n



*Microsoft Onboarding Accelerator*

*Active Directory Certificate Services*

*Deployment & Migration*

*CA Setup Helper- Manual*

Prepared for

Microsoft internal usage only

16. Jan. 2020

Version 1.0

Prepared by

Andreas Luy

anluy@microsoft.com

Premier Field Engineering  
[Microsoft Premier Services](http://premier.microsoft.com/)

**Conditions and Terms of Use**

**Microsoft Confidential**

This training package is proprietary and confidential, and is intended only for uses described in the training materials. Content and software is provided to you under a Non-Disclosure Agreement and cannot be distributed. Copying or disclosing all or any portion of the content and/or software included in such packages is strictly prohibited.

The contents of this package are for informational and training purposes only and are provided "as is" without warranty of any kind, whether express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

Training package content, including URLs and other Internet Web site references, is subject to change without notice. Because Microsoft must respond to changing market conditions, the content should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication. Unless otherwise noted, the companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

**Copyright and Trademarks**

© 2018 Microsoft Corporation. All rights reserved.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

For more information, see Use of Microsoft Copyrighted Content at

<http://www.microsoft.com/en-us/legal/intellectualproperty/Permissions/default.aspx>

DirectX, Hyper-V, Internet Explorer, Microsoft, Outlook, OneDrive, SQL Server, Windows, Microsoft Azure, Windows PowerShell, Windows Server, Windows Vista, and Zune are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other Microsoft products mentioned herein may be either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are property of their respective owners.

Revision and Signoff Sheet

Change Record

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Version | Change reference |
| January 10, 2020 | Andreas Luy | 1.0 | Initial Release |
| January 15, 2020 | Andreas Luy | 1.1 | Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 1: Change Record

Reviewers

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Version approved | Position | Date |
| Dagmar Heidecker | 1.1 |  | January 14, 2020 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 2: Reviewers

Table of Contents

Table of Contents

[Revision and Signoff Sheet 3](#_Toc30060246)

[Change Record 3](#_Toc30060247)

[Reviewers 3](#_Toc30060248)

[Table of Contents 4](#_Toc30060249)

[Introduction 5](#_Toc30060250)

[How to use the OA ADCS Setup Helper 6](#_Toc30060251)

[Prerequisites 6](#_Toc30060252)

[Getting started 6](#_Toc30060253)

[Creating Customer-specific CA Configuration files 6](#_Toc30060254)

[Running the Script 7](#_Toc30060255)

[Appendix 11](#_Toc30060256)

[Arguments accepted by the Script 11](#_Toc30060257)

[List of Figures 12](#_Toc30060258)

[List of Tables 12](#_Toc30060259)

Introduction

The OA ADCS CA Setup Helper provides a convenient way of installing and configuring ADCS by automating installation and applying settings from customer-specific configuration files.

Setup of ADCS based on XML-configuration files is a two-step approach:

Step 1 installs the binaries and creates the certificate requests, while step 2 installs the CA certificate and applies all the necessary configuration. Active Directory-integrated CAs must be rebooted between step 1 and 2 to ensure that Kerberos tokens get refreshed.

Customer configuration files may define one or more Certification Authorities and their configuration.

The customer configuration files are written in the XML Format.

1. For the future, it is planned to include more PKI-related services as OCSP and NDES.

How to use the OA ADCS Setup Helper

Prerequisites

* You will need Windows PowerShell Version 4.0 or newer installed on your machine.
* You need to be local administrator to run the script in **“Install”** mode.
* You don’t need to have administrative permissions to run the script in **“View”** mode.

Getting started

The base component to setup a CA is the CA configuration XML file. Each configuration file starts with a config node. The config node contains information on the schema file that is used to verify the configuration. Each CA configuration must contain one CA node. The node contains all configuration settings for a single CA.

Creating Customer-specific CA Configuration files

To create customer-specific configuration files, take the sample configuration files that can be found in the “conf” Subdirectory of the Script.

1. **Never rename or edit the sample configuration files directly**. Make a copy of them and store them separately.



Figure : Sample CA Configuration files

Each CA configuration starts typically with the configuration file for the root CA. Create a copy of the file and rename it as desired.

1. You should use an **XML** **editor with syntax highlighting** to edit the customer configuration files. You may otherwise have serious trouble properly formatting them. Give **Visual Studio Code** a try!
2. The CA configuration files are mostly self-explanatory and/or inline commented.

We strongly recommend making yourself familiar with the structure of the configuration files before running this in a customer’s environment.

Running the Script

Copy the provided OA ADCS Setup Helper ZIP file onto the designated new CA and unpack.

Open an elevated Windows PowerShell and navigate to the folder where the OA ADCS Setup Helper files are stored. You run the script with:

.\Setup-CA.ps1

If you do specify **-Help** as argument on the command line, it will print the help section of the script.

.\Setup-CA.ps1 -Help

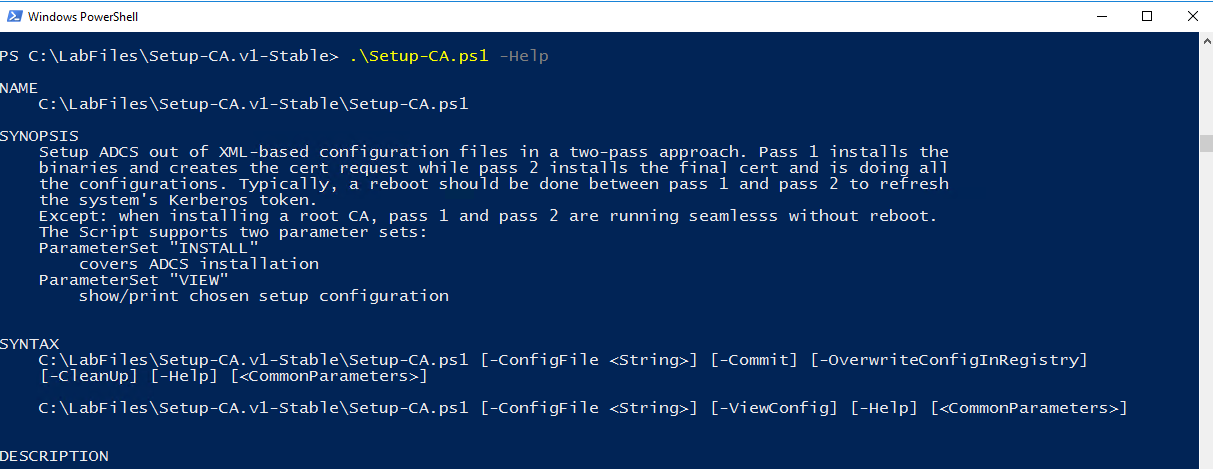


Figure : Running the Script with -Help Argument

1. For a general understanding, we first work with sample data.

Install and configure ADCS out of XML-based configuration files in a two-step approach. Step 1 installs the binaries and creates the cert request while step 2 installs the final cert and is doing all the configurations. Typically, a reboot should be done between step 1 and step 2 to refresh the system's Kerberos token.

**Except:** when installing a root CA, steps 1 and 2 are running seamlesss without reboot.

The following is only valid for installing a subordinated CA

After the step 1 run, the script pops out the request file for the CA certificate and asks for a reboot. The request file can be found at the root of the system drive (typically c:\):

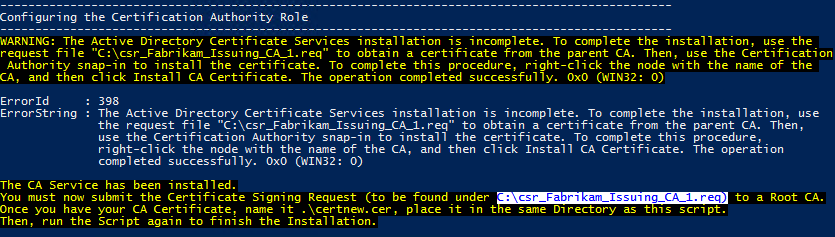


Figure : Result of step 1 run of the script. The text includes the CSR file name and location.

Now, you need to transfer the CSR file to your upper tier CA. At your upper tier CA, submit the request file, sign and issue the final certificate. Now export the certificate in DER binary format and get the final certificate back to your system (**we currently don’t support PKCS#7 files within the script**).

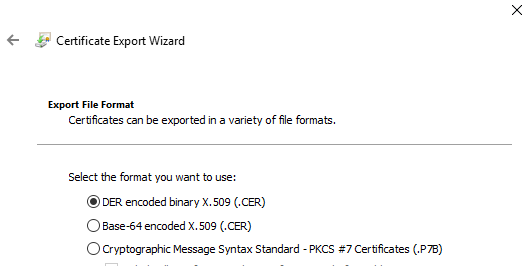


Figure : Export the issued certificate in DER encoded binary format.

Save the certificate file in the script’s root folder and rename it to “**certnew.cer**”.

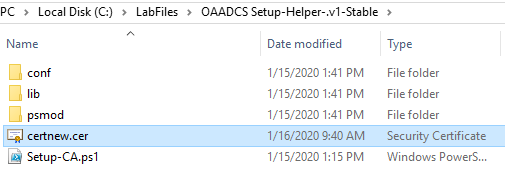


Figure : Save the certificate file as “certnew.cer” into the script’s root folder.

Before you continue with the step 2 run, ensure you established the appropriate CA certificates chain trust (AIA/root CA trust), especially if you are installing a non-domain joined CA system. For enterprise CAs, ensure you reboot them before continuing.

1. Ensure you are using the same account for step 2 as you did for step 1.

The Command Line Arguments

ConfigFile

To specify customer configuration files, you can add the **-ConfigFile** argument.

.\Setup-CA.ps1 -ConfigFile “Path-to\Customer-CAConfig.xml”

1. If you omit the **-ConfigFile** Argument, a file selection dialogbox will open to select the config file.

The configfile selection is only important for step 1. During the installation process, the value of “ConfigFile” will be written into the registry and reused for step 2.

Commit

The script will not do anything if there are no additional arguments given. To make the script do something, you will have to specify at least either the **-Commit** or the **-ViewConfig** argument.

.\Setup-CA.ps1 -ConfigFile “Path-to\Customer-Config.xml” -ViewConfig

.\Setup-CA.ps1 -ConfigFile “Path-to\Customer-Config.xml” -Commit

The **-Commit** argument will make the script to install ADCS, but it is recommended to **first run it without the -Commit argument**, it will verify the configuration files and display the configuration and any errors detected.

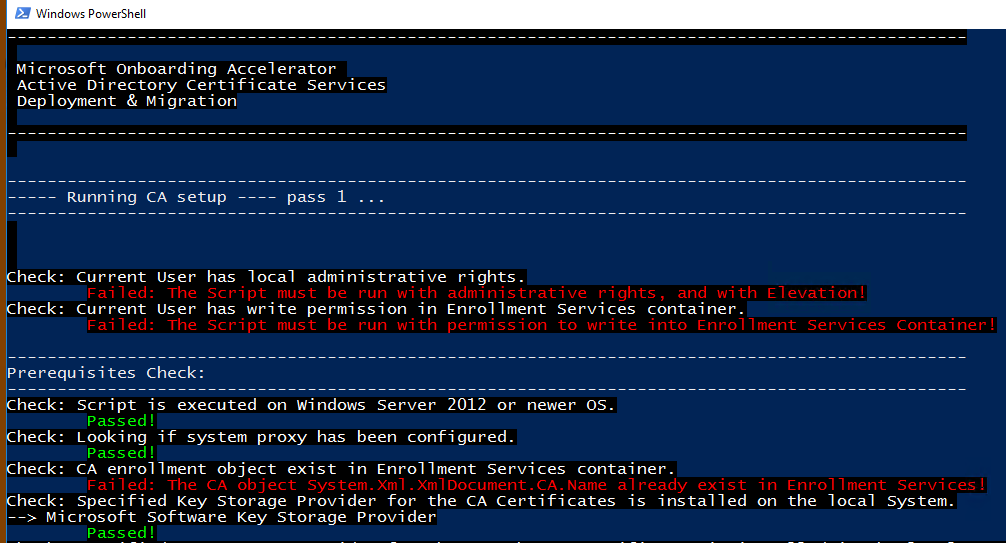


Figure : Script Run without the -Commit Argument, to check for errors

In both cases, the script will verify the input configuration files against XML schemata. If there is an error, the script will halt and display the verification error.

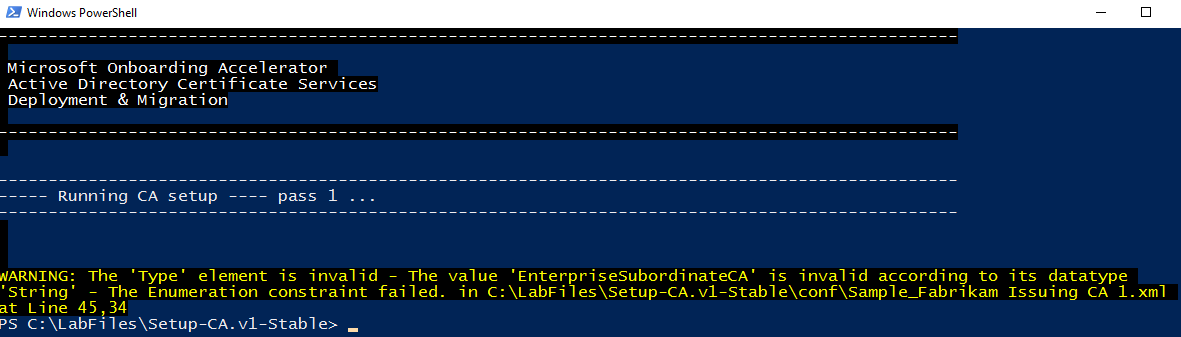


Figure : Script Run without the -Commit Argument, using sample Data with Errors in the XML File

Therefore, it is recommended to run the script first without the **-Commit** argument and then re-run it with the **-Commit** argument if there were no errors found.

OverwriteConfigInRegistry

During step 1 the used **ConfigFile** and **setup account** are written into the system’s registry and reused/validated automatically during step 2. If **-OverwriteConfigInRegistry** switch is defined for step 2, the setup scripts ignore the previously written values in registry and applies the one from the -**ConfigFile** (or file selection dialogue) provided with the **-OverwriteConfigInRegistry** argument.

1. This parameter should be used with care as it allowes to mix different **ConfigFiles and setup accounts.**

CleanUp

The **-CleanUp** argument is used to clean up the registry from temporary setup status entries done in step 1. Can be used in any step 1 ( in step 1 only together with the "**-Commit**" switch) to automaticall remove the temporary setup status entries after sucessful installation. When calling **CleanUp** after successful installation, it removes left over registry entries from ADCS installation.

ViewConfig

ViewConfig let you verify and print a specific CA configuration. ViewConfig will not check for any installation prerequirements and cannot be used with any of the other switches above.

Appendix

Arguments accepted by the Script

|  |  |  |
| --- | --- | --- |
| Argument | Required | Description |
| ConfigFile | optional | Specify the Customer Configuration file here. It may either be a relative or an absolute path. if setup is running pass 2, the ConfigFile entered in pass 1 has been written into registry and will be used as default. Only be used when entered together with "OverwriteConfigInRegistry" switch, ConfigFile parameter will be used in pass 2. Otherwise ConfigFile parameter will be ignored in pass 2.  If no ConfigFile will be entered on the command line, a file selection dialogbox will open to select the config file. |
| Commit | optional | Add this Parameter to make the Script actually do anything. If missing, the configuration based on the used ConfigFile will be written to output window to get verified before application.  **Note:** commit will be automatically $true when in pass 2 except when changing the parameterset to "View" |
| OverwriteConfigInRegistry | optional | Has only effect if running pass 2. ignores registry setting for ConfigFile and setup account and uses that one in the command parameter instead.  (this parameter should be used with care as it allowes to mix different ConfigFiles) |
| CleanUp | optional | This will clean up the registry from temporary setup status entries. Can be used either in any step (during step 1 only together with the "commit" switch) to automaticall remove temporary setup status entries after sucessful installation. When calling CleanUp after successful installation, it removes left over registry entries from ADCS installation. |
| ViewConfig | optional | Allows to any time reviewing the configuration. ViewConfig will not check for any installation prerequirements and cannot be used with any of the other switches above. |
| Help | optional | Display help |

Table : Arguments accepted by the Script

List of Figures

[Figure 1: Sample CA Configuration files 6](#_Toc30060260)

[Figure 2: Running the Script with -Help Argument 7](#_Toc30060261)

[Figure 3: Result of step 1 run of the script. The text includes the CSR file name and location. 8](#_Toc30060262)

[Figure 4: Export the issued certificate in DER encoded binary format. 8](#_Toc30060263)

[Figure 5: Save the certificate file as “certnew.cer” into the script’s root folder. 8](#_Toc30060264)

[Figure 6: Script Run without the -Commit Argument, to check for errors 9](#_Toc30060265)

[Figure 7: Script Run without the -Commit Argument, using sample Data with Errors in the XML File 10](#_Toc30060266)

List of Tables

[Table 1: Change Record 3](#_Toc30060267)

[Table 2: Reviewers 3](#_Toc30060268)

[Table 3: Arguments accepted by the Script 11](#_Toc30060269)