Sync LDAP / AD User – Installation Guide

1 Checklist

Before following this installation Guide, you should check if you received all files that are needed to install the tool. You should have received a zip-File that contains the following files:

- Configuration.ps1
- DirectoryFunctionsLDAP.psm1
- do-ldap-sync.bat
- Logging.psm1
- Test.ps1
- UserSync.ps1

Furthermore, you should make sure that **.Net-Framework 4.5** (or higher) is installed on the corresponding aqua Server.

2 Installation

2.1 Configuration

1. Unzip the zip file on your aqua Server, e.g. to the following folder (you can also use a different folder):

   `C:\Programs\SyncLDAPUser`

2. Right-Click **Configuration.ps1** and select **edit**
3. Adopt the following settings to your needs (samples can be found in the configuration file):

aqua	
$ErrorActionPreference	Defines if you want to abort on any error
$webserviceUrl	http://**<yourAquaServerUrl>**/aquaAPI/MainService.asmx
$apiUser	Login to aqua api - Username (must be a user that has permission to create users)
$apiPassword	Login to aqua api - Password
$licenseProfileCode	Internal technical name of license to be assigned (ask andagon support if you are not sure)
$floatingLicense	true/false (related to $licenseProfileCode)
$defaultProjectName	Default project to which the user is assigned to
$defaultRoleName	Default role to which the user is assigned to
LDAP	
$directoryServer	A string specifying the server which can be a domain name, LDAP/AD server name or dotted strings representing the IP address of the LDAP/AD server. Optionally, this parameter may also include a port number, separated from the ID itself by a colon (:). If $null, it represents the identity of any domain controller in the domain associated with the computer account.
$directoryServerUseTLS	Set to $true if you want to use an encrypted connection
$lookupUsername	Username for connecting to the directory server, leave empty if no authentication is required
$lookupUserPassword	Password for connecting to the directory server, leave empty if no authentication is required
$baseDN	defines a distinguished-name of branch where to look for aqua users (recursively)

$aquaUsersQuery	an LDAP/AD query intended to match all users (in scope of $baseDN) that should be synchronized with aqua
$aquaLoginAttribute	Defines mapping of LDAP/AD attribute to aqua username
$aquaEmailAttribute	Defines mapping of LDAP/AD attribute to aqua user email
$aquaFirstNameAttribute	Defines mapping of LDAP/AD attribute to aqua user's firstname
$aquaLastNameAttribute	Defines mapping of LDAP/AD attribute to aqua user's lastname
Configuration	
$deactivateKnownMissingUsers	When $true, users which were found in the LDAP/AD before but are no longer included in the LDAP/AD result will be deactivated in aqua.
$deactivateAllMissingUsers	When $true, all users which are not found in the LDAP/AD and are not on the whitelist $userWhiteList will be deactivated
$userWhiteList	List of aqua usernames which should never be deactivated by this synchronization script

4. Save and close **Configuration.ps1**
5. Optional: For testing, you can now execute **Test.ps1** via PowerShell to test connectivity to your LDAP/AD Server. ('cd' into the folder first!) Test is read-only and does not apply any changes. The flag -DumpEntries can be used to dump the raw result received from the LDAP/AD server.

2.2 Disable user in aqua

If you want to deal with disabled users in your LDAP /AD so that they are disabled in aqua as well, then you need to implement the method 'IsUserDisabled' in file 'DirectoryFunctionsLDAP.psm1'. You can test your implementation by executing 'Test.ps1'. The test output can be found in logs\Test.log. You should see a list of users and Disabled-flag accordingly.

Furthermore, it is possible to deactivate aqua users when they are not present in the LDAP/AD results anymore. You can choose one of two options:

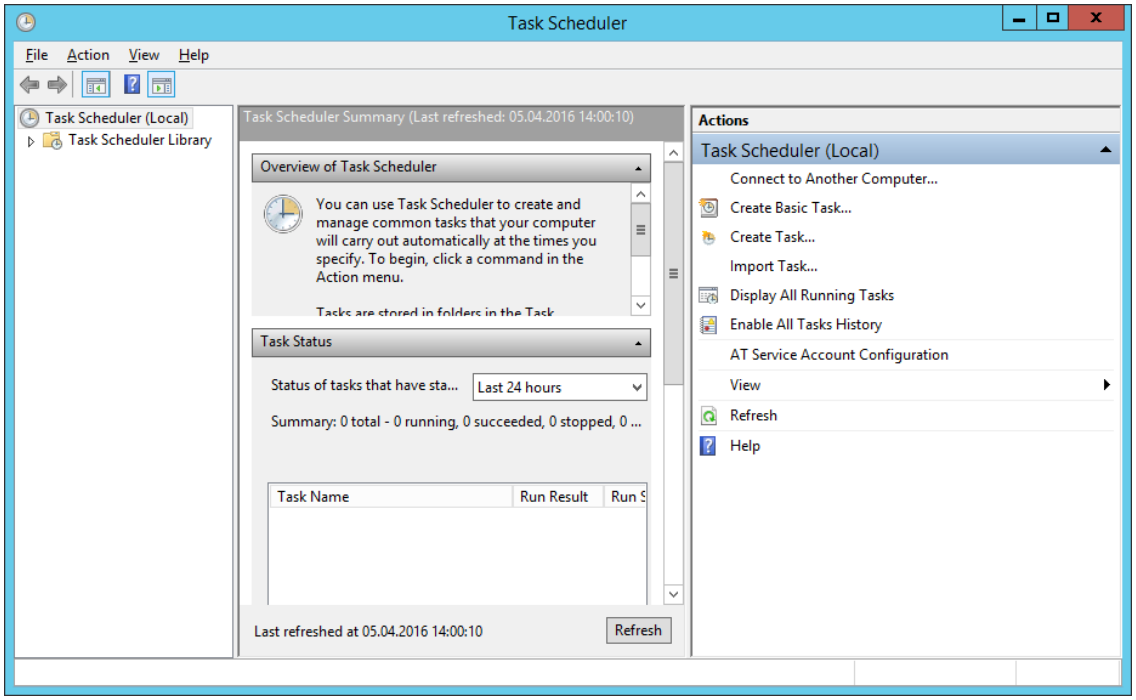
- Deactivate known missing users ($deactivateKnownMissingUsers): aqua users which were found by this sync before but are now missing in the results will be disabled. aqua users on the whitelist will not be deactivated
- Deactivate all missing users ($deactivateAllMissingUsers): **all** aqua users which are not present in the LDAP/AD results will be disabled

Before enabling $deactivateAllMissingUsers carefully review your LDAP/AD query and the user whitelist. You can potentially lock yourself out. We strongly recommend adding at least a couple of global administrators to the whitelist.

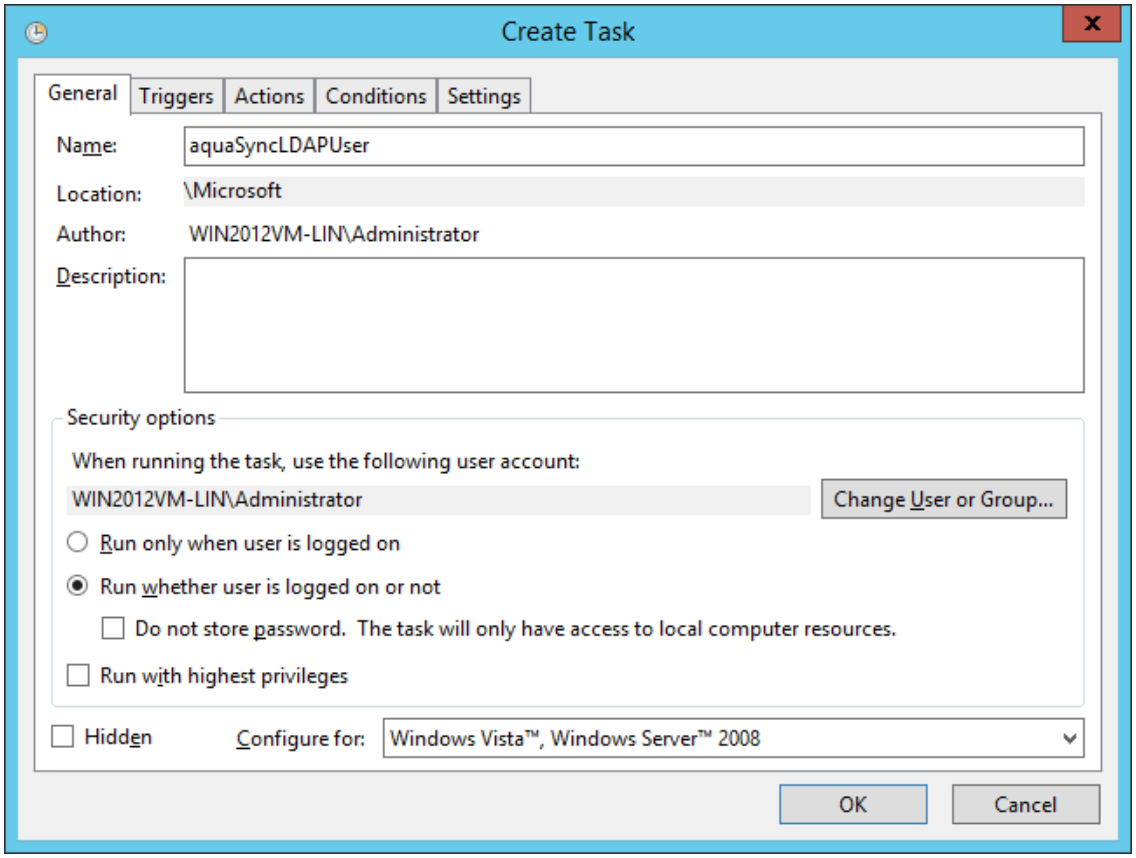
Users which are included in the whitelist $userWhiteList will never be deactivated.

2.3 Setting up nightly schedule

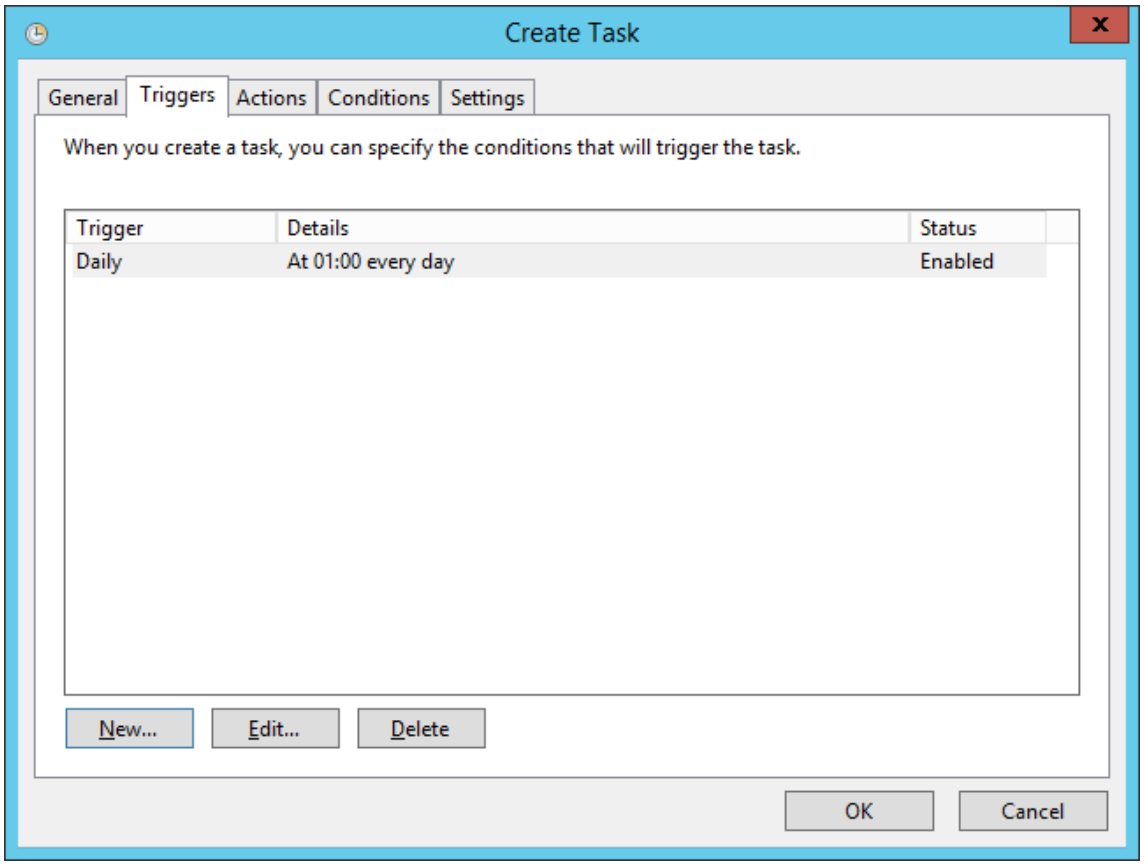
1. Open Windows Task Scheduler (e.g. press Windows-Key on Keyboard and start typing “scheduler”)



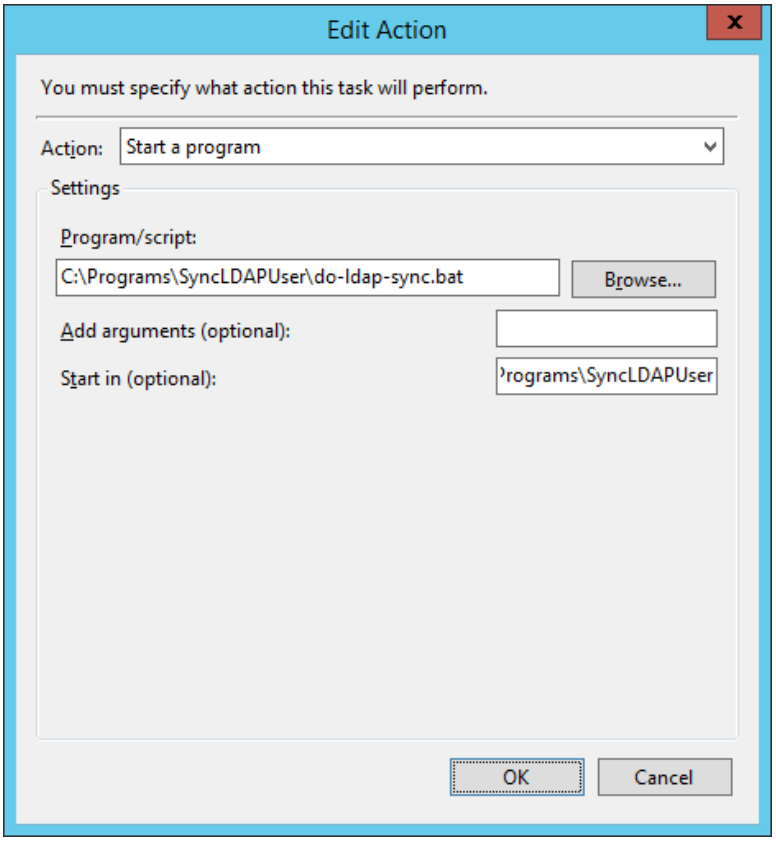
2. “Create Task” from the menu on the right
3. On tab “General” please apply the following settings



4. On tab “Triggers” please create the following entry:

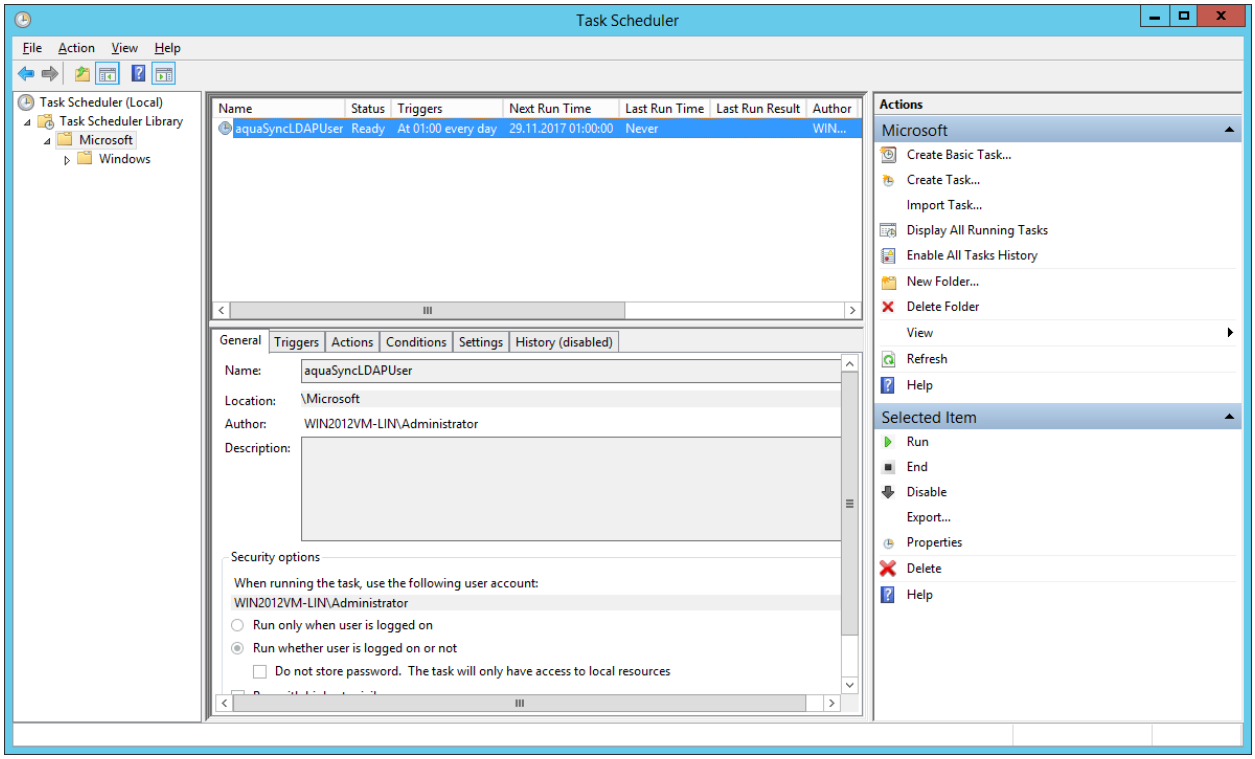


5. On tab “Actions” please create the following entry (do **not** forget “Start In”):

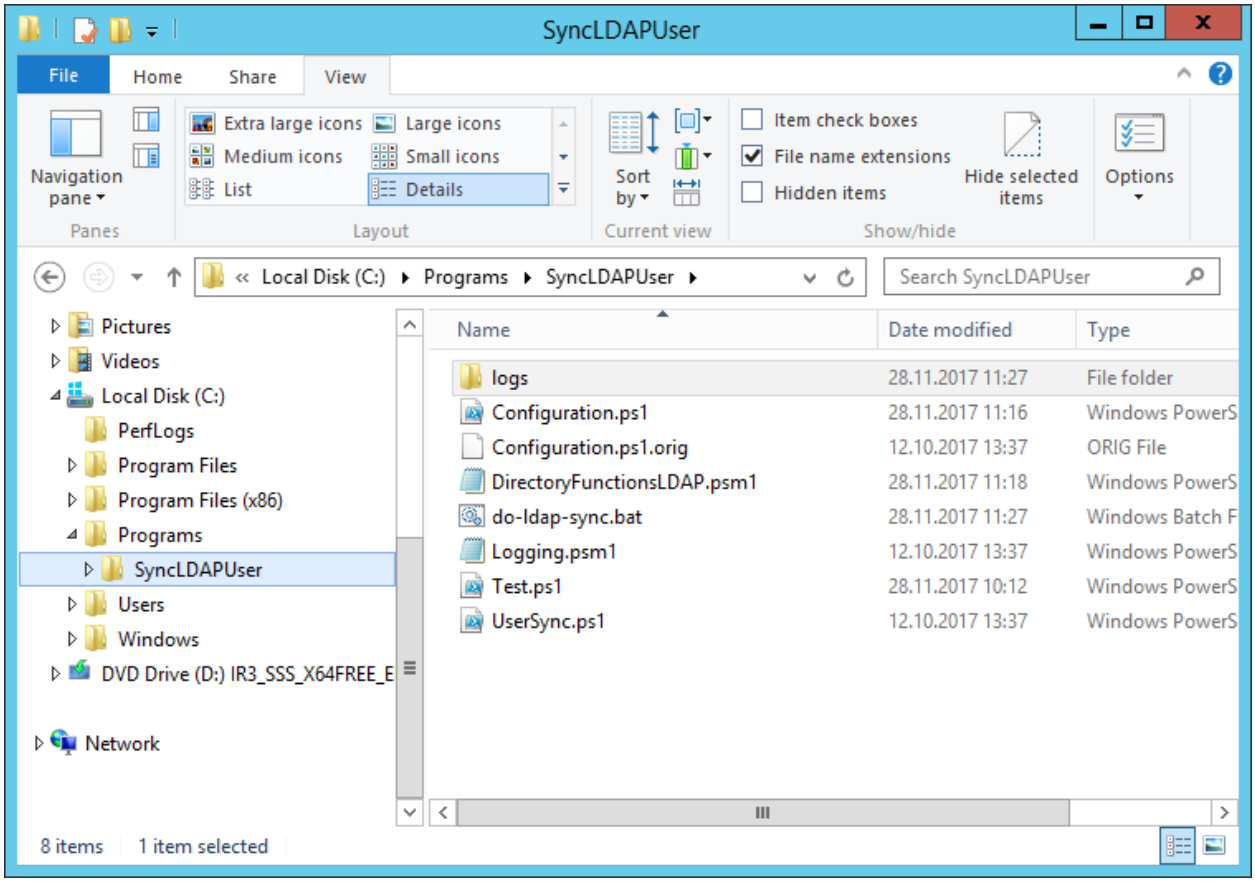


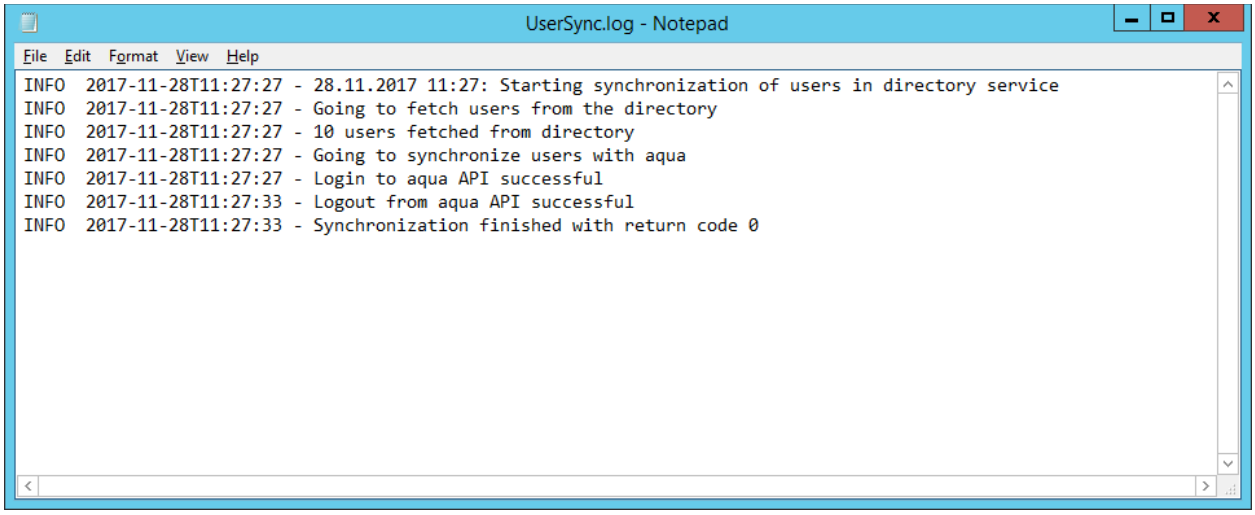
6. Finally press “ok” and provide user credentials so that your task can be executed.

7. That's it. Your task should have been created and should run automatically during the next night. You can find your created Task in Task Scheduler Library. Run it manually by doing a Right-click.



8. If sync was successful, you should see a logs-folder containing a file "UserSync.log".




UserSync.log - Notepad
File Edit Format View Help
INFO 2017-11-28T11:27:27 - 28.11.2017 11:27: Starting synchronization of users in directory service
INFO 2017-11-28T11:27:27 - Going to fetch users from the directory
INFO 2017-11-28T11:27:27 - 10 users fetched from directory
INFO 2017-11-28T11:27:27 - Going to synchronize users with aqua
INFO 2017-11-28T11:27:27 - Login to aqua API successful
INFO 2017-11-28T11:27:33 - Logout from aqua API successful
INFO 2017-11-28T11:27:33 - Synchronization finished with return code 0