Udcx Remediation

# Introduction

This is a client-side Console application that reads input .csv file and it filters non-empty data of “Authentication” column. It iterates filtered data and reads each of the udcx file, and then comments the <udc:Authentication> node if present. A status report .csv file is generated at the end of the application, which will show whether the nodes are being commented for each file or not.

# Authentication

The console will prompt the user for an administration account.

Be sure to specify an account that has Admin permissions on the target SharePoint environment. This account will be used to generate Authenticated User Context instances that will be leveraged to access the target environments.

* If you wish to target an **SPO-D (or On-Prem)** farm:
  + use the **<domain>\<alias>** format for the administrator account.
* If you wish to target an **SPO-MT (or vNext)** farm:
  + use the **<alias>@<domain>.com** format for the administrator account.

Example is shown in the below screen shot:

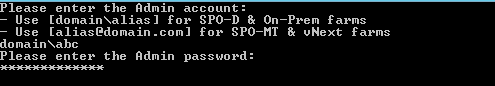


Figure Prompts for username and password for authentication

# Other Inputs to be provided on Command prompt.

The console will also prompt the user for the input file path.

Be sure to specify the input file path in the following format:

* If you wish to target an **SPO-D (or On-Prem)** farm:
  + use the **<folder path containing the file>\<csv file name>** format for the input file path
* Example:
  + One such example is as shown in the figure:

Path should contain the folder name as well as the file name:

Example location: E:\UdcxRemediation\UDCXReport.csv

Where E:\UdcxRemediation🡪represents the folder containing the file.

And, UDCXReport.csv🡪represents the file name.

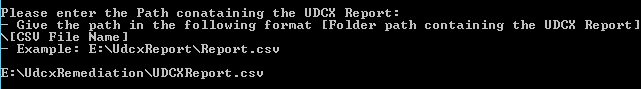


Figure Prompt for the input file path and Example showing the same

**Note**: If the path entered is incorrect or the file doesn’t exist, or nothing is entered, then respective error message is shown and it will again prompt for the file path. This can be shown in the below figure:

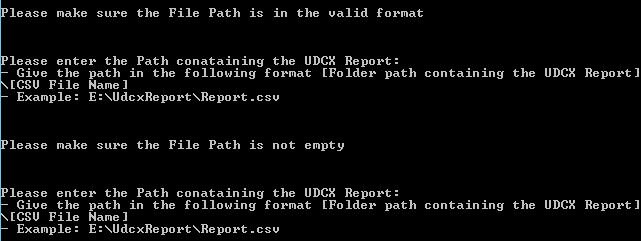


Figure Error message on invalid file path and consequent prompt after the same

## Operation – CommentUDCXFileNodes

This operation reads data from an input file and iterates through each row and connects to corresponding site collection. It reads the udcx file and comments out the <udc:Authentication> node, if it is present or not commented already. (Based on the column ‘Authentication’ in the input csv file)

Also it generates status report indicating whether the commenting of the nodes was successful or not.

### Input

* **UDCXReport.csv**
  + The file defines the udcx files of interest and is generated by the script: **Get-UdcxReport.ps1**
  + This is a CSV that follows in which the header row is atleast expected with the following format and columns:
    - SiteId, WebId, DirName, LeafName, Id, SiteUrl, WebUrl, Authentication
  + SiteId
    - This column should contain the GUID of the site
  + WebId
    - This column should contain the GUID of the Web
  + DirName
    - This column should contain the serverRelative path of the data connection library where the .udcx file is present.
  + LeafName
    - This column should contain the name of the .udcx file
  + Id
    - This column should contain the GUID of the .udcx file
  + SiteUrl
    - This column should contain the Url of the site
  + WebUrl
    - This column should contain the Url of the Web.
  + Authentication
    - This column should contain the data present in the <udc:Authentication> node in the .udcx file

### Output

* **UDCXStatusReport\_27\_12\_2015\_10\_53\_17\_dd\_mm\_yyyy\_hh\_mm\_ss.csv**

A status report .csv file is generated at the end of the application, which will show whether the nodes are being commented for each file or not.

**Note:** This file will be generated in the same folder where the input file is present.

This CSV that follows the header format and columns as mentioned below:

* + - Status, ErrorDetails, DirName, LeafName, Id, SiteUrl, WebUrl, Authentication
  + Status
    - This column contains the Status of commenting the node which will be either ‘Success’ or ‘Error’ depending on the status of the .udcx file
  + ErrorDetails
    - This column contains Error Details or Message if the commenting of the desired node is unsuccessful. On the success of the same, the corresponding data would be blank.
  + DirName
    - This column contains the serverRelative path of the data connection library where the .udcx file is present.
  + LeafName
    - This column contains the name of the .Udcx file
  + Authentication
    - This column contains the data present in the <udc:Authentication> node in the .udcx file
  + Id
    - This column contains the GUID of the .udcx file
  + SiteUrl
    - This column contains the Url of the site
  + WebUrl
    - This column contains the Url of the Web.

# Samples

## Input:

## UDCXReport.csv



## Udcx files before console run

|  |  |  |
| --- | --- | --- |
| File Name | Test Files | Comments |
| Country.udcx |  | Authentication node is available |
| groups.udcx |  | Authentication node was commented |
| List1.udcx |  | Authentication node available |
| Read.udcx |  | Invalid udcx file for testing |
| RESTWebService.udcx |  | Authentication node was commented |

## Output:

## UDCXStatusReport\_28\_12\_2015\_10\_24\_59.csv

## Udcx files after console run

|  |  |  |
| --- | --- | --- |
| File Name | Test Files | Comments |
| Country.udcx |  | Authentication node is commented and updated file in data connection library successfully. |
| List1.udcx |  | Authentication node is commented and updated file in data connection library successfully. |