Adobe Consulting

User Synchronization Tool Documentation

29 Jan 2017

Table of Contents

1 Introduction

Adobe User Sync is a command-line tool that moves user and group information from your organization's enterprise directory system (such as Active Directory or other LDAP system) to the Adobe User Management system.

Each time you run the tool it looks for differences between the user information in the two systems, and updates the Adobe system to match the enterprise directory.

1.1 Prerequisites

The User Sync Tool is run from a server that you operate. That system must have an internet connection to be able to contact Adobe's user management systems. It must also be able to access your enterprise directory system.

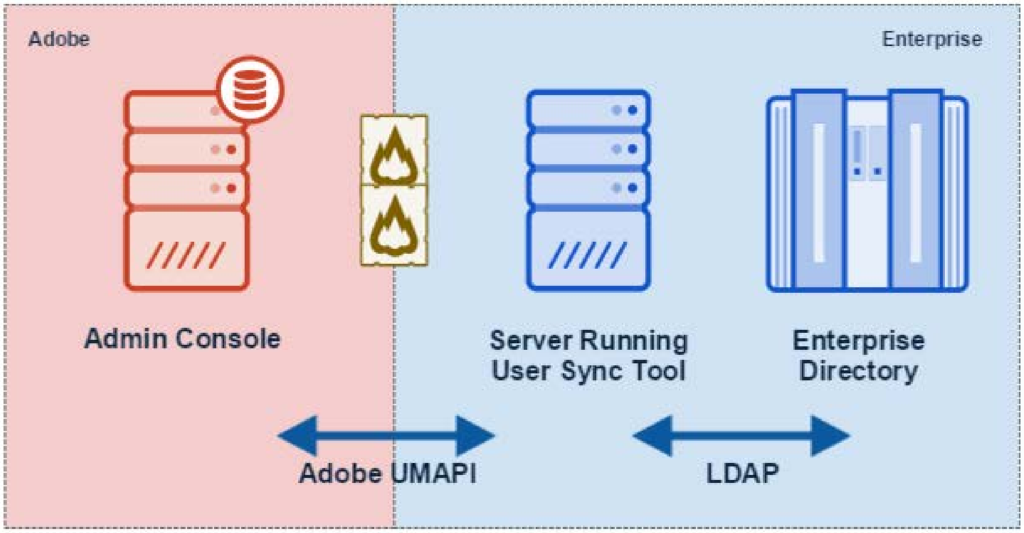
Python 2.7.9 or higher needs to be installed.

The User Sync tool is a client of the User Management API (UMAPI). In order to use it, you must first register it as an API client in the Adobe.io Console, then install and configure the tool, as described below.

The operation of the tool is controlled by local configuration files and command invocation parameters that provide support for a variety of configurations. You can control, for example, which users are to be synced, how directory groups are to be mapped to Adobe user groups and product configurations, and a variety of other options.

1.2 Operation overview

The User Sync Tool communicates with your enterprise directory through LDAP protocols, and with Adobe's Admin console through the Adobe User Management API (UMAPI) in order to update the user account data for your organization. The following figure illustrates the data flow between systems.



Each time you run the tool:

- User Sync Tool requests employee records from an Enterprise Directory System.
- User Sync Tool requests current users and associated product configurations from the Adobe Admin Console through the User Management API.
- User Sync Tool determines which users need to be created, deleted, or updated, and what user-group and product configuration memberships they should have, based on rules you have defined in the User Sync configuration files.
- User Sync Tool makes the required changes to the Adobe Admin Console through the User Management API.

2 Setup and Installation

The use of the User Sync tool depends on your enterprise having set up Product License Configurations in the Adobe Admin Console. For more information about how to do this, see the Configure Services help page.

2.1 Setup a User Management API integration on adobe.io

The User Sync tool is a client of the User Management API. Before you install the tool, you must register it as a client of the API by adding an *integration* in the Adobe I/O Developer Portal. You will need to add an Enterprise Key integration in order to obtain the credentials the tool needs to access the Adobe User Management system.

The steps required for integration are described in the Getting Started section of the Adobe.io User Management API website.

- The registration process requires that you create a certificate and a JWT (Java Web Token). Instructions are included in the documentation on Adobe.io.
- For complete information about the integration setup process and certificate requirements, see https://www.adobe.io/products/usermanagement/docs/setup.

When the process is complete, you will get an **API key**, a **client ID**, and a **client secret** that the tool will use to communicate securely with the Admin Console. When you install the User Sync tool, you must provide these as initial configuration values that the tool requires to access your organization's user information store in Adobe.

2.2 Map product configurations to your directory groups

Adobe users are granted access to Adobe products and services by creating Product Configurations in the Adobe Admin Console and then adding users to those configurations (which behave much like groups). The User Sync tool can grant product access to users by adding users to those product configurations based on their directory group memberships.

You will need to create the product configurations in the Adobe Admin Console before User Sync can run.

Checking your products and product configurations

Before you start configuring User Sync, you must know what Adobe products your enterprise uses, and what Product License Configurations are defined in the Adobe User Management system, and create them if someone else hasn't already set them up.

Please visit this link for more information.

Mapping between systems

Once you have defined user groups and product configurations in the Adobe Admin Console, you must create a mapping to those groups and configurations from groups in your own Enterprise Directory.

1. Identify the types of product access that users will need (such as All Access, Individual Product Access).
2. Ensure that product configurations exist in the Admin Console to reflect the product access needs of Creative Cloud Enterprise users.
3. Ensure that corresponding directory groups exist in the Enterprise Directory. For example, a Directory group corresponding to an "All Access" product configuration might be called "ADOBE-ALL-ACCESS".

2.3 Install the User Sync tool

System requirements

The User Sync tool is implemented using Python and supports version 2.7.9 or higher. For each environment in which you intend to install, configure and run the script, you must make sure that Python has been installed on the operating system before moving to the next step. For more information, see the Python web site.

The tool is built using a Python LDAP package, `pyldap`, which in turn is built on the `OpenLDAP` client library. Windows Server, Apple OSX and many flavors of Linux have an `OpenLDAP` client installed out of the box. However, some UNIX operating systems, such as OpenBSD and FreeBSD do not have this included in the base installation.

Check your environment to be sure that an `OpenLDAP` client is installed before running the script. If it is not present in your system, you must install it before you install the User Sync tool.

Installation

The User Sync Tool is available on github. The main page is located here: https://github.com/adobe-apiplatform/user-sync.py

Click the Releases link to locate the latest release. From there, download the .tar.gz file and locate a build for your environment. If you are building from source, you can download the Source Code package which corresponds to the release, or use the latest source off the master branch.

Builds for Windows, OSX, and Ubuntu are available. Locate the file UserSync (UserSync.PWX for windows) and place it in a directory of your choosing.

Example configuration files are also available in the .tar.gz file. You will want to copy these out and edit and rename them to make your own configuration files.

To run the User Sync tool, run the Python package file, UserSync (or `UserSync.PEX`. on Windows).

Installing in Windows

A file path length limit of 260 characters is enforced the Windows operating system. When executing the PEX file, it creates a temporary location to extract the contents of its package. It is possible that with path exceeds 260 characters which will cause the script to fail.

To work around this issue, create an environment variable in Windows called PEX_ROOT to override the behavior of creating the cache in the home directory. Set the path to "C:\user-sync\.pex ". By doing so, you now have a shortened path for the PEX cache and will avoid the file path length limit issue on Windows. This should be done prior to executing the PEX file.

3 Configuration

3.1 Configuration overview

Typically, three configuration files must be setup to run the User Sync Tool. One contains credentials and access information for calling the Adobe User Management APIs. A second contains credentials and access information for accessing the Enterprise Directory. The third is the main configuration file that defines the mapping of directory groups to Adobe product configurations and user groups. In complex configurations, a fourth configuration file is sometimes needed to setup access to multiple organizations.

A good approach to creating the configuration files is to copy the example configuration files, then rename and edit them for your environment. The example configuration files contain comments showing all possible configuration items. You can uncomment items that you need to use.

Configurations files are in YML format. You can read all about yml here. Just a couple of important things to note: indentation matters. Sections and hierarchy in the file are based on indentation and indentation MUST BE spaces, not tabs. Make sure you don't have tabs in the indentation.

A second thing is that the dash character "-" is used to form a list of values. For example,

```
dashboard_groups:
 - Photoshop Users
 - Lightroom Users
```

defines a list named "dashboard_groups" with two items in it. This looks confusing sometimes when there is only one item in the list:

```
dashboard_groups:
 - Photoshop Users
```

Create the dashboard-config.yml file

To setup a configuration specific to your enterprise requirements, please refer to `example.dashboard-config.yml` for a configuration template. Make a copy of this file and rename it to `dashboard-config.yml` and open it with a plain text editor of your choice. You can go ahead and do this for the other two files you will need, example.connector-ldap.yml and example.user-sync-config.yml, at this point also.

Once you have access to the adobe.io console and an integration is set up in adobe.io, please take note of the following configuration items:

1. Organization ID
2. API Key
3. Client Secret
4. Technical Account ID
5. Private Certificate

Edit the dashboard-config.yml file and put the values in their respective places in the "enterprise" section.

```
enterprise:
  org_id: "Organization ID goes here"
  api_key: "API key goes here"
  client_secret: "Client Secret goes here"
  tech_acct: "Tech Account ID goes here"
  priv_key_path: "Path to Private Certificate goes here"
```

An enterprise will usually have one admin console instance. However, it is possible for an enterprise to have more than one Admin Console in complex situations with multiple contracts or divisions. In that case there may be more than one console that wants to claim the same domain as the one claimed by an owning console. In this situation, the Admin Consoles can be linked together with an owning console and an accessor relationship. The User Sync Tool is equipped to synchronize between all of the Admin Consoles. Refer to Section 5 for more detail.

Important note: This file contains credentials used to access your Adobe organization on your behalf. You should protect this file at least as you would a password. Limit access to authorized individuals or store the data in a credential management system.

Create the connector-ldap.yml file

Edit this file and set username, password, host, and base_dn to enable access to your enterprise directory system.

Important note: This file contains credentials used to access your enterprise directory on your behalf. You should protect this file at least as you would a password. Limit access to authorized individuals or store the data in a credential management system.

Create the user-sync-config.yml file

Finally, edit the user-sync-config.yml file. This is the main configuration file and is divided into three sections: dashboard, directory, and logging. The most important part is the group map that is part of the directory section. The group map defines the correspondence between directory groups and Adobe product configurations and user groups. You must have an entry here for each directory group that represents access to an Adobe product or products. For that group, list the product configuration(s) to which users in that directory group should be granted access.

`dashboard`

```
dashboard:
  owning: dashboard-config.yml
```

The dashboard section specifies the file containing the configuration values that will used by the User Sync tool to connect to the Adobe Admin Console through the User Management API.

`directory`

```
directory:
  connectors:
    ldap: connector-ldap.yml
```

This entry specifies the name of the file that contains the credentials and host address needed to access the enterprise directory system.

Note: The service account used to access the LDAP should only be read access and never be able to write to the directory.

```
groups:
    - directory_group: Acrobat
      dashboard_groups:
        - Default Acrobat Pro DC configuration
```

The directory group to product configuration or user group mappings are set up under the “groups:” header. These mappings determine, for each user who is a member of the “directory_group” (from the Enterprise Directory), the

Adobe user group or product configuration that user will be made a member of. If this mapping is defined, it is required that the product configuration names are created in the Admin Console prior to running the User Sync Tool. **It will not create the product configuration on the Adobe side at run time.**

Also note that this mapping and group membership adjustment are done only if the –process-groups command line parameter is present. If this parameter is not present, the group mapping is not used.

`logging`

```
logging:
  log_to_file: True
  file_log_directory: logs
  file_log_level: debug
  console_log_level: debug
```

Specifies an audit trail path as well as the verbosity of the log statements stored.

The *file_log_directory* indicates where the log files will be written. It is a required field if the *logToFile* property is set to true. The application will throw an exception if you try to execute the application without providing a proper path for logging. A new log file is date-time stamped for every execution of the application. Please ensure the proper read/write permissions are set on the file and directory where this log will be written.

The *file_log_level* value defines the log level that is written to file and the *console_log_level* value defines the log level that is written to the console during User Sync Tool execution. For the logging level, you can specify "debug", "info", "warning", "error", or "critical" depending on the detail of the logs you require. This is in ascending order, meaning "debug" < "critical". The *file_log_level* can be different than the *console_log_level*. The level of detail can be adjusted to suit your needs.

When reviewing the logs for errors, pay attention to any log entries that contain WARNING, ERROR or CRITICAL as they will have a description that accompanies the status.

```
2017-01-19 12:54:04 7516 WARNING dashboard.trustee.org1.action - Error
requestID: action_5 code: "error.user.not_found" message: "No valid users were
found in the request"
```

In the example above, a warning was logged on 2017-01-19 at 12:54:04 during execution. An action ran into an error with the code "error.user.not_found". The description of the associated error code is also included as well. This is an example of a message that is returned by the User Management API if it has detected any issues executing the action requested by the User Sync Tool.

When using the log message to troubleshoot, make use of the requestID value to help narrow down the data in the log to only show information that is associated to that request. Based on the above example, searching for "action_5" returns the following detail,

```
2017-01-19 12:54:04 7516 INFO dashboard.trustee.org1.action - Added action:
{"do": [{"add": {"product": ["default adobe enterprise support program
configuration"]}}], "requestID": "action_5", "user": "cceuser2@ensemble.ca"}
```

It now returns some more information about the action that was performed which resulted in the WARNING message being returned. In this case, the User Sync Tool was attempting to tell the User Management API to add the "default adobe enterprise support program configuration" to the user cceuser2@ensemble.ca and the API returns saying that "cceuser2@ensemble.ca" doesn't exist in the Admin Console yet so it's unable to do so.

Note that when modifying this file, it is important to keep all indentation consistent and in line with the example.user-sync-config.yml file. Further, tabs are not allowed. Use spaces for all indentation. You may need to check your text editor settings to ensure that tabs are not used.

4 Command Parameters

The User Sync command accepts parameters that support various behaviors to meet your synchronization needs.

Once the User Sync Tool has been configured, open up a command line or command prompt and execute the following line to run the application,

```
user-sync [optional arguments]
```

The information for each parameter is as follows,

Optional arguments:

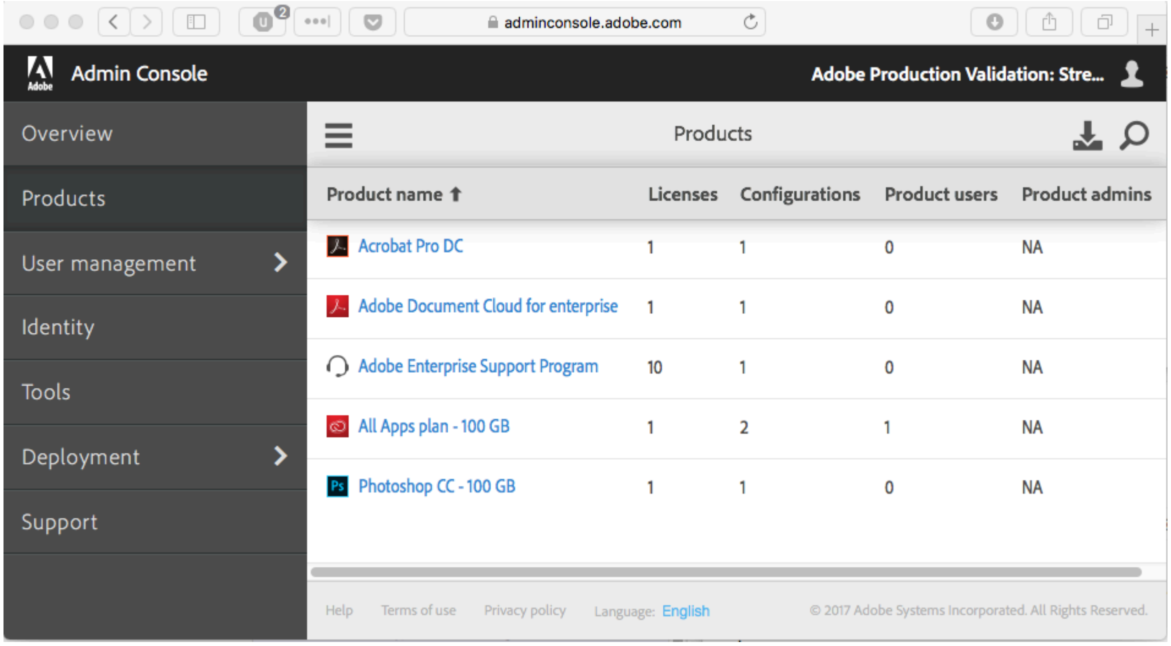
-h, --help	Show this help message and exit
-v, --version	Show program's version number and exit
-t, --test-mode	Run API action calls in test mode (does not execute changes). Logs what would have been executed.
-c path, --config-path path	Specify path to config files. (default: "")
--config-filename filename	Main config filename. (default: "user-sync-config.yml")
--users all|file|group [arg1 ...]	Specify the users to be considered for sync. Legal values are 'all' (the default), 'group name or names' (one or more specified AD groups), 'file f' (a specified input file).
--user-filter pattern	Limit the selected set of users that may be examined for syncing, with the pattern being a regular expression. Refer to https://docs.python.org/2/library/re.html for guidance on constructing regular expressions in Python

--source-filter connector:file	Reference a separate file that contains a ldap query filter. (for example, --source-filter ldap:foo.yml). This parameter is used to limit the scope of the LDAP query.
	Ex: A file foo.yml that contains an ldap filter
	`# specifies the source filter while retrieving the users from ldap.  The filter is a ldap query string that will be passed as is to the ldap server`
	`#`
	`# Example:`
	`all_users_filter: <insert ldap filter>`

--update-user-info	If user information differs between the customer side and the Adobe side, the Adobe side is updated to match. This includes the firstname and lastname fields. Specifying this option can have a significant impact on performance. It is recommended to run with this option only occasionally.
--process-groups	If the membership in mapped groups differs between the customer side and the Adobe side, the group membership is updated on the Adobe side so that the memberships in mapped groups matches the customer side.
--remove-nonexistent-users	Causes the user sync tool to remove Federated users that exist on the Adobe side if they are not in the customer side Directory. This has the effect of removing the user account from the organization.
--generate-remove-list output_path	Processing similar to --remove-nonexistent-users except that rather than performing removals, a file is generated (with the given pathname) listing users who would be removed. This file can then be given in the --remove-list argument in a subsequent run.
-d input_path, --remove-list input_path	Specifies the file containing the list of users to be removed. Users on this list are removeFromOrg'd on the Adobe side.

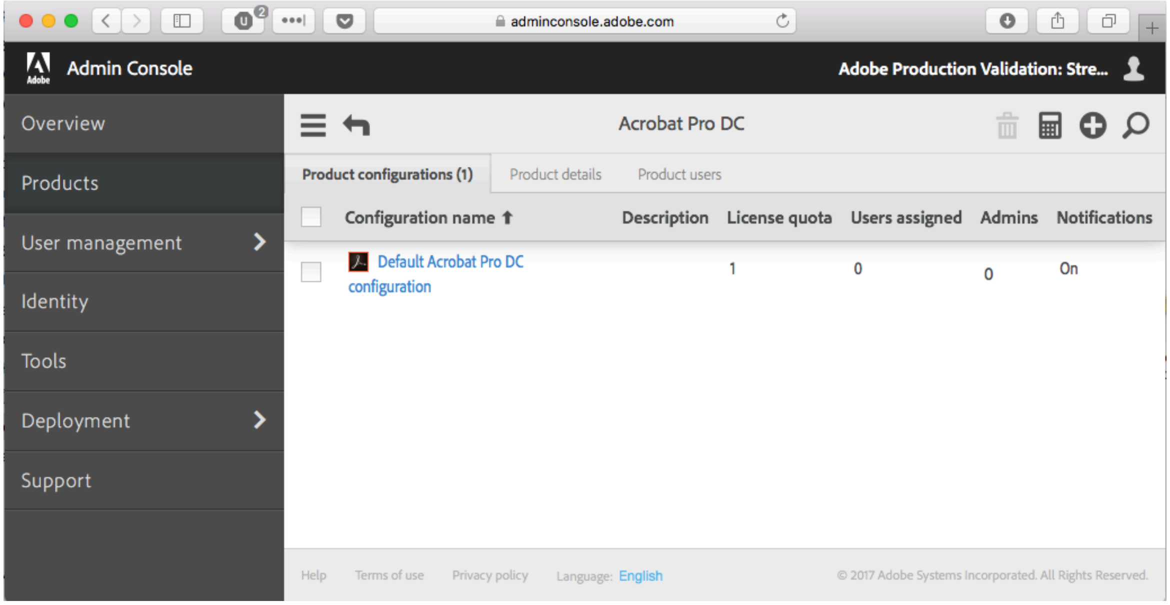
5 Example Configurations

Go to the Products section in the Adobe Admin Console to see the products that are enabled for your enterprise.



Product name ⬆	Licenses	Configurations	Product users	Product admins
Acrobat Pro DC	1	1	0	NA
Adobe Document Cloud for enterprise	1	1	0	NA
Adobe Enterprise Support Program	10	1	0	NA
All Apps plan - 100 GB	1	2	1	NA
Photoshop CC - 100 GB	1	1	0	NA

Click a product to see the details of Product License Configurations that have been defined for that product.



	Configuration name	Description	License quota	Users assigned	Admins	Notifications
□	Default Acrobat Pro DC configuration		1	0	0	On

4. If you do not yet have any configurations, you must use the Console to create them. Go ahead and create them as they are required to exist prior to configuring the User Sync Tool.

Once the configuration is complete, the User Sync Tool is now ready to be configured.
Below is an example of a basic user sync configuration.

user-sync-config.yml

```
dashboard:
  owning: dashboard-config.yml
  user_identity_type: federatedID

directory:
  connectors:
    ldap: connector-ldap.yml

  groups:
    - directory_group: Acrobat
      dashboard_groups:
        - Default Acrobat Pro DC configuration

    - directory_group: Photoshop
      dashboard_groups:
        - "Default Photoshop CC - 100 GB configuration"
        - "Default All Apps plan - 100 GB configuration"
        - "Default Adobe Document Cloud for enterprise configuration"
        - "Default Adobe Enterprise Support Program configuration"

logging:
  log_to_file: True
  file_log_directory: logs
  file_log_level: debug
  console_log_level: debug
```

connector-ldap.yml

```
username: “ldap username”
password: “ldap password”
host: ldap://<ldap host>
base_dn: “base dn”

all_users_filter: “(&(objectClass=person)(objectClass=top))"
```

dashboard-config.yml

```
server:
  # This section describes the location of the servers used for the
dashboard. Default is:
  # host: usermanagement.adobe.io
  # endpoint: /v2/usermanagement
  # ims_host: ims-na1.adobelogin.com
  # ims_endpoint_jwt: /ims/exchange/jwt

enterprise:
  org_id: "Org ID goes here"
  api_key: "API key goes here"
  client_secret: "Client secret goes here"
  tech_acct: "Tech account ID goes here"
  priv_key_path: "Path to private.key goes here"
```

6 Usage Scenarios

There are various ways to integrate the User Sync tool into your enterprise processes. This section provides some examples of how you might configure and run the tool for the following typical scenarios:

- Sync users and group memberships by adding, updating, and deleting users.
- Sync only specific users.
- Sync only users. Product access then needs to be handled using the Admin Console.
- Sync users (and groups) but do not delete users; instead keep a list of users to be deleted.
- Sync user deletions based on the list.
- Limit syncing to users matching certain patterns
- Sync against a csv file rather than a directory system.
- Process user updates and deletions in response to push notifications or other file-based updates.

6.1 Add users and groups to Adobe Admin Console

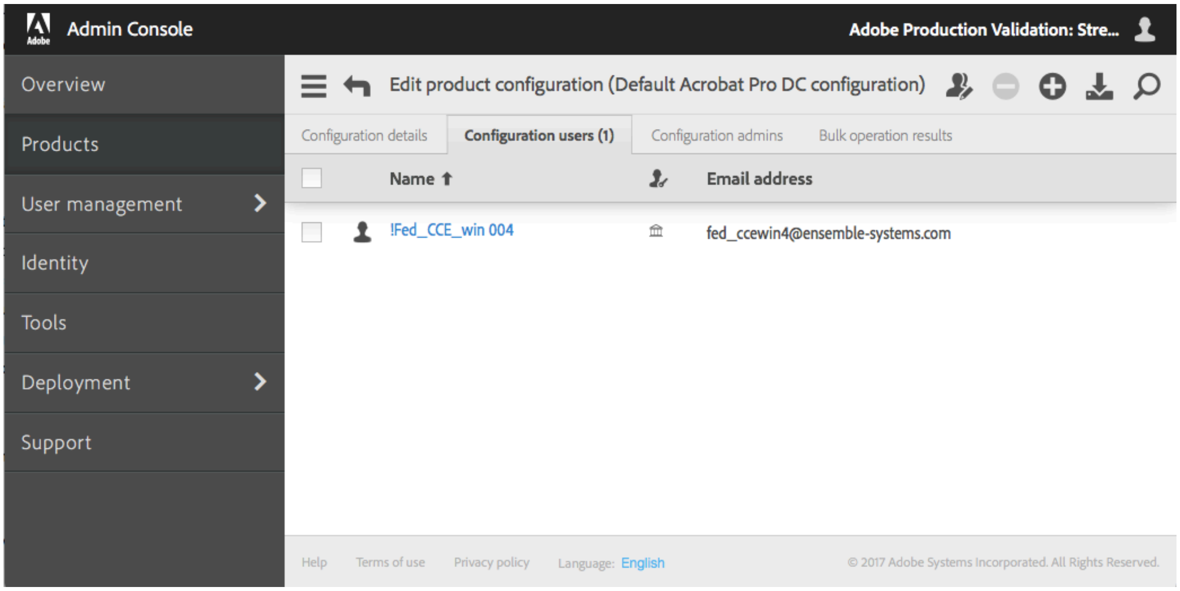
This action synchronizes all the users from the customer side with the Adobe side and also applies groups membership synchronization. It will also delete users that are on the Adobe side but no longer on the customer side if the user has a federated identity type. This is the most typical invocation.

```
      ./user-sync –c user-sync-config.yml --users all --process-groups --remove-
nonexistent-users
```

As the User Sync Tool synchronizes the Admin Console with the enterprise directory, the log output to the console updates as it runs.

```
2017-01-20 16:51:02 6840 INFO main - ========== Start Run ==========
2017-01-20 16:51:04 6840 INFO processor - ---------- Start Load from Directory ---------
--------------
2017-01-20 16:51:04 6840 INFO connector.ldap - Loading users...
2017-01-20 16:51:04 6840 INFO connector.ldap - Total users loaded: 4
2017-01-20 16:51:04 6840 INFO processor - ---------- End Load from Directory (Total
time: 0:00:00) ---
2017-01-20 16:51:04 6840 INFO processor - ---------- Start Sync Dashboard --------------
--------------
2017-01-20 16:51:05 6840 INFO processor - Adding user with user
key: fed_ccewin4@ensemble-systems.com
2017-01-20 16:51:05 6840 INFO dashboard.owning.action - Added action: {"do":
[{"createFederatedID": {"lastname": "004", "country": "CA", "email":
"fed_ccewin4@ensemble-systems.com", "firstname": "!Fed_CCE_win", "option":
"ignoreIfAlreadyExists"}}, {"add": {"product": ["default acrobat pro dc
configuration"]}}], "requestID": "action_5", "user": "fed_ccewin4@ensemble-systems.com"}
2017-01-20 16:51:05 6840 INFO processor - Syncing trustee org1...
/v2/usermanagement/action/210D419534321495E53@AdobeOrg HTTP/1.1" 200 77
2017-01-20 16:51:07 6840 INFO processor - ---------- End Sync Dashboard (Total time:
0:00:03) --------
2017-01-20 16:51:07 6840 INFO main - ========== End Run (Total time: 0:00:05) ==========
```

After this command is executed, reviewing the dashboard will show that a user has been added to the "Default Acrobat Pro DC configuration" as instructed by the group mapping in the user-sync-config.yml. The synchronization has been successful and the Admin Console is now updated.



Name	Email address
!Fed_CCE_win 004	fed_ccewin4@ensemble-systems.com

6.2 Executing sync looking at users in specified groups

This action synchronizes users in the specified groups from the customer side. It does not look at any other users in the customer's directory service. It does not perform any product configuration or user group management. That would still have to be done using the Adobe Admin Console.

```
./user-sync –c user-sync-config.yml --users groups “group1, group2, group3”
```

6.3 Executing sync for users only

This action synchronizes all the users from the customer side with the Adobe side. It only applies to the users. Groups membership is not processed. It ignores users that are on the Adobe side but no longer on the customer side. It does not perform any product configuration or user group management. That would still have to be done using the Adobe Admin Console.

```
./user-sync –c user-sync-config.yml --users all
```

6.4 Add users and generate a list of users to delete

This action synchronizes all users from the customer side with the Adobe side and also generates a list of users that no longer exist in the customer directory but still exist on the Adobe side.

```
      ./user-sync –c user-sync-config.yml --users all --generate-remove-list users-to-
remove.csv
```

6.5 Delete users from separate list

This operation will compare the users on the Adobe side against the users on the enterprise directory side to determine the list of users for removal. If a user exists on the Adobe side but not on the enterprise directory side, they will be flagged for deletion. Running the first command will generate the list of users that will be removed from the Adobe side. The second command will execute the deletion from the Adobe side.

```
1. ./user-sync –c user-sync-config.yml --generate-remove-list users-to-
   remove.csv
2. ./user-sync –c user-sync-config.yml --remove-list users-to-remove.csv
```

There is also the ability to manually create a csv with a list of users to be removed from the Adobe side. The steps are as follows,

```
1. manually create the file users-to-remove.csv
2. ./user-sync –c user-sync-config.yml --remove-list users-to-remove.csv
```

Note: If the users removed using a manually created csv still exist on the enterprise directory side, the next time a sync is executed to add users, this previously removed users will be added back to the Adobe side again.

6.6 Sync from file

```
./user-sync –c user-sync-config.yml --users file <file.csv>
```

6.7 Configure to add users to groups in different organizations

A domain can only be claimed by a single organization. So consider the following scenario:

A company, Geometrixx, has multiple departments, each of which has their own unique Enterprise Dashboard. Also, each department wants to use either Enterprise or Federated user IDs, all utilizing the geometrixx.com domain. In this case, the system administrator for each of these departments would want to claim this domain for identity use. The Enterprise Dashboard prevents multiple departments from claiming the same domain. However, once claimed by a single department, other departments can request access to it through the domain claim process.

The first department to claim the domain (owner) will be responsible for approving any requests for access by other departments (accessors).

Requirements:
Provide the dashboard configurations for the owning dashboard and the accessor dashboards. (In this configuration, there could be one or more accessor orgs)

Group mappings
When specifying the group mappings, if the dashboard group exists in a accessor organization, prefix the dashboard group name with "<org name>::" The <org name> to be used is the same org name that is used as the org name for the accessor dashboard configuration file.

For example:
In the user-sync-config.yml, if the accessor yml configurations are specified,

```
trustees:
  org1: dashboard-org1-config.yml
  org2: dashboard-org2-config.yml
```

An example of a group mapping for a group in an accessor org would be as follows,

```
  - directory_group: CCE Friends Group
    dashboard_groups:
      - "org1::Default Adobe Enterprise Support Program configuration”
```

If the user sync tool is configured to look for accessor yml files using a filename format as specified in the user-sync-config.yml,

```
trustee_config_filename_format: "dashboard-{organization_name}-config.yml”
```

And the config yml files for the accessing orgs included are,

```
  dashboard-org1-config.yml
  dashboard-org2-config.yml
```

The group mapping in an accessing org would be as follows,

```
  - directory_group: CCE Friends Group
    dashboard_groups:
      - "org1::Default Adobe Enterprise Support Program configuration”
      - "org2::Default Acrobat Pro DC configuration"
```

7 Deployment Best Practices

The User Sync Tool is designed to run with limited/no human interaction once properly configured. It can be setup to run on an environment using a scheduler and the frequency of the execution of the solution can be configured as desired.

Given the nature of the data in the configuration and log files, a server should be dedicated for this task and locked down with industry best practices. It is recommended that a server that sits behind the enterprise firewall be provisioned for this application. A system service account should be created with restricted privileges that is specifically intended for running the application and writing log files to the system. In addition to that, it should be locked down such that only privileged users will be able to connect to this machine.

Actions on the User Management API are executed using GET and POST requests against a HTTPS endpoint. JSON data that represent the changes that need to be written to the Admin console is constructed by the User Sync Tool. This data is then attached in the body of a POST request to the User Management API.

Please note that the first few executions of the User Sync Tool may take a long time depending on how many users need to be added into the Adobe Admin console. The initial run of the application should be executed manually before setting it up to run as a scheduled task to avoid having multiple instances running. Subsequent executions should be quicker, as it will only be performing updates to users as necessary. The frequency that this application gets executed depends on how often an enterprise directory changes.

To protect the availability of the Adobe back-end user identity systems, the User Management API imposes limits on client access to the data. Limits apply to the number of calls that an individual client can make within a time interval, and global limits apply to access by all clients within the time period. The User Sync Tool implements back off and retry logic to prevent the script from continuously calling the User Management API when it hits the rate limit. It is normal to see messages in the console indicating that the script has paused for a short amount of time before trying to execute again.

7.1 Scheduled Task Examples

Cron

Run the SyncTool at the top (minute 0) of each hour.

```
0 * * * * [python bin directory]/python [path to UserSyncTool executable]/user-sync --
config=[path to config]/config.yml > [path to logs]/cron-logs/$(date +\%Y-\%m-\%d).log 2>&1
```

Windows Task Scheduler

Run the SyncTool every hour starting at 4PM

```
C:\> schtasks /create /tn "Adobe_User_Sync" /tr "[path to python]\python.exe [path to
UserSyncTool executable]\user-sync.PEX --config=[path to config]\config.yml" /sc HOURLY /st
16:00
```

8 Security Considerations

8.1 Configuration Values

In order for the User Sync Tool to properly establish a connection with the Enterprise Directory, the User Sync Tool should be configured to read from the directory server using a service account. In addition to those values, the API key and path to private certificates also must exist in the configuration file. The service account used to access the enterprise Active Directory (or other LDAP directory system) should only be able to read from the system. Please take necessary steps to protect this configuration file to ensure that only authorized users will be able to access this file.

Please note that it is critical to keep a backup copy of private cert in a secure location since if it is lost, the account will be blocked until a new certificate is created and installed.

8.2 Logging

Logging outputs all transactions against the User Management API to the console. This is enabled by default and can be set to write to a log file as well. The User Management API treats a user's email address as the unique identifier. When an action is executed against the User Management API, that action along with the email address associate with the user is written to the log. The files created during execution are date stamped and written to the file system. For admins opting to log data to file, these files will live on the server forever as the utility does not provide any log rotation or management. Please take necessary precautions to manage the lifetime and access to these files.

If your company's security policy does not allow any personally identifiable information to be persisted on disk, disabling the log to file will stop the log from being saved on disk but it will still output the log transactions to console where it's stored temporarily in memory during the execution.

9 Test Cases

The following test cases can be followed to ensure that the User Sync Tool is configured correctly, and that the product groups are correctly mapped to Directory security groups.

9.1 User Creation

Step 1

Create one or more test users in Enterprise Directory

Step 2

Add user(s) to one or more configured security groups

Step 3

Run the User Sync Tool

Step 4

Ensure that test users were created in Adobe Admin Console

9.2 User Update

Step 1

Modify product group membership of one or more test users

Step 2

Run the User Sync Tool

Step 3

Ensure that users in Adobe Admin Console were updated to reflect new product group membership

9.3 User Disable

Step 1

Remove or Disable one or more existing test users in Enterprise Directory

Step 2

Run the User Sync Tool

Step 3

Ensure that users deleted or disabled in Enterprise Directory were removed from configured product groups in the Adobe Admin Console.

10 Support

For additional support with this utility, please open an issue in GitHub at https://github.com/adobe-apiplatform/user-sync.py/issues.

Please include any log files that are generated during the application execution in your support request. This will help with the debugging process.

Adobe Customer Support is currently unable to provide support for the User Sync Tool.