How-To- Set AutoDiscoverSiteScope using a sample script I created

Content table

1. Script Introduction

I won't describe the AutoDiscover Site Scope (also called "Site Affinity") principles here, there are some nice and clear TechNet articles or MVP articles like this one...
I created a sample script here, which sets the **AutodiscoverSiteScope** according to a CAS-To-Site map that we must define in a CSV file. The input CSV file must have the following headers:

```
YourServer_1, YourServer_2, YourServer_3
Site_Name1, Site_Name1, SiteName2
ClientOnly_Site1, ClientOnly_Site1,ClientOnlySite2
…
```

HINT: You can use Excel to create your CSV file with your CAS servers as headers, and for each header (=each server) the column will contain the AD sites you wish to associate with these CAS servers.

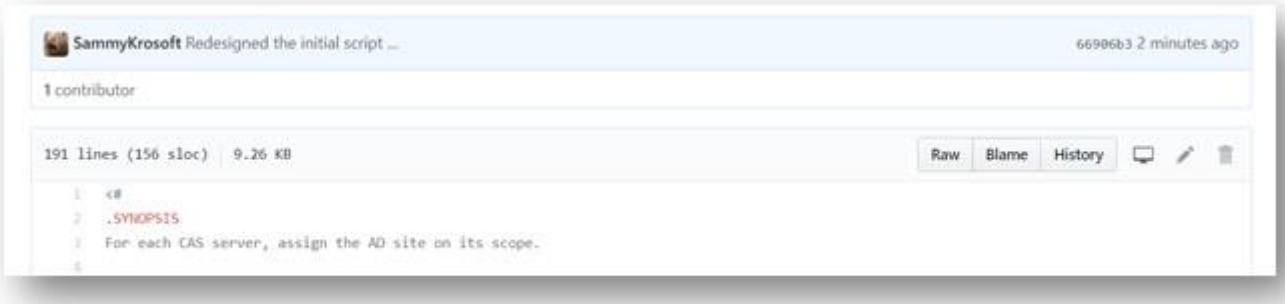
NOTE: The script generates a sample CSV file for you with 4 CAS servers to give you a better idea of how if should look like – see later on this page.

2. Script Download

- To get the script, go to the following link:

https://github.com/SammyKrosoft/PowerShell/blob/master/Sample-CASSiteScopes.ps1

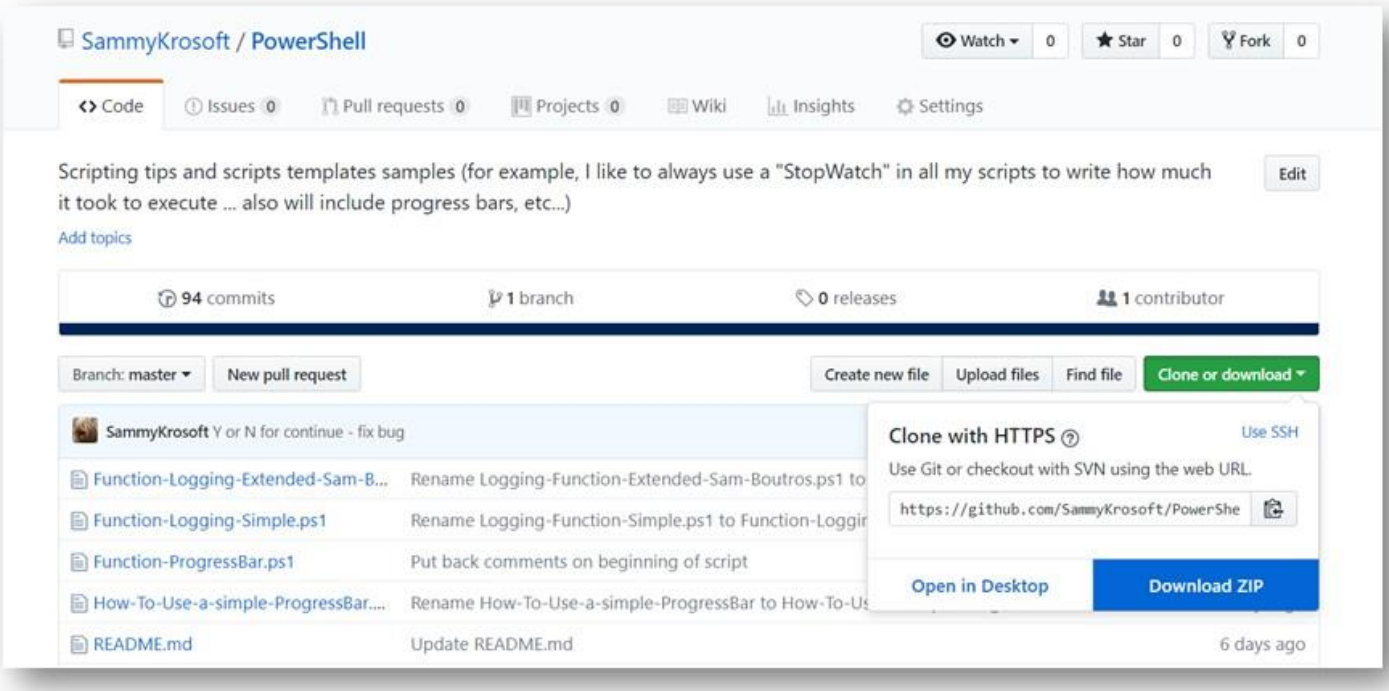
- NOTE: you can click on the "**RAW**" button (see below), and just **copy/paste the text on your own "Sample-CASSiteScopes.PS1" file through NOTEPAD**



To get the RAW content of the script directly from an URL, to be able to copy/paste the text into your NOTEPAD and save it as Sample-CASSiteScopes.ps1 (or the name of your choice), here is the link: https://raw.githubusercontent.com/SammyKrosoft/PowerShell/master/Sample-CASSiteScopes.ps1
Or you can download the whole scripts repository and use the "Sample-CASSiteScopes.ps1" script directly – it's up to your preferences.

To download the whole scripts repository (mostly templates for now) :

- Go to my GitHub space http://aka.ms/pwsh
- Click on the below "Clone or download" green drop down list, and choose "Download Zip"



3. Integrated script help

Save the script on your computer, and get the synthetized, full help or examples-only by typing :

`get-help .\Sample-CASSiteScopes.ps1` => that will get you the basic help text
`get-help .\Sample-CASSiteScopes.ps1 -Examples` => that will get you only the examples
`get-help .\Sample-CASSiteScopes.ps1 -Full` => that will show you the full help with parameters, and examples

Here is the output with the `Get-Help <Script_Path\Script_name.ps1> -Examples`, try yourself with the `-Full` parameter of the `Get-Help` (2nd line above) to see the magic happening :-)

```
get-help .\Sample-CASSiteScopes.ps1 -Examples

NAME
    C:\scripts\Sample-CASSiteScopes.ps1

SYNOPSIS
    For each CAS server, assign the AD site on its scope.

    -------------------------- EXAMPLE 1 --------------------------

    C:\PS>.\Sample-CASSiteScopes.ps1 -UseSampleCASServers


    Will generate a sample CSV file from within the script with CAS1, CAS2,
CAS3 and CAS4, and fake AD Site names, just
     to show you how the script work a little bit.
    The Sample-AutoDiscoverSiteScope.csv will be located on the same
directory where you are executing the script, and
    will show you how your CSV file for your real server must be configured.



    -------------------------- EXAMPLE 2 --------------------------

    C:\PS>.\Sample-CASSiteScopes.ps1


    This will launch the script against your production servers using the
default <Script Directory>\Classeur.csv file
    containing your CAS servers (as file header) and your AD sites for each
CAS server (under each CAS server header)
    This will NOT execute the AD Site scope setting because we don't specify
the -EXECUTE parameter -> that's to test i
    f we' re good before executing the changes.
    If Classeur.CSV doesn't exist, the script will tell you and exit.



    -------------------------- EXAMPLE 3 --------------------------

    C:\PS>.\Sample-CASSiteScopes.ps1 -Execute


    Same as the above example, but this time it will execute the commands and
assign the sites to the CAS servers, as d
    efined in the default CLASSEUR.CSV.
    If Classeur.CSV does not exist, the script will tell you and exit.
```

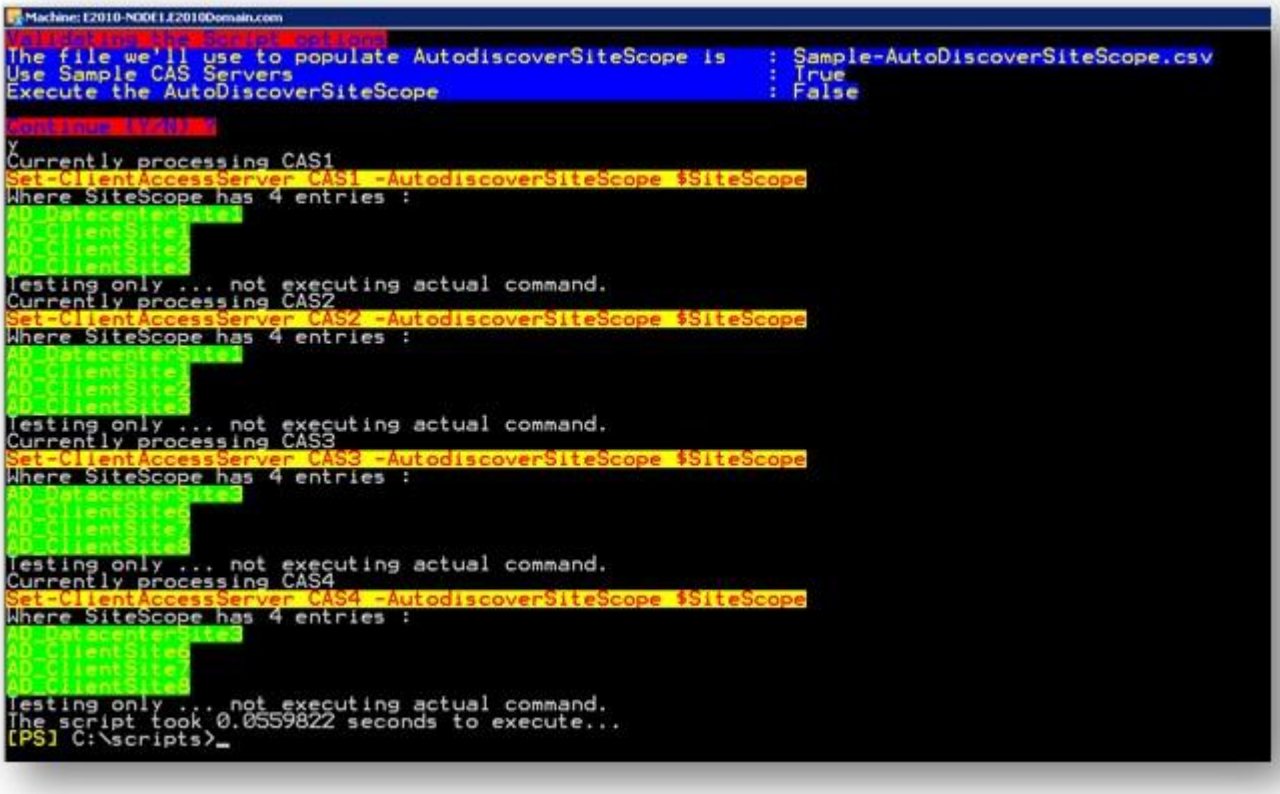
```
    -------------------------- EXAMPLE 4 --------------------------

    C:\PS>.\Sample-CASSiteScopes.ps1 -CSVFileName "C:\temp\AutreCSVFile.csv"
-Execute


    Will not only display the command that will set the AD sites from default
CSV file (note : no -CSVFileName parameter used, then
    will take the default C:\temp\Classeur1.csv file if it exists (otherwise
if Classeur1.csv doesn't exist, will output an error message),
    but will also Execute the actual command to set the CAS
AutodiscoverSiteScope (aka Site Affinity) as per the map de
    fined in the file defined in the -CSVFileName parameter (in this example,
C:\temp\AutreCSVFile.csv)
```

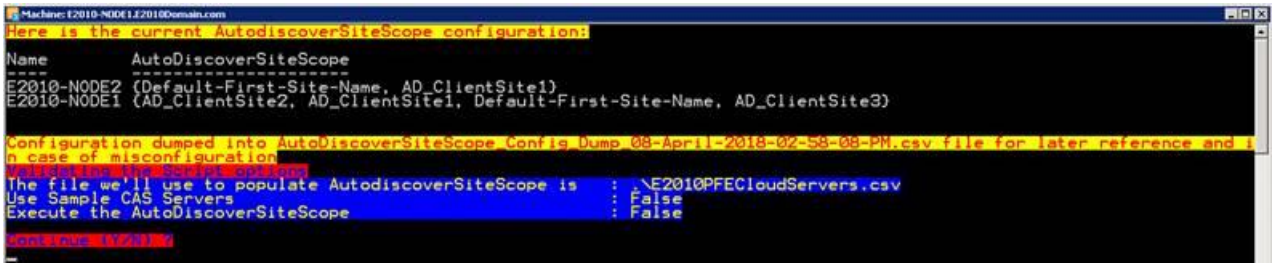
4. Script execution modes illustrations

Below is an execution example applied with the **`-UseSampleCASServers`** – will show in yellow background the type of PowerShell command it would execute (without executing it), and list the sites from the sample CSV file auto-generated by the script:



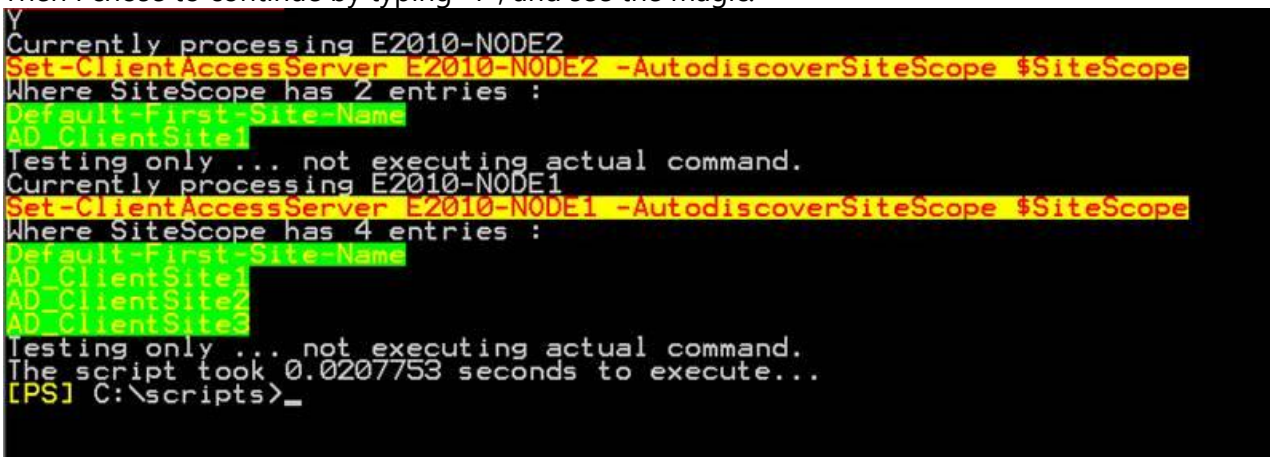
And using my own CSV file with my LAB servers names as headers and the AD sites I want to set, but I don't want to execute (I don't specify the -EXECUTE parameter) :

It first shows me the current configuration of my Autodiscover Sites scopes (and save it in a file for future reference, documentation, or just in case I want to revert because I made mistake in my CSV file), then in blue background shows the current script actions, and ask you if you wish to continue or not :



NOTE: typing anything else than Y or N will keep asking you if you wish to continue or not

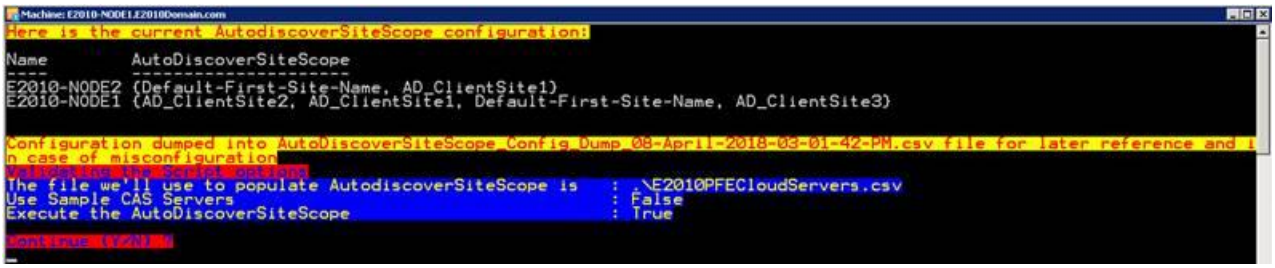
Then I chose to continue by typing "Y", and see the magic:



NOTE: since I didn't use the -Execute switch with the script, it will only display the Set- cmdlet with the options it would do if you launch the script using the -EXECUTE switch.

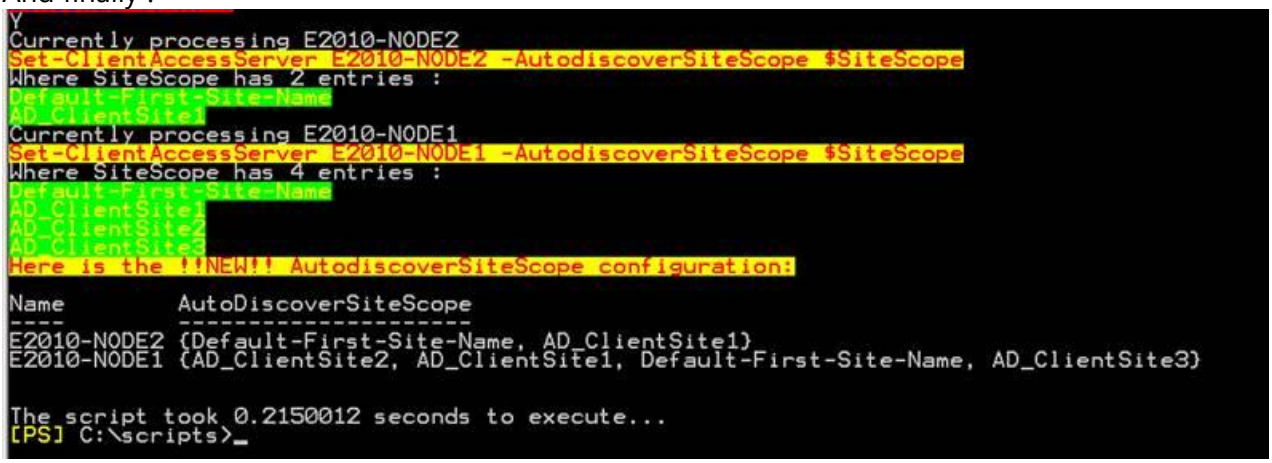
And now last example, I triple-checked above that these are the sites I want to set for my CAS servers, I now execute the script again but with the -EXECUTE parameter so that the changes are applied.
NOTE: the initial configuration before the change is always saved in a file for security and future reference, in the current directory and with the name and date/time tag of :
AutoDiscoverSiteScope_Config_Dump_08-April-2018-02-58-08-PM.csv

NOTE: same command line as above, but with the -EXECUTE parameter this time



NOTE: see the "Execute the AutoDiscoverSiteScope" shows "TRUE" now because we specified the -Execute parameter – compare with above, without specifying the -Execute parameter, it said "False"

And finally :



NOTE: we also dump the resulting configuration of the AutoDiscoverSiteScope just to compare with the previous configuration – on my example it's the same because I ran the script multiple times already with the same CSV file