

# Analytics Script Solution Guide

## EPA – Analytics 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 – Analytics 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. Site Collection .....	9
2. Sub-Sites .....	9
3. Communities .....	10
4.2 Configuring Logging .....	10

# 1 Introduction

This document is aimed at describing the solution implemented to capture site analytic data on EPA's O365 tenant.

The function of the EPA's O365 analytic script is to capture environmental data used to monitor user adoption on EPA's SPO implementation. The Analytics script work by enumerating the sites and capturing key metrics including:

- Site creation date
- Storage used
- User count
- Sub-site count
- Site collection version
- Community count(s)
- Last modified Date

The data is used to build spreadsheet report(s) that provide management with high level view of users' adoption as well as the ability to slice the data and generate custom reports.

As of the current iteration, the script outputs the data into a text file using a comma delimited format. The log file(s) is imported into the pre-built spreadsheet template. The spreadsheet template contains the BI reports as well as tables to host the captured data (included in attachment).

The EPA's O365 Analytics 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 collects the data needed for the report and outputs it on the standard output.

## 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
archive	1/14/2015 12:04 PM	File folder	
log	1/14/2015 12:04 PM	File folder	
EPA-Analytics	1/14/2015 12:10 PM	Application	15 KB
EPA-Analytics.exe	1/14/2015 10:20 A...	XML Configuration...	1 KB
EPA-Analytics	1/14/2015 12:10 PM	Program Debug D...	22 KB
Microsoft.Online.SharePoint.Client.Tenant...	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-Analytics.exe.config (See screenshot below).

Name	Date modified	Type	Size
archive	1/14/2015 12:04 PM	File folder	
log	1/14/2015 12:04 PM	File folder	
EPA-Analytics	1/14/2015 12:10 PM	Application	15 KB
EPA-Analytics.exe	1/14/2015 10:20 A...	XML Configuration...	1 KB
EPA-Analytics	1/14/2015 12:10 PM	Program Debug D...	22 KB
Microsoft.Online.SharePoint.Client.Tenant...	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

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-Analytics.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 connectionString "<PathToFileDirectory>" -prov DataProtectionConfigurationProvider



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



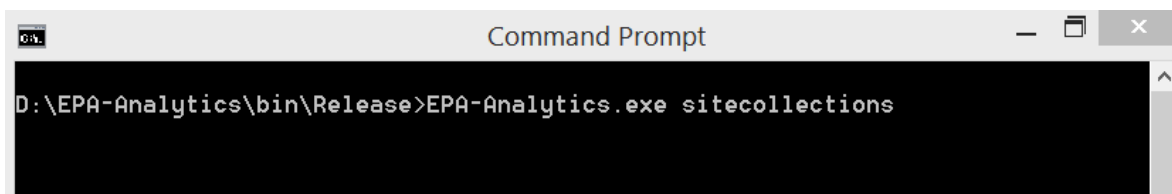
## 4 Usage

### 4.1 Executing the Script

The script should be executed through the command prompt. The script can be executed in 3 modes uses command line argument:

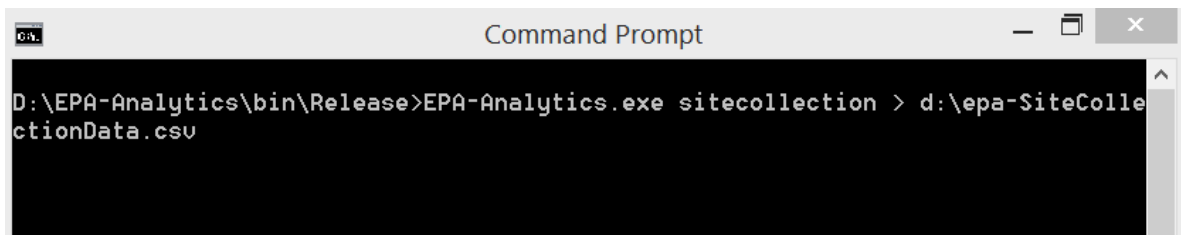
#### 1. Site Collection

The "sitecollection" option instructs the script to process only the root sites and will collect data used for the site collections report.



```
Command Prompt
D:\EPA-Analytics\bin\Release>EPA-Analytics.exe sitecollections
```

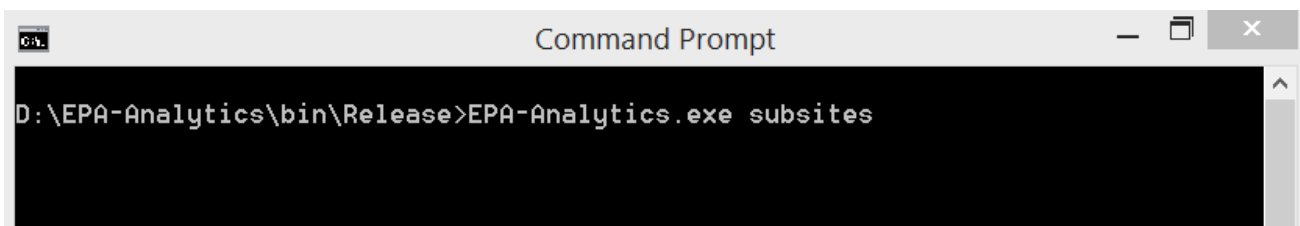
In order to capture the report, pipe the script into your text file of choice.



```
Command Prompt
D:\EPA-Analytics\bin\Release>EPA-Analytics.exe sitecollection > d:\epa-SiteCollectionData.csv
```

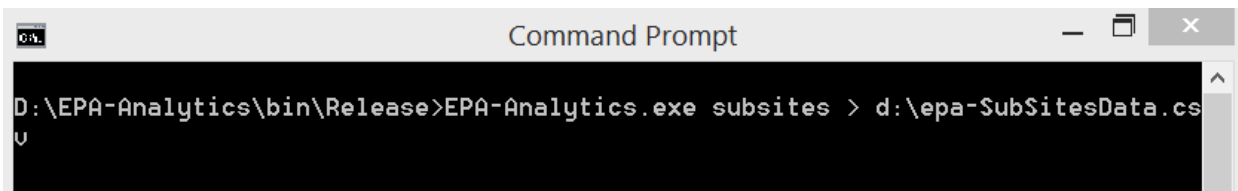
#### 2. Sub-Sites

The "subsites" option instructs the script to process all sub-sites of all site collection. The root site is not processed. The script will collect data used for the subsites report.



```
Command Prompt
D:\EPA-Analytics\bin\Release>EPA-Analytics.exe subsites
```

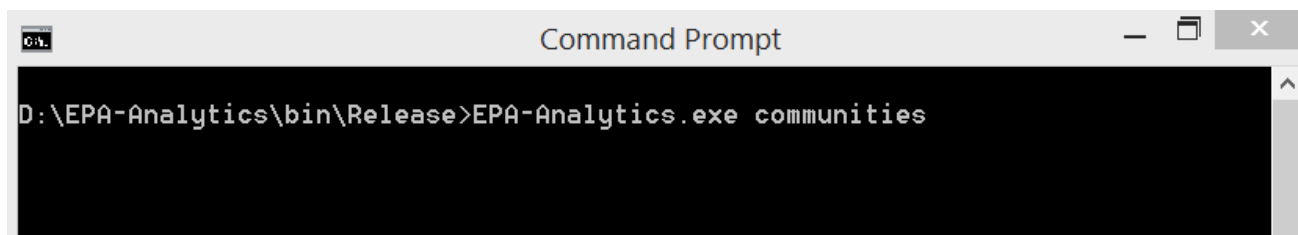
In order to capture the report, pipe the script into your text file of choice.



```
D:\EPA-Analytics\bin\Release>EPA-Analytics.exe subsites > d:\epa-SubSitesData.csv
```

### 3. Communities

The “communities” option instructs the script to process only community sites. The script collects data to be used for the communities report.



```
D:\EPA-Analytics\bin\Release>EPA-Analytics.exe communities
```

## 4.2 Configuring Logging

The script uses nlog as its underlying 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	1/14/2015 12:46 PM	File folder	
obj	7/27/2014 3:05 PM	File folder	
Packages	7/17/2014 12:52 PM	File folder	
Properties	7/17/2014 12:54 PM	File folder	
App	1/14/2015 10:20 A...	XML Configuration...	1 KB
EPA-Analytics	1/14/2015 11:58 A...	Visual C# Project fi...	6 KB
EPA-Analytics.csproj	7/6/2014 12:16 AM	Visual Studio Proje...	1 KB
EPA-Analytics	1/14/2015 12:03 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	1/14/2015 12:03 PM	Visual C# Source fi...	18 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>