Table Of Content

[1. Requirements](#_heading=h.gjdgxs) 1

[2. Script](#_heading=h.30j0zll) 2

[3. Permission Mapping](#_heading=h.1fob9te) 3

Table Of Figures

Figure 1. Status Requirements 1

Figure 2. Tenant ID 2

Figure 3. Tenant ID 3

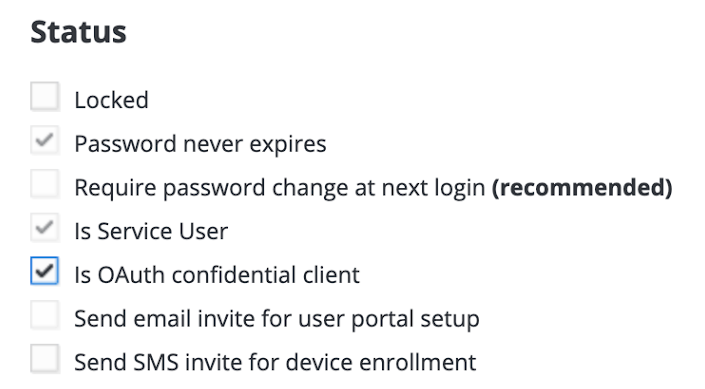
This document provides the detailed description on the use of the powershell scripts to migrate from Delinea to CyberArk PCloud. It covers the creation of the Safes as well as mapping the folder and secret permissions from delinea to the CyberArk Safes with appropriate member rights.

# Requirements

* 1. Service User

Create a service user in the identity portal

* + 1. Navigate to Core Services > Users , then click Add User
    2. Fill in these three categories: Login name, Display name, and Password
    3. Check these boxes (“Password never expires” is optional but update the expiration time if u don’t check this box)



*Figure 1: Status requirements*

* + 1. Give suitable permissions to the user (best possible: Privilege Cloud Administrator)
    2. Refer to this link if you have any questions for this part:

<https://docs.cyberark.com/PrivCloud-SS/Latest/en/Content/ISPSS/ISPSS-API-Authentication.htm#Overview>

* + 1. Use this user to perform automation tasks with the following scripts
  1. Scripts

Download the following two scripts from the Drive

* Safe\_Member\_Permissions.ps1
* Safe\_Creation.ps1

<https://drive.google.com/drive/folders/1mrsnKgGXheQiCvpeU3_2zKRGUTLhuWsI?usp=sharing>

* 1. CSV file

Download the csv template from the drive

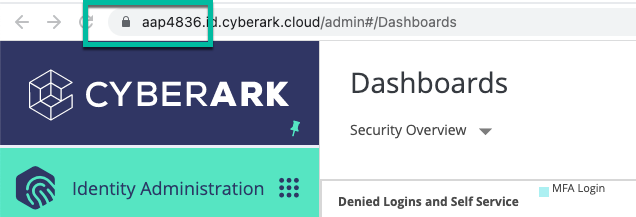
* CSV\_Templet.csv

<https://drive.google.com/drive/folders/1mrsnKgGXheQiCvpeU3_2zKRGUTLhuWsI?usp=sharing>

* 1. Powershell ISE

# Script

* 1. Upload Safes
     1. Open Safe\_Creation.ps1 in Powershell ISE.
     2. Update the log file location on line 2 (local to the machine and not on one drive)
     3. Update only the identity-tenant-id (ex. aat4921) on line 4 which you will get from the below image.



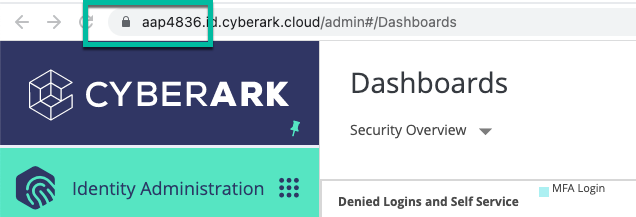
*Figure 2: Tenant ID*

* + 1. Update the PVWA subdomain on line 6 (only the clango-ispss part)
    2. Update CSV file location on line 8

Note: - Make sure that the column name for safe name in the CSV matches.

Note: - Update $\_."CyberArk Safe" with the column name of the file which contains safe names if any changes are made.

* + 1. Run the script
  1. Safe Member Permission
     1. Open Safe\_Member\_Permission.ps1 in Powershell ISE.
     2. Update the log file location on line 2 (local to the machine and not on one drive)
     3. Update only the identity-tenant-id (ex. aat4921) on line 4 which you will get from the below image.



*Figure 3: Tenant ID*

* + 1. Update the PVWA subdomain on line 6 (only the clango-ispss part)
    2. Update CSV file location on line 8

Note: - Make sure that the column name for safe name in the CSV matches.

Note: - Update $\_."CyberArk Safe" with the column name of the file which contains safe names and other properties respectively if any changes are made.

* + 1. Run the script

# Permission Mapping

* 1. Folder Permission Mapping

| View/Add Secret/Edit/Owner | addAccounts = $true  ManageSafe = $true  ViewSafemembers = $true    viewauditlog = $true    backupsafe = $true\  managesafemembers = $true |
| --- | --- |
| View/Add Secret/Edit | addAccounts = $true  ManageSafe = $true  ViewSafemembers = $true    viewauditlog = $true    backupsafe = $true\ |
| View/Add Secret | addAccounts = $true |

* 1. Secret Permission Mapping

| List/View/Edit/Owner | useAccounts = $true  retrieveAccounts = $true    listAccounts = $true    addAccounts = $true  updateAccountContent = $true  updateAccountProperties = $true  initiateCPMAccountManagementOperations = $true  specifyNextAccountContent = $true  renameAccounts = $true    deleteAccounts = $true    unlockAccounts = $true |
| --- | --- |
| List/View/Edit | useAccounts = $true  retrieveAccounts = $true    listAccounts = $true  updateAccountContent = $true  updateAccountProperties = $true    unlockAccounts = $true  renameAccounts = $true  initiateCPMAccountManagementOperations = $true  specifyNextAccountContent = $true |
| List/View | useAccounts = $true  retrieveAccounts = $true    listAccounts = $true |
| List | listAccounts = $true |