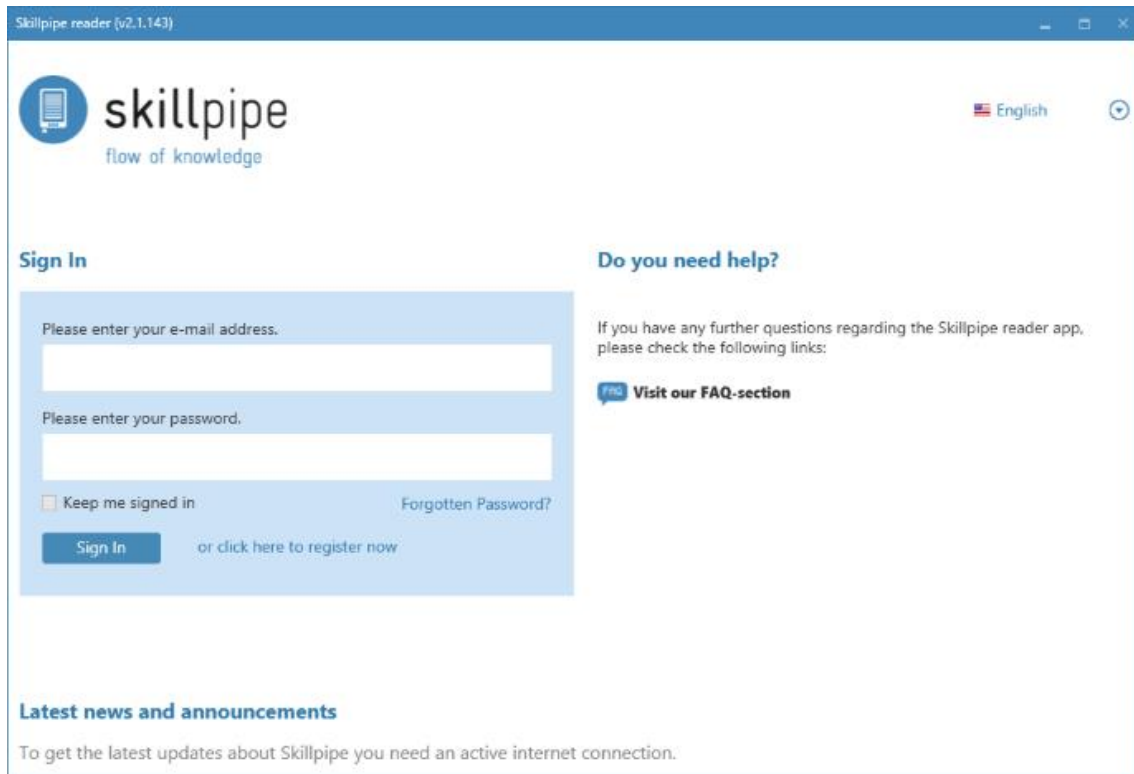


Figure 9: Skillpipe desktop App - Initial screen

First thing eyes pointed was to the version, will be the latest? Appears to be the more up to date under the Windows 7 edition [7]. It's odd that an offline app asks for a login, right? If so, always a direct connection is mandatory, and maybe then you can reside offline until the next re-execution or use the *“Keep me signed in”* (as seen in figure 9) to avoid the credentials in every time.

The mouse was moving doing circles while thinking about it, and we find the *“help”*. The app has a FAQ section that can be queried locally. Seems you need one-time connection to internet to create and synchronize the profile, once done, reading offline can be done. In next figure 10, it describes this process well in two sections under chapter *Skillpipe* reader app for Windows Vista / 7 / 8.

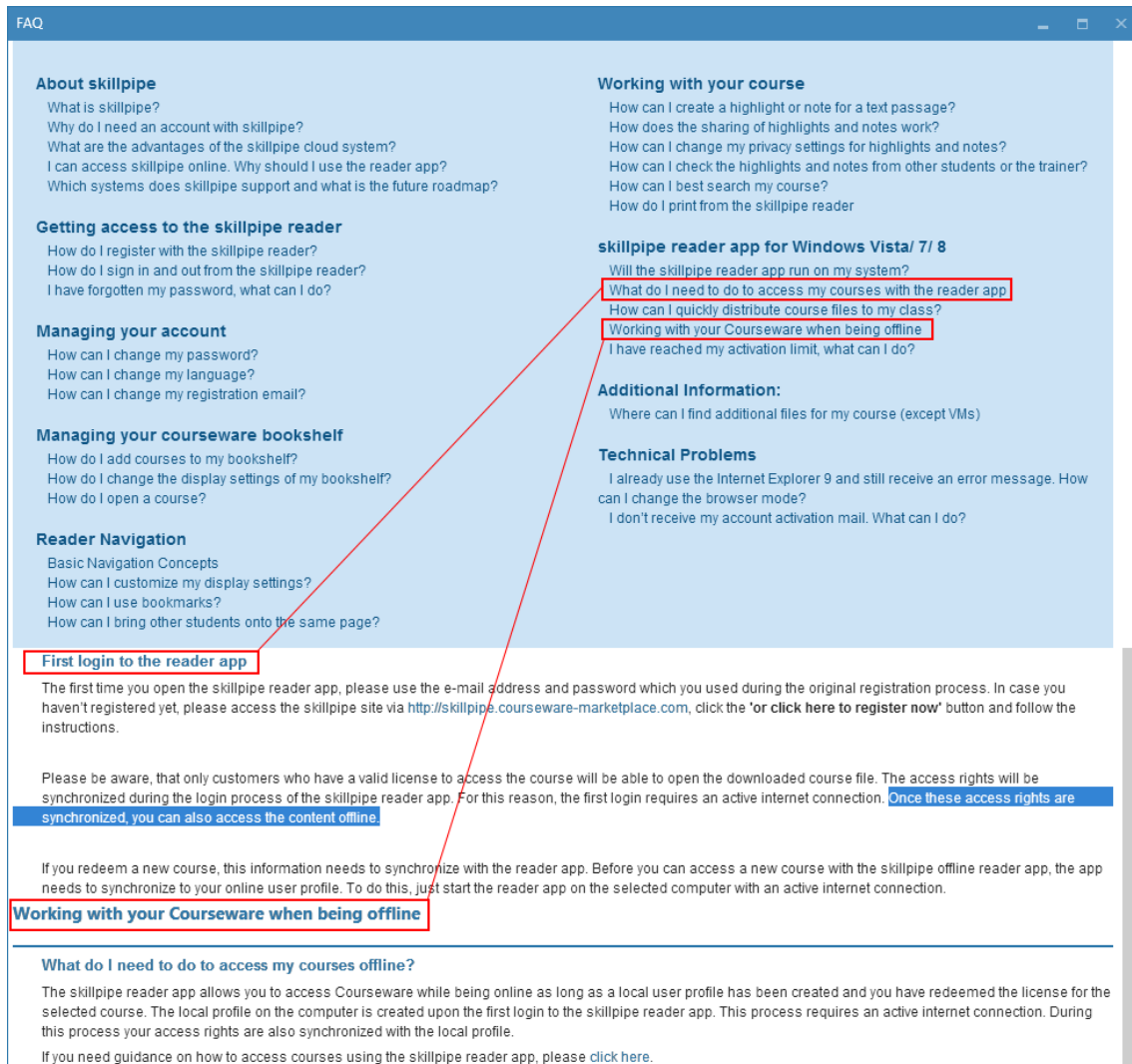


Figure 10: Skillpipe desktop App – FAQ

I had no plans to document this part, but the interest aroused in me to know how the mechanism works to read the book using the credentials.

2.3.2. Caching It

I remember accessing to the app and download a book (in *.bdb2* format) to start interacting with. You can easily open the file with the corresponding extension (software has the extension associated). The steps can be found in a *YouTube* video [8]: Access to the *Skillpipe* Web App, then choose the book course, then download the file version and open it. Finally, you need introduce again the credentials (on every app execution).

As a test, we tried open a *.bdb2* we had, and login with our email as user but different password to ensure if the connection against old *Skillpipe* desktop infrastructure is still alive, and... Check your internet connection?

Sign In

Please enter your e-mail address.

Please enter your password.

Keep me signed in [Forgotten Password?](#)

or [click here to register now](#)

ⓘ Unable to verify username and password. Please check your internet connection.

Figure 11: Skillpipe desktop App – Login Unchecked

Internet connection at the moment of the test was fine! Let's see what happen if try to open the links *Forgotten Password?* and *or click here to register now*. Links are: <https://www.skillpipe.com/reader/en-GB/Account/ForgottenPassword>, <https://www.skillpipe.com/reader/en-GB/Account/Register> respectively. Links appear to be down, no longer exist (and at all moment, the *Keep me signed in* box is disabled).

Tried using the correct credentials and... Access granted! All the book functionalities appear working OK, you can select the different modules and no error was found. The profile should be store in some place of the computer...

In the config options, there is a function to retain the cache. It seems also contemplates the credentials.

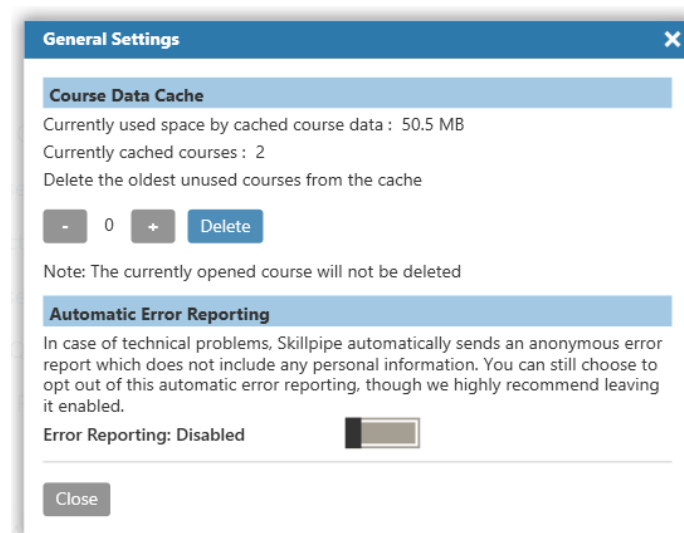


Figure 12: Skillpipe desktop App – Cache Options

The tree of folders and content it's as follows:

```
%USERPROFILE%\APPDATA\LOCAL\ARVATO\SKILLPIPE READER  
  |--cache
```

```

/ aps.v2
/ cd.v4
/ ebm.v2
/ n.v1
/
+---Books
/ +---619fb4b7-e582-487d-9bc1-8fb028766fc4
/ / \---7
/ / / SI.v1
/ / /
/ / +---additions
/ / / \---images
/ / / *.png
/ / /
/ / /
/ / +---html
/ / / chap*.html
/ / /
/ / \---meta
/ / additionalcontents.dat
/ / meta.xml
/ / thumbnail.png
/ / toc.xml
/ /
/ \---7f09d2fa-bb31-48eb-9e48-2fb04b3aa562
/ / \---2
/ / / SI.v1
/ / /
/ / +---additions
/ / / +---images
/ / / / *.png
/ / / / cover.png
/ / / /
/ / / \---json
/ / / *.json
/ / /
/ / +---html
/ / / chap*.html
/ / /
/ / \---meta
/ / additionalcontents.dat
/ / meta.xml
/ / thumbnail.png
/ / toc.xml
/ /
\---Users
  \---user__account
    / ad.v1
    / c.v1
    / l.v2

```

```

/ p.v1
/ ubi.v1
/ us.v1
/
\---uc
+---619fb4b7-e582-487d-9bc1-8fb028766fc4
/ ac.v1
/ s.v1
/ ucs.v1
/
\---7f09d2fa-bb31-48eb-9e48-2fb04b3aa562
s.v1
ucs.v1

```

Inside of **SKILLPIPE READER** folder, there is a cache folder with files which have a non-related extension with a software (unknown). There are other important folders; **Books** and **Users** but the extensions .v1, .v2, appear in, not only under **cache** folder, also spread in the rest. The exception is with .v4 only is found in **cache**. All the .v* appears to be encrypted (major of content is illegible).

The **Books** seems contains all the digital documents associated with the student. Folders **619fb4b7-e582-487d-9bc1-8fb028766fc4** and **7f09d2fa-bb31-48eb-9e48-2fb04b3aa562** are associated with *20331b-core-solutions-of-microsoft-sharepoint-server-2013-v7.bdb2* and *20462d-administering-microsoft-sql-server-v2.bdb2* books respectively. The images located in **images** folder are not encrypted, neither files *meta.xml* and *toc.xml* but *chap*.html* yes. Other folders from main **Books** folder, seems to be the document structure.

Users folder contains other folder with the **username** as a name, with .v*, but the most interesting is there are the folders of the books (same name) that are under **Books** folder, in that case, only .v1 inside. One of them has one more .v1, the *ac.v1* maybe it reference to last read position.

Well, can't spend time on reverse engineering, but one thing to try is copy the **cache** folder and paste into other user account created in same computer to see if still works. And worked! Next challenge was copy the contents but into other computer, and... it fails! The error is the same as seen in figure 11 (and yes, the user and password was correct!).

My idea was virtualizing the machine just in case some registry value or other file in wherever were missing. Using this method, we could be sure that it is not about leaving any files along the way.

The objective is converting it to a *VirtualBox* machine as destination. We used the *Disk2vhd* from *Sysinternals* [9]. We unchecked the options of 'use Vhdx', and 'Use Volume Shadow Copy' [10]. Conversion was OK, taking some long time but without any error. Turned on and after a while, no difference from original *Windows*, so the clone was almost perfect. Now, you want to know how was the result? Another fail!!

OKI Doki! So, the encryption mechanism is using something specific for every machine (For what it seemed). And what about the inspection of a book in their *BDB2* format?

2.3.3. BDB2

It consists, in see if there is any tool to dissection the book and *7-Zip* was chosen for this purpose.

For the test, we used the *20462d-administering-microsoft-sql-server-v2.bdb2*. The content is the same as seen in the tree of the chapter 2.3.2. the only difference is that the file *SI.v1* is not in the *BDB2* skeleton.

I wanted to do a quick comparison taking the structure of an *EPUB* Before inspecting in depth. Let's show if *BDB2* has some similitudes [11]:

```
/ Mimetype file (Archive)
+---META-INF
/   container.xml
\---DOCUMENT folder
     contains HTML, CSS, image files, plus OPF and NCX files
```

Despite having a similar structure, *Skillpipe* seems to adopt other different mechanism (plus something with *DRM*) although it seems to have good intention/inspiration with its appearance in xml as the toc file (*toc.xml* from *Skillpipe* against *toc.ncx* from standard *EPUB*).

In *.bdb2*, almost all the important content is encrypted like the html files. You can extract other content, part of the book, but is like having “*souvenir postcards*”, if you don't have all the pieces, you never see the entire “*picture*” and many hypothesis should appear.

To be a collector of *Skillpipe BDB2* books, your files are **.json*, **.png* and **.xml*:

- *JSON* (*\additions\json*): files *.json* shows the *Check Your Knowledge* section. There are many type of challenges (from the book), you need choose correct answers or drag & drop...
- *XML* (*\meta*): *meta.xml* shows the description of the course. *toc.xml* the table of contents (The chapters are related to the html files).
- *PNG* (*\additions\images*): All the *.png* images of the modules.

Nothing to do with encrypted books but... Why not look to the book without dismembering it? Let's go for the magnifying glass!

We opened with *notepad++* and we review first words at the beginning of the first line. Three sections can be defined here.

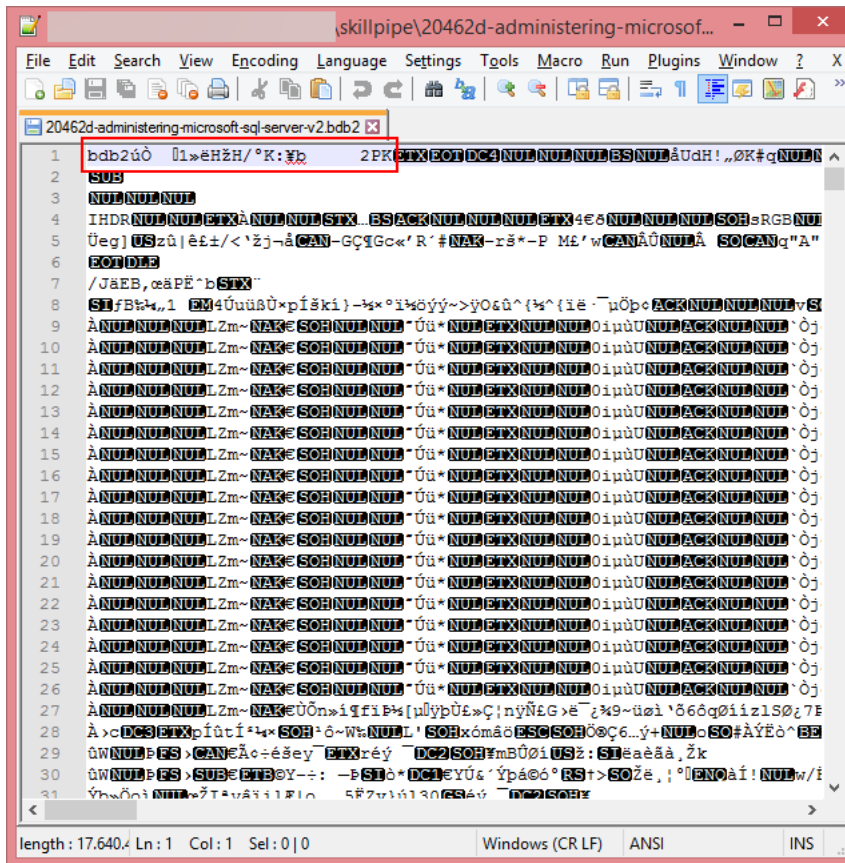


Figure 13: Opening/inspection a .bdb2 with notepad++

According to the figure 13, the first understandable letters are *bdb2*. This is the extension that software will use to identify as their book.

The “*strange*” following letters could be something unique related with the license of the student.

The number, 2 in this case, is the code of the update of the book, this code can be seen in the tree, above in chapter 2.3.2 following the folder with name of the book id *7f09d2fa-bb31-48eb-9e48-2fb04b3aa562*. This can be also seen in the *meta.xml* file, as the following figure 14 shows:

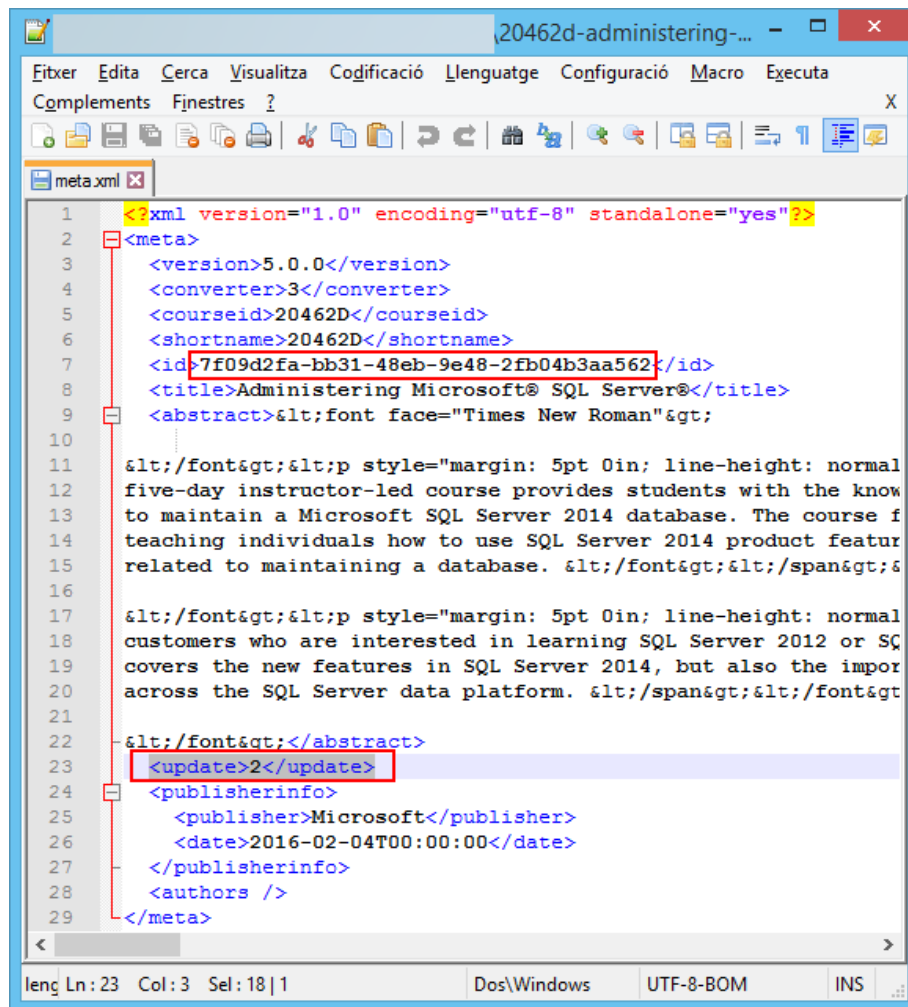


Figure 14: Opening/inspection *meta.xml* with *notepad++*

It demonstrates number and *id* are used as identifiers.

The two last words, *PK* are not used when we have the books in cache. We created a blank file with any name and extension *.bdb2* with this content:

```
bdb2úÒ    □ l»ëHžH°K:¥b  2
```

Open – put the credentials in *Skillpipe*, and it worked like a charm :D ! (Seems data is consumed from the cache instead from the book file).

For curiosity, we could know what files are implicated from the login until the book is open.

2.3.4. Tracking It

In order to know how the encryption mechanism is working, need some tool that helps on this research, something that show us the accessed files by *Skillpipe* ordered by time.

If you work in this beautiful IT world, and you challenge with servers, networks... (like a system administrator) You may know the great tools from *NirSoft* [12]. In all the variety of tools *NirSoft* has developed, the suitable for this job is *FileActivityWatch* [13]. This

tool will help to display what files are accessed by *Skillpipe* in time order, and of those, if is read, write, bytes on each action... Simple, good GUI and easy to track.

We started with the origin computer where is working well and then with the Virtualized one.

Origin computer:

Before proceeding, need to prepare the filter correctly to avoid “*noise*” (because *FileActivityWatch* gather all info of the files access in computer from all the processes). In advanced options, the option can be set up. Now need filter the filter, we mean, capture only in the moment of do a correct login avoiding the entire opening of the .exe-cutable. To accomplish this, capturing can be stopped with *F2* key, and if for some reason there are “*garbage*” can be cleaned by *Ctrl+X*.

Time to click on Sign In *Skillpipe* button, and the following files has been accessed:

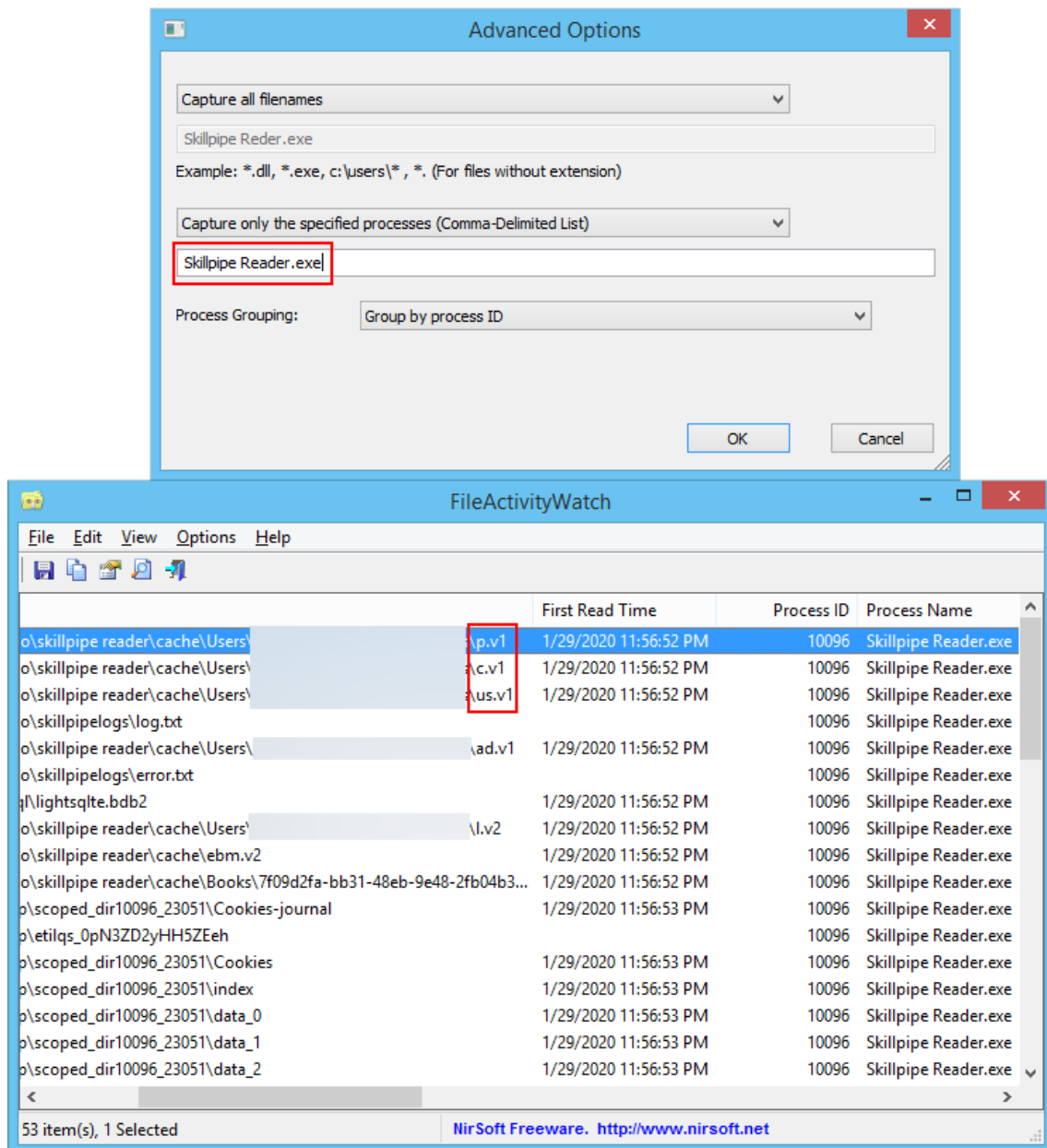


Figure 15: FileActivityWatch – Filtering process in Advanced Options and File inspection

From top to down, the *p.v1* is first file accessed preceded by the files *c.v1* and *us.v1*. Seems a check for the password, and these in conjunction a check with some computer algorithm to be unique. Then at line 4 in same figure 15, *log.txt* file is touched and after two more lines, is the turn of *error.txt*.

That is all? Why not take a look in logs of the application? The log folder is found in: *%localappdata%\arvato\skillpipelogs* and from here, two text files (*.txt*). One is *log.txt* and the other *error.txt* both maintain six more versions, in this prefix/format; *log.txt.year-month-day* and *error.txt.year-month-day*.

A good idea could be correlate by time, so if *FileActivityWatch* say that at *11:56:52 PM* let's see by this time, opening the *log.txt* (because the *error.txt* don't show important because all was fine!):

2020-01-29 23:56:52:5652 [5] INFO Remote logging disabled by user preference.
 2020-01-29 23:56:52:5652 [1] WARN Calling SaveBookUserSettingsAsync when
 UserContext.IsComplete = false
 2020-01-29 23:56:52:5652 [7] INFO The operation 'Load book index' took '185' ms
 2020-01-29 23:56:53:5653 [1] INFO The operation 'chapter loading' took '631' ms

The information we obtain is three *INFO* (information) and one *WARN* (warning). Good news is book could load correctly, and take less than a second. The *WARN* appears to be a function of the software that seems not to be more important.

Don't get tired yet, there is another tool that could be great to use... is *Skillpipe* using some port for connections (internally or externally)? *Process Hacker* [14] has "power glasses" to say something about, and can filter by the application name:

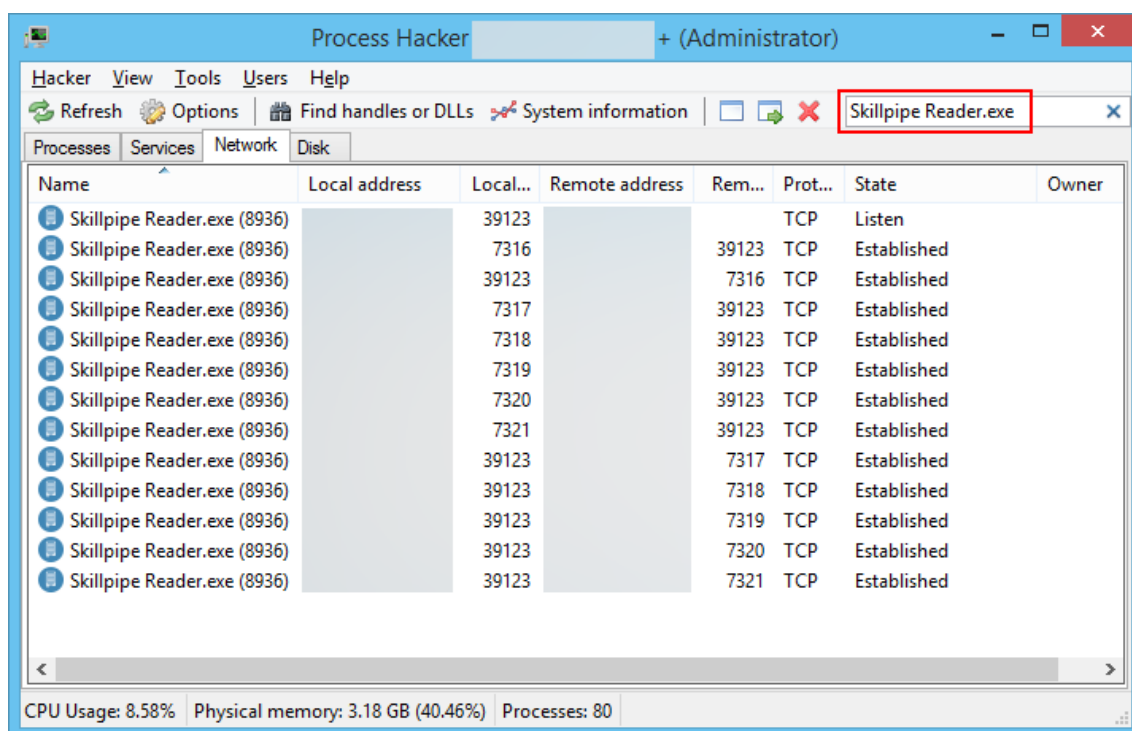


Figure 16: Process Hacker – Network filtering

As note; the local and remote addresses are covered because is localhost, but shows the FQDN of my test machine.

The port 39123 is open at the beginning (listen), and while book is open, there is a bidirectional internal communication. Six connections are open. 39123 against 7316, 7317, 7318, 7319, 7320, 7321 and vice versa. It could open six connections to balance the load or to distribute parts of functions in the program.

Maybe doing same operation in the cloned virtual machine more context could get, who knows!

Cloned virtual Computer:

Let's use again the *FileActivityWatch* with same steps described above. And the result is shorten:

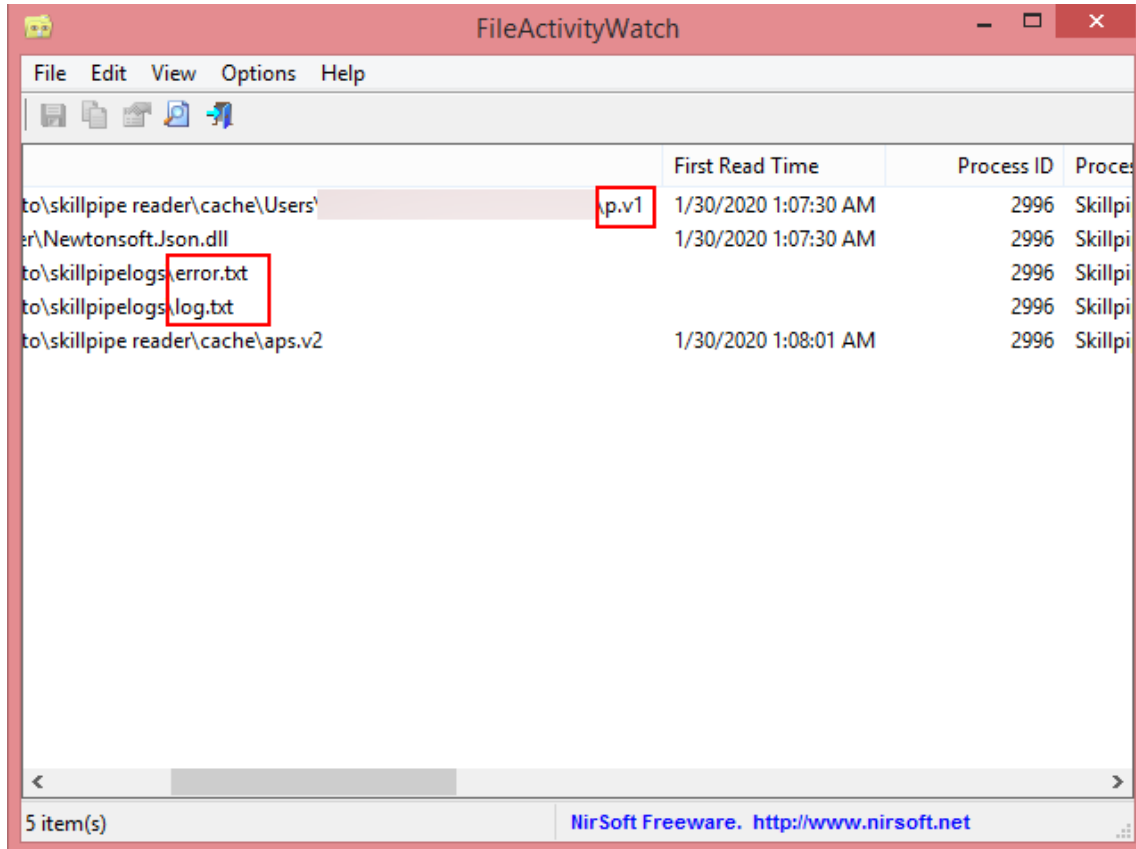


Figure 17: FileActivityWatch – Filtering process at virtual cloned machine

It pretends read and interpret the *p.v1* file but seems without success because we see the *error.txt* is first wrote than the *log.txt*. Next step? See what happen in *error.txt* (again, correlate by same time):

```
2020-01-30 01:07:30:730 [5] ERROR Error occured while reading or deserializing a
file: C:\Users\user\AppData\Local\arvato\skillpipe reader\cache\Users\user\p.v1
System.Runtime.Serialization.SerializationException: Deserializing '? ??h*gf??'??{-
??A
o!z??????<??(?C?hZJ?m???-Ku3??Z?????%Pu??T7?
```

```

?H?z?!l??• ??;?D?(n?P/?m??B??r???=????????8?-
??----->?J• ?• ??#?????m? ?!6{??\??' failed --->
Newtonsoft.Json.JsonReaderException: Unexpected character encountered while
parsing value: ?. Path "", line 0, position 0.
  at Newtonsoft.Json.JsonTextReader.ParseValue()
  at Newtonsoft.Json.JsonTextReader.ReadInternal()
  at Newtonsoft.Json.JsonTextReader.Read()
  at
Newtonsoft.Json.Serialization.JsonSerializerInternalReader.ReadForType(JsonReader
reader, JsonContract contract, Boolean hasConverter)
  at
Newtonsoft.Json.Serialization.JsonSerializerInternalReader.Deserialize(JsonReader
reader, Type objectType, Boolean checkAdditionalContent)
  at Newtonsoft.Json.JsonSerializer.DeserializeInternal(JsonReader reader, Type
objectType)
  at Newtonsoft.Json.JsonConvert.DeserializeObject(String value, Type type,
JsonSerializerSettings settings)
  at Newtonsoft.Json.JsonConvert.DeserializeObject[T](String value,
JsonSerializerSettings settings)
  at Arvato.DTBS.Common.Infrastructure.JsonNetJsonConverter.GKC`1.C()
  at System.Threading.Tasks.Task`1.InnerInvoke()
  at System.Threading.Tasks.Task.Execute()
--- End of stack trace from previous location where exception was thrown ---
  at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)
  at
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotific
ation(Task task)
  at Arvato.DTBS.Common.Infrastructure.JsonNetJsonConverter.HKC`1.I()
  --- End of inner exception stack trace ---
  at Arvato.DTBS.Common.Infrastructure.JsonNetJsonConverter.HKC`1.I()
--- End of stack trace from previous location where exception was thrown ---
  at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)
  at
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotific
ation(Task task)
  at System.Runtime.CompilerServices.TaskAwaiter.ValidateEnd(Task task)
  at A.WP`1.TP.M()

```

Amount of words, more difficult to read (lot of errors). Basically, it fails to validate the correct routines for *p.v1* file (important lines are underlined in code).

Is important look for the exact times, but if we see the before log time? Let's see so:

```

2020-01-30 01:07:09:79 [5] ERROR Error occured while reading or deserializing a file:
C:\Users\fuguet-gerard\AppData\Local\arvato\skillpipe_reader\cache\n.v1
System.OverflowException: Arithmetic operation resulted in an overflow.
  at Arvato.DTBS.Crypto.Implementation.BlowfishCrypter.Decrypt(Byte[] key, Byte[]
dataToDecrypt)
  at Arvato.DTBS.Crypto.Implementation.BlowfishCrypter.G.E()

```

```

    at System.Threading.Tasks.Task`1.InnerInvoke()
    at System.Threading.Tasks.Task.Execute()
--- End of stack trace from previous location where exception was thrown ---
    at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)
    at
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotific
ation(Task task)
    at Arvato.DTBS.Common.Core.Crypto.MachineSpecificCrypter.MDC.H()
--- End of stack trace from previous location where exception was thrown ---
    at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)
    at
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotific
ation(Task task)
    at System.Runtime.CompilerServices.TaskAwaiter.ValidateEnd(Task task)
    at A.WP`1.TP.M()

```

It is talking about Blowfish algorithm designed by *Bruce Schneier* [15]. And about MachineSpecific... So, the algorithm uses something of your machine to be able to decrypt the content. File *n.v1* appears to be “*an essential key*” in conjunction with the machine/computer to achieve the “*clear work*” of decryption.

We better “*not bothering the blowfish*”, lest we get poisoned!

3. Water-Engine-Mark

The native application does not work; it is not our mobility solution, not only because we cannot move freely between devices, but because printing seems impossible...

Returning to the watermark; it’s digital and hides the content using an steganographic technique. Aims the intellectual protection of the digital object (copyright). [16]

We are going to simplify the process adapting it to the specific case of the application, we will not deepen much in all its aspects, but some components need to be exposed, must be explained to understand its mechanics.

3.1. Pipe Factory

Let’s show first a diagram and going to explain the legend:

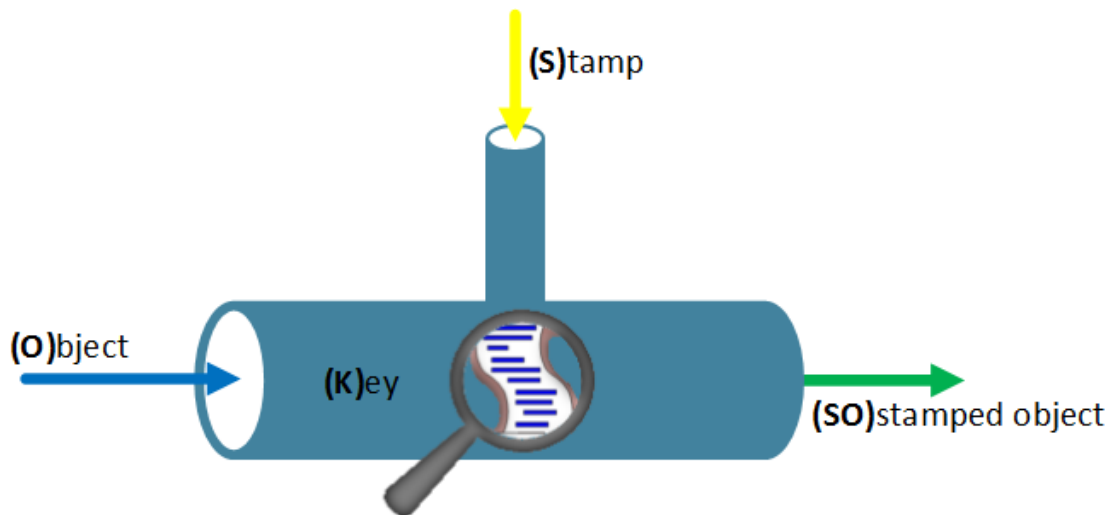


Figure 18: Watermarked Process

The pipe simulates the marked of the process (where watermark is made), is like the processes of a factory:

- (O)bject: Is the document to be subjected to stamping.
- (K)ey: Is who let you access to the marked of the process and to the object/document. In this case the key are the student credentials to access to *Skillpipe* portal. Without the credentials, no object (nor stamped).
- (S)tamp: Is the watermark to be inserted in the object/document.
- (SO)stamped object: Is the result of (O)bject & (S)tamp, it means, it produces the object with the watermark.

3.2. Types

There are different watermark styles (it depends on the object and if it is hidden or uncovered), for those, we will identify what *Skillpipe* uses:

- Private: Original (O)bject is needed
Type 1: $SO \times O \times K \rightarrow S$
Type 2: $SO \times O \times K \times S \rightarrow$ Stamped/NoStamp
- Semi-private: The (S)tamp is needed
 $SO \times K \times S \rightarrow$ Stamped/NoStamp
- Public: Not need the original object (O) neither the stamp (S)
 $SO \times K \rightarrow S$
- Visible: The watermark here is visible, through the key (K), the original object can be retrieved without losses. The stamp (S) can be pulled off as if it were another layer
 $SO \times K \rightarrow O$

After view all the types, is easy to identify the case of the *Skillpipe*, visible type is the correct formula, because with the credentials, you have access to the material and the

content can be seen without the watermark. The problem still exists (and persists) at the printing stage.

4. Pipe Leak

Have you ever had a leak in a pipe? Well, it depends on the home and the age of it or quality of pipe materials, but surely, you have already seen it in the street (water pipe as most typically). No one would think of breaking one of them, because the disaster that is created is very scary! But...! the pipe of *Skillpipe* "is dirty", we need to break it to repair and clean it.

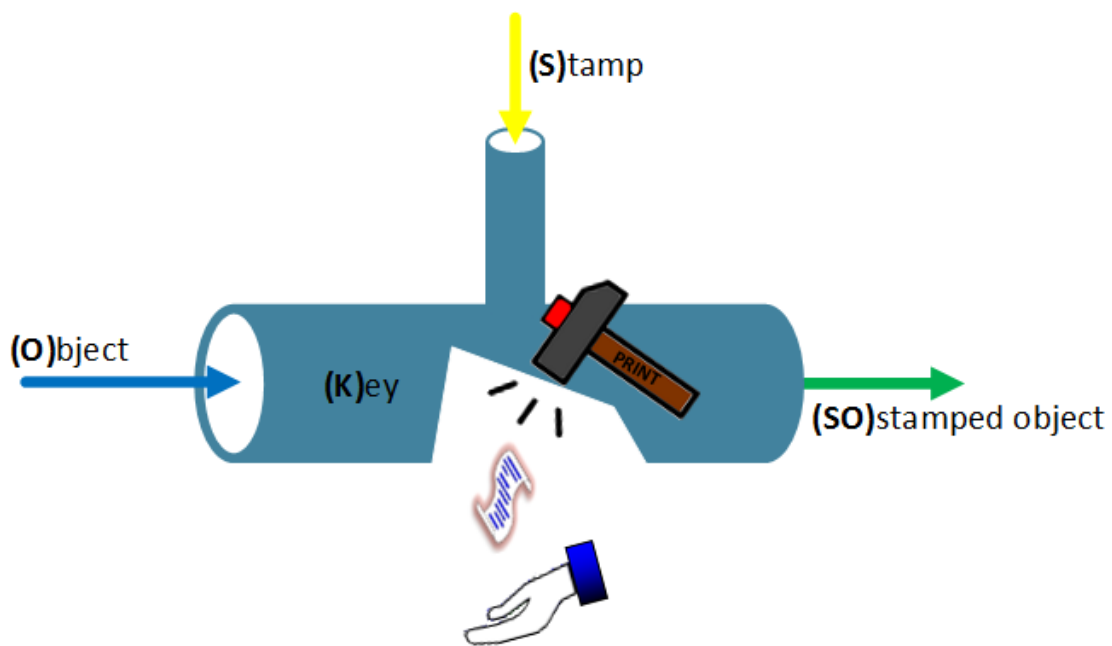


Figure 19: Stripping the watermark

If we interrupt the process of watermarking, we will be able to catch the original object (O) without being stamped (S). This is the theory, the work plan, now, we must see how to put it into practice.

4.1. Manual Repair

This is the procedure of the hack (make the watermark disappear) deleting manually the node that corresponds to watermark code in web. We will show the old and nowadays mode to achieve this.

4.1.1. Cast Iron (Old)

I used this method in 2017 so we can say is little old and is no longer available, but for historical reasons, we put here.

Step 1: Have you the **(K)**ey? Enter to the portal.

1. Go to *skillpipe.com* and enter the credentials.

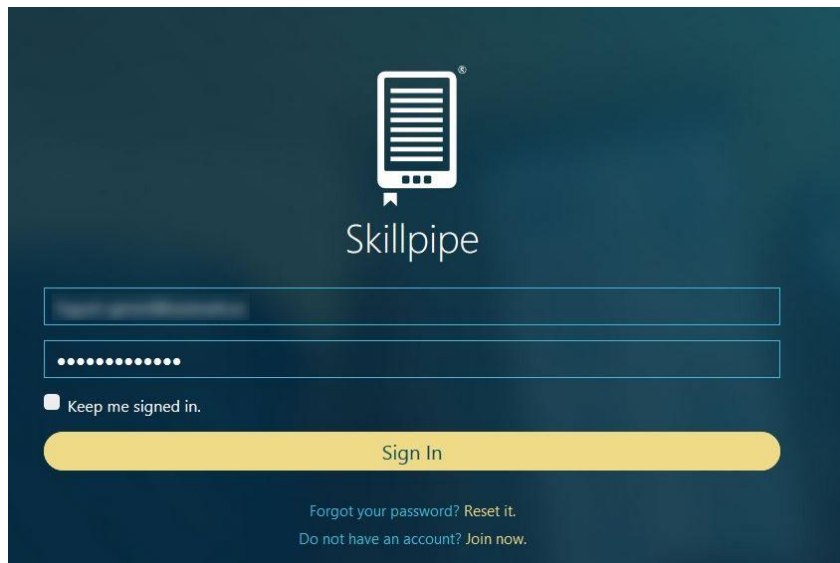


Figure 20: *Skillpipe* – Sign In

2. Choose the course you want to keep offline.

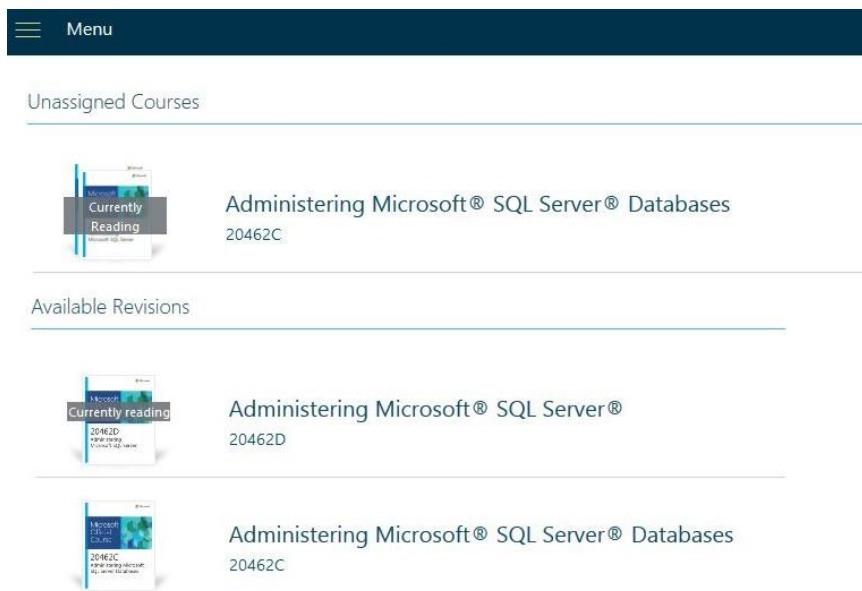


Figure 21: *Skillpipe* – Choosing Course

Step 2: Take out the water from the Mark.

1. Choose a desired module to print.

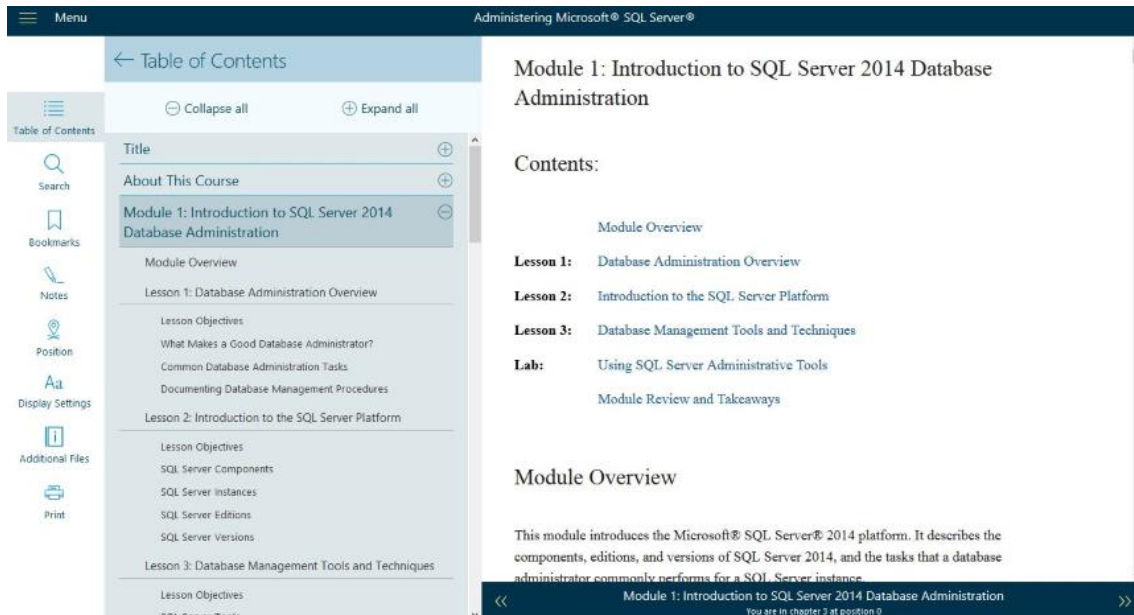


Figure 22: Skillpipe – Choosing Module

2. At the bottom – left, you can localize the print button, click and again click to print in the window that appear to the right (seen in figure 4 of the section 2.1.).
3. A new tab is created, and the print wizard appears, cancel the print wizard window (seen in figure 5 of the section 2.1.).
4. Access to the developer tools of the browser (we used *Firefox*) options pushing *F12* and look for the node: *print_watermark_wrapper*. Inspector option can be helpful.

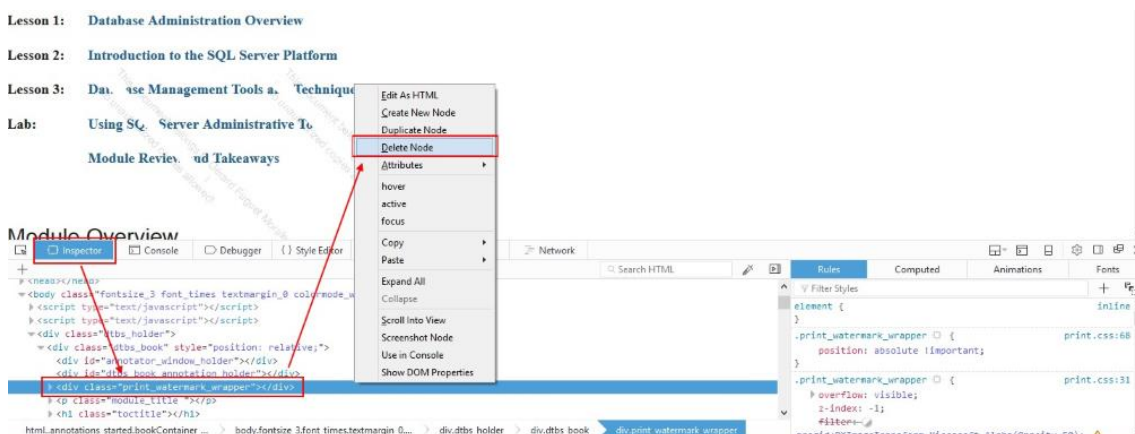


Figure 23: Deleting the Watermark through Developer Tools

5. Right click – Delete Node.

Step 3: Get the unstamped object.

1. You can invoke again the print wizard with *Ctrl + P* and you will be surprised that the watermark is gone.

If you print following the process until Step 2 at point 3 without cancelling the print wizard and without deleting the *print_watermark_wrapper* node, you can compare the two documents, and you be realized the size of the non-watermarked file is too heavy!!

A video of all the process was made, here you have the link (in Spanish but with English subtitles):

SkillPipe - Eliminar Marca de Agua (Lector de Cursos usado por Microsoft)

<https://youtu.be/wqXxGh3D-CI>

4.1.2. PVC (New)

A very similar method to the previous one but with two differences; the watermark node is somewhat deeper in the code and, there is no new tab to show you the result of the “*finish of the product*”. Print wizard appears in the same tab and you must have “*faith*” at the end of the printing process... Or you can use the *Chrome* browser to see if all is going well (viewing print preview)!

0. Let’s see the version of the new Web App first.

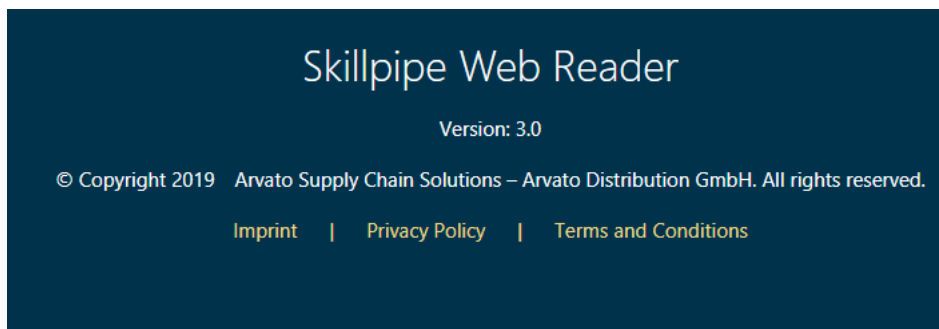


Figure 24: *Skilpipe* Web App Reader - About

Repeat all the steps of the *step 1, step 2* until complete the pass 1. We are going to continue in pass 2 of the *step 2* (seen on section 4.1.1.).

2. Before printing, the watermark node must be cleaned/deleted. Open *DevTools* of the browser (we used *Chrome* now) options pushing *F12* and search for the node: *#watermark*. Then Right click – Delete element as shown in the following figure 25.

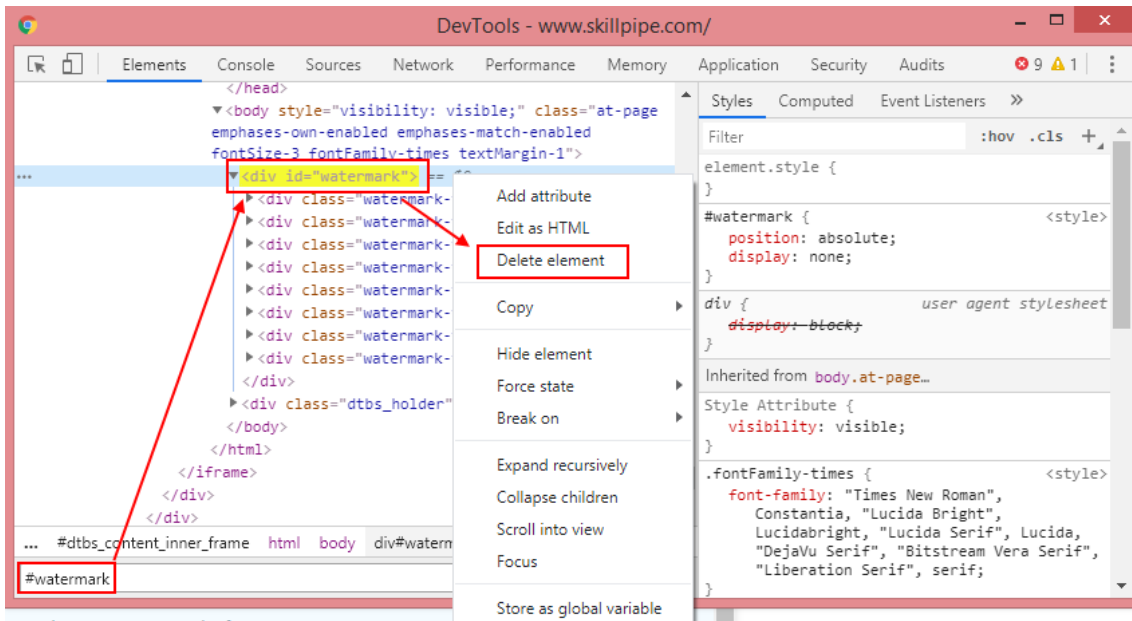


Figure 25: Chrome – DevTools – Deleting watermark node

Step 3: Drink water without mark.

1. You can close the *DevTools* window and print using the Web App print button (seen in figure 6 of the section 2.2.) and you will aware, watermark is not shown in document!

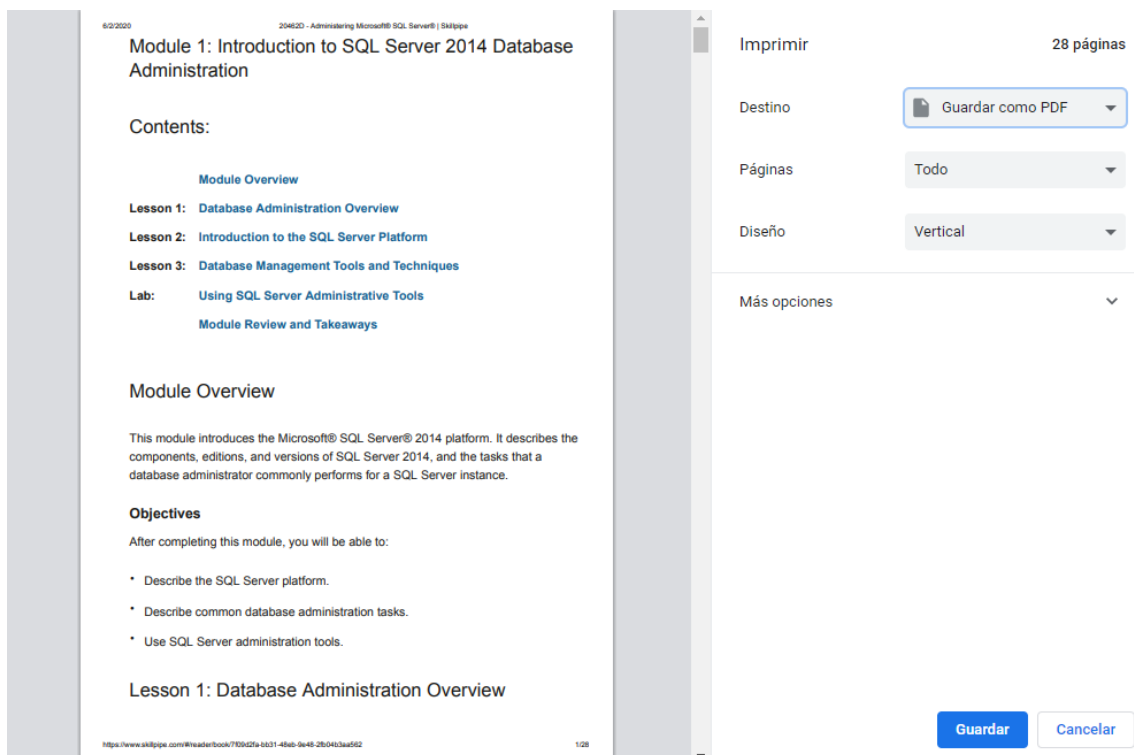


Figure 26: Chrome – Print Wizard – Watermark is gone

4.2. Semi-Auto Repair

The manual method of the “*pipe repairing*”, works well, watermark is removed entirely in all the book module, however, you need repeat the process on each module and can be tedious (It is already quite tedious to generate a PDF for each module...). So decided thinking in something, some kind of script. I still remember when the teacher says that a script at the beginning consumes you a significant amount of time but, when is build, your working life turns a little easier. Need you always create one? Not always. A workmate who has very good skills in development, before creating a new software, he searches if someone created it. Why do the job twice? This is we did, and we found a software able to defeat the watermark.

4.2.1. Ad Blocking

An *Ad Blocker* is used usually as a plugin of a browser to defeat the advertisings. Not only can block advertisings, can also filter and not show other things, things like watermarks. The PoC was done with *ABP (Adblock Plus)* [17]. Let’s see how it works!

0. Showing the version of *Chrome* browser & *ABP* used in this PoC.

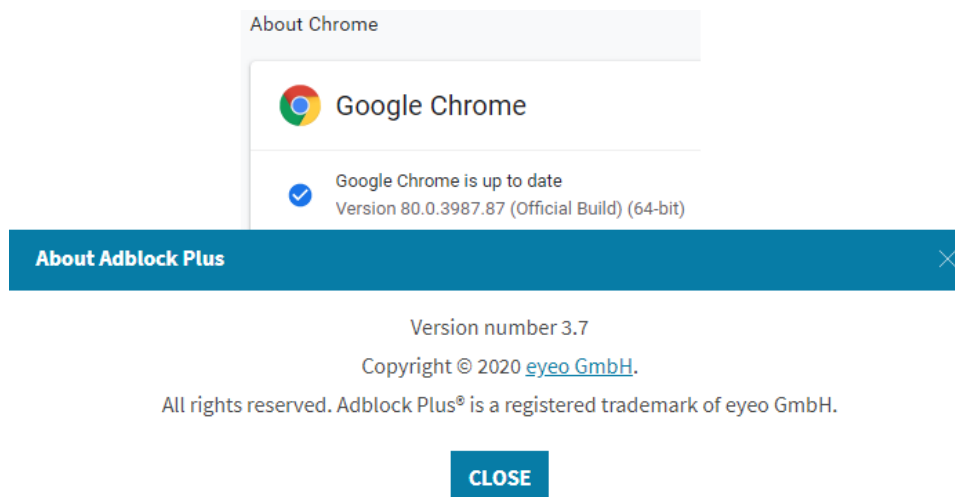


Figure 27: *Chrome* & *ABP* versions respectively

Step 1: Imprisoning to the watermark.

1. Open the *ABP* icon located at top – right of *Chrome* and click on settings button.

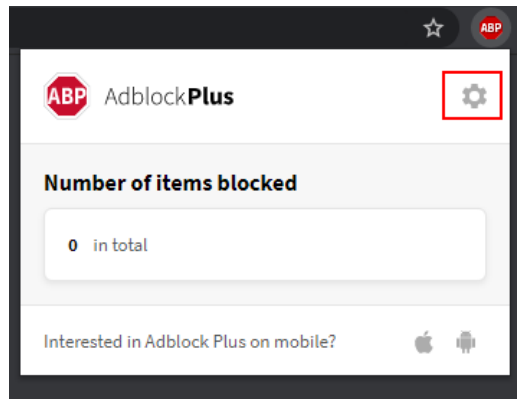


Figure 28: Opening ABP plugin in Chrome (Incognito mode)

2. Go to *Advanced* – Type the following filter in *My Filter List*: `skillpipe.com###watermark` [18] and push +ADD to add it.

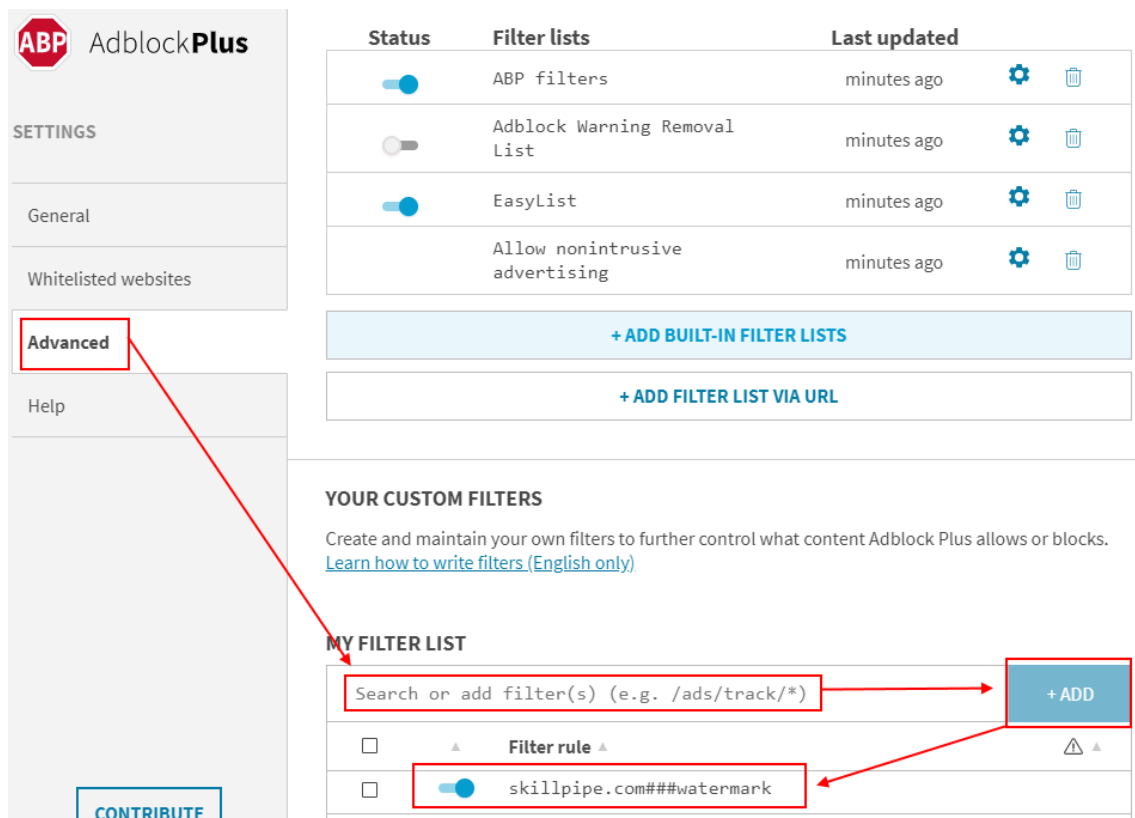


Figure 29: Opening ABP plugin in Chrome (Incognito mode)

Step 2: Ready to print clearly.

1. All is ready to print without the watermark, so situate in the module of a course, and print it with the printing functionality of the *Skillpipe* Web App (seen in figure 6 of the section 2.2.).

2. That's all! You can print next modules without the watermark.

This new method is covered in a video showing all the process (including the new manual method seen in section 4.1.2.). Here you have the link:

Skillpipe - Remove the Watermark (Course Reader used by Microsoft) using Ad Blocking
<https://youtu.be/Ok1UmRFWoLY>

5. Conclusions

Watermark in digital content is still fragile and lacks robustness.

In 2017 we initiate the notify, we reach at the webform of the *Skillpipe* official support:

Microsoft

By product ▾ Downloads ▾ Store ▾ Contact us

Your support ticket number: [redacted]

Contact Us: Microsoft Training & Certification

In what country/region do you reside?
Spain

Please specify your Microsoft Program level or the level of membership you have a question on:
Courseware

Please provide your Customer ID:
[redacted]

Please provide your company name (if applicable):
[redacted]

First name:
Gerard

Last name:
Fuguet

Contact e-mail address:
[redacted]

Phone number:
[redacted]

What is the subject of your comment or question?
Vulnerability found on the watermark of skillpipe documents

Reason for contacting us:
Solution Finder

Please provide the details of your query below, including all the information relating to it (eg. order or payment information) so that we can quickly resolve the problem.
I have a concern with the digital documents that Skillpipe offers to the students. I discovered that you can remove the copyright (the watermark) easily. When you print the document to PDF format through a virtual printer, you can cancel the native printing window of the navigator, then through the developer tools, you can remove the node called: "print_watermark_wrapper" and you can print again the document without the watermark.

What URL were you using when you experienced the problem?
<https://www.skillpipe.com>

E-mail address used to log in, if different from your contact e-mail address:
[redacted]

Figure 30: Contacting with *Microsoft Training & Certification* Support

And we got an answer from the *Arvato Bertelsmann* staff (by *Arvato Distribution GmbH*, an independent reseller of *Microsoft*) [19]:

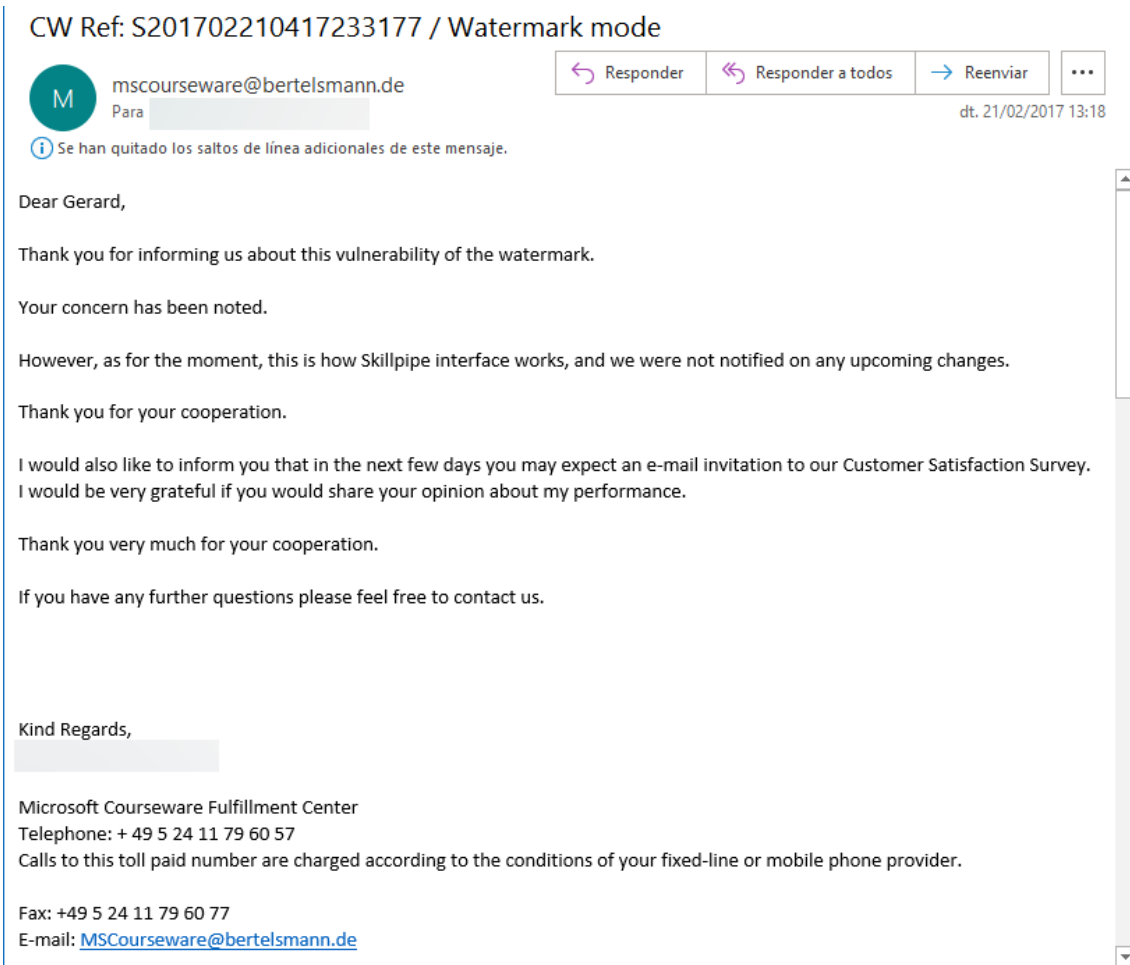


Figure 31: Answer of *Microsoft Courseware Fulfillment Center*

In addition, they are saying this is normal behavior so, no changes?! Well... It is clear that the answer did not convince me at all. Like in *Windows OS's*, for do some operations you can achieve it by many ways, this is we did, and another email went to the *MSRC* [20]:

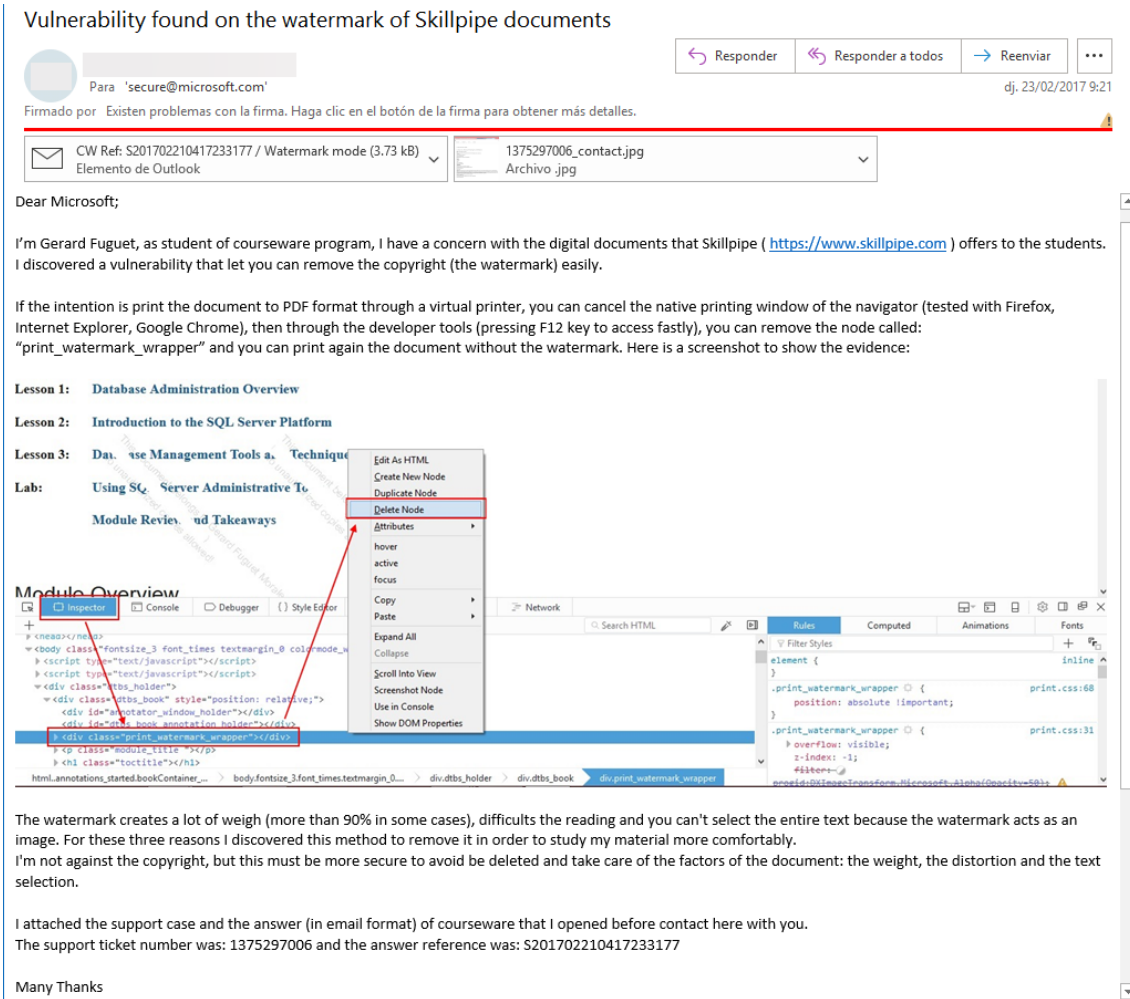


Figure 32: Email request to *Microsoft Security Response Center*

We attached screenshot and attached the email response evidence from *Courseware*, including the ticket number, ... The answer came:

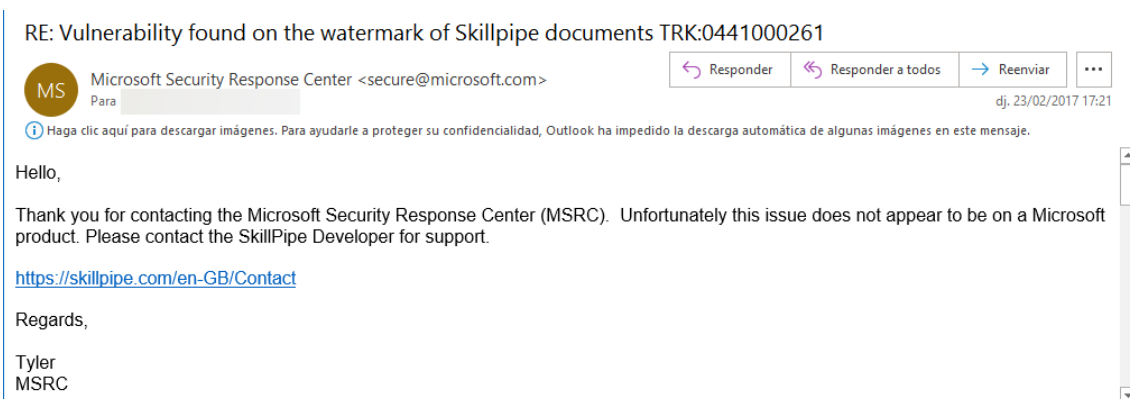


Figure 33: Email answer from Microsoft Security Response Center

Issue is not related with *Microsoft* product, so again go to *Skillpipe* was our scape, but we entered in an infinite loop... until we decided to stop it, doing this white paper, with good intention (always) of course.

Is true *Skillpipe* developers made some changes in code (comparison is in section 2.1.1. and 2.2.1.) but again they were insufficient to protect against total disassembly of the watermark.

As I mentioned in section 1. The initial problems and objective were trying to read the books/courses in a comfortable way, avoiding watermark problems (and we aren't against the copyright). The mechanism to protect these courses is immature, and probably we wouldn't try to break or clean it if the watermark had the characteristics like; be imperceptible and mustn't degrade the object intended to protect. The comparisons with and without the watermark are enough evidences to address a solution to the watermarking problem, but *Google Chrome* seems to know how to handle the problem of the weight of *PDF*, vertical bars in the corners and text selection. This implied us to test using the *PDF Creator* [21] (used with *Firefox* combination at the beginning, in 2017) with *Chrome* and the built-in *PDF* that *Firefox* doesn't have [22] (only through add-ons).

The following table show the results in bytes for each case (we did the tests with and without watermark printing the first Module 1 of the SQL course):

Size on PDF in Bytes	PDF Creator 2.5.2		Built-in PDF
	Firefox	Chrome	Chrome
Watermark	5.969.265	3.658.241	830.608
No Watermark	594.228	688.974	821.876

Figure 34: Comparison table of PDF sizes

Note that under watermark the clear winner is *Chrome* but in “No watermark” the best results are taken by *Firefox* (which uses *PDF Creator*) sizing 581kB against 803kB in Built-in *PDF*. Even so, *PDF Creator* appears to be better algorithm working without the watermark, *Chrome* resized to 673kB.

About the vertical bars and the selection text, not all the culprit is for *PDF Creator*. *Firefox* manipulates it in one way and *Chrome* in another different. See the differences at the following figure 35:

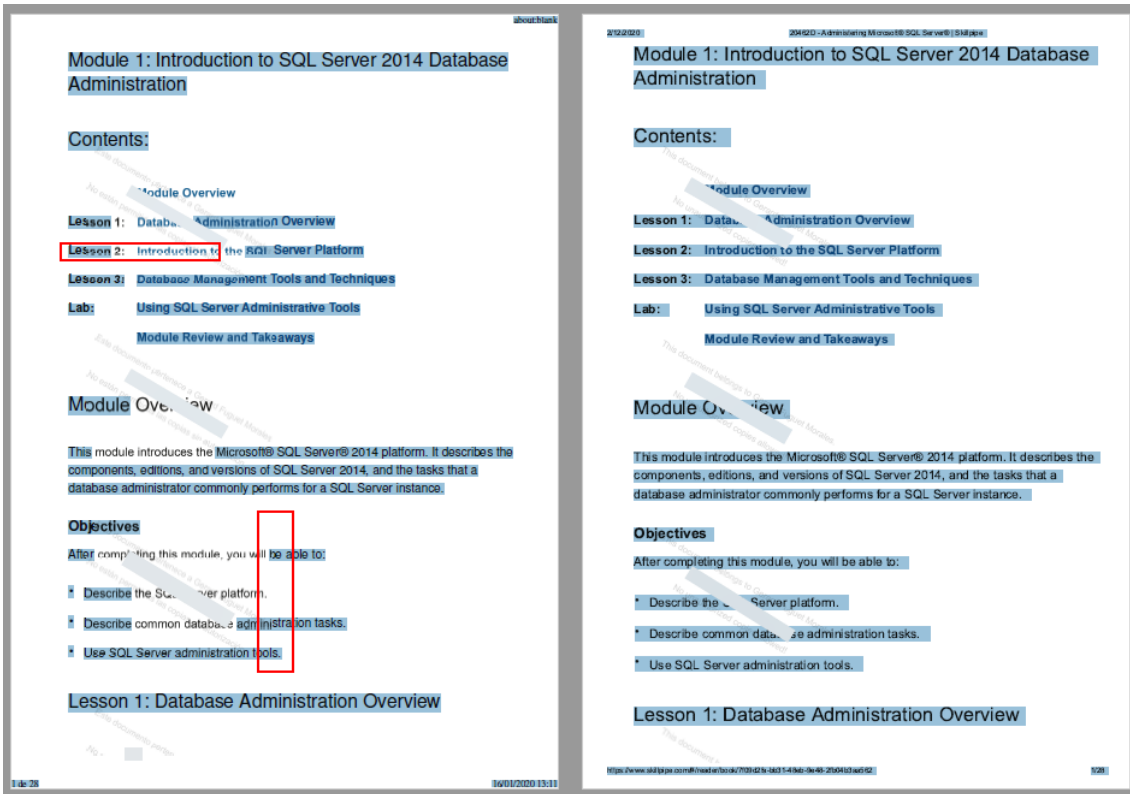


Figure 35: Comparison with *Firefox* (at left) and *Chrome* (at right) using *PDF Creator*

In *Firefox* version, the vertical bars and text distortion persists along the module and not all the text can be selected. On the other hand, in *Chrome*, all is “smoothly” including the size of the *PDF* file, indicating the watermark is more text than a raw image.

As final words to add for the end... How can we strengthen the watermark?

Most important is take care of the integrity of the digital document, must be robust and secure but not produce a high distortion of the object. We saw some tools are useful for developers (like the *DevTools* of the browsers) but also are dangerous allowing tamper it. Taking this into account, some method in code could do the job.

We took out the watermark before producing into *PDF*. So, *PDF* here is not the key element to put into check, maybe a review on the mechanism used could be good, or the creation of another format, be subject to another object, etc.

And other question could be, is the watermark the right way to protect the documents against illegal copies, should it be abandoned in the digital world and keep using in the physical?

Either way, this document is free of any watermark. Keep it clean! 😊

Remember...

Be Good, Be Hackers.

6. References

- [1] Computer Training, Computer Certifications, Microsoft Learning. <https://www.microsoft.com/learning>
- [2] Skillpipe eReader. <https://skillpipe.com>
- [3] Skillpipe - Support for the offline apps has ended. <https://policies.skillpipe.com/en/faq/#news>
- [4] Skillpipe – About this app. <https://www.skillpipe.com/#/account/about>
- [5] SkillPipe Frequently Asked Questions – Directions Training Center. <https://directionstraining.zendesk.com/hc/en-us/articles/200606785-SkillPipe-Frequently-Asked-Questions>
- [6] Skillpipe Mobile - From Native App to Progressive Web App. <https://pmskillpipeweu.blob.core.windows.net/archive/Skillpipe%20Mobile%20From%20Native%20App%20to%20Progressive%20Web%20App.pdf>
- [7] Skillpipe Reader - Software Informer. <https://skillpipe-reader.software.informer.com>
- [8] YouTube - Training & Education Services - Download A Book. <https://youtu.be/foSx0pi4uS8>
- [9] Disk2vhd - Windows Sysinternals. <https://docs.microsoft.com/en-us/sysinternals/downloads/disk2vhd>
- [10] How to Convert Physical Windows Computer into a VirtualBox Image - Jozef Jarosciak Blog. <https://www.joe0.com/2017/09/27/how-to-convert-physical-windows-computer-to-virtualbox-virtual-machine/>
- [11] Learning About EPUB: Structure and Content - Altova Blog. <https://www.altova.com/blog/learning-about-epub-structure-and-content/>
- [12] NirSoft - freeware utilities: password recovery, system utilities, desktop utilities. <https://www.nirsoft.net/>
- [13] FileActivityWatch - View read / write / delete file activity on Windows 10/8/7/Vista. https://www.nirsoft.net/utills/file_activity_watch.html
- [14] Overview – Process Hacker. <https://processhacker.sourceforge.io/>
- [15] Schneier on Security: The Blowfish Encryption Algorithm. <https://www.schneier.com/academic/blowfish/>
- [16] Wikipedia - Digital Watermark (Spanish). https://es.wikipedia.org/wiki/Marca_de_agua_digital
- [17] Adblock Plus - The world's # 1 free ad blocker. <https://adblockplus.org>

- [18] How to write filters - Adblock Plus Help Center .
<https://help.eyeo.com/adblockplus/how-to-write-filters>
- [19] Courseware Marketplace. <https://shop.courseware-marketplace.com>
- [20] MSRC - Microsoft Security Response Center. <https://www.microsoft.com/msrc>
- [21] Create, edit and merge PDFs easily - pdfforge. <https://www.pdfforge.org>
- [22] How to print a page on firefox in pdf - Firefox Support Forum - Mozilla Support.
<https://support.mozilla.org/en-US/questions/1109428>