Tresorit Active Directory Connector

V2.0

User’s Guide

tresorit

Contents

About Tresorit Active Directory Connector

Tresorit Active Directory Connector is an administrative tool developed by Tresorit for their Enterprise and Business customers. The tool can synchronize user accounts, tresors and tresor memberships from the customer's own directory service (or other data source) to the customer's Tresorit Business subscription.

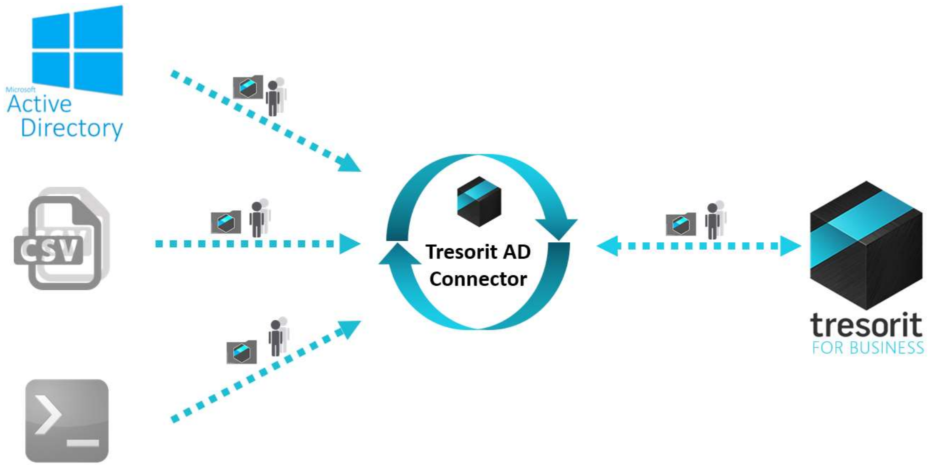


Figure 1 - Tresorit AD Connector high level data flow

Features

- Available for Business customers
- Can be installed on a Windows machine on premise
- One-way sync tool: only syncs data from the data source to the Tresorit business subscription
- Can sync subscription memberships
- Can sync tresors and tresor memberships [OPTIONAL]
- Out-of-the-box sync from Active Directory
- Supports custom data sources through file of command line API
- Can distinguish users and tresors synchronized from the data source and manually created ones. Even after enabling sync, Tresorit subscription members can invite members and create tresors manually. (Depending on subscription settings)
- Can auto-update to keep infrastructure secure
- Can be scheduled to run automatically time-to-time and keep subscription up-to-date

Synchronization logic

The synchronization process runs in a specific, well defined order by applying predefined sync rules.

The process always starts with a subscription membership synchronization followed by an optional tresor and tresor membership synchronization. This chapter describes the exact logic and order of the synchronization process.

About managed users and tresors

Using the sync tool does not remove or limit existing features of Tresorit. Users with proper permissions can still invite subscription members or create and manage tresors manually. To support this behavior Tresorit service automatically maintains a "management" state for all objects (both users and tresors) in a subscription.

Any subscription object (user or tresor) which was touched by the sync tool marked is marked as "managed", and remains in that state until it is deleted or removed from the subscription. More precisely:

A **tresorit user is managed** if she was invited by the sync tool OR invited manually but later the sync tool has found her in the subscription membership data source.

A **tresor is managed** if it was created by the sync tool OR created manually but later sync tool has found it in the tresor membership data source.

Tresorit AD Connector only modifies managed objects in the subscription. This ensures that if a non-managed (manually invited) user exists in a subscription the tool won't remove it. This also applies for non-managed tresors or non-managed members of managed tresors.

Notes:
Please note, that a non-managed object can become managed as soon as the tool founds it in the data source. In this case the previously non-managed object will be marked as managed.

Please also note, that the managed property is maintained fully automatically, and cannot be seen or modified by subscription members or administrators.

Logging and simulation mode

During the sync process the tool excessively logs all operations on both the screen (in interactive mode) and into a log file in the configured directly. All steps and operations of the sync can be found in the logs.

The tool also has a simulation mode in which no modifications are made to the subscription or tresors, but still all operations are logged. (These log lines will be marked as "simulated"). This mode is useful to test the system configuration before the production deployment of the sync tool in the company architecture.

The simulation mode can be activated both in the config file and by command line parameter. (See Appendix B – Configuration file and Appendix C – Input file and command line formats for details.)

Note: *Please note that the simulation mode of subscription membership sync and tresor membership sync is not connected. (Both sync steps will be simulated according to the current state of the Business subscription).*

Subscription membership sync

The subscription membership synchronization is always the first step of a synchronization cycle.

At first the tool will read and parse the data source (invalid lines / objects will be discarded by the tool). The information read should contain the email address, first- and last name and the account status of the user. Then the tool will retrieve the current subscription membership information from Tresorit servers. The users read from the data source will be matched with the current subscription users ***by their email*** address case insensitively. After the match completed, the following ruleset will be applied to each user.

Subscription sync rules

During the synchronization:

- The users will be processed sequentially, one after another
- Rules are applied in the given order for each user
- If multiple rules match, all of them will be applied for the user (in-order)
- The subscription admin is always skipped by the sync tool, even if found in the data source.
- Users are never deleted from the subscription, only suspended by the tool
- The data source must provide an email address and an account state for each user. First- and last names are optional, they are only used for sending out properly addressed invitations

Rule #	**Condition**	**Applied changes**
I.	A user (either invited or a current member) found both in the data source and in the current subscription, but the user is not managed.	The user will be marked as managed.
II.	New user found in the data source (but not in the current subscription)	User will be invited into the subscription as a “managed” user.
III.	A “managed” user is found in the current Tresorit subscription but not in the data source (user was deleted from the data source)	The user will be suspended. (She cannot log in to Tresorit while she is in disabled state.)
IV.	A “managed”, invited user is found in the current Tresorit subscription but not in the data source (user was invited but later deleted from the data source)	User’s subscription invitation will be revoked and her “managed” state will also be removed.
V.	An invited user is found both in the data source and in the current subscription, but the user’s account is disabled in the data source.	The user’s subscription invitation will be revoked and her “managed” state will also be removed.
VI.	A “managed” user is found both in the data source and in the current subscription, but the user’s account is disabled in the data source and enabled in Tresorit.	The user will be suspended. (She cannot log in to Tresorit while she is in disabled state.)
VII.	A user is found both in the data source and in the current subscription, but the user’s account is enabled in the data source and suspended in Tresorit.	The user will be re-enabled in Tresorit.

Table 1 - Subscription membership synchronization rules

The same rules can be described with the following state transition chart:

Source	Tresorit	Not found (Not a user)	Found, not a subscription member	Found, already invited into subscription		Found, already a subscription member			
				managed	unmanaged	managed		unmanaged	
						enabled	disabled	enabled	disabled
Found	**enabled**	Invite user to tresorit and subscription as managed	Invite user to subscription as managed	No action	Set user as managed	No action	Enable user	Set user as managed	Enable user and set her as managed
	disabled	No action	No action	Revoke invitation	Revoke invitation	Disable user	No action	Disable user and set her as managed	Set user as managed
Not found		No action	No action	Revoke invitation	No action	Disable user	No action	No action	No action

Table 2 - Subscription membership synchronization - user state transitions

Example

In the following example the first two columns show the data found in the data source, the third and fourth columns are show the users found in the Tresorit business subscription and the fifth column shows the action that will be performed by the sync process. (The users are already mapped by their email address in the chart for easier understanding.)

Data source (Active Directory or files)		Tresorit subscription		Action(s) performed by Tresorit AD Connector
User	**Status**	**User**	**Status**	
user1@example.com	enabled	*not found*		Invite user to subscription as managed
user2@example.com	disabled	*not found*		*no action*
user3@example.com	enabled	user3@example.com	invited, managed	*no action*
user4@example.com	disabled	user4@example.com	invited, managed	Revoke invitation
not found		user5@example.com	invited, managed	Revoke invitation
user6@example.com	enabled	user6@example.com	invited, unmanaged	Set user as managed
user7@example.com	disabled	user7@example.com	invited, unmanaged	Revoke invitation
not found		User8@example.com	invited, unmanaged	*no action*
user9@example.com	enabled	user9@example.com	member, managed, enabled	*no action*
user10@example.com	disabled	user10@example.com	member, managed, enabled	Disable user

not found		user11@example.com	member, managed, enabled	Disable user
user12@example.com	enabled	user12@example.com	member, managed, disabled	Enable user
user13@example.com	disabled	user13@example.com	member, managed, disabled	*no action*
not found		user14@example.com	member, managed, disabled	*no action*
user15@example.com	enabled	user15@example.com	member, unmanaged, enabled	Set user as managed
user16example.com	disabled	user16example.com	member, unmanaged, enabled	Disable user and set her as managed
not found		user17@example.com	member, unmanaged, enabled	*no action*
user18@example.com	enabled	user18@example.com	member, unmanaged, disabled	Enable user and set her as managed
user19@example.com	disabled	user19@example.com	member, unmanaged, disabled	Set user as managed
not found		user20@example.com	member, unmanaged, disabled	*no action*

Table 3 - Subscription membership sync examples

Tresor sync

The tresor synchronization is an optional feature of the tool. This helps in managing tresor memberships via the directory service. If it is enabled, it must run *after* a successful subscription membership synchronization, because the sync tool only syncs the tresor memberships of managed users. If email addresses of non-managed users found in the data source, those users will be discarded.

To enable Tresor sync, a user must log into the sync tool. During tresor sync, the tool will act a desktop Tresorit client and will create and modify tresors. The sync user must be at least a co-admin of the Tresorit Business subscription. We recommend creating a separate user for this purpose.

Tresorit AD Connector sync synchronizes only the tresors ***owned by*** the sync user (the user who is selected and logged into the tool). The tool can sync both tresors (it can create them) and the memberships and permissions of the managed tresors (by inviting or kicking other users from them or by setting user permissions).

When the synchronization starts the tool will read and parse the data source at first (invalid lines / objects will be discarded). The information chunks read should contain a tresor name, and optionally a permission level and a user email address.

Note: *Please note that this data is represented differently by Active Directory and file data sources. Read Appendix C – Input file and command line formats and Appendix D – Active Directory sync schema for details about data source schema.)*

Warning: *The tresors are matched by **name** (exact, case-sensitive match), while users are matched by **email** (case-insensitive match). If the sync user owns tresors with the exact same names the tool's behavior is undefined. Please never create multiple tresors with the same name for the sync user.*

The synchronization process runs in two consecutive phases. The first phase called "Tresor state sync" takes care about the tresors (it will match the existing tresors with the found ones and create the missing tresors) while the second phase called "tresor membership synchronization" will sync the members of the tresors with the data source.

Tresor state sync

The tresor state sync at first reads the data source and gathers all tresors found in it, then it queries all tresors accessible by the sync user and matches the two lists.

Notes:

Please note that the tool never deletes a tresor to prevent data loss. If a tresor should be deleted, then the tool will kick out all other users from it except the owner during Tresor membership sync.

Also note, that an empty tresor may be defined in the data source by omitting to define a permission and any users. This is a valid use case. The tresor will be created, but won't be shared with any other users.

The sync process will apply the following changes to the found tresors:

Rule #	Condition	Applied changes
I.	A tresor is found in the data source but not in the cloud in the list of the sync user's owned tresors.	*The tresor will be created by the sync user and will be marked as managed.* *→ Tresor member sync will take place*
II.	A tresor is found in the data source and also in the cloud as a managed tresor owned by the sync user	*Tresor won't be changed* *→ Tresor member sync will take place*
III.	A tresor is found in the cloud as a managed tresor owned by the sync user but not found in the data source	*→ Tresor member sync will take place* *(all members will be kicked out in phase 2)*
IV.	A tresor is found in the data source and also in the cloud but as an unmanaged tresor owned by the sync user	Tresor will be marked as managed *→ Tresor member sync will take place*

Table 4 - Tresor state sync rules

The same rules can be described with the following state transition chart:

Tresorit Source	Not found	Found	
		managed	unmanaged
Found	Tresor created (as managed)*	No change[1]	Tresor will be marked as managed*
Not found		No change[1,2]	No action

Table 5 - Tresor state sync state transitions

1 Tresor member sync (phase 2) will take place after these state transitions only. 2 - Tresor won't be deleted to prevent data loss, but all managed users will be kicked out during membership sync (except owner)

Tresor membership sync

Tresor membership sync takes place after a successful state sync of a tresor. During the sync the managed members of the tresors are synchronized with the data source.

Note: *the tresor membership sync phase may be skipped for a tresor. Check the previous tables for more information.*

During the synchronization:

- Managed users may be invited, kicked out or their permissions are altered according to the data source.
- Tresor membership sync always discards the tresor owner (owner 's permission is never altered)
- Tresor state sync may only set ***Viewer*** and ***Editor*** permissions.
- Tresor state sync may only change ***Viewer, Editor*** and ***Manager*** permissions
- Existing unmanaged tresor members will be left unchanged

The membership sync process will apply the following changes to the synced tresor:

Rule #	Condition	Applied changes
I.	A new member is found in the data source for the tresor, the user is a managed subscription user. (User is not found in the managed tresor.)	The user will be invited into the tresor with the proper permission read from the data source.
II.	A managed tresor member is found in an existing managed tresor, but not in the data source.	The user will be kicked out from the tresor.
III.	A managed, invited user is found in an existing managed tresor, but not in the data source.	The user's invitation will be revoked.
IV.	A managed tresor member is found in an existing managed tresor and also in the data source, but with different permission levels.	The user's permission level will be adjusted according to the data source.

Table 6 - Tresor state sync rules

The same rules can be described with the following state transition chart:

Tresorit Source	Not found	Found	
		managed	unmanaged
Found	Invited (with proper permission)	Permission will be changed if needed	Tresor will be marked as managed*
Not found		User will be kicked out OR her invitation will be revoked	No action

Table 7 - Tresor membership sync state transitions

Example

Consider the following example where the first three column shows the data read from the data source the columns from four to six show the tresors owned by the sync user in the cloud and the seventh column shows the sync actions done by the tool. The managed tresors and users are underlined in the table. (The owner user is not shown for tresors.)

Data source			Cloud (tresors of sync user)			Sync actions
Tresor	**Perm.**	**User email**	**Tresor**	**Perm.**	**User**	
Tresor A			Tresor A			*no action*
	Editor	user1@example.com		Editor	user1@example.com	*no action*
	Editor	user2@example.com		*Not found*		User invited as editor
	Viewer	user3@example.com		Manager	user3@example.com	Permission changed to viewer
	Editor	user4@example.com		Viewer	user4@example.com	Discarded, user is not managed
	Not found			Editor	user5@example.com	Discarded, user is not managed
	Not found			Editor (invited)	user6@example.com	Invitation revoked
	Not found			Viewer (invited)	user7@example.com	*no action*
Tresor B			*Not found*			Tresor created as managed
Tresor C			Tresor C			Tresor marked as managed
	Editor	user3@example.com		Editor	user3@example.com	*no action*
Not found			Tresor D			*no action*
				Viewer	user2@example.com	User kicked out
Not found			Tresor E			*no action*
				Editor	user3@example.com	*no action*
				Viewer	user4@example.com	*no action*

Table 8 - Tresor sync example

Install and configuration

This chapter will guide you through the installation and configuration of the tool.

Tresorit requirements

The Tresorit Active Directory connector is available for Tresorit Business customers and is downloadable from the settings tab of the admin center of the account owner. If you have a Business subscription but you cannot access the tool, please contact our sales or support team for further information.

System requirements

Operation system	Windows (either 32 or 64 bit edition)
Environment	Installed .net framework 4.5 or newer
Disc space	At least 100 Mb free disk space.
Permissions	Administrative permission for installation and configuration, standard permission for sync.

Network requirements

The tool communicates with our Account Management Public API at https://accountapi.tresorit.com/public through standard HTTPS protocol, using the default port 443 (uses requests with GET and POST verbs only, with *application/json* as content type).

The updates will be checked automatically from https://log.tresorit.com and downloaded from https://installerstorage.blob.core.windows.net, https://installerstorage-internetrouting.blob.core.windows.net, https://installerstorage-microsoftrouting.blob.core.windows.net and https://installer.tresorit.com. It is required to allow HTTPS traffic to both locations as well.

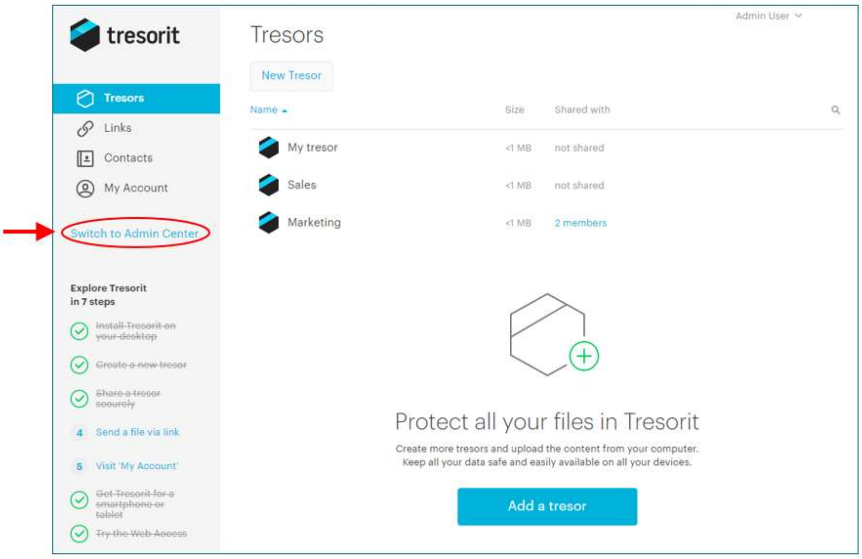
For full functionality (Tresor Member Sync) the tool must access https://tresorit.com and all its subdomains (https://*.tresorit.com) via https.

If Active Directory is used as data source, the directory server must be accessible from the computer running the tool.

Obtaining the tool and credentials

The tool can be downloaded by the Tresorit Business subscription owner through our Admin center.

Log in to our web client with the account owner and select “Switch to Admin Center” on the left tab.

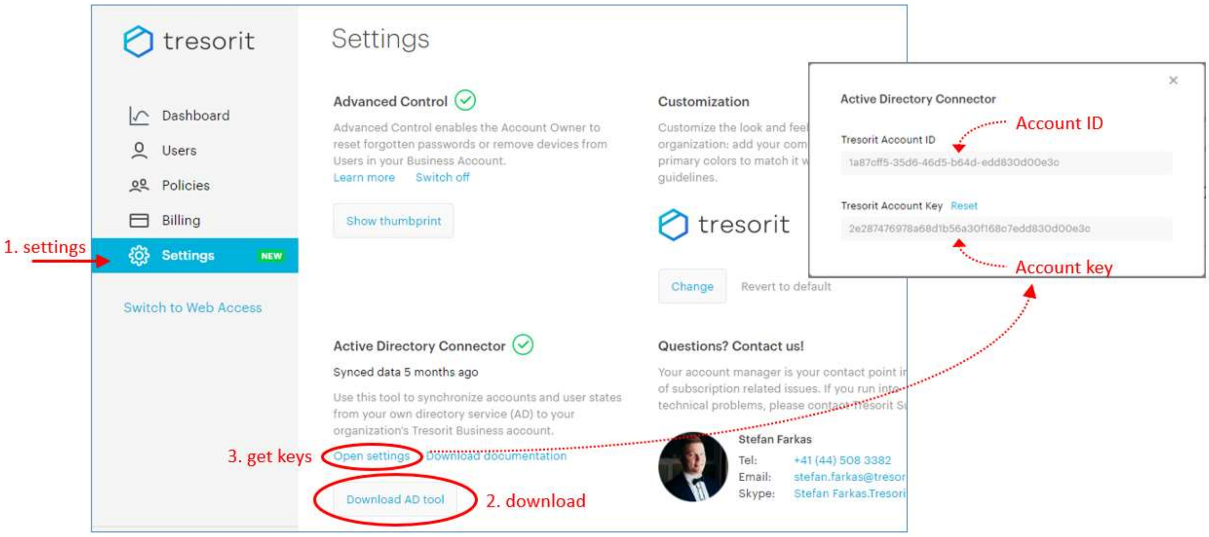


Protect all your files in Tresorit

Create more tresors and upload the content from your computer.
Keep all your data safe and easily available on all your devices.

Figure 2 - Switching from Tresorit web client to Admin center

In the admin center select the “Settings” menu. There you can find the “Active Directory Connector” section, where the tool can be downloaded along with the documentation and the access credentials for the tool can be obtained. You can come back any time to download the tool or the documentation again or the copy the credentials.



Advanced Control enables the Account Owner to reset forgotten passwords or remove devices from Users in your Business Account.

Customize the look and feel organization: add your com primary colors to match it w guidelines.

Synced data 5 months ago

Use this tool to synchronize accounts and user states from your own directory service (AD) to your organization's Tresorit Business account.

Your account manager is your contact point i of subscription related issues. If you run into technical problems, please contact Tresorit S

Stefan Farkas
Tel: +41 (44) 508 3382
Email: stefan.farkas@tresor
Skype: Stefan Farkas.Tresori

Active Directory Connector

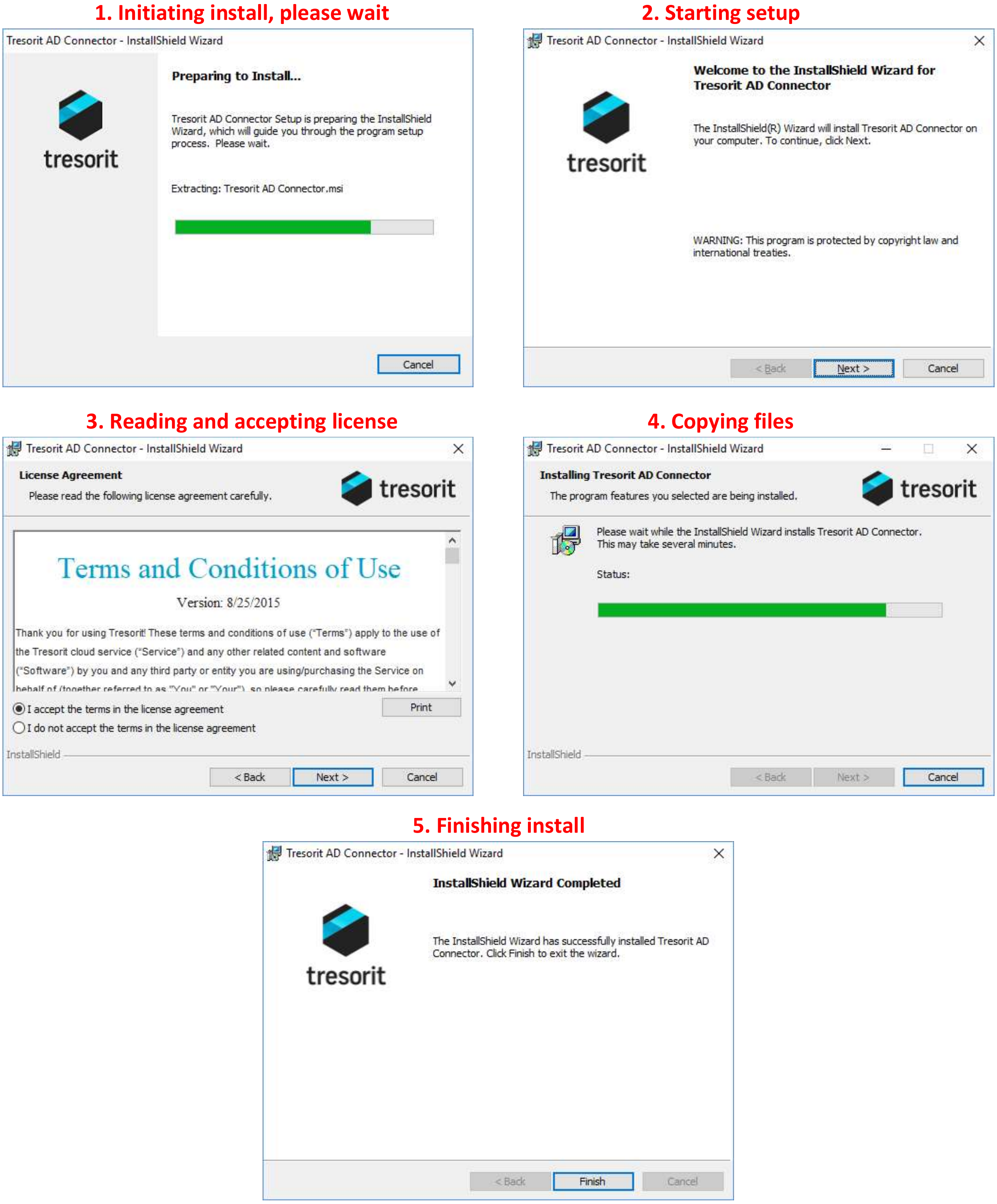
Tresorit Account ID: 1a87cff5-35d6-46d5-b64d-edd830d00e3c

Tresorit Account Key Reset: 2e287476978a68d1b56a30f168c7edd830d00e3c

Figure 3 - Downloading Active Directory Connector, its documentation and obtaining access credentials

Installation

Download the installer according to the previous section and copy it to the destination machine where it should be installed. Start the installer. You will need administrative privileges to complete the setup. Please follow the onscreen guide to install software.



1. Initiating install, please wait

Tresorit AD Connector - InstallShield Wizard

Preparing to Install...

Tresorit AD Connector Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.

Extracting: Tresorit AD Connector.msi

2. Starting setup

Tresorit AD Connector - InstallShield Wizard

Welcome to the InstallShield Wizard for Tresorit AD Connector

The InstallShield(R) Wizard will install Tresorit AD Connector on your computer. To continue, click Next.

WARNING: This program is protected by copyright law and international treaties.

3. Reading and accepting license

Tresorit AD Connector - InstallShield Wizard

License Agreement

Please read the following license agreement carefully.

Terms and Conditions of Use

Version: 8/25/2015

Thank you for using Tresorit! These terms and conditions of use ("Terms") apply to the use of the Tresorit cloud service ("Service") and any other related content and software ("Software") by you and any third party or entity you are using/purchasing the Service on behalf of (together referred to as "You" or "Your"), so please carefully read them before

● I accept the terms in the license agreement

○ I do not accept the terms in the license agreement

4. Copying files

Tresorit AD Connector - InstallShield Wizard

Installing Tresorit AD Connector

The program features you selected are being installed.

Please wait while the InstallShield Wizard installs Tresorit AD Connector. This may take several minutes.

Status:

5. Finishing install

Tresorit AD Connector - InstallShield Wizard

InstallShield Wizard Completed

The InstallShield Wizard has successfully installed Tresorit AD Connector. Click Finish to exit the wizard.

Figure 4 - Installation procedure

Configuration

After successful installation the tool must be configured. This could be done directly through its configuration file (see Appendix B – Configuration file for details) or through the built-in configuration utility.

To launch the config utility, open up a command windows and type “`C:\>`**`tresorit-connector init`**” into the command prompt.

The utility intelligently asks for each required configuration value and skips those which are not relevant according to the previously configured values. The tool can be re-run at any time. It will show the actual value for each key. Then the key can be modified by typing a new value and pressing enter, if enter is pressed without typing a new value the setting will be unchanged. To delete a setting, please press Ctrl + Del and then enter.

For further information about configuration and the available keys, please see Appendix B – Configuration file.

***Note:** Please note, that the tool can be used without a configuration file at all. For more information about this mode, check Appendix B – Configuration file and Appendix C – Input file and command line formats.*

Testing (simulation)

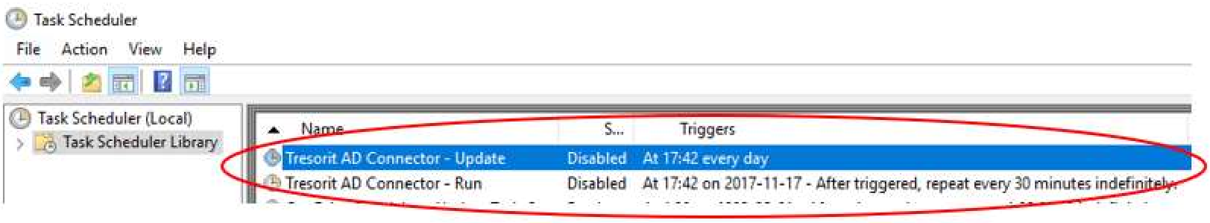
The tool has a simulation mode which can be set up either in the configuration file or activated by a command line option. (For further information please see Logging and simulation mode section in the previous chapter.)

We highly recommend setting up the tool at first in simulation mode and activate actual sync only when you are satisfied with the simulation results. During simulation the tool logs excessively into the configured log directory.

Scheduling

When the tool is properly configured and activated (simulation mode turned off), the tool can be scheduled for continuous synchronization and / or updated with scheduled tasks. During installation the tool creates two, originally disabled Windows Scheduler tasks. One for the continuous sync and one for the auto-update. To enable scheduling, you can start Windows task Scheduler and activate or re-configure those tasks.

The tool can automatically check-for updates and install them by calling it with the “update” subcommand. We recommend activating the scheduler task for the auto-update to keep the tool up-to-date and the system secure.



Name	S...	Triggers
Tresorit AD Connector - Update	Disabled	At 17:42 every day
Tresorit AD Connector - Run	Disabled	At 17:42 on 2017-11-17 - After triggered, repeat every 30 minutes indefinitely.

Figure 5 - Tresorit Active Directory Connector scheduler tasks

Frequently Asked Questions

In this chapter you can quickly find answers for the most common questions about Tresorit Active Directory Connector tool and its requirements and behavior.

Availability

Who can use Tresorit Active Directory Connector?
Owners of Tresorit Business subscriptions with large seat count are eligible for using the product. If you cannot access the tool on the admin center portal, please contact support.

Who can download Tresorit Active Directory Connector?
Tresorit Business subscription admins and co-admins can access the admin center and the download page of the tool.

System requirements

What is the minimal configuration for running the tool?
The tool can be installed on a Windows machine with installed .Net 4.5 and at least 100 Mb free disk space.

Do I need admin privileges for installing or running the tool?
Yes, admin privileges are required to install or configure the tool, but it can run in the name of a normal account.

Network requirements

Does the tool need internet access?
Yes, for correct operation the tool must access Tresorit services. For subscription membership sync the tool requires to access https://accountapi.tresorit.com/public.

For auto updates it also requires:

- https://log.tresorit.com
- https://installer.tresorit.com
- https://installerstorage.blob.core.windows.net

If tresor synchronization is also enabled, then the tool must access https://*.tresorit.com and all its subdomains via https protocol.

Sync capabilities

Can the tool sync from Tresorit?
No, the Active Directory Connector is a one-way sync tool. It can synchronize from external data sources into Tresorit.

What can the tool sync?
The tool can synchronize Tresorit Business subscription members and optionally tresors and tresor memberships of a specified sync user.

Can the tool sync files?
No, the tool is for subscription, tresor, and tresor membership synchronization only.

Subscription membership sync

If I turn on the sync, do I have to define all subscription users in the Active Directory?
No, you can still invite users into the subscription who are not members of the Active Directory.

Can I use subscription membership sync only (without tresor synchronization)?
Yes, the membership sync can be used without tresor sync.

Can the tool modify the subscription owner?
No, the tool will not change the subscription owner user's membership.

Can the tool remove users from the subscription?
No, the tool will not remove users from the subscription, it will rather suspend the users.

I try to invite a user into the subscription, but her invitation revoked automatically.
The user is probably already a managed user and she is not found by the tool in the data source. The tool will revoke the invitation after each run.

I try to enable a user account in the subscription, but he is disabled automatically.
The user is probably already a managed user and she is either not found or disabled in the data source. The tool will disable the Tresorit account after each run.

I try to disable a user account in the subscription, but he is enabled automatically.
The user is probably already a managed user and she is enabled in the data source. The tool will enable the Tresorit account after each run.

Tresor sync

If I turn on sync, do I have to define all tresors and tresor members in the Active Directory?
No, the users can still create tresors and invite people into them manually.

Who will be the owner of the synced tresors?
A specific subscription user, who must log into the tool permanently.

Do I have to use tresor sync?
No, tresor synchronization is an optional feature.

Can I use tresor sync only (without subscription membership synchronization)?
No, tresor sync must run after subscription membership synchronization, as it handles the same managed users found by the membership sync.

What permission levels can be synced by the tool?
The tool can set ***Viewer*** and ***Editor*** permissions.

Can the tool delete tresors?
No, the tool will not delete tresors to prevent data loss. If a managed tresor is not found in the data source anymore, the tool will kick out all users from it except the tresor owner (sync) user.

By which property do the tresors are matched in the data source and in the cloud?
The matching is done by the ***name*** of the tresors.

I invited a member into a tresor, but he kicked out after a short time again and again.

The tresor is probably a managed tresor and the user is a managed user. If the user is managed and not found as a member in the data source, the tool will kick him out after each run.

Data sources

Can the tool synchronize from Active Directory to Tresorit Business subscription?
Yes, sync from AD is fully supported out of the box.

Can the tool synchronize from other directory services?
Yes, through file or command line input API, but the data query and formatting must be done by a custom script provided by the customer.

Can the tool sync from files?
Yes, the tool can use csv files as input instead of an active directory.

The "managed" property of users and tresors

What does "managed" property of a user or tresor mean?
A tresor is managed if it is created by the sync tool OR created manually but later the tool found it in the data source (the tresor became managed).

Can I see or alter the "managed" property of a user or tresor?
No, the "managed" property is maintained automatically by Tresorit systems.

What is a "managed" tresor?
A tresor is managed if it is created by the sync tool OR created manually but later the tool found it in the data source (the tresor became managed).

What is a "managed" user?
A subscription user is managed if it is invited by the sync tool OR invited manually but later the tool found it in the data source (the user became managed). If a user is removed from a subscription (or her subscription invitation is revoked), her managed state is automatically deleted.

Appendix A – Command line interface

The application has several starting options and parameters to support all custom use-cases and configurations. This section describes the available interface options and return codes.

Application

The application by default installs into ***C:\Program Files (x86)\Tresorit AD Connector*** folder as **Tresorit.Tools.DirectoryConnector.exe**. We recommend referencing this executable from other scripts and from scheduled tasks. The installation also creates two launchers:

"tresorit-connector.exe" This app is also registered into the machine path for easy command line access. (You can simply call from command line.)

"Tresorit AD Connector.exe" Has the same name as the previous (v1.x) versions of the application. If the tool is called through this executable, then the app input and output will be converted to the old format.

Available sub-commands

```
tresorit-connector sync
```

Starts an automatic sync according to the config file or options.

```
tresorit-connector sync subscription
```

Starts a subscription membership synchronization according to the config file or options. (Tresor sync will be skipped, even if it is enabled in the config or by a command line option).

```
tresorit-connector sync tresors
```

Starts a tresor synchronization according to the config file or options. (Tresor subscription membership sync will be skipped, even if it is enabled in the config or by a command line option).

```
tresorit-connector sync all
```

Starts a full sync (subscription membership sync followed by a tresor sync) according to the config file or options. (If the tool is not configured for tresor sync the command will fail.)

```
tresorit-connector login
```

Logs in interactively into a tresorit account. Required for enabling tresor synchronization.

```
tresorit-connector logout
```

Logs out from the logged in tresorit account. After logout the tresor synchronization will not be possible.

```
tresorit-connector update
```

Checks for updates and automatically starts an updater in the background if a new version of the software was found.

```
tresorit-connector init
```

Starts the interactive command line configuration tool. The application will prompt for each setting. You can

- Change the setting by enter a new value and pressing enter
- Delete a value by pressing Ctrl + Del and enter
- Leave the value unchanged by pressing enter

You can re-run init tool any time. Please note that init command should be executed with elevated permissions, as it must access the config file with write permissions.

```
tresorit-connector version
```

Prints out the current installed software version.

Command line options

Note: *Please note that command line options may override the values set in the config file.*

Option	Description	Available for sub-commands
-k **--key**	Sets the Tresorit subscription account key.	*All commands*
-a **--account**	Sets the Tresorit subscription account id.	*All commands*
-l **--log-dir**	Sets the log directory.	*All commands*
--sync-tresors	Indicates to perform a tresor sync.	sync
-n **--dry-run**	If set, the engine will run in simulation mode (will not modify the subscription nor the tresors).	sync sync subscriptions sync tresors sync all
-d **--data-source**	Sets the data source type to use. Available values: ***stdi / file / ad***	sync sync subscriptions sync tresors sync all
--stdi	Forces the app to use command line input as data source. Please note that in standard input mode only one sync phase can run at a time (either subscription membership OR tresor sync). Do not use together with options: -d Should be used together with –subscription-sync / --tresor-sync	sync sync subscriptions sync tresors
--ad	Forces the app to use Active Directory as data source. Please note that in this mode ad credentials and group / ou names should be configured either in config file or by command line options.	sync sync subscriptions sync tresors sync all
--file	Forces the app to use csv files as data source. Please note that in this mode data source file paths should be configured either in config file or by command line options.	sync sync subscriptions sync tresors sync all
--ad-address	Sets the address of the active directory to use as data source	sync sync subscriptions sync tresors sync all
--ad-group	Sets the distinguished name of the subscription membership sync group in the active directory.	sync sync subscriptions sync tresors sync all
--ad-ou	Sets the distinguished name of the organizational unit in the active directory which contains the tresor sync groups.	sync sync subscriptions sync tresors sync all

--ad-password	Sets the password of the active directory to use as data source	sync sync subscriptions sync tresors sync all
--subscription-file	Sets the path of the subscription membership data file to use as data source	sync sync subscriptions sync tresors sync all
--tresor-file	Sets the path of the tresor membership data file to use as data source	sync sync subscriptions sync tresors sync all

Table 9 - List of available command line options

Return codes

Note: *For backward compatibility with the original version, if the program is called with the original method (directly invoking C:\Program Files (x86)\Tresorit AD Connector\Tresorit AD Connector.exe), the return codes will be altered according to the original version.*

Code	**Description**
0	Successful run
-1	Another instance of the tool is running.
-2	The tool started with invalid arguments
-3	Client is unsupported. Please run updater scheduled task, or update manually (run with “update” argument). Please remember, if you update manually, you need to run the application with as a user who has write permission to the install folder.
-4	Forbidden by policy. See logs for the specific reason.
-5	Unexpected error happened during the tresor member sync.
-6	The file source path is invalid in the config file, or does not have permission to read.
-7	Tresorit member sync group not found in the AD
-8	Unknown exception happened.
-9	The logged in user is not validated yet in Tresorit.
-10	Unable to access to the Tresorit Business Subscription.
-11	Client IP address is forbidden.
-12	Client location is forbidden.
-13	Client platform is forbidden.
-14	User is disabled.
-15	Relogin with password is required by policy. You must start the tool with “login” argument, and login again.
-16	You must setup two factor authentication. Please login into your Account (https://web.tresorit.com) and set up two factor authentication.
-17	Tresorit service currently is unavailable.
-18	Password has been changed. Please sign in again.
-19	This device has been expired. Please sign in again.
-20	This device has been unlinked remotely. Please sign in again.
-21	The logged in tresorit user has no Admin or CoAdmin permission in Tresorit Business Subscription.

-22	File access denied.
-23	File not found.
-24	File path too long. Please install at least .NET 4.6.2 when using a path that is longer than 250 characters.
-25	Unexpected file access error.

Table 10 - List of application return codes

Appendix B – Configuration file

The configuration file of the tool should be placed in the same folder where the executable is stored. The filename is “adconnector.config”. If the installation was used with the default settings, then the file will be installed to:

***C:\Program Files (x86)\Tresorit AD Connector*\adconnector.config**

Note: *The older versions of the tool used a config file called “Tresorit AD Connector.exe.config” in the same location. This file is no longer used. If the older version of the tool was installed, then the updater already migrated the contents of the file. The two file formats are not fully compatible.*

The configuration file can be edited by any text editor, or by using the AD connector tool’s build in config editor with the “init” subcommand.

Available configuration keys

Key	**Description**
DirectoryAddress	URL of the directory service. If not provided, the tool binds to the default node in the executing user’s subscription.
DirectoryUsername	Username for accessing the directory service. If not provided, tool will run in the name of the executing user. If provided, DirectoryAddress is also required.
DirectoryPassword	Password for accessing the directory service with the user above.
DirectorySyncGroup	Distinguished name of the group to be scanned by the tool for subscription users to feed subscription membership synchronization
DirectoryOrganizationalUnit	Distinguished name of the OU to be scanned by the tool for tresor-groups. Required only if AD is used as data source and SyncTresorMembers is enabled.
TresoritAccountId	Identifier of the Tresorit Business account, provided by Tresorit. (Can be obtained from Tresorit Admin Center by subscription admins and co-admins)
TresoritAccountKey	Access key provided by Tresorit for using Account Management Public API. (Can be obtained from Tresorit Admin Center by subscription admins and co-admins)
Simulation	If enabled, the connector tool runs in simulation mode. In simulation mode no actual change is made in the Tresorit subscription, but the log files are written properly. Useful for testing integration before activating the tool.
DataSource	Data source to use. Available values: ***stdi*** *(standard input)* ***/ file*** *(files)* ***/ ad*** *(Active Directory)*
SubscriptionMemberSourceFile	Full path of the subscription membership source file. Required only if file data source is used.
SyncTresorMembers	True if syncing of tresor members is needed (false by default) note: login is required for this operation
TresorMemberSourceFile	Full path of the tresor membership data source file. Required only if file data source is used and the SyncTresorMembers flag is enabled.
LogDirPath	Full path of the directory where the tool will be creating the log files.

Table 11 - Available configuration file keys

File format

The file format is standard .Net application setting format. Internally the file is an XML file where each setting should be added as a new child node of the “appSetting” node.

Example:

```xml
<?xml version="1.0" encoding="utf-8"?>
<appSettings>
  <add key="DirectoryAddress" value="dc01.customer.local"/>
  <add key="DirectoryUsername" value="tresoritadsyncuser@customer.local" />
  <add key="DirectoryPassword" value="accountpwd123"/>
  <add key="DirectorySyncGroup" value="CN=Tresorit,CN=Users,DC=ldap,DC=local"/>
  <add key="DirectoryOrganizationalUnit" value="OU=Tresorit,DC=ldap,DC=local"/>
  <add key="TresoritAccountId" value="c0628421-4a66-4b02-8dd7-2f54e480bc98" />
  <add key="TresoritAccountKey" value="tresoritapikey123"/>
  <add key="Simulation" value="true"/>
  <add key="SyncTresorMembers" value="true"/>
</appSettings>
```

Figure 6 - Configuration file example

Without a configuration file

It is not recommended, but the tool can be used without a configuration file at all. In this case, all options should be supplied through command line options.

Appendix C – Input file and command line formats

The tool can use external data files or direct command line input as data source. The data file and the command line input formats are the same for both subscription membership and tresor membership input files.

Both files / streams should correspond to the following criteria:

- The data is provided as comma separated values (CSV)
- There is at most one data record per line (line ending marker can be both \n or \r\n)
- The used separator character is the comma → **,**
- The data fields may be escaped in double quotes → “
- The file/stream should not contain any header rows
- Empty rows are omitted
- Rows beginning with **#** character are treated as comments (omitted)

Subscription membership data file

For subscription membership input the tool uses a file / stream referenced as subscription membership source file. Each line of the data set should contain four fields:

```
<email>,<first name>,<last name>,<status>
```

The email address is obligatory, Tresorit identifies users by email address.

If the first name / last name properties are unavailable, the fields can be left empty, but must be present. These values are only used for sending subscription invitations to new users.

Status is a logical field used to indicate the users account status. If the user is disabled, she cannot log even into her account. (Flag has the same meaning as account status in Active Directory). Available values for status flag: *enabled*, *disabled* (you can also use *true*, *yes* or *1* as enabled value and *false*, *no* or *0* as disabled value)

Note: *If a user presents multiple times in the data source, the tool will adjust her account status after reading each occurrence, and the last occurrence will be the file state of the user.*

Example data:

```
john.doe@example.com,John,Doe,true
jane.doe@example.com,Jane,Doe,false
# This line is a comment and will be discarded, as well as the following empty line

“unknown@example.com”,,,enabled
“john.john.jr@example.com”,”John, Jr.”,”John”,disabled
```

Example usages:

Assumed that the script “produce-subscription-data.bat” processes an external data source and writes a subscription membership data file to its standard output, the tool can be called either directly, or with the produced file.

Produce the file and then call the tool

```
C:\produce-subscription-data.bat >> subscriptiondata.csv
C:\tresorit-connector sync subscription --subscription-file subscriptiondata.csv
```

Or the tool can be fed directly with the output of the script

```
C:\produce-subscription-data.bat | tresorit-connector sync subscription -stdi
```

Tresor membership data file

For tresor sync input the tool uses a file / stream referenced as tresor membership source file. Each line of the data set should correspond to one of the free following formats (line formats can be mixed within a single file):

```
<tresor name>
<tresor name>,<permission>,<email>
<tresor name>,<permission>,<email>;<email>;...;<email>
```

The tresor name is obligatory, Tresorit AD Connector identifies tresors by exact name match. If a tresor name appears only in a line which only contains the name of the tresor, the sync tool will create the tresor without adding any other users (except the sync user as the owner).

Permission field describes the permission of the users(s). The available values are *Viewer* and *Editor*. Note, that the sync tool currently does not sync “Manager” permission level. This field is obligatory if any user is present in the third field. If the field is invalid, the line will be discarded and an error will be logged.

The third field can contain one or more email addresses separated by semicolons → **;**

The line formats described above can be mixed in a single data file as desired. The tool will read the entire dataset and process it logically before the actual sync will take place.

Note: *If a user if found multiple times in the data source as a member of a single tresor with both Viewer and Editor permissions, then the tool will accept the higher permission.*

Example data:

```
My tresor,Viewer,john.doe@example.com
My tresor,Viewer,jane.doe@example.com
# This line is a comment and will be discarded, as well as the following empty line

My other tresor,Editor,“user1@example.com”;“user2@example.com”;user3@example.com
My other tresor,Editor,“john.john.jr@example.com”
My empty tresor
```

Example usages:

Assumed that the script “produce-tresor-data.bat” processes an external data source and writes a tresor membership data file to its standard output, the tool can be called either directly, or with the produced file.

Produce the file and then call the tool

```
C:\produce-tresor-data.bat >> tresordata.csv
C:\tresorit-connector sync tresors --tresor-file tresordata.csv
```

Or the tool can be fed directly with the output of the script

```
C:\produce-tresor-data.bat | tresorit-connector sync tresors -stdi
```

Appendix D – Active Directory sync schema

When the tool uses an Active Directory instance directly as data source, the active directory must contain groups and organizational units properly named and prepared to enable the sync.

Access control

The AD can be accessed with two different credential modes:

- By providing a username and a password for the tool, where the given user has at least read access for the synced objects
- Or if the computer which executes the sync tool is the part of the same subscription as the AD, the tool can use the executing user's windows identity to access the AD.

Subscription membership sync group

For subscription membership synchronization the tool requires a single group. All users who should be managed (synced with) the Tresorit business subscription should be added to this group either directly or indirectly (through another group).

If a user account is disabled, then the corresponding Tresorit user will also be suspended automatically. (If the user is later re-enabled in the AD, then the Tresorit user will also be enabled after a sync.)

The tool will need the distinguished name of this group in its settings. The AD access user must have read permissions for this group.

Example:

If the sync group is called "TresoritSubscription", imagine the following logical structure:

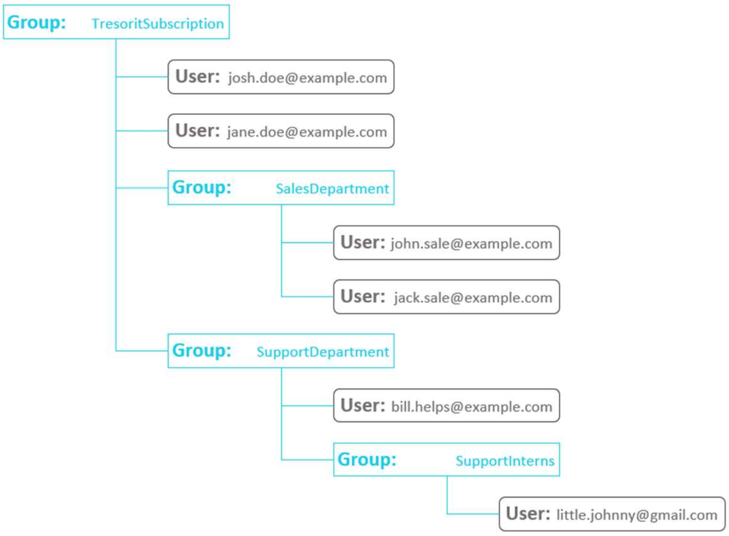

Group: TresoritSubscription
User: josh.doe@example.com
User: jane.doe@example.com
Group: SalesDepartment
User: john.sale@example.com
User: jack.sale@example.com
Group: SupportDepartment
User: bill.helps@example.com
Group: SupportInterns
User: little.johnny@gmail.com

Table 12 - Example AD sync group structure

After a successful sync run all five users present directly or indirectly in the group (josh.doe@example.com, jane.doe@example.com, john.sale@example.com, jack.sale@example.com, bill.helps@example.com, little.johnny@gmail.com) will be added to the tresorit subscription.

Tresor membership sync groups

If tresor sync is also enabled, Tresorit AD connector will look for specially named groups in a selected organizational unit in the Active Directory. The tool will need the distinguished name of the organizational unit as setting or command line parameter.

All groups in this organizational unit will represent a permission level in a tresor. Users found in those groups will be synced as members of the linked tresor in tresorit.

The name of these groups should be corresponding to the following format:

“{Tresor name}_{Permission}”

Where permission can “Viewer” or “Editor”. If a group’s name does not end with “_Viewer” or “_Editor”, then the group will be discarded during sync.

Example:

If the sync organizational unit is called “TresorGroups”, consider the following structure:

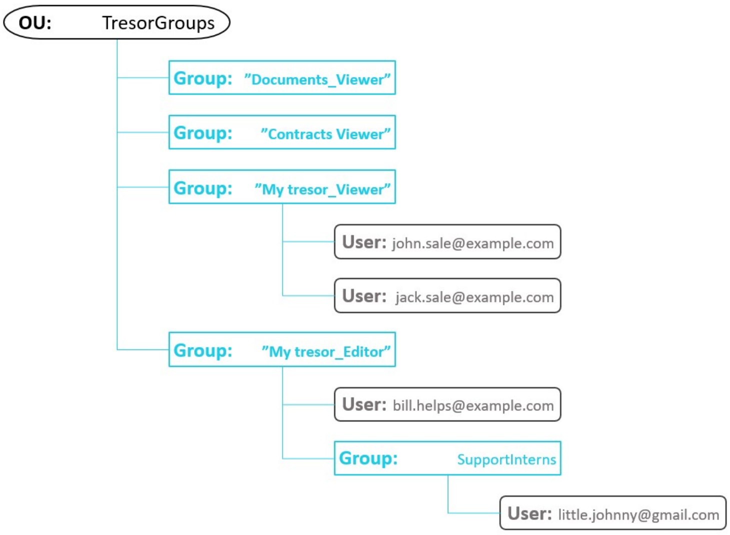

OU: TresorGroups
Group: ”Documents_Viewer”
Group: ”Contracts Viewer”
Group: ”My tresor_Viewer”
User: john.sale@example.com
User: jack.sale@example.com
Group: ”My tresor_Editor”
User: bill.helps@example.com
Group: SupportInterns
User: little.johnny@gmail.com

Table 13 - Example AD sync OU

After a successful sync run:

- A tresor with name „Documents” will be created, but no users will be added.
- The group “Contracts” will be discarded because it has an invalid name. (Missing underscore.)
- A tresor with name “My tresor” will be created
    - Users john.sale@example.com and jack.sale@example.com will be added to the tresor with “Viewer” privilege
    - Users bill.helps@example.com, little.johnny@gmail.com will be added to the tresor with “Editor” access level

Appendix E – User export

Summary

Upon successful validation of your config file, the command will list users under the current subscription with details about their status. The output of the command is the following:

```
Email,FirstName,LastName,AccountStatus,MembershipStatus,SubscriptionRole,Managed,AdvancedControlAccepted,PolicyTemplate,LastActiveTime,LastInviteSentDate
“john.doe@example.com”,John,Doe,enabled,member,admin,TRUE,TRUE, 1ace45d9-de34-41ca-a05e-3a522b9c92de,2019-06-03T09:50:59,2019-06-03T09:50:59
```

Description for the fields:

- Email: e-mail address of the subscription member (hereinafter referred to as user)
- First name: first name of user
- Last name: last name of user
- AccountStatus: whether the user is active or suspended
- MembershipStatus: whether the user is a member or just have been invited
- SubscriptionRole: whether the user is an admin, coadmin or member
- Managed: whether the user was synced with this tool or not
- AdvancedControlAccepted: whether the user accepted the advenced control feature or not
- PolicyTemplate: Name of the policy template
- LastActiveTime: The last time the user was active on one of their device
- LastInviteSentDate: The last time the invitation email was sent to the user

Usage

Options

- Output: name of the file to export to (-o <filepath>, --output <filepath>)
- File: Change mode to file output in csv format (--file)
- Stdi: Change mode to standard output (--stdi)

Example 1:

```
> tresorit-connector export users --file --output export.csv
> Get-Content export.csv
Email,FirstName,LastName,AccountStatus,MembershipStatus,SubscriptionRole,Managed,AdvancedControlAccepted,PolicyTemplate,LastActiveTime,LastInviteSentDate
“john.doe@example.com”,John,Doe,enabled,member,admin,TRUE,TRUE, 1ace45d9-de34-41ca-a05e-3a522b9c92de,2019-06-03T09:50:59,2019-06-03T09:50:59
```

Example 2:

```
> tresorit-connector export users --stdi
Email,FirstName,LastName,AccountStatus,MembershipStatus,SubscriptionRole,Managed,AdvancedControlAccepted,PolicyTemplate,LastActiveTime,LastInviteSentDate
“john.doe@example.com”,John,Doe,enabled,member,admin,TRUE,TRUE, 1ace45d9-de34-41ca-a05e-3a522b9c92de,2019-06-03T09:50:59,2019-06-03T09:50:59
```

Appendix F – User management

Summary

User management makes basic operations like add and remove possible on users. The full operation list includes invite, remove, suspend and unsuspend.

Currently supported commands:

- Add user: invite a user to the subscription
- Suspend user: suspends a user
- Unsuspend user : unsuspends a user
- Remove user: removes a user from the subscription

All commands share the same signature which means they need at least one email address as input parameter but you can list as much as you want by separating them with a single space.

Add user

Add user has a slightly different behavior as you have the option to set the policy template of the invited user. When you add a user to the subscription they will be assigned to the Default policy template. To override this you have to use the -p/--policytemplate <name> parameter. If the template does not exist, you cannot add the user to it.

Add user sets the membership type of the invited user to *member*.

Examples

```
> tresorit-connector add user john.doe@example.com
Adding users succeeded
```

```
> tresorit-connector add user john.doe@example.com jane.doe@example.com -p Default
Adding users succeeded
```

```
> tresorit-connector suspend user john.doe@example.com jane.doe@example.com
Suspending users succeeded
```

```
> tresorit-connector unsuspend user john.doe@example.com
Unsuspending users succeeded
```

```
> tresorit-connector remove user john.doe@example.com
User removal succeeded
```