HowTo: Powershell Script to cleanup expired certificates from a Microsoft CA Part 2 of 2

Written by: 12/19/2014 2:01:00 PM

In this second post I explain *how to query or delete certificate entries from a MS CA DB*. For this I wrote a PowerShell script with 4 functions. I don't use any modules!
The only "tool" I use is the certutil.exe.
The complete script is available in this post!

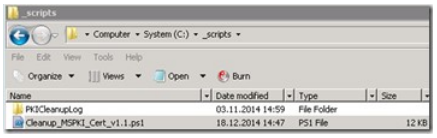
>> Here is the link to the first part of this series:
How To: Cleanup expired certificates from a Microsoft CA with Powershell and Shrink the DB Part 1 of 2

Here in part2 I explain the powershell script in detail (the script is used in step 2 of part 1 of this series) :

The Microsoft Enterprise CA I'm responsible for is running on a Microsoft Windows Server 2008 Enterprise Server
- with PowerShell 2.0 installed
- no 3rd party PS modules are used
- the certutil.exe is used by the powershell (PS) script

- the PS script I created is "Cleanup_MSPKI_Cert_v1.1.ps1" and contains 3 functions

On this CA Server in the C:\ root drive **I create a folder** "*_scripts* " (I don't use PS remoting) and copy my powershell script "*Cleanup_MSPKI_Cert_v1.1.ps1*" into this folder



Per default the functions "*Remove-ExpiredCertFromDB*" writes the temporary files to a subfolder within C:_scripts\PKICleanupLog.

You can change this default folderpath with the parameter "CleanedFolderLogPath"

```
[string]$CleanedFolderLogPath ="C:\_scripts\PKICleanupLog",
```

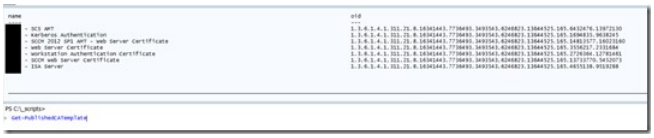
The 3 functions I have implemented are:

A.) Get-PublishedCATemplate

B.) Get-IssuedCert

C.) Remove-ExpiredCertFromDB

A.) Get-PublishedCATemplate



When you run this function without a parameter, it displays all Templates from the "Certificate Templates" folder with it's OID. This OID is used by the other to functions to display or delete certificates issued with this certain template. In the following picture you see the corresponding templates from the PKI SnapIn


```
function Get-PublishedCATemplate{
    [CmdletBinding()]
    Param (
        [parameter()]
        [string]$filter
    )
    $FilterLen = ("msPKI-Cert-Template-OID =").length+3
```

```
    $AllPublishedTemplates = Invoke-Expression "certutil.exe -catemplates -v | select-string msPKI-Cert-Template-OID"
    $AllPublishedTemplates | foreach{
        $tmp= ($_.line).Substring($FilterLen)
        $splitarr = $tmp.split(" ",2)
        $obj = New-Object PSObject
        Add-Member -Input $obj -Name "name" -MemberType Noteproperty -Value $Splitarr[1].trim()
        Add-Member -Input $obj -Name "oid" -MemberType Noteproperty -Value $Splitarr[0].trim()
        if ($PSBoundParameters["filter"]){
            if ($Splitarr[1].trim() -match $filter){
                write-output $obj
            }
        }
        else{
            write-output $obj
        }
    }
}
```

below I run the script with the *–filter* parameter and so I only get templates with "SCCM" in their name

I assign the oid of ONE template (=> change filter that you get only one result!) to the variable *WSTemplate*

$WSTemplate = (Get-PublishedCATemplate -filter workstation).oid

```
PS C:\_scripts> $WSTemplate = (Get-PublishedCATemplate -filter Workstation).oid

PS C:\_scripts> $WSTemplate
1.3.6.1.4.1.311.21.8.16341443.7736493.3493543.6246823.13644525.165.2726364.12781461
```

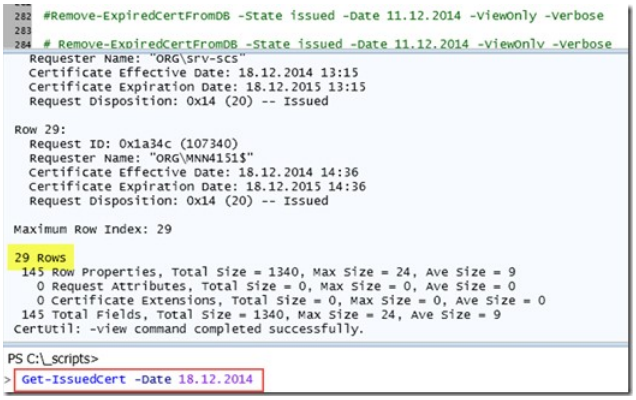
B.) Get-IssuedCert

With this function you can list the certificates issued from all templates or a certain template (specified with it's oid = $CertTemplate variable) which are issued beginning at a certain date.

```
function Get-IssuedCert{
  [CmdletBinding()]
  Param (
    [ValidatePattern('^([0-9\.\s])+$')]
    [string]$CertTemplate,
    [ValidatePattern('^\d\d[\./]{1}\d\d[\./]{1}\d\d\d\d$')]
    [string]$Date
  )
  if ($PSBoundParameters["CertTemplate"]){
    Invoke-Expression "certutil.exe -view -restrict 'certificate template=$CertTemplate,disposition=20,notbefore>=$Date' -out 'Request.Requ
  }
  else {
    # displays Certificates issued with any custom template
    Invoke-Expression "certutil.exe -view -restrict 'disposition=20,notbefore>=$Date' -out 'Request.RequestID,Request.RequesterName,NotBefo
  }
}
```

The following example lists all 29 certificates (from ALL templates) issued from December 18. 2014 and later …. (with this version it's not possible to select a time range / only a "start-date")

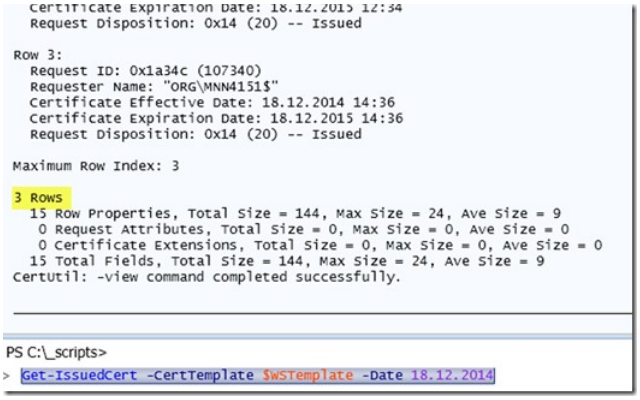
Get-IssuedCert -Date 18.12.2014



```
#Remove-ExpiredCertFromDB -State issued -Date 11.12.2014 -ViewOnly -Verbose

# Remove-ExpiredCertFromDB -State issued -Date 11.12.2014 -ViewOnly -Verbose
  Requester Name: "ORG\srv-scs"
  Certificate Effective Date: 18.12.2014 13:15
  Certificate Expiration Date: 18.12.2015 13:15
  Request Disposition: 0x14 (20) -- Issued

Row 29:
  Request ID: 0x1a34c (107340)
  Requester Name: "ORG\MNN41515$"
  Certificate Effective Date: 18.12.2014 14:36
  Certificate Expiration Date: 18.12.2015 14:36
  Request Disposition: 0x14 (20) -- Issued

Maximum Row Index: 29

29 Rows
  145 Row Properties, Total Size = 1340, Max Size = 24, Ave Size = 9
    0 Request Attributes, Total Size = 0, Max Size = 0, Ave Size = 0
    0 Certificate Extensions, Total Size = 0, Max Size = 0, Ave Size = 0
  145 Total Fields, Total Size = 1340, Max Size = 24, Ave Size = 9
CertUtil: -view command completed successfully.

PS C:\_scripts>
> Get-IssuedCert -Date 18.12.2014
```

The following example lists ONLY the 3 certificates which are issued with the Template $WSTemplate (OID of "…- Workstation – Authentication Certificate") beginning December 18. 2014

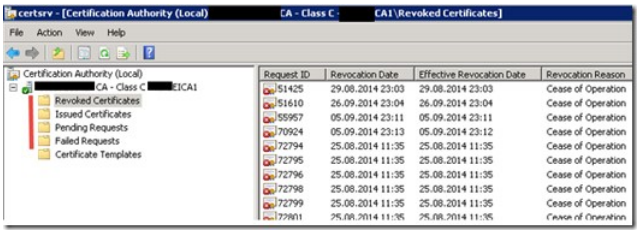
$WSTemplate = (Get-PublishedCATemplate -filter workstation).oid
Get-IssuedCert -CertTemplate $WSTemplate -Date 18.12.2014



C.) Remove-ExpiredCertFromDB

This is an advanced function and all available parameters are displayed with the get-help command

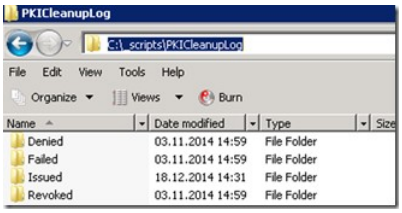
- the expired certificates to view (1st step) and then delete are in one of 4 folders



- I select this "folder" with the –state parameter

```
.DESCRIPTION
   You must choose one of the following 4 types
   - revoked
   - issued
   - failed
   - denied (also in "Failed Requests" folder)

   and also define a date to cleanup all old entries older than that "date"

.PARAMETER state
   type of record you want to delete
   issued | revoked | failed | denied
```

- the script creates a log file (also needed for further parsing!) in a separate folder



these folders are created automatically if they don't exist yet.

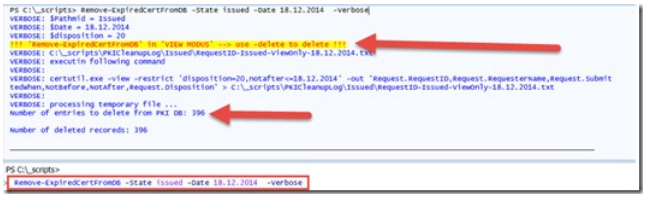
In a first step always run the cmdlet without the "**-delete**" parameter so nothing is really deleted

I also **recommend the ISE** instead of the shell

And I also always run this cmdlet with the "**-verbose**" parameter!

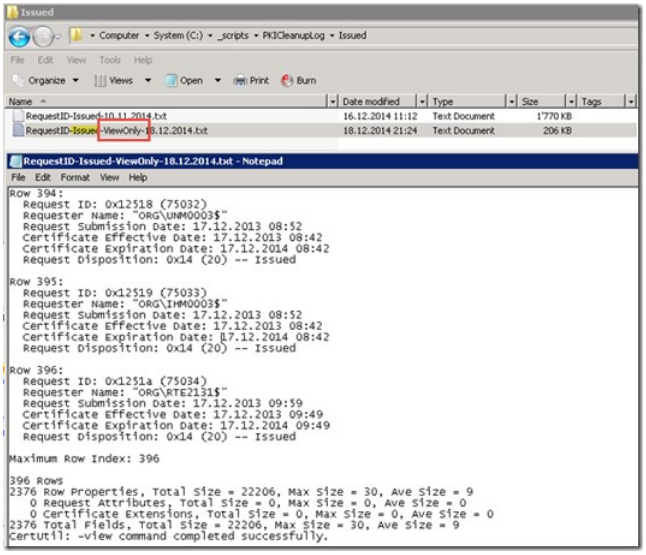
The following example displays all issued (and expired) certificates till 18.12.2014 --- they are not really deleted yet!

Remove-ExpiredCertFromDB -State issued -Date 18.12.2014 -Verbose



without the –delete switch parameter - the log file has "-ViewOnly" in it's name

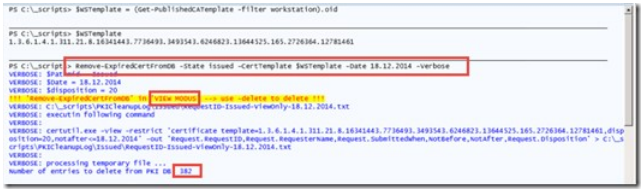
below is the output from the example above / 396 entries "would be" deleted from the "issued folder" (or category) if you run the cmdlet with "-delete"



```
RequestID-Issued-ViewOnly-18.12.2014.txt - Notepad
File  Edit  Format  View  Help
Row 394:
  Request ID: 0x12518 (75032)
  Requester Name: "ORG\UNM0003$"
  Request Submission Date: 17.12.2013 08:52
  Certificate Effective Date: 17.12.2013 08:42
  Certificate Expiration Date: 17.12.2014 08:42
  Request Disposition: 0x14 (20) -- Issued

Row 395:
  Request ID: 0x12519 (75033)
  Requester Name: "ORG\IHM0003$"
  Request Submission Date: 17.12.2013 08:52
  Certificate Effective Date: 17.12.2013 08:42
  Certificate Expiration Date: 17.12.2014 08:42
  Request Disposition: 0x14 (20) -- Issued

Row 396:
  Request ID: 0x1251a (75034)
  Requester Name: "ORG\RTE2131$"
  Request Submission Date: 17.12.2013 09:59
  Certificate Effective Date: 17.12.2013 09:49
  Certificate Expiration Date: 17.12.2014 09:49
  Request Disposition: 0x14 (20) -- Issued

Maximum Row Index: 396

396 Rows
2376 Row Properties, Total Size = 22206, Max Size = 30, Ave Size = 9
   0 Request Attributes, Total Size = 0, Max Size = 0, Ave Size = 0
   0 Certificate Extensions, Total Size = 0, Max Size = 0, Ave Size = 0
2376 Total Fields, Total Size = 22206, Max Size = 30, Ave Size = 9
CertUtil: -view command completed successfully.
```

The following example lists / deletes certificates from a certain (workstation authentication) template expired up to Dec. 18 2014

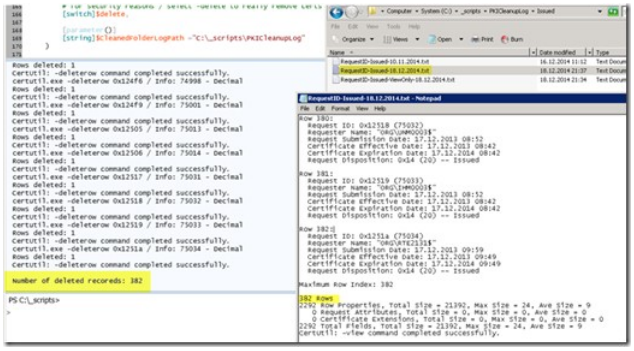
$WSTemplate = (Get-PublishedCATemplate -filter workstation).oid
Remove-ExpiredCertFromDB -State issued -CertTemplate $WSTemplate -Date 18.12.2014 -Verbose



With the added –delete switch parameter you really delete the entries…this step can take some time if there are a lot of entries!

$WSTemplate = (Get-PublishedCATemplate -filter workstation).oid
Remove-ExpiredCertFromDB -State issued -CertTemplate $WSTemplate -Date 18.12.2014 -Verbose ***-delete***

the output at the end (and the log file)



when you run the same cmdlet again, you see that there aren't any entries to delete from the db

The latest (full) version of this script with the 3 functions you can download from the Microsoft Script Gallery: go to download

---