Site Metadata Script

Solution Guide

EPA – Site Metadata Script

Prepared for EPA

20-Jan-15

Version .1 Draft

Prepared by

Sofiane ben

Sr. Consultant

sofianeb@microsoft.com

Contributors

David Weinstein

Principal Consultant, daweins@microsoft.com

Revision and Signoff Sheet

Change Record

Date	Author	Version	Change Reference
01/20/2015	Sofiane Ben	.1	Initial draft for internal review/discussion

Reviewers

Name	Version Approved	Position	Date

Table of Contents

Contents

1 Introduction

This document is aimed at describing the solution implemented to identify SharePoint sites where the metadata list used for the sire directory is either missing or miss-configured.

The function of the EPA's O365-SPO Site Metadata script is to identify sites where the Metadata list is not properly set up. Sites missing the metadata list are not listed in EPA's site directory search portal which is not conform to EPA governance policies and collaboration strategy.

The EPA's O365-SPO Site Metadata 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 Metadata list exist and populated with one item. Sites are missing the Metadata list and/or with an empty list are reported on the site information configuration portal.

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

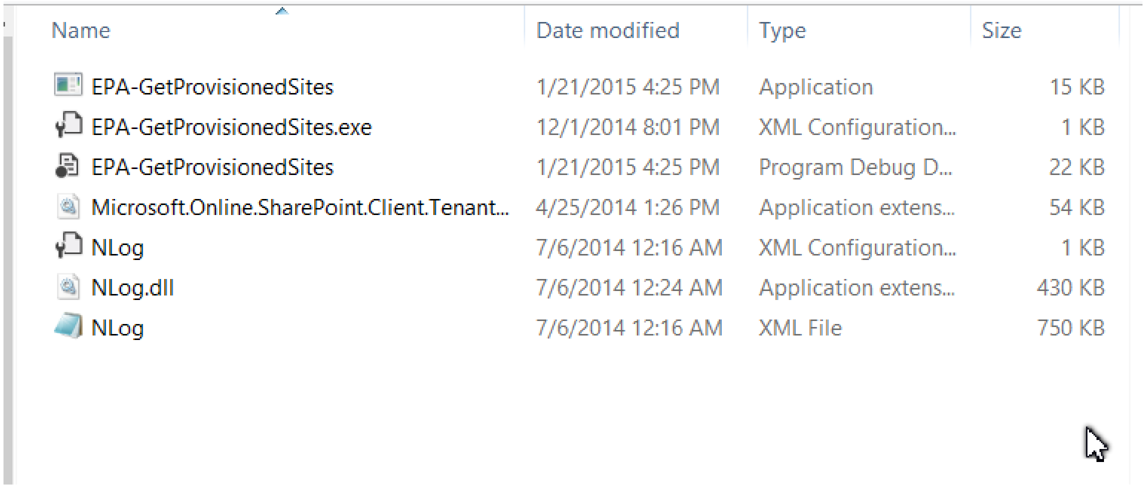
Note:

- 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 excitable binary can be found. In addition, the folder contains the script's supporting files.



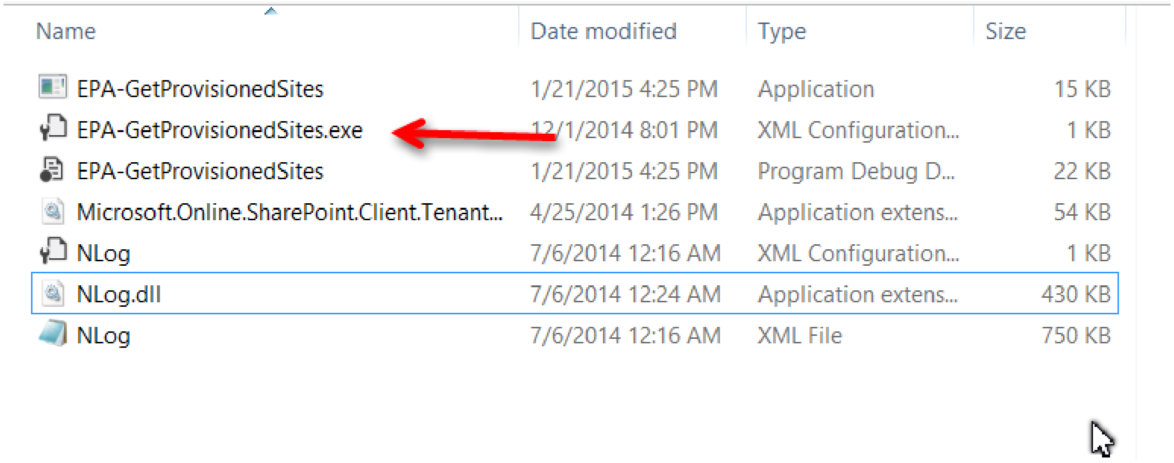
Name	Date modified	Type	Size
EPA-GetProvisionedSites	1/21/2015 4:25 PM	Application	15 KB
EPA-GetProvisionedSites.exe	12/1/2014 8:01 PM	XML Configuration...	1 KB
EPA-GetProvisionedSites	1/21/2015 4:25 PM	Program Debug D...	22 KB
Microsoft.Online.SharePoint.Client.Tenant...	4/25/2014 1:26 PM	Application extens...	54 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.

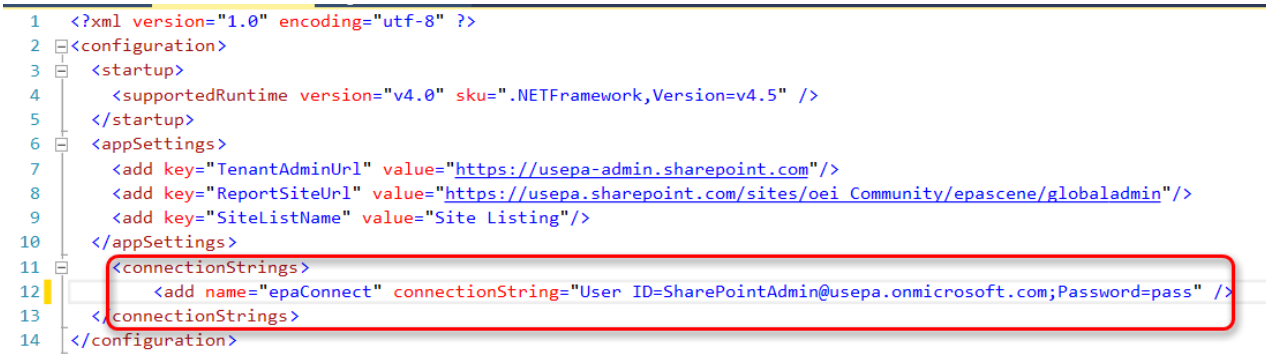
In order for the script to successfully connect to EPA's O365 tenant, the connection script needs to be updated with the appropriate userid and password.

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



Name	Date modified	Type	Size
EPA-GetProvisionedSites	1/21/2015 4:25 PM	Application	15 KB
EPA-GetProvisionedSites.exe	12/1/2014 8:01 PM	XML Configuration...	1 KB
EPA-GetProvisionedSites	1/21/2015 4:25 PM	Program Debug D...	22 KB
Microsoft.Online.SharePoint.Client.Tenant...	4/25/2014 1:26 PM	Application extens...	54 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

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



```xml
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
  </startup>
  <appSettings>
    <add key="TenantAdminUrl" value="https://usepa-admin.sharepoint.com"/>
    <add key="ReportSiteUrl" value="https://usepa.sharepoint.com/sites/oei Community/epascene/globaladmin"/>
    <add key="SiteListName" value="Site Listing"/>
  </appSettings>
    <connectionStrings>
        <add name="epaConnect" connectionString="User ID=SharePointAdmin@usepa.onmicrosoft.com;Password=pass" />
  </connectionStrings>
</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.

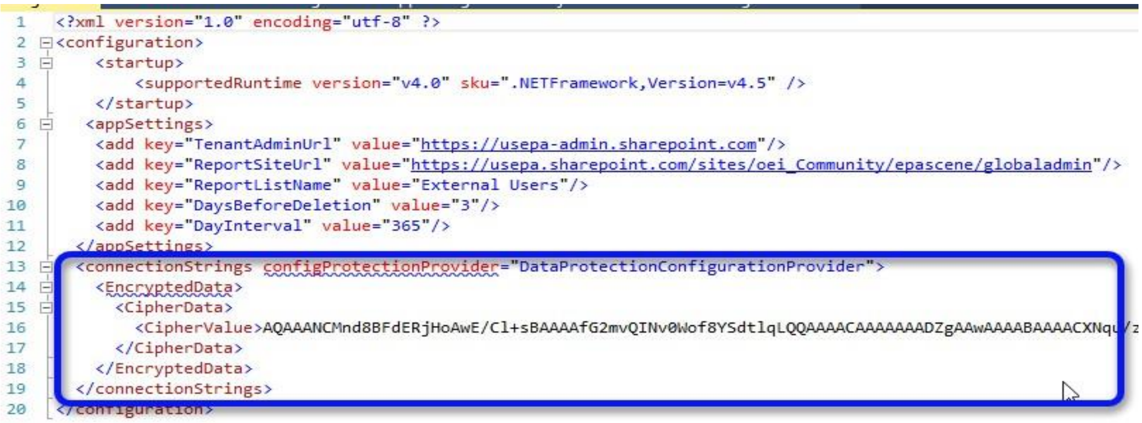
In order to encrypt the configuration file:

- Rename EPA-GetProvisionedSites.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.



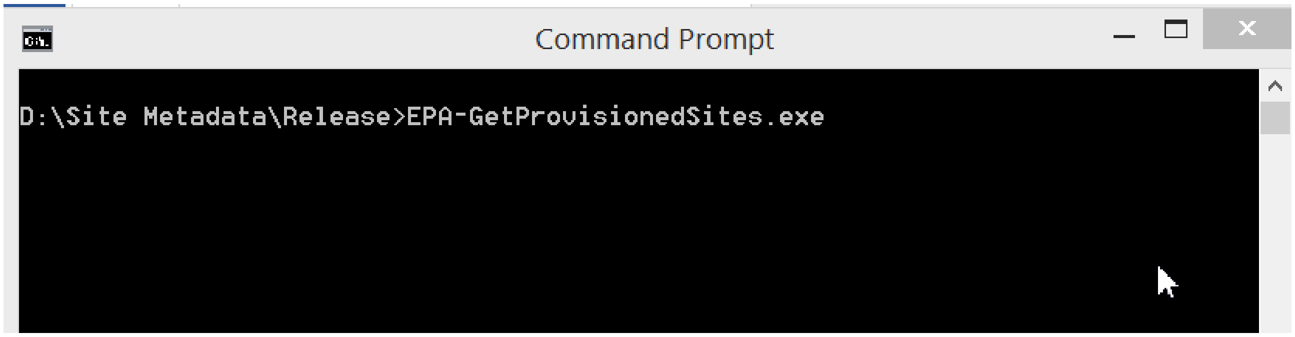
```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
    </startup>
   <appSettings>
    <add key="TenantAdminUrl" value="https://usepa-admin.sharepoint.com"/>
    <add key="ReportSiteUrl" value="https://usepa.sharepoint.com/sites/oei_Community/epascene/globaladmin"/>
    <add key="ReportListName" value="External Users"/>
    <add key="DaysBeforeDeletion" value="3"/>
    <add key="DayInterval" value="365"/>
  </appSettings>
  <connectionStrings configProtectionProvider="DataProtectionConfigurationProvider">
    <EncryptedData>
      <CipherData>
        <CipherValue>AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAfG2mvQINv0Wof8YSdtlqLQQAAAACAAAAAADZgAAwAAAABAAAACXNqu/z
      </CipherData>
    </EncryptedData>
  </connectionStrings>
</configuration>
```

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

4 Usage

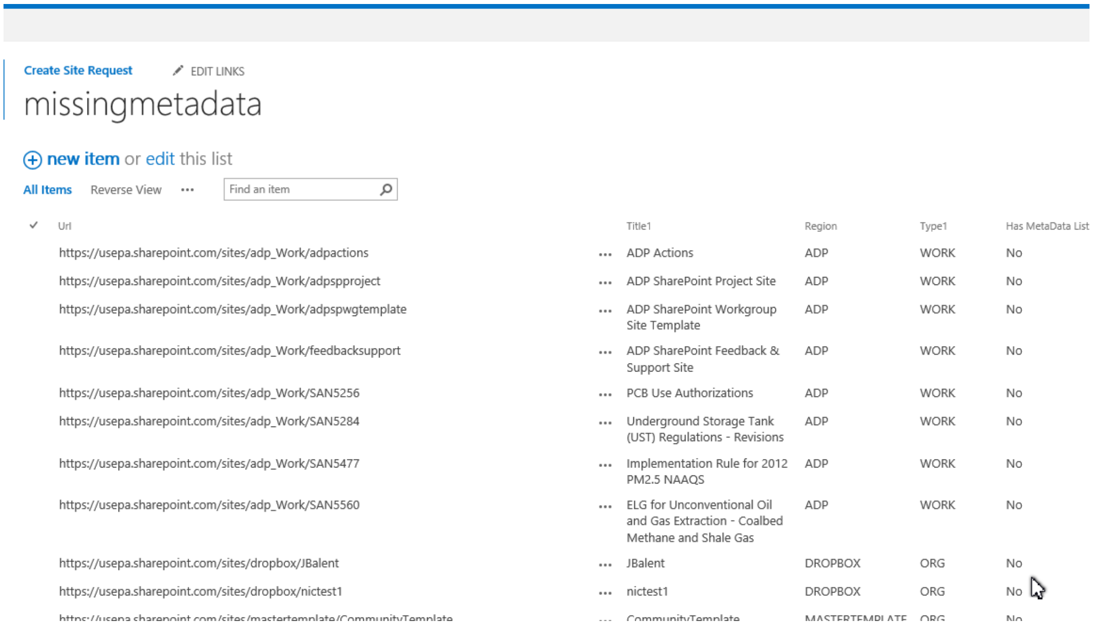
4.1 Executing the Script

The script should be executed through the command prompt. The script doesn't accept any command line parameter and is executed without any option.



```
D:\Site Metadata\Release>EPA-GetProvisionedSites.exe
```

If new problematic sites are found, the script updates the "missingmetadata" list on the configuration site



missingmetadata

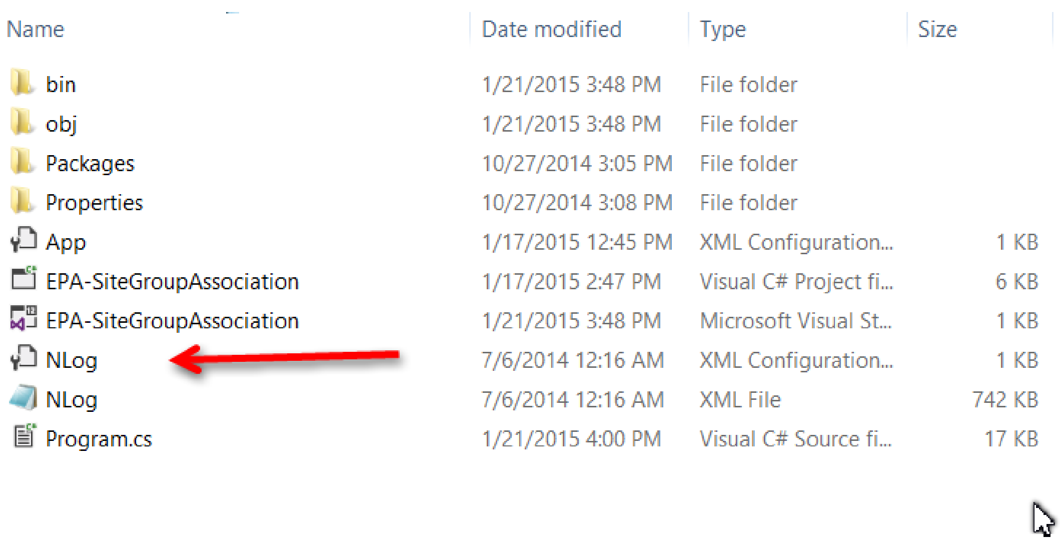
✓ Url		Title1	Region	Type1	Has MetaData List
https://usepa.sharepoint.com/sites/adp_Work/adpactions	•••	ADP Actions	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/adpspproject	•••	ADP SharePoint Project Site	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/adpspwgtemplate	•••	ADP SharePoint Workgroup Site Template	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/feedbacksupport	•••	ADP SharePoint Feedback & Support Site	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/SAN5256	•••	PCB Use Authorizations	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/SAN5284	•••	Underground Storage Tank (UST) Regulations - Revisions	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/SAN5477	•••	Implementation Rule for 2012 PM2.5 NAAQS	ADP	WORK	No
https://usepa.sharepoint.com/sites/adp_Work/SAN5560	•••	ELG for Unconventional Oil and Gas Extraction - Coalbed Methane and Shale Gas	ADP	WORK	No
https://usepa.sharepoint.com/sites/dropbox/JBalent	•••	JBalent	DROPBOX	ORG	No
https://usepa.sharepoint.com/sites/dropbox/nictest1	•••	nictest1	DROPBOX	ORG	No
https://usepa.sharepoint.com/sites/mastertemplate/CommunityTemplate	•••	CommunityTemplate	MASTERTEMPLATE	ORG	No

4.2 Configuring Logging

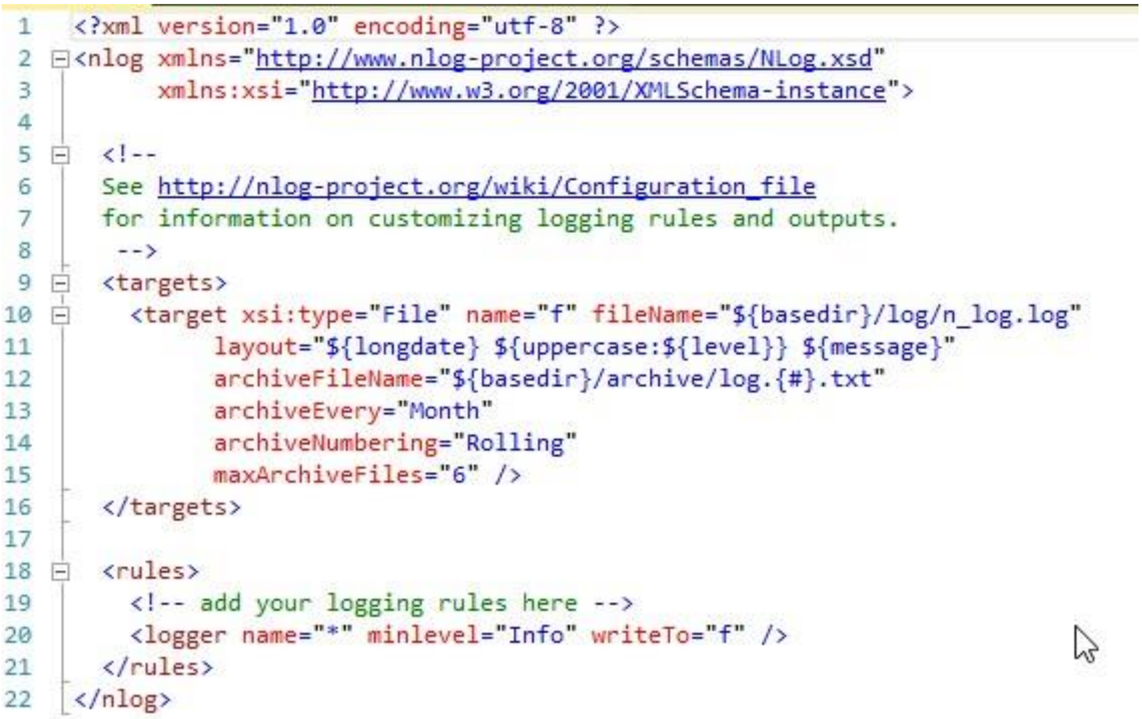
The script uses nlog as its underlining logging framework. This enables the tool to have highly configurable logging capabilities including:

- 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



```
<?xml version="1.0" encoding="utf-8" ?>
<nlog xmlns="http://www.nlog-project.org/schemas/NLog.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <!--
  See http://nlog-project.org/wiki/Configuration_file
  for information on customizing logging rules and outputs.
   -->
  <targets>
    <target xsi:type="File" name="f" fileName="${basedir}/log/n_log.log"
          layout="${longdate} ${uppercase:${level}} ${message}"
          archiveFileName="${basedir}/archive/log.{#}.txt"
          archiveEvery="Month"
          archiveNumbering="Rolling"
          maxArchiveFiles="6" />
  </targets>

  <rules>
    <!-- add your logging rules here -->
    <logger name="*" minlevel="Info" writeTo="f" />
  </rules>
</nlog>
```

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