

DRIVEEXPORT 4

Introduction.....	2
How DriveExport works	3
G Suite Configuration	5
A - Set up of a Google Project + Service Account	5
Setup a Google Project and enable the APIs.....	5
Create the Service Account inside the project	9
B - Setup of the G Suite / Google Apps domain	14
Set up domain-wide access for Service Account.....	14
DriveExport install & setup	18
The local configuration file	18
Details	20
Too long path or file name.....	20
Long paths on Windows 10 and Windows Server 2016	21
Files names collision	22
Orphaned items.....	22
Unknown parent files	23
Command line version exit codes	23
Large file export - Work-around WA1	23
Binary file verification.....	24
Silent mode.....	24
Scheduling with Windows Task Scheduler	24
Obtaining the Unique ID of the Google Apps subscription.....	26
Run DriveExport on Linux	26
Send the report by email	27
Configuration parameters.....	29

Introduction

DriveExport is a tool aimed to export data massively and efficiently from Google Drive. It connects to Google Drive for Business or Education and dumps to the local hard-disk the files stored on the cloud. It is available as a pure command line application, designed to be easily scheduled or automatically launched.

DriveExport runs on Windows and requires Microsoft .NET Framework 4.6.2 or later. It also runs on Mono 4 on Linux machines, but it is not officially supported. On both environments, it is suggested to use the latest version available of .NET / Mono with the latest updates and patches. DriveExport communicates directly with Google G Suite servers. To do that, it requires a correctly configured GSuite environment. This document will guide you through the steps required to setup and run DriveExport.

How DriveExport works

The purpose of DriveExport is to create a local copy of all items on Google Drive. To do that, DriveExport follows a sequence of steps:

- Connects to Google Drive servers
- Authenticates
- Gets the list of the users from the configured G Suite subscription
- For each user in the list, gets the list of documents
- For each document in the list, verifies if the document is still locally available and the modification date is not changed. If not, the document is exported.
- If a local file is no longer present on Google Drive, DriveExport can be configured to ignore, delete or move it (refer to parameter DeleteExtraFiles under Configuration parameters paragraph).
- The export procedure works in three different ways based on the type of the document:
 - Google native formats (Google Documents, Spreadsheets, Presentations, etc.): the document is converted and exported to a similar format (docx for Documents, xlsx for Spreadsheet, etc.)
 - Generic binary formats (images, pdf files, specific format, binary files, etc.): the files are downloaded exactly as they are without any conversion.
 - Third-party native format: Google Drive allows third-party apps to store specific file in proprietary formats. These files are not managed by DriveExport and are skipped. (The limit comes directly from Google: the Google API does not allow to export or download custom formats).

DriveExport creates a folders structure like the following:

```
X:\
  MyDriveExport\
    logfiles
  MyDriveExportFiles\
    usera@mydomain\
      file_a
      file_b
      dir_1\
        file_x
    userb@mydomain\
      .....
    .....
    userc@mydomain\
      .....
      .....
```

The log folder contains:

- one or more files (.log)
- the DriveExport.tsv file. It is a tab-separated-values text file containing the full list of items processed by DriveExport.
- the DriveExport_report.html file, containing a summary of the activity.

DriveExport

Version: 1.99.7.0
 Start time: 18/09/2016 07:51:41
 End time: 18/09/2016 07:51:50
 Total users: 9
 Users processed: 2
 User fatal errors: 0
 Downloads: 54
 Downloaded bytes: 13048
 Deleted extra files: 3

Username	Start	End	Status	Docs	Folders	Download ok	Download not managed	Download skip	Download error	User error details
user1@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user2@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user3@mydomainin.test	18/09/2016 07:51:43	18/09/2016 07:51:47	Processed	44	28	3	1	41	0	
user4@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user5@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user6@mydomainin.test			Suspended	0	0	0	0	0	0	
user7@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user8@mydomainin.test			NoFilterMatch	0	0	0	0	0	0	
user9@mydomainin.test	18/09/2016 07:51:47	18/09/2016 07:51:50	Processed	9	8	0	0	9	0	

<http://www.driveexport.com>

Under the user folders, where DriveExport places files coming from Google Drive, there are some special folders and files:

- **__DeletedItems** : folder containing deleted files, i.e. files previously present in the output folder but no longer present on Google Drive.
- **__OrphanedItems** : folder containing the “orphaned items” (refer to the specific paragraph for more details) (very rare cases).
- **__SharedWithMe**: folder containing items shared with the current user but not placed in a folder visible to the user.
- **__UnknownParent**: folder containing items placed in particular folders not directly accessible by the current user (very rare cases). Refer to the specific paragraph for more details.
- **__fileList.txt** : file containing the full list of item processed and the action performed
- **__mapper.xml** : file used by DriveExport to manage “too long” file and folder names.

G Suite Configuration

To gain access to your Google Drive documents, DriveExport does not use the credential of an administrator user. Instead, it uses an ad-hoc created Google Service Account. The account is used to access the Directory service and the Drive service exposed by Google APIs. The account must be granted access to these services. A read-only access is required.

The creation of a Google Service account is mandatory and required by official Google guidelines. It ensures a secure and reserved communication between DriveExport and Google services.

Macro steps:

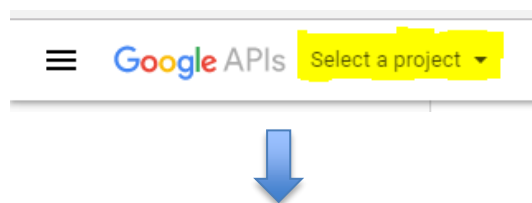
- A. On the Google Developers Console:
 - setup of a project
 - creation of a service account
 - APIs setup
- B. On the G Suite Admin Console:
 - domain-wide authorization
 - APIs setup

A - Set up of a Google Project + Service Account

DriveExport needs a **Google Service Account** along with configured APIs access. To do that, a Google Project is required. The following activities must be performed inside the Google Developers Console <https://console.developers.google.com>

Setup a Google Project and enable the APIs

1. Go to the Google Developers Console - <https://console.developers.google.com>
2. Select an existing project or create a new one (suggested option). Choose an available name that you like.



Google APIs

Search for APIs and Services

New Project

Project name *

DriveExport4

Project ID: driveexport4. It cannot be changed later. [EDIT](#)

Organization *

test-driveexport.com

Select an organization to attach it to a project. This selection can't be changed later.

Location *

test-driveexport.com

[BROWSE](#)

Parent organization or folder

[CREATE](#)

[CANCEL](#)



Notifications

✓ Create Project: DriveExport4 in 1 minute

[SELECT PROJECT](#)

[SEE ALL ACTIVITIES](#)



Google APIs

DriveExport4

Search for APIs and Services

APIs & Services

APIs & Services

[+ ENABLE APIS AND SERVICES](#)

[Dashboard](#)

[Library](#)

[Credentials](#)

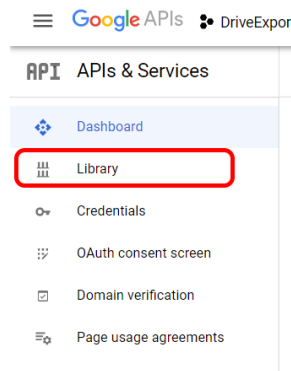
[OAuth consent screen](#)

[Domain verification](#)

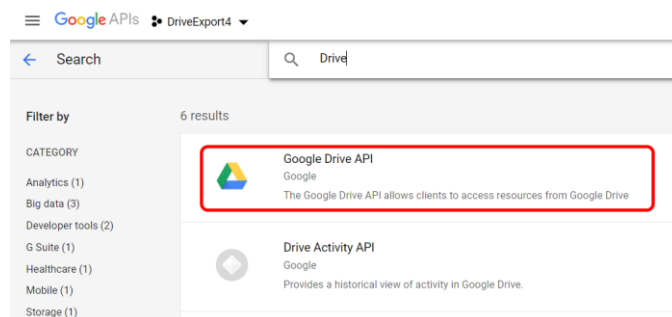
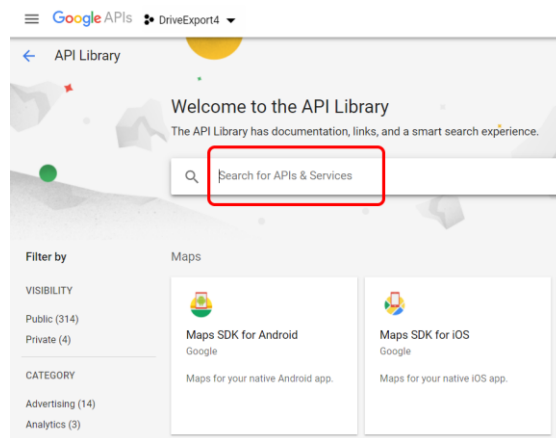
[Page usage agreements](#)

! You don't have any APIs available to use yet. To get started, click "Enable APIs and services"

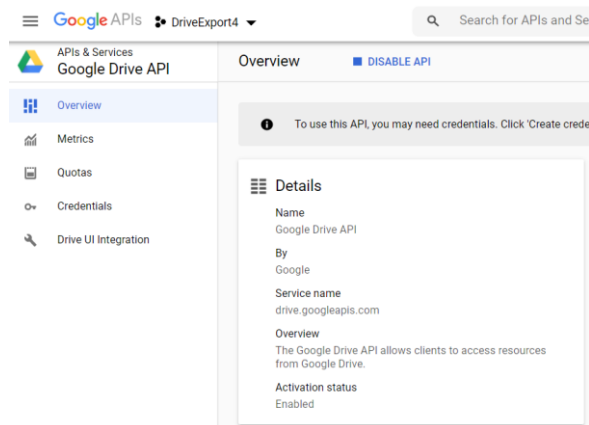
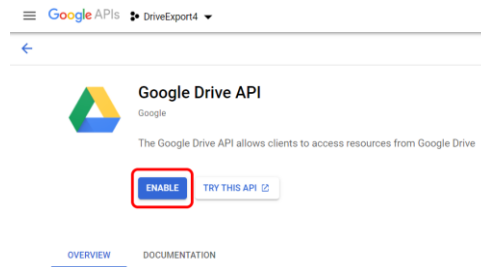
- From the left panel, select “**Library**”



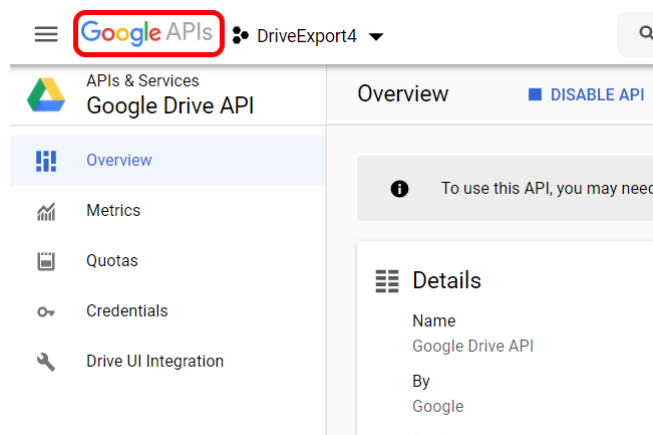
- Select the **Drive API**: type “Drive” in the search box and the API will appear. Click on it.



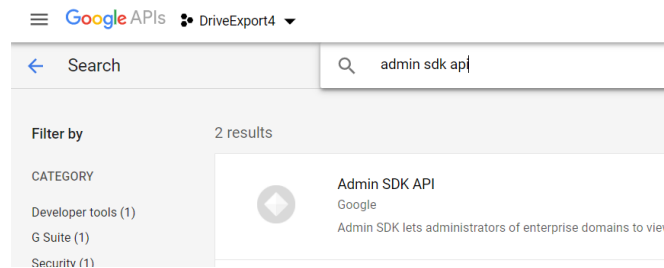
- Enable the Drive API pressing the button “Enable” near the top-bar. Wait some seconds while the API is enabled. At the end, you will be showed the default “About this API” page with details about its usage.



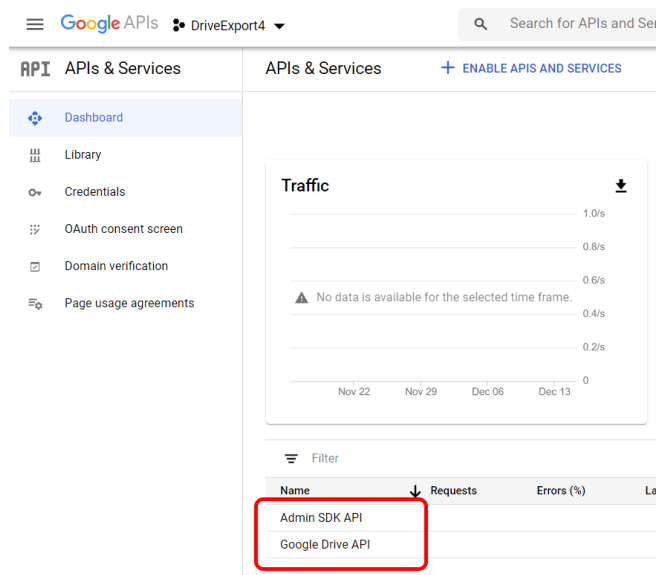
6. Click Google APIs logo to come back to main dashboard.



7. Repeat the same steps done for **Google Drive API** also for the **Admin SDK API**:
 - a. Select "Library" on the left bar
 - b. Enter "admin sdk" in the search box
 - c. The box Admin SDK appears. Click on it.
 - d. Press "enable" to activate the API



8. After enabling **Drive API** and **Admin SDK**, come back to the main “Dashboard”. It will look like as below (notice the two enabled API at the bottom)



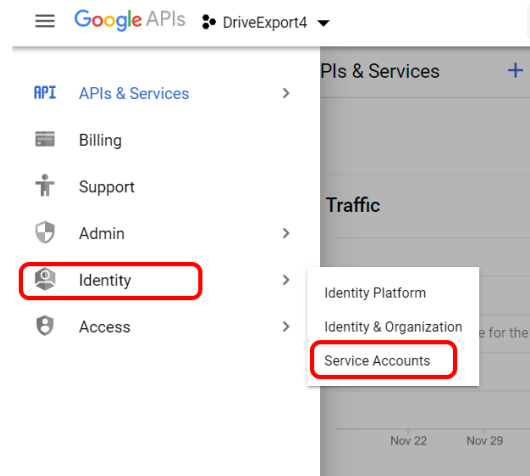
Create the Service Account inside the project

At the end of this activity, you will have a **Google Service Account**, composed by:

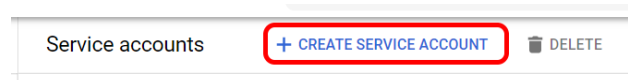
- a Client ID
- a private key file (a p12 file)
- a virtual email address

Warning: these steps should be done using **Google Chrome**. The other browsers are not well supported by Google Developer Console website.

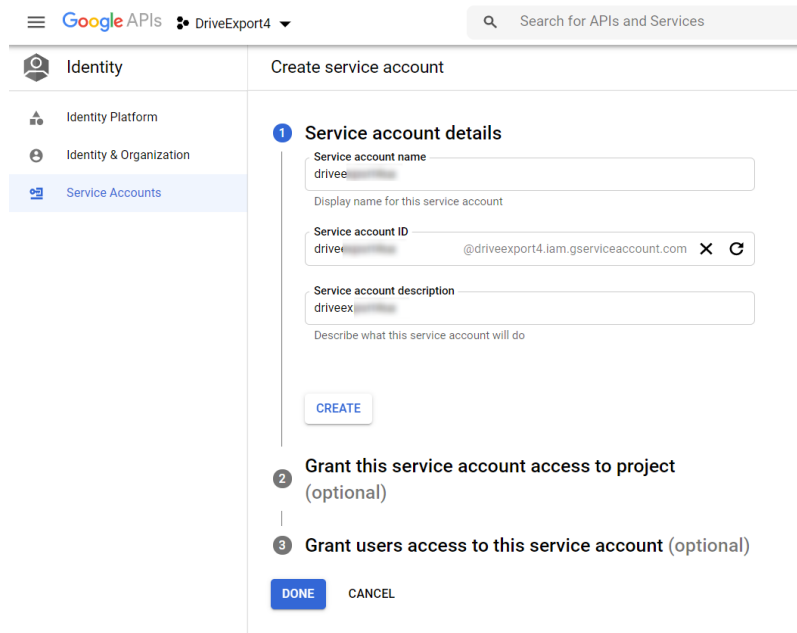
1. On the left bar select “Credentials”, then “Identity”, then “Service Accounts”



2. Press **Create Service Account**



3. Choose a name and then press **Done**.



Google APIs DriveExport4

Search for APIs and Services

Identity

Identity Platform

Identity & Organization

Service Accounts

Create service account

1 Service account details

Service account name
driveex

Display name for this service account

Service account ID
driveex @driveexport4.iam.gserviceaccount.com

Service account description
driveex

Describe what this service account will do

CREATE

2 Grant this service account access to project (optional)

3 Grant users access to this service account (optional)

DONE CANCEL

- The service account appears in the list. Click the 3-dots and press Edit

Service accounts [+ CREATE SERVICE ACCOUNT](#) [DELETE](#)

Service accounts for project "DriveExport4"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. [Learn more about service accounts.](#)

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. [Learn more about organization policies.](#)

Filter table	Email	Status	Name ↑	Description	Key ID	Key creation date	Actions
<input type="checkbox"/>	drive[redacted]@driveexport4.iam.gserviceaccount.com	✓	drive[redacted]	drive[redacted]	No keys		<ul style="list-style-type: none"> Edit Disable Create key Delete

- Select **Add Key** and **Create new key**

Google APIs DriveExport4

Search for APIs and Services

Identity

Identity Platform

Identity & Organization

Service Accounts

driveexport4sa

DETAILS PERMISSIONS

Service account details

Name: drive[redacted]

Description: drive[redacted]

Email: drive[redacted]@driveexport4.iam.gserviceaccount.com

Unique ID: 10843[redacted]263

Service account status

Disabling your account allows you to preserve your policies without having to delete it.

✓ Account currently active

[DISABLE SERVICE ACCOUNT](#)

✓ SHOW DOMAIN-WIDE DELEGATION

Keys

Add a new key pair or upload a public key certificate from an existing key pair. Please note that public certificates need to be in RSA_X509_PEM format. [Learn more about upload key formats](#)

[ADD KEY](#)

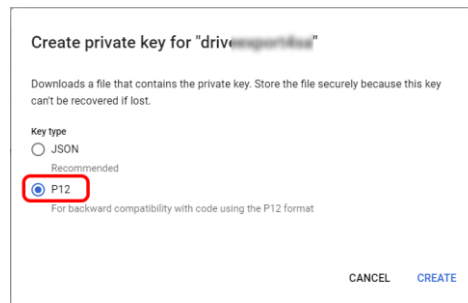
Create new key

Upload existing key

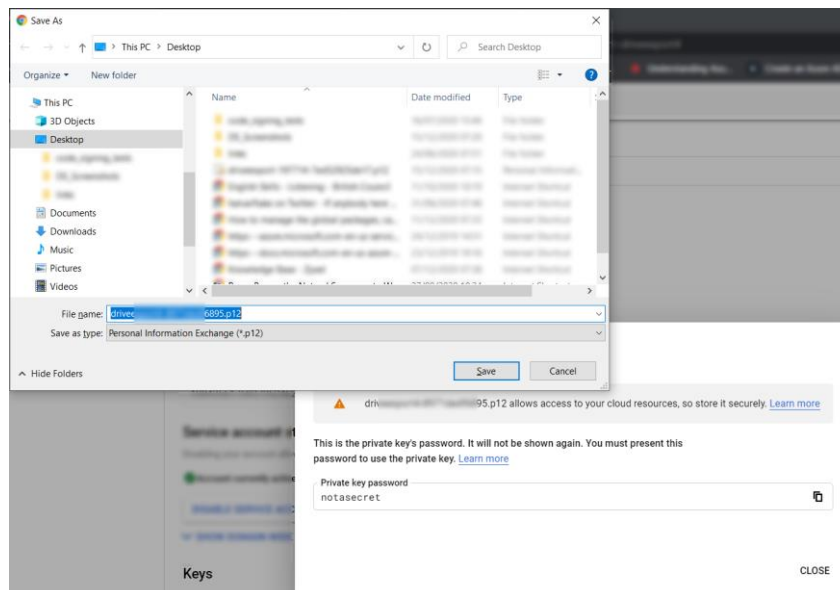
Key creation date Key expiration date

[SAVE](#) [CANCEL](#)

- Select P12 and press **Create**



7. After few seconds, a pop-up appears, and the browser ask to save a **p12** file. Save it on your local hard-disk. The process also creates a “private key’s password”. For Google Service Account, it is always “notasecret”. So, it is not necessary to annotate it. After saving the p12 file, press **Close**.



8. In the service account details, expand **Show Domain-Wide Delegation** and then activate **Enable G Suite Domain-wide Delegation**

Service account status

Disabling your account allows you to preserve your policies without having to delete it.

✔ Account currently active

[DISABLE SERVICE ACCOUNT](#)

☐ Enable G Suite Domain-wide Delegation


Allows this service account to be authorized to access all users' data on a G Suite domain without manual authorization on their parts. [Learn more](#)

⤴ HIDE DOMAIN-WIDE DELEGATION

Keys

Add a new key pair or upload a public key certificate from an existing key pair. Please note that public certificates need to be in RSA_X509_PEM format. [Learn more about upload key formats](#)

[ADD KEY](#)

Type	Status	Key	Key creation date
	✔ Active	89[REDACTED]5e	Dec 15, 2020

[SAVE](#)

[CANCEL](#)

9. Add a brief description under Product name for the consent screen

☒ Enable G Suite Domain-wide Delegation

Allows this service account to be authorized to access all users' data on a G Suite domain without manual authorization on their parts. [Learn more](#)

i To change domain wide delegation, a product name for the OAuth consent screen must be configured. You can enter the product name below. On some platforms, the email address is shown with the developer information. To select a different email address, configure consent screen.

[CONFIGURE CONSENT SCREEN](#)

Product name for the consent screen

Dr[REDACTED]

Assign product name.

10. Press **Save**

[SAVE](#)

[CANCEL](#)

11. The list of service account appears. Scroll to the right and press **View Client ID**

Service accounts for project "DriveExport4"

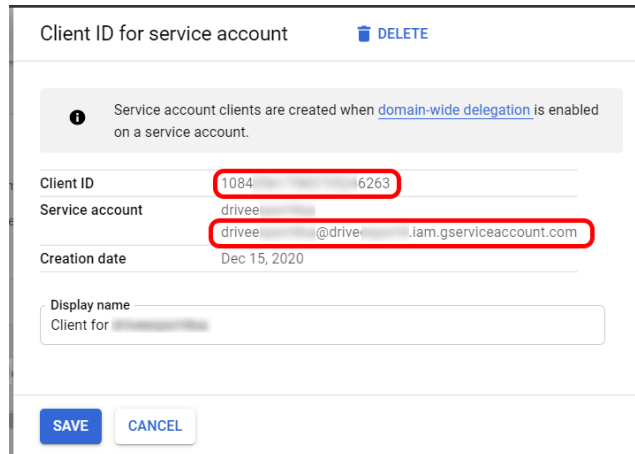
A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. [Learn more about service accounts.](#)

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. [Learn more about organization policies.](#)

Filter table

	Status	Name ↑	Description	Key ID	Key creation date	Domain wide delegation ?
report[REDACTED]@driveexport4.com	✔	dr[REDACTED]	dr[REDACTED]	89[REDACTED]5e	Dec 15, 2020	Enabled View Client ID

12. The pop-up shows the **ClientID** and the Service account email addresses. They are needed when you will setup DriveExport.



Client ID for service account [DELETE](#)

i Service account clients are created when [domain-wide delegation](#) is enabled on a service account.

Client ID 1084 6263

Service account drivee...@drive...iam.gserviceaccount.com

Creation date Dec 15, 2020

Display name
Client for

[SAVE](#) [CANCEL](#)

B - Setup of the G Suite / Google Apps domain

The following activities must be performed in the **G Suite admin console**.

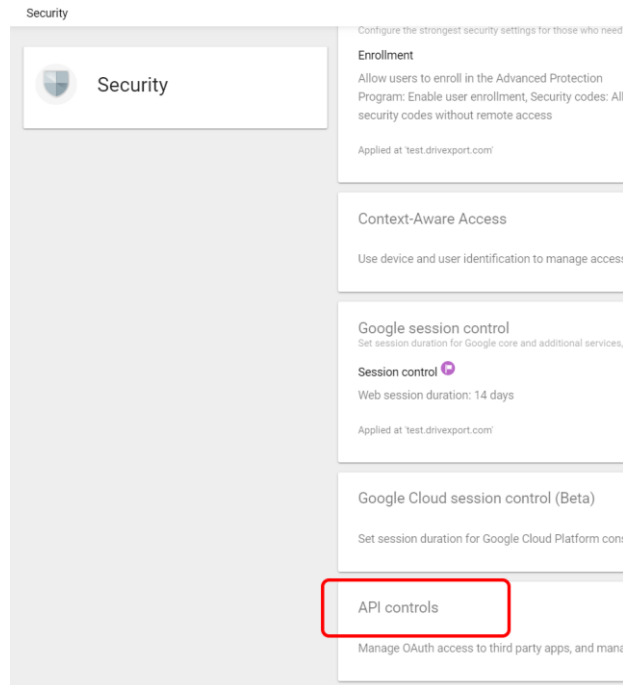
Set up domain-wide access for Service Account

The following steps delegate domain-wide authorization to the previously created Service Account. Only required authorization must be assigned.

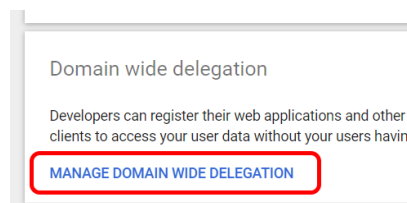
1. Go to your G Suite admin console <https://admin.google.com>
2. Login with an administrative account
3. Select **Security** from the list of controls. If you don't see Security listed, select More controls from the gray bar at the bottom of the page, then select Security from the list of controls.



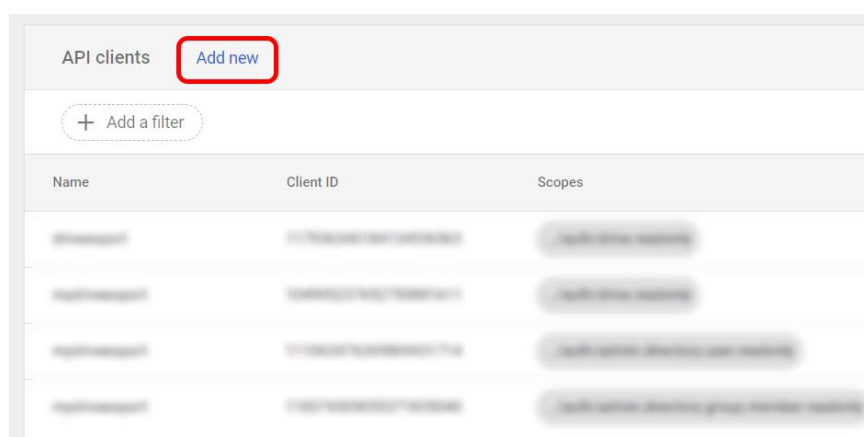
4. Select **Advanced settings** from the list of options (if you don't see it, expand Show more).
5. Select **API controls**



6. Select **Manage Domain Wide Delegation**



7. The list of registered client appears. Select **Add New**



8. Add the Client ID previously created and the required OAuth scopes and then press **Authorize**

- <https://www.googleapis.com/auth/admin.directory.user.readonly>
- <https://www.googleapis.com/auth/drive.readonly>
- <https://www.googleapis.com/auth/admin.directory.group.member.readonly>

Add a new client ID

Client ID

1084 6263

☐ Overwrite existing client ID ?

OAuth scopes (comma-delimited)

https://www.googleapis.com/auth/admin.directory.i

×

OAuth scopes (comma-delimited)

https://www.googleapis.com/auth/drive.readonly

×

CANCEL

AUTHORIZE

Add a new client ID

https://www.googleapis.com/auth/admin.directory.i

OAuth scopes (comma-delimited)

https://www.googleapis.com/auth/drive.readonly

×

OAuth scopes (comma-delimited)

https://www.googleapis.com/auth/admin.directory.i

×

OAuth scopes (comma-delimited)

CANCEL

AUTHORIZE

- The newly added client appears in the list. If you need, click on it to review the assigned OAuth scopes.

API clients Add new		
+ Add a filter		
Name	Client ID	Scopes
DriveExport	1084 6263	.../auth/admin.directory.user.readonly .../auth/drive.readonly +1 More
DriveExport	1084 6263	.../auth/admin.directory.user.readonly
DriveExport	1084 6263	.../auth/admin.directory.user.readonly
DriveExport	1084 6263	.../auth/admin.directory.user.readonly
DriveExport	1084 6263	.../auth/admin.directory.user.readonly

× DriveExport

Client ID

1084[REDACTED]6263

Scopes

https://www.googleapis.com/auth/admin.directory.user.readonly

https://www.googleapis.com/auth/drive.readonly

https://www.googleapis.com/auth/admin.directory.group.member.readonly

EDIT

DriveExport install & setup

To install DriveExport 4.0 follow the following steps:

- obtain a free-test license file or a paid one from <https://www.driveexport.com/>
- download the application as a zip file from <https://www.driveexport.com/>
- un-block the zip (if blocked) [Right click on it, properties, unblock]
- create three folders on your hard-disk:
 - a folder for the tool (e.g. D:\DriveExport\)
 - a folder for log files (e.g. D:\DriveExport\logs)
 - a folder for the exported files (e.g. D:\GAppsDriveFiles)
- unzip the file and place the files in D:\DriveExport
- optional: place the sample_config.xml in D:\DriveExport and create a copy of the xml file (e.g. myconfig.xml)
- follow instructions for creating a “Service Account” on your Google Apps domain (see “Google Apps Configuration” chapter)
- edit the xml configuration. More details about the configuration file are available in the chapter “The local configuration file”.
- place the license-file in the same folder of the configuration file (e.g. D:\DriveExport\)
- open a command line prompt (e.g. at D:\DriveExport\)
- run the tool passing myconfig.xml as parameter
 - on Windows: driveexportcmd.exe myconfig.xml [enter]
 - on Linux: mono ./driveexportcmd.exe myconfig.xml [enter]

The local configuration file

The configuration file is an XML file containing parameters needed by DriveExport to run properly. The filename must be specified on the command line as the first parameter:

```
DriveExportcmd.exe myconfig.xml
```

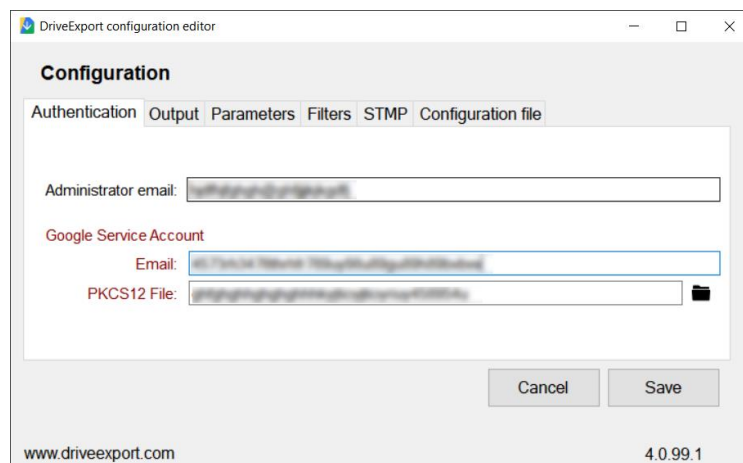
Details about the parameters are available in the chapter “Configuration parameters”

Example of a configuration file:

```
<DriveExportConfig>
  <LicenseCode>1234....</LicenseCode>
  <OverwriteLogs>true</OverwriteLogs>
  <DomainAdminEmail>adminuser@mydomain</DomainAdminEmail>
  <DegreeOfParallelism>4</DegreeOfParallelism>
  <ManageExternalItems>false</ManageExternalItems>
  <MaxFilenameLength>50</MaxFilenameLength>
  <MaxFoldernameLength>30</MaxFoldernameLength>
  <OutputFolder>c:\temp\driveexport\</OutputFolder>
</DriveExportConfig>
```

```
<LogFolder>c:\temp\driveexportlog\</LogFolder>
<GoogleServiceAccountEmail>...@...gserviceaccount.com
  </GoogleServiceAccountEmail>
<GoogleServiceAccountPKCS12FileName>path_to\myfile.p12
  </GoogleServiceAccountPKCS12FileName>
<DeleteExtraFiles>1</DeleteExtraFiles>
<SendReportEmail>true</SendReportEmail>
<SMTPServer>mysmtp.local</SMTPServer>
<SMTPPort>587</SMTPPort>
<SMTPUsername>smtpusername</SMTPUsername>
<SMTPPassword>smtppassword</SMTPPassword>
<SMTPPasswordEncrypt>false</SMTPPasswordEncrypt>
<EmailFromAddress>from@mydomain</EmailFromAddress>
<EmailToAddresses>to1@mydomain,to2@mydomain</EmailToAddresses>
</DriveExportConfig>
```

The configuration file can be edited with a text/xml editor or using the DriveExport Configuration Editor (DriveExportConfigEditor.exe)



Details

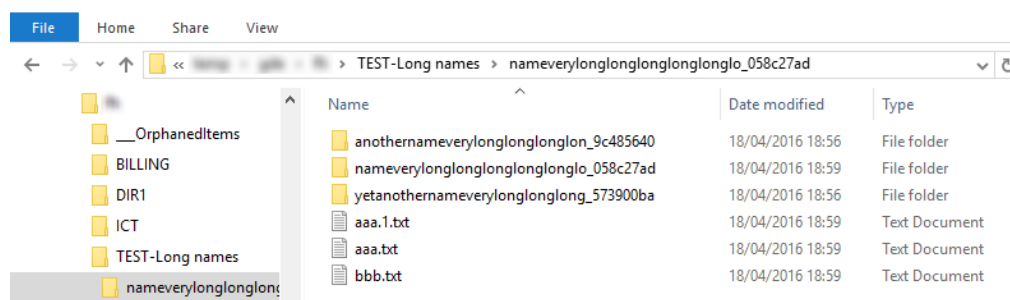
Too long path or file name

Windows has a limit on the maximum length of the names of directories files and on the resulting full path. In case of deeply nested directories with long names, it's easy to exceed the limit of **260 characters**. In such a case, Windows generates errors. (**Note:** the limit can be bypassed starting from Windows 10 – version 1607 and Windows Server 2016. Details in the next chapter “Long paths on Windows 10 and Server 2016”).

Google Drive doesn't have any documented limit about the maximum length of files and directories names nor any limit about levels of nested directories. DriveExport tries to replicate and to map this structure to Windows hard-disk in a “best effort way”. But sometime errors could occur and DriveExport show errors like this:

"Exception_FolderTooLongPath [PathTooLongException: The specified path, file name, or both are too long. The fully qualified file name must be less than 260 characters, and the directory name must be less than 248 characters.]"

There is not a complete solution to the problem. DriveExport implements an algorithm aimed to mitigate this issue: it cuts and partially renames too long file and folder names. The default limits are 30 characters for folder names and 50 characters for file names. The values can be modified through the configuration options MaxFilenameLength and MaxFoldernameLength.



Unfortunately, that's not a global solution. Consider a simple example: you have 10 level of subfolders. Every folder name length is more than 30 characters. The default value of MaxFoldernameLength is 30. So DriveExport cuts every folder names around 30th characters. But unfortunately, the resulting output full path is still longer than the limits of Windows (260 chars). Some suggestions to solve the issue:

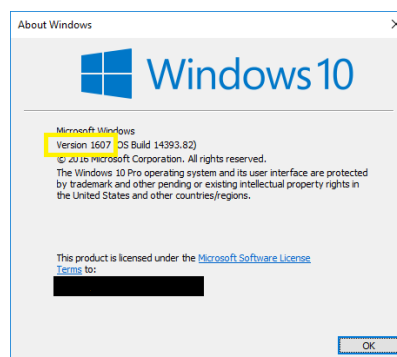
- ask the users to reduce the levels of nested folders and/or reduce the length of the folder names
- set the OutputFolder as short as possible. Example: <OutputFolder>d:\DE-OUT\</OutputFolder>
- reduce the value of MaxFoldernameLength

Warning: changing the values of MaxFilenameLength or MaxFoldernamLength after having already downloaded files from Google Drive, could cause DriveExport to export again a lot of files on the next run. In fact, changing those parameters, force DriveExport to recalculate too long files of directories names, resulting in many not yet downloaded files.

Long paths on Windows 10 and Windows Server 2016

Starting from version 1607 (also known as “Anniversary update”), Windows 10 can manage paths longer than 260 characters. Windows Server, starting from version 2016, has the same capabilities. DriveExport, if executed on Windows 10 1607 or Windows Server 2016, can create very long file and folder names, exceeding the limits highlighted in the previous chapter.

To verify your Windows 10 version, run winver.exe from command line or directly from the Start menu. The window shows the version.

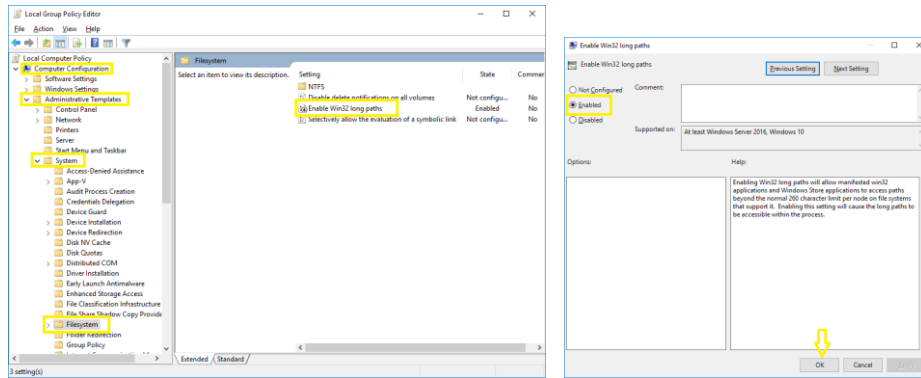


If your version is lesser than 1607, update your system from “Windows Settings” → “Update & Security” → “Windows Update”. **At the time of writing**, if you use Windows 10 Enterprise, the update could be not yet available through Windows Update. You need a full iso / DVD to install or update an existing installation of windows 10 enterprise.

On Windows Server 2016 and Windows 10 version 1607, before running DriveExport, you must activate the "long paths management" (Microsoft chose to design it as an "opt-in" functionality). If available, use the Local Group Policy Editor. Alternatively, you can use the Registry Editor.

Local Group Policy Editor steps:

- from the command line or from start menu, run the Local Group Policy Editor, typing gpedit
- double-click on "Enable Win32 long paths" under Computer configuration --> Administrative templates --> System --> Filesystem, select "Enabled" and press "OK"
- close the Local Group Policy Editor
- reboot the computer



Registry Editor steps:

- from the command line or from start menu, type **regedit** and press enter
- select HKEY_LOCAL_MACHINE\ SYSTEM\ CurrentControlSet\ Control\ FileSystem\
- set the key LongPathsEnabled to 1 (DWORD)

Note: with the release of Windows 10 1607 and on Windows Server 2016, Windows File Explorer has been updated to manage long paths. You will be able to explore folders, open/copy/delete files with it. But in few cases, it could be in troubles with very long paths (we suppose that there are still some issues that will be fixed by Microsoft in the near future). If you have problems browsing deeply nested folders, you can use an alternative file manager, like XYplorer. <https://www.xyplorer.com/free.php>

Files names collision

Google Drive allows to have more files with the same name in the same folder. When DriveExport encounters these files, it detects a file name collision. The first file is download with its original name and the followings are partially renamed, near the end of the file name, adding a progressive counter. (e.g. file.txt, file.1.txt, file.2.txt, etc.)

Orphaned items

In some cases, items under Google Drive could become “orphaned”, i.e. not belonging to any folders but still present in the Google Drive cloud storage. In such a case the items are not directly visible in the Google Drive web-site but they can be managed and exported by DriveExport. “Orphaned items” is known behavior of Google Drive. From the official support documentation:

If an item in Google Drive loses all of its parent folders, it becomes an orphan. The item still exists, but it's now harder to find. For example, this can happen if you create a file in someone else's folder, and then that folder is deleted. Your file isn't deleted along with the folder because no one else can delete your file, but it no longer has a home.

<https://support.google.com/a/answer/6008339>

If DriveExport encounters “orphaned items”, it places them under the special folder **__OrphanedItems** under the root folder of each user. This behavior can be controlled through the option ManageOrphanedItems of the configuration file:

<ManageOrphanedItems>true</ManageOrphanedItems>

Unknown parent files

In some rare cases, Google does not give or is not able to link some files to the corresponding containing folders. Those files are placed under “__UnknownParent” special folder. It is unclear if it is a bug, an incomplete design of Google Drive APIs service or an intrinsic behavior because of the nature of relationship between files and corresponding containing folder.

At the moment, two cases are known that could trigger this behavior:

- a shared file owned by UserA is placed under a folder owned by UserB
- shared files for which the parent folder was deleted.

Normally few files should be present in those folders. As a general suggestion, for shared files it is better to use Shared Drives (aka Team Drives).

Command line version exit codes

0	OK
1	Unmanaged or severe error
10	Configuration file not specified
11	Configuration file does not exist
12	Cannot read or invalid configuration file
13	License validation was unsuccessful
14	There is one or more user error (possible presence also of document errors)
15	There is one or more document error (no user error present)
16	Over quota (processing Shared Drives items)
17	There are one or more errors processing Shared Drives files
18	Over quota (processing users): some users have not been processed

Large file export - Work-around WA1

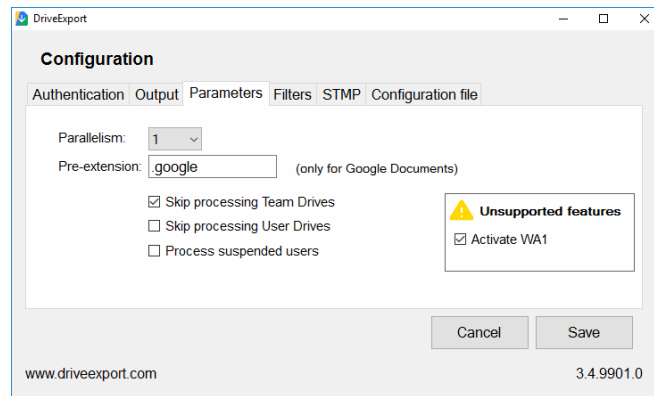
There is a known limit (bug?) of the official Google Drive API. It is not possible to export Google native items – such as documents, spreadsheet and presentations – which are greater than 10 MBytes. In such a case, the user receives errors like:

```
This file is too large to be exported. [403]
Errors [ Message[This file is too large to be exported.]
Location[ - ] Reason[exportSizeLimitExceeded] Domain[global]]
```

DriveExport implements a work-around for managing those files. The work-around interacts with a Google export API normally used by Google Drive web-site and not officially documented by Google alongside their official APIs.

The WA1 can be activated from the GUI interface or directly in the configuration file, adding the entry:

```
<ActivateWA1>true</ActivateWA1>
```



Important: the tests showed a correct behavior in all use-cases but the work-around WA1 is not officially supported and there is no guarantee against wrong behaviors. Its underlying API could be discontinued or modified without notice by Google, making the workaround ineffective. You agree that you use it at your own risk.

Binary file verification

DriveExport verifies the MD5 hash of every “binary” files it manages, both during the download phase from Google services and during verification in the following runs. This guarantees that local files are the exact copy of the original files on Google servers. Native Google documents (Documents, Sheets, Presentations, etc.) cannot be verified because Google does not give the MD5 hash of them.

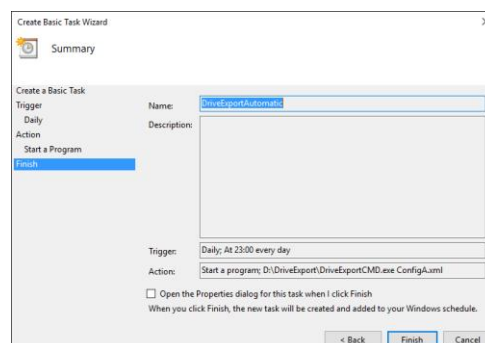
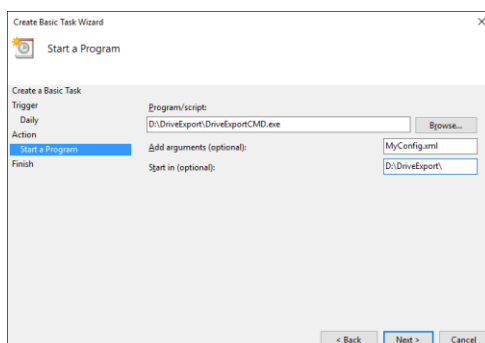
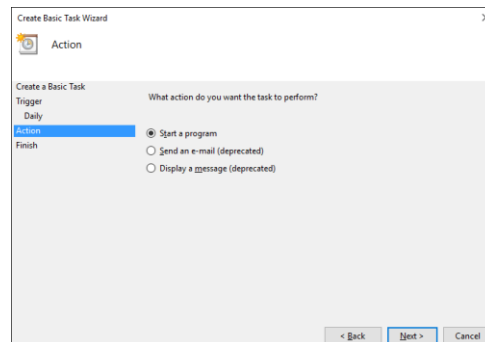
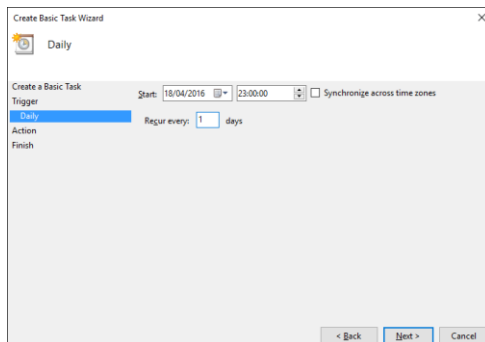
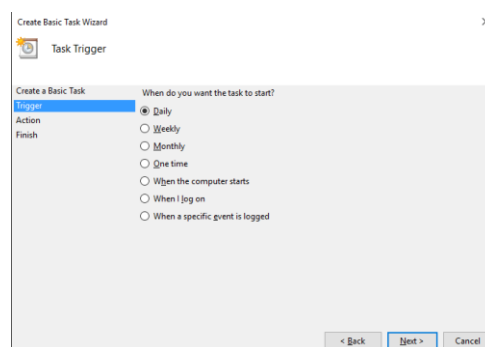
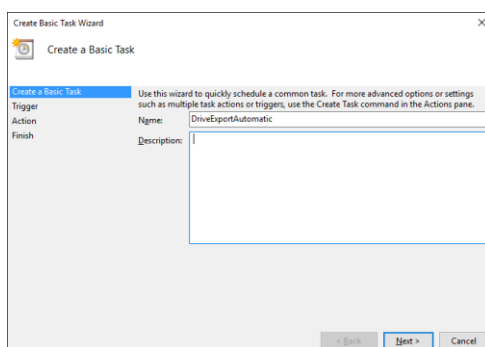
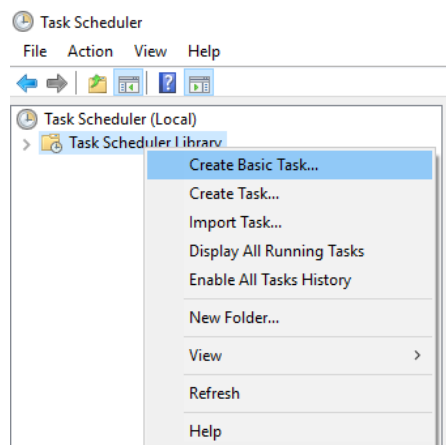
Silent mode

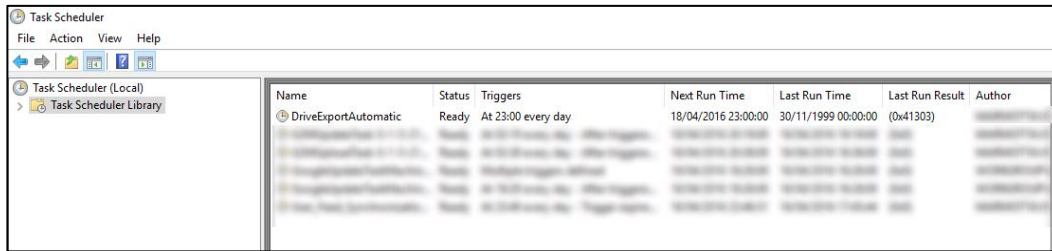
DriveExportCMD can be easily run by scheduler system (like Windows Task Scheduler). Some of these systems are able to capture the output generated by DriveExport on the standard out of the console. Under some systems there could be issues. To solve them, it's useful to run DriveExportCMD in Silent Mode. Using this mode, DriveExport does not generate output on the console standard output. The output is sent to the file DriveExportSilent.txt.

```
DriveExportCMD.exe configfilename.xml -silentmode
```

Scheduling with Windows Task Scheduler

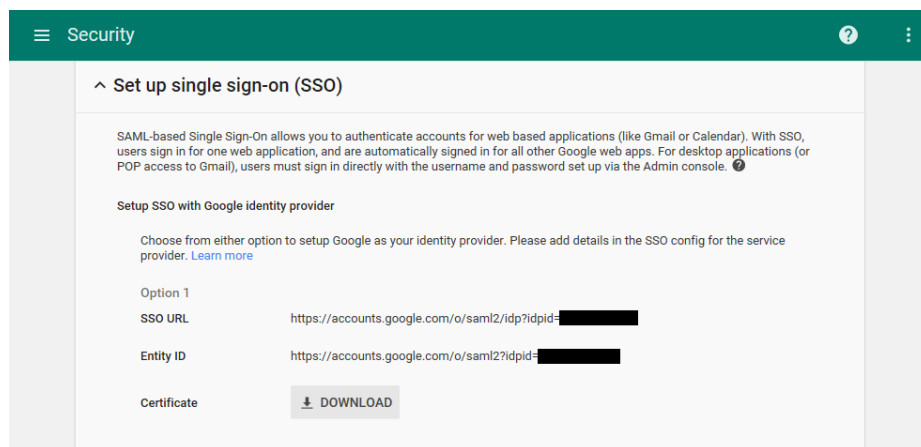
DriveExportCMD, the command line version, can be automatically run using the Windows Task Scheduler. In few steps, it's possible to setup the task.



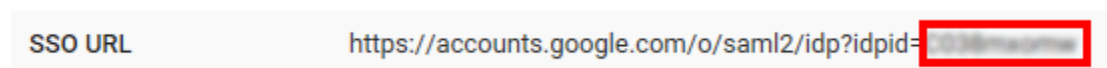


Obtaining the Unique ID of the Google Apps subscription

To obtain the Unique ID of a Google Apps subscription, open the Admin Console at <https://admin.google.com>, select “Security” and then “Set up single sign-on (SSO)”.



Your Unique ID is the code after the equality sign at the end of the SSO URL:



Run DriveExport on Linux

DriveExport can run on Linux although it is not officially supported. It requires Mono installed and https certificates repository updated. On Ubuntu 24.04 these commands set up the environment:

```
sudo apt-get update
sudo apt-get install mono-complete
```

On some older or different Linux distros, it is also required to update root certificates with:

```
mozroots --import --ask-remove
```

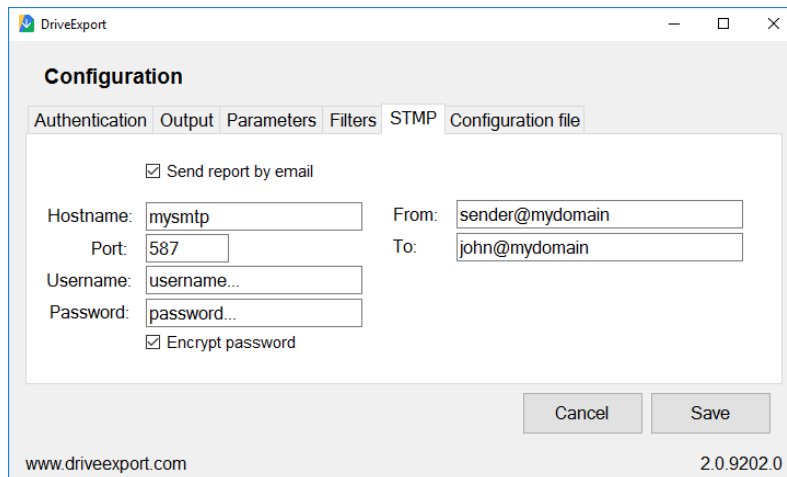
DriveExport can run from command line as follow:

```
mono ./DriveExportCMD.exe config.xml
```

Send the report by email

DriveExport can send the final html report by email. To activate the feature, set to “true” the option `SendReportEmail`, directly in the configuration xml file or by the configuration panel. It’s also necessary to set the SMTP parameters (host, port, username and password), the “from” address to use and the destination address/addresses (comma separated).

```
<SendReportEmail>false</SendReportEmail>
```



Security note: the SMTP password is stored in the xml configuration file. It can be stored encrypted, using the Windows DPAPI, or in clear-text mode (not secure!!!).

The encryption is controlled by the option “Encrypt password” (or `<SMTPPasswordEncrypt>true/false</SMTPPasswordEncrypt>` directly in the xml configuration file).

Example of XML for storing the password in clear text:

```
<SMTPPassword>mypassword</SMTPPassword>
<SMTPPasswordEncrypt>>false</SMTPPasswordEncrypt>
```

If you want to use the DPAPI protection, enter the password and active the option “Encrypt password” using the graphical Configuration panel. The password will be encrypted before saving. This option is available only using the Configuration panel of the graphical interface.

The encrypted data can only be decrypted by user that encrypted it and on the same computer. So, the XML configuration file cannot be created on a computer and run on another computer. And cannot be created by a user and used by another one.

Example of resulting encrypted data:

```
<SMTPPassword>AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAvdAltGfhE0eS/iyCB8It3gAAAAACAAAAAAQZgAA
AAEAACAAAAA9SoebZn/0rBfikZICp25X/vFN8bIuhjybQRnch/L7wAAAAA0gAAAAAIAACAAAAAxdTvVcjBYII
vB06ypDagZYPJ+Nz2GPRhCUj1jtiRW8DAAAADybzt8TQB1ekKD8gL1MUCHDgRMC6bBrzCHCXfI/2V/sL78jKog
32wqJL6askRGH15AAAAAnKuTBiuLFbf0mW5/fkGWLfD7xTUWu/TCJsp6s/fLa/gxv2V/w1K0YhDhZ0Z5024aAM
fXMoqRh0Tj4jjuj7t6VQ==</SMTPPassword>
```


Configuration parameters

The easiest way to modify the configuration parameters is through the DriveExport Configuration Editor app (DriveExportConfigEditor.exe). Some configuration parameters can be modified only by editing the xml file directly.

Note: some parameter names refer to “Team Drives”. It’s the former name of G Suite Drive “Shared Drives”.

WARNING: filters options (GroupFilter, UserFilter, UserListFileName, etc.) cannot be used to work around the number of users allowed by the license subscription.

	(1)	(2)	
LicenseCode			the license code assigned to you and obtained from DriveExport web site www.driveexport.com
DomainAdminEmail			the email of one of the administrators of your G Suite domain
OutputFolder			the folder on your computer where DriveExport will place exported files
LogFolder			the log output folder
OverwriteLogs	X		“true”: log files are overwritten on every run. “false”: every run generates a new set of log file.
GoogleServiceAccountEmail			the virtual email of the Google Service Account to be used (you created it during the steps in “Google Apps Configuration”)
GoogleServiceAccountPKCS12FileName			The full path to the certificate (p12 file) of the Google Service Account.
GoogleAcknowledgeAbuse			by default, Google does not allow to download files identified as abusive (malware, virus, etc.). If you want to bypass the block and download those files, activate this option. The value of the tag is exactly the full phrase below. <u>Remember: you do that at your own risk!!!</u> <code><GoogleAcknowledgeAbuse>ACTIVE - I understand and accept the risks of using it.</GoogleAcknowledgeAbuse></code>
DegreeOfParallelism			Number of parallel processes to be used to perform data export. Use a value between 1 and 4.

MaxFilenameLength			The maximum length of files names on the local hard-disk. Longer names are truncated and added with a pseudo-random sequence. (default is 50 characters)
MaxFoldernameLength			The maximum length of folders names on the local hard-disk. Longer names are truncated and added with a pseudo-random sequence. (default is 30 characters)
DeleteExtraFiles			How to manage the extra files no longer present on Google Drive but still present on local hard-disk. Values: 0 : Do not delete: the extra files will remain at the current location 1 : Move to special folder: the extra files will be moved to the special folder “__DeletedItems” 2 : Delete immediately: the extra files will be immediately delete from your local hard-disk. WARNING: be careful about activating this option. DriveExport could delete important files from your hard disk if not properly configured.
ManageExternalItems	X		(true/false) export also items shared by other users to the current user.
RemoveDomainFromFolderName	X		(true/false) DriveExport creates one folder for each user. If the option is false or omitted, the full primary email address is used as folder name (e.g. joe@example.com, hillary@example.com, etc.). If the option is set true, DriveExport uses only the username, omitting the @ and the domain (e.g. joe, hillary, etc.)
CustomExtensionPrefixForDocuments	X		allows to insert a custom prefix after the filename and before the standard file extension when DriveExport exports Google native documents (Documents, SpreadSheets and Presentations)
SendReportEmail	X	X	(true/false) If true, DriveExport sends the final report by email, too. SMTP parameters, required only if SendReportEmail is true. <ul style="list-style-type: none"> • SMTPServer: hostname or IP address of the SMTP server • SMTPPort: tcp port of the SMTP server • SMTPUsername: SMTP uername • SMTPPassword: SMTP password • SMTPPasswordEncrypt: “true” for encrypted password • SMTPEnableSSL: (true/false) use SSL for connecting to SMTP server • EmailFromAddress: sender email address • EmailToAddresses: list of destination email addresses (comma separated)
ProcessSuspendedUsers			if set to “true”, DriveExport will also process documents of suspended users.

SkipUserDrives	X		if set to “true”, DriveExport will not process users personal Google Drive areas (i.e. the “classic” Google Drive repositories).
SkipTeamDrives	X		if set to “true”, DriveExport will not process Google Shared Drives (formerly Team Drives).
GroupFilter	X	X	the list of Groups to be processed. The names are comma separated. If filled, DriveExport only processes users belonging to these groups and sub-groups.
UserFilter	X	X	the list of Users to be processed. The names are comma separated. If filled, DriveExport only processes users in the list.
OrgUnitsProcessList	X	X	List of organizational units, # separated, to be processed. The not matching ones, will be ignored. The units must be expressed as a “path” starting from root. <OrgUnitsProcessList>/MyOrgA#/OrgX/My org B</OrgUnitsProcessList>
OrgUnitsSkipList	X	X	List of organizational units, # separated, to be skipped. The units must be expressed as a “path” starting from root. <OrgUnitsSkipList>/MyOrgA#/OrgX/My org B<OrgUnitsSkipList>
UserListFileName	X	X	name of the text file containing the list of usernames to be processed. The file contains one username per line, in the format “username@domain”. DriveExport only process users in the list.
SkipUserListFileName	X	X	name of the text file containing the list of usernames to be skipped during processing. The file contains one username per line, in the format “username@domain”.
TeamDriveListFileName	X	X	name of the text file containing the list of Shared Drives names to be processed. The file contains one Shared Drive name per line.
SkipTeamDriveListFileName	X	X	name of the text file containing the list of Shared Drives names to be skipped. The file contains one Shared Drive name per line.
ActivateWA1	X		if set to “true”, activates “work around 1”
StopOnTooLongPathErrors	X		“true” → stops processing user or Team Drive when a too long path is found. Otherwise the process continues marking the user as “error”.
ProcessMyFoldersOnly			DO NOT CHANGE
RebuildMissingFolders			DO NOT CHANGE
GoogleAPILimit			DO NOT CHANGE
GoogleAPIMaxParallelism			DO NOT CHANGE
RevertToOldAPI			DO NOT CHANGE

RevertToOldTDCollect		DO NOT CHANGE
DebugMode		"true" → more information in the output log file
DumpLogToConsole		"true" → dump log rows to console, too

(1) optional

(2) only for paid license