

# Site Group Association Script

## EPA – Site Group Association Script

*Prepared for EPA*

14-Jan-15

Version .1 Draft

*Prepared by*

**Sofiane ben Sr.**

Consultant

sofiane.b@microsoft.com

Contributors

**David Weinstein**

Principal Consultant, dawein@microsoft.com

# Revision and Signoff Sheet

## Change Record

Date	Author	Version	Change Reference
01/14/2015	Sofiane Ben	.1	Initial draft for internal review/discussion

## Reviewers

Name	Version Approved	Position	Date

# Table of Contents

## Contents

EPA – Site Group Association Script.....	1
Revision and Signoff Sheet.....	2
Table of Contents.....	3
1 Introduction .....	4
2 Requirements .....	5
3 Deployment & Configuration.....	6
3.1 Deploying the Script.....	6
3.2 Set up the Configuration File .....	6
4 Usage.....	9
4.1 Executing the Script .....	9
1. No Option.....	9
2. Repair .....	9
4.2 Configuring Logging .....	10

# 1 Introduction

This document is aimed at describing the solution implemented to identify SharePoint sites where the group association is not properly configured.

The function of the EPA's O365-SPO Site Group Association script is to identify sites with problematic group association settings and optionally properly set the configuration. SharePoint sites contain a configuration setting indicating which groups are associated with the three built-in levels (owners, members and visitors). A missing and/or problematic configuration leads to a number of issues from which, the inability of site owners to perform some key administrative tasks.

The EPA's O365-SPO Site Groups Association script is a console application written in .Net 4.5. The application leverages the APIs provided by Office 365 and SharePoint Online. The script communicates with EPA's O365 tenant through SharePoint's Client Side Object Model API (CSOM) and via secure https connection.

The script enumerates through the sites using a site listing provided by a configuration list and process each site as well as its sub-sites. For each site, the script checks if the group association properties are properly set up. The script also provides an option to correct the configuration settings. The script uses the site's title and/or name to determine which groups are to be set.

## 2 Requirements

The table below list the script's system requirement.

Requirement	Category	Comment
Windows Server OS	Platform	(any current version)
.Net 4.5	Platform	
Task scheduling solution	Platform	Preferably Windows Tasks Scheduler
Internet connectivity with port 443 enabled	Network	Communication with EPA 's O365 is done via https
Microsoft.SharePoint.Client.dll	API	Included in package
Microsoft.SharePoint.Client.Runtime.dll	API	Included in package
Microsoft.Online.SharePoint.Client.Tenant	API	Included in package
Nlog.exe	API	Included in package

**Note:**

- The script does not require additional technology such as an IIS/ASP.net sever or a SharePoint server.
- The script does not require a Server OS and can be executed on a Client OS as well.

## 3 Deployment & Configuration

### 3.1 Deploying the Script

The script packages contain the release folder where the executable binary can be found. In addition, the folder contains the script's supporting files.

Name	Date modified	Type	Size
log	1/9/2015 1:19 PM	File folder	
EPA-SiteGroupAssociation	1/21/2015 12:05 PM	Application	16 KB
EPA-SiteGroupAssociation.exe	1/17/2015 12:45 PM	XML Configuration...	1 KB
EPA-SiteGroupAssociation	1/21/2015 12:05 PM	Program Debug D...	24 KB
Microsoft.Online.SharePoint.Client.Tenant.dll	4/25/2014 1:26 PM	Application extens...	54 KB
Microsoft.SharePoint.Client.dll	4/25/2014 11:25 A...	Application extens...	460 KB
Microsoft.SharePoint.Client.Runtime.dll	1/23/2014 6:53 AM	Application extens...	283 KB
NLog	7/6/2014 12:16 AM	XML Configuration...	1 KB
NLog.dll	7/6/2014 12:24 AM	Application extens...	430 KB
NLog	7/6/2014 12:16 AM	XML File	750 KB

### 3.2 Set up the Configuration File

The configuration file is used to store the account credentials used to access EPA's O365 tenant as well as other configuration parameters used by the script.

In order for the script to successfully connect to EPA's O365 tenant, the connection script needs to be updated with the appropriate userid and password.

Locate and open the configuration file named EPA-SiteGroupsAssociation.exe.config (See screenshot below).

Name	Date modified	Type	Si
log	1/9/2015 1:19 PM	File folder	
EPA-SiteGroupAssociation	1/21/2015 12:05 PM	Application	
EPA-SiteGroupAssociation.exe	1/17/2015 12:45 PM	XML Configuration...	
EPA-SiteGroupAssociation	1/21/2015 12:05 PM	Program Debug D...	
Microsoft.Online.SharePoint.Client.Tenant.dll	4/25/2014 1:26 PM	Application extens...	
Microsoft.SharePoint.Client.dll	4/25/2014 11:25 A...	Application extens...	
Microsoft.SharePoint.Client.Runtime.dll	1/23/2014 6:53 AM	Application extens...	
NLog	7/6/2014 12:16 AM	XML Configuration...	
NLog.dll	7/6/2014 12:24 AM	Application extens...	
NLog	7/6/2014 12:16 AM	XML File	

Using your text editor of choice, update the user string with the appropriate userid and password.

```

1  <?xml version="1.0" encoding="utf-8" ?>
2  <configuration>
3    <startup>
4      <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
5    </startup>
6    <appSettings>
7      <add key="TenantAdminUrl" value="https://usepa-admin.sharepoint.com"/>
8      <add key="ReportSiteUrl" value="https://usepa.sharepoint.com/sites/oei_Community/epascene/globaladmin"/>
9      <add key="SiteListName" value="Site Listing"/>
10   </appSettings>
11   <connectionStrings>
12     <add name="epaConnect" connectionString="User ID=SharePointAdmin@usepa.onmicrosoft.com;Password=pass" />
13   </connectionStrings>
14 </configuration>

```

**Note:** For production environments, we recommend that the configuration file be encrypted using the ASP.net registration tool (aspnet\_regiis.exe) which ships with .Net 2.0.

In order to encrypt the configuration file:

- Rename EPA-SiteGroupsAssociation.exe.config to web.config
- Open web.config with your XML editor of choice
- Replace the encrypted element with a new connection string element

- Execute aspnet\_regiis.exe

Execute the following command prompt: <Path-To-Windows>\Microsoft.NET\Framework\v2.0.50727\aspnet\_regiis.exe -pef connectionStrings "<PathToFileDirectory>" -prov DataProtectionConfigurationProvider

This will create a cypher of the connection string which should look similar to the below.

```

1 <?xml version="1.0" encoding="utf-8" ?>
2 <configuration>
3   <startup>
4     <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
5   </startup>
6   <appSettings>
7     <add key="TenantAdminUrl" value="https://usepa-admin.sharepoint.com"/>
8     <add key="ReportSiteUrl" value="https://usepa.sharepoint.com/sites/oei_Community/epascene/globaladmin"/>
9     <add key="ReportListName" value="External Users"/>
10    <add key="DaysBeforeDeletion" value="3"/>
11    <add key="DayInterval" value="365"/>
12  </appSettings>
13  <connectionStrings configProtectionProvider="DataProtectionConfigurationProvider">
14    <EncryptedData>
15      <CipherData>
16        <CipherValue>AQAAANCmnd8BFdERjHoAwE/C1+sBAAAFG2mvQINv0Wof8YSdt1qLQAAAAACAAAAAADZgAAwAAAAABAAAAACXNqu/z
17      </CipherData>
18    </EncryptedData>
19  </connectionStrings>
20 </configuration>

```

- Rename web.config to EPA-SiteGroupsAssociation.exe.config



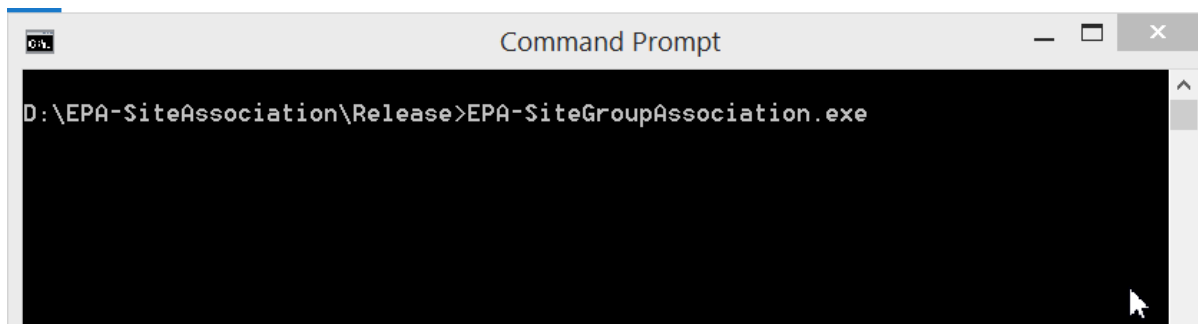
## 4 Usage

### 4.1 Executing the Script

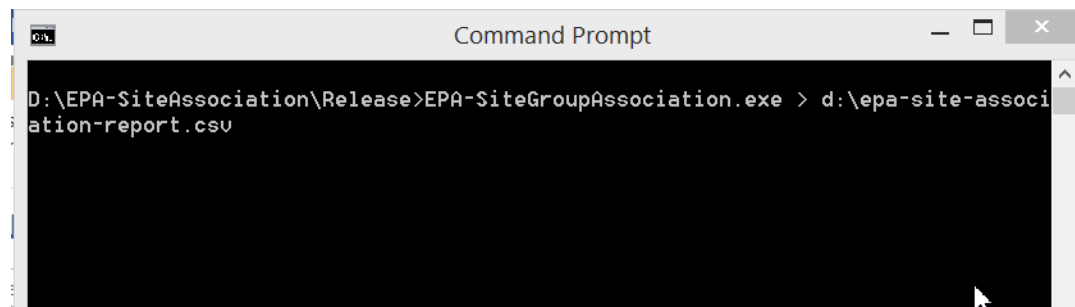
The script should be executed through the command prompt. The script can be executed in 2 modes uses command line argument:

#### 1. No Option

Running the script without option will generate a report of the sites' group association setting. The report is used to locate the problematic sites.



In order to capture the report, pipe the script into your text file of choice.



#### 2. Repair

The "repair" option instructs the script to attempt to set the groups association of problematic sites. Probably configured site will be left untouched.

```
Command Prompt
D:\EPA-SiteAssociation\Release>EPA-SiteGroupAssociation.exe repair
```

## 4.2 Configuring Logging

The script uses nlog as its underlining logging framework. This enables the tool to have highly configurable logging capabilities including:

- Minimum level
- Logs directory
- Logs files maintained criteria

The nlog dll and configuration files are found within the script folder and referenced in the tool.

Name	Date modified	Type	Size
bin	1/21/2015 3:48 PM	File folder	
obj	1/21/2015 3:48 PM	File folder	
Packages	10/27/2014 3:05 PM	File folder	
Properties	10/27/2014 3:08 PM	File folder	
App	1/17/2015 12:45 PM	XML Configuration...	1 KB
EPA-SiteGroupAssociation	1/17/2015 2:47 PM	Visual C# Project fi...	6 KB
EPA-SiteGroupAssociation	1/21/2015 3:48 PM	Microsoft Visual St...	1 KB
NLog	7/6/2014 12:16 AM	XML Configuration...	1 KB
NLog	7/6/2014 12:16 AM	XML File	742 KB
Program.cs	1/21/2015 4:00 PM	Visual C# Source fi...	17 KB

```

1  <?xml version="1.0" encoding="utf-8" ?>
2  <nlog xmlns="http://www.nlog-project.org/schemas/NLog.xsd"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
4
5      <!--
6      See http://nlog-project.org/wiki/Configuration\_file
7      for information on customizing logging rules and outputs.
8      -->
9      <targets>
10         <target xsi:type="File" name="f" fileName="${basedir}/log/n_log.log"
11             layout="${longdate} ${uppercase:${level}} ${message}"
12             archiveFileName="${basedir}/archive/log.{#}.txt"
13             archiveEvery="Month"
14             archiveNumbering="Rolling"
15             maxArchiveFiles="6" />
16     </targets>
17
18     <rules>
19         <!-- add your logging rules here -->
20         <logger name="*" minlevel="Info" writeTo="f" />
21     </rules>
22 </nlog>

```

Detailed guide to nlog can be found @ <https://github.com/nlog/nlog/wiki>