SHA1 Key Migration to SHA256 for a two tier PKI hierarchy

http://blogs.technet.com/b/askds/archive/2015/10/26/sha1-key-migration-to-sha256-for-a-two-tier-pki-hierarchy.aspx

Server Authentication certificates: CAs must begin issuing new certificates using only the SHA-2 algorithm after January 1, 2016. Windows will no longer trust certificates signed with SHA-1 after January 1, 2017.

In this post, I will be following the steps documented here with some modifications: Migrating a Certification Authority Key from a Cryptographic Service Provider (CSP) to a Key Storage Provider (KSP) - https://technet.microsoft.com/en-us/library/dn771627.aspx

The steps that follow in this blog will match the steps in the TechNet article above with the addition of screenshots and additional information that the TechNet article lacks.

Additional recommended reading:

The following blog written by Robert Greene will also be referenced and should be reviewed - http://blogs.technet.com/b/askds/archive/2015/04/01/migrating-your-certification-authority-hashing-algorithm-from-sha1-to-sha2.aspx

This Wiki article written by Roger Grimes should also be reviewed as well - http://social.technet.microsoft.com/wiki/contents/articles/31296.implementing-sha-2-in-active-directory-certificate-services.aspx

Microsoft Trusted Root Certificate: Program Requirements - https://technet.microsoft.com/en-us/library/cc751157.aspx

The scenario for this exercise is as follows:

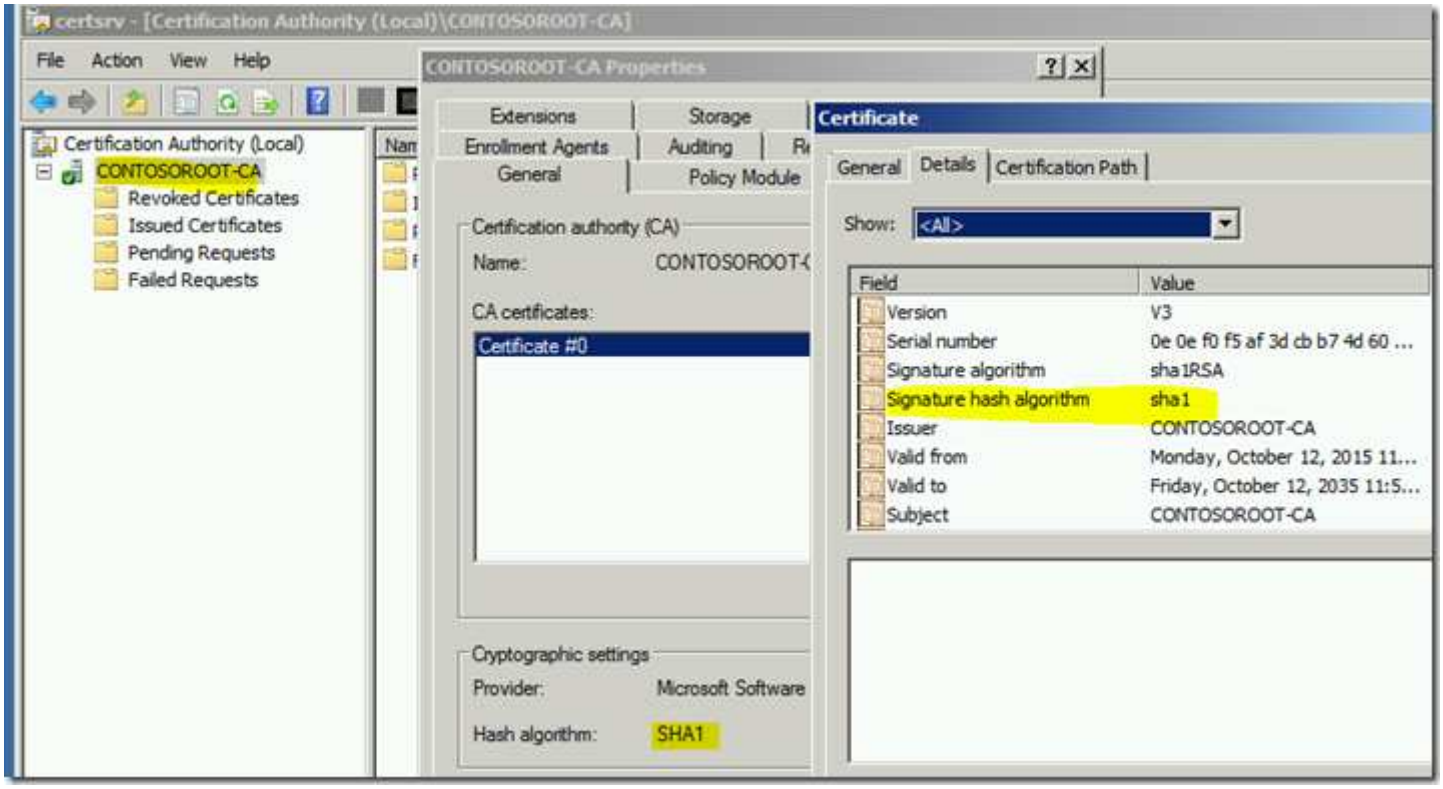
A two tier PKI hierarchy consisting of an Offline ROOT and an Online subordinate enterprise issuing CA.

Operating Systems:

Offline ROOT and Online subordinate are both Windows 2008 R2 SP1

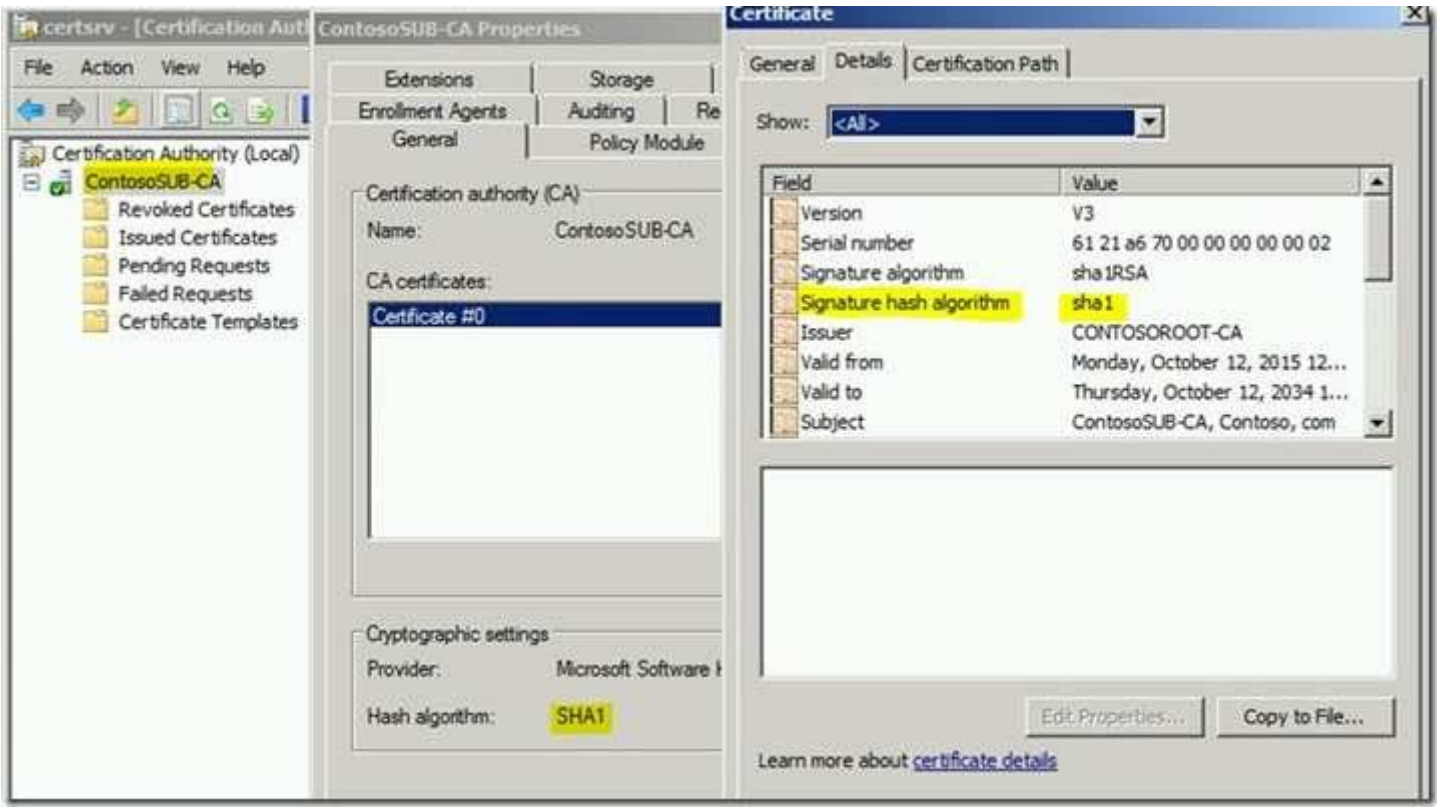
OFFLINE ROOT

CANAME - CONTOSOROOT-CA



ONLINE SUBORDINATE ISSUING CA

CANAME – ContosoSUB-CA



First, you should verify whether your CA is using a Cryptographic Service Provider (CSP) or Key Storage Provider (KSP). This will determine whether you have to go through all the steps or just skip to changing the CA hash algorithm to SHA2. The command for this is in step 3. The line to take note of in the output of this command is “Provider =”. If the **Provider = line** is any of the top five service providers highlighted below, the CA is using a CSP and you must do the conversion steps. The RSA#Microsoft Software **K**ey **S**torage **P**rovider and everything below it are KSP’s.

Here is sample output of the command - Certutil –store my <Your CA common name>

As you can see, the provider is a CSP.

```
================ Certificate 0 ================
Serial Number: 611065b1000000000002
Issuer: CN=CONTOSOROOT-CA
 NotBefore: 10/12/2015 3:06 PM
 NotAfter: 10/12/2034 3:16 PM
Subject: CN=CONTOSOSUB-CA, DC=Contoso, DC=com
CA Version: V0.0
Certificate Template Name (Certificate Type): SubCA
Non-root Certificate
Template: SubCA, Subordinate Certification Authority
Cert Hash(sha1): 66 05 68 98 9b 9d 7a d9 cd 45 29 38 d0 5a 3f 58 65 b0 8c 84
  Key Container = CONTOSOSUB-CA
  Unique container name: bc7d8d0361c8121c7e2a6fa4c58b2cdb_6a5e786f-23f5-46c1-80be-7bc667467eed
  Provider = Microsoft Enhanced Cryptographic Provider v1.0
Signature test passed
CertUtil: -store command completed successfully.
```

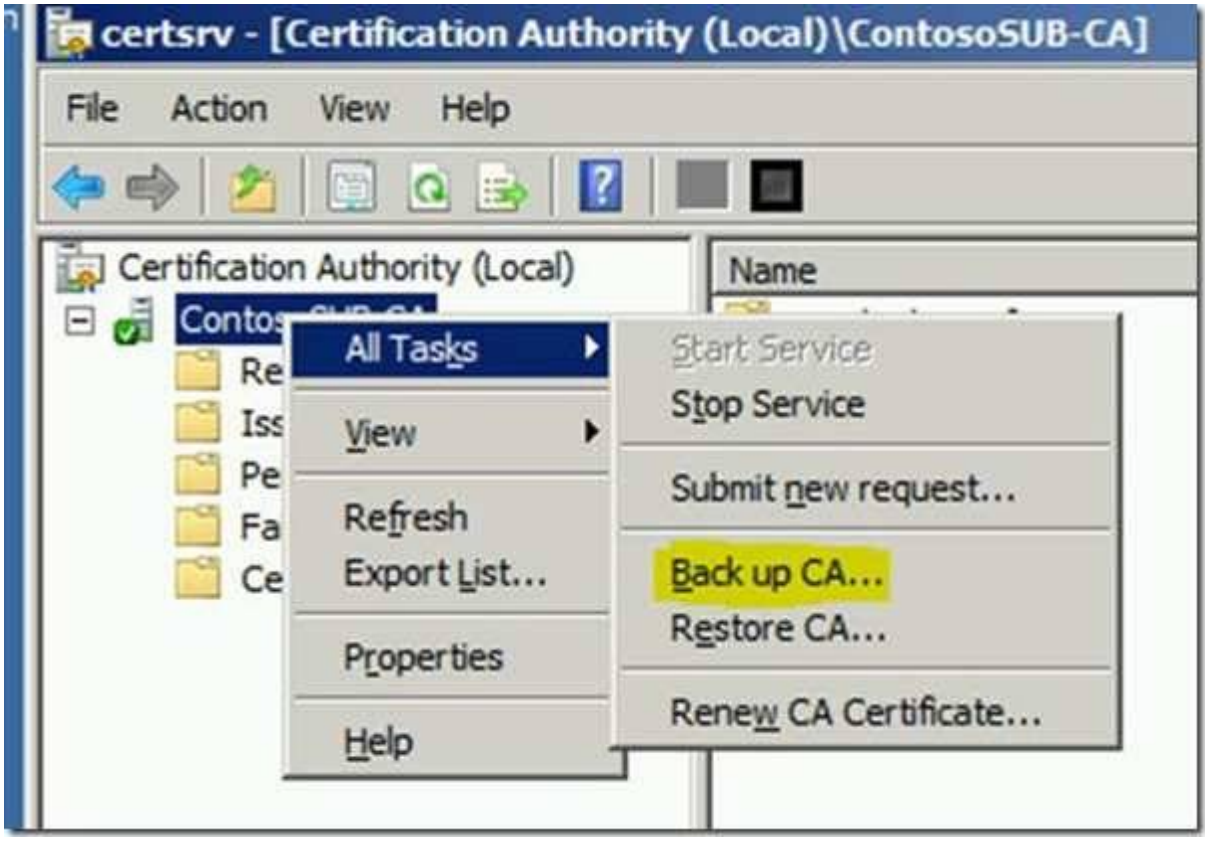
If you are using a Hardware Storage Module (HSM) you should contact your HSM vendor for special guidance on migrating from a CSP to a KSP. The steps for changing the Hashing algorithm to a SHA2 algorithm would still be the same for HSM based CA's.

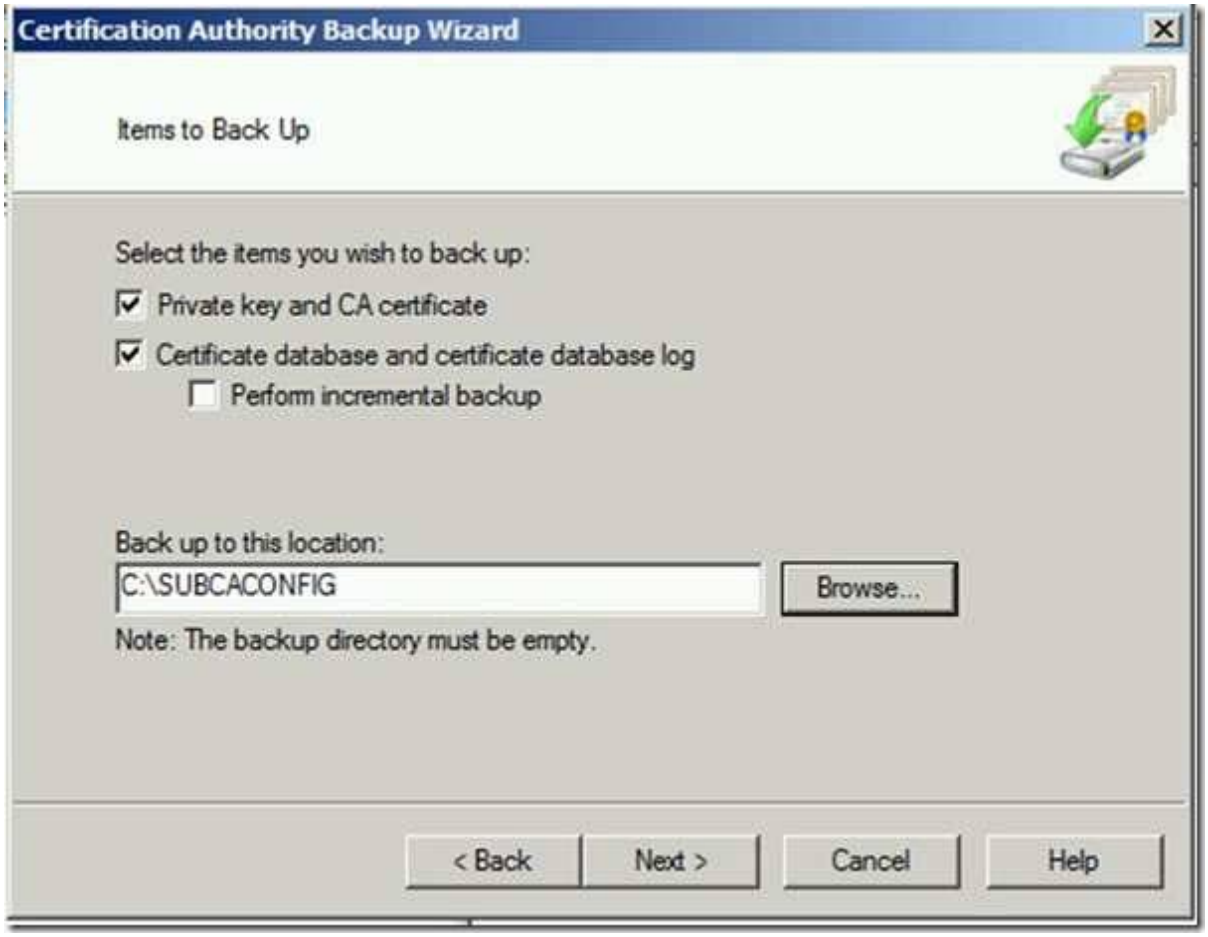
There are some customers that use their HSM for the CA private / public key, but use Microsoft CSP's for the Encryption CSP (used for the CA Exchange certificate).

We will begin at the OFFLINE ROOT.

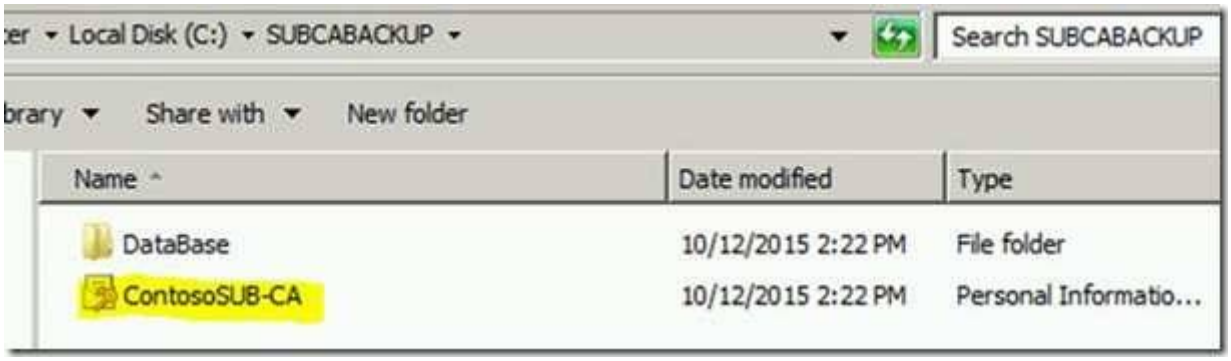
BACKUP! BACKUP! BACKUP the CA and Private KEY of both the OFFLINE ROOT and Online issuing CA. If you have more than one CA Certificate (you have renewed multiple times), all of them will need to be backed up.

Use the MMC to backup the private key or use the CERTSRV.msc and right click the CA name to backup as follows on both the online subordinate issuing and the OFFLINE ROOT CA's –





Provide a password for the private key file.



You may also backup the registry location as indicated in step 1C.

Step 2– Stop the CA Service

Step 3- This command was discussed earlier to determine the provider.

- Certutil –store my ***<Your CA common name>***

Step 4 and Step 6 from the above referenced TechNet articleshould be done via the UI.

a. Open the MMC - load the Certificates snapin for the LOCAL COMPUTER

b. Right click each CA certificate (If you have more than 1) - export

c. Yes, export the private key

d. Check - Include all certificates in the certification path if possible

e. Check - Delete the private key if the export is successful

f. Click next and continue with the export.

Step 5

Copy the resultant .pfx file to a Windows 8 or Windows Server 2012 computer

Conversion requires a Windows Server 2012 certutil.exe, as Windows Server 2008 (and prior) do not support the necessary KSP conversion commands. If you want to convert a CA certificate on an ADCS version prior to Windows Server 2012, you must export the CA certificate off of the CA, import onto Windows Server 2012 or later using certutil.exe with the -KSP option, then export the newly signed certificate as a PFX file, and re-import on the original server.

Run the command in Step 5 on the Windows 8 or Windows Server 2012 computer.

- Certutil –csp ***<KSP name>*** -importpfx ***<Your CA cert/key PFX file>***

Step 6

a. To be done on the Windows 8 or Windows Server 2012 computer as previously indicated using the MMC.

b. Open the MMC - load the Certificates snapin for the LOCAL COMPUTER

c. Right click the CA certificate you just imported – All Tasks – export

*I have seen an issue where the “Yes, export the private key” is dimmed after running the conversion command and trying to export via the MMC. If you encounter this behavior, simply reimport the .PFX file manually and check the box **Mark this key as exportable** during the import**.** This will not affect the previous conversion.

d. Yes, export the private key.

e. Check - Include all certificates in the certification path if possible

f. Check - Delete the private key if the export is successful

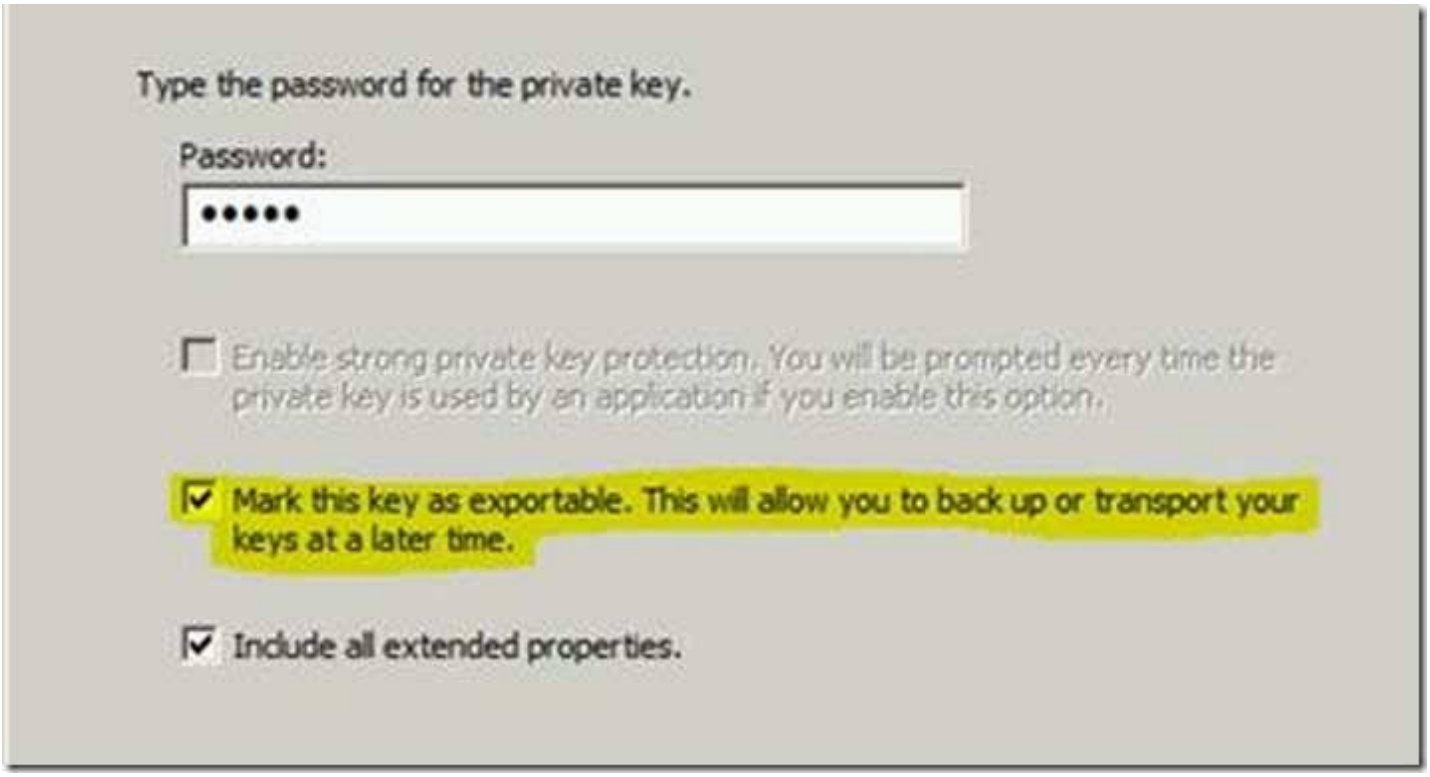
g. Click next and continue with the export.

h. Copy the resultant .pfx file back to the destination 2008 R2 ROOTCA

Step 7

You can again use the UI (MMC) to import the .pfx back to the computer store on the ROOTCA

***Don’t forget during the import to Mark this key as exportable.**



*****IMPORTANT*****

If you have renewed you CA multiple times with the same key, after exporting the first CA certificate as indicated above in step 4 and step 6, you are breaking the private key association with the previously renewed CA certificates. This is because you are deleting the private key upon successful export. After doing the conversion and importing the resultant .pfx file on the CA (remembering to mark the private key as exportable), you must run the following command from an elevated command prompt for each of the additional CA certificates that were renewed previously:

certutil –repairstore MY serialnumber

The Serial number is found on the details tab of the CA certificate. This will repair the association of the public certificate to the private key.

Step 8–

Your CSP.reg file must contain the information highlighted at the top –

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration\CONTOSOROOT-CA\CSP]
"ProviderType"=dword:00000000
"Provider"="Microsoft Software Key Storage Provider"
"CNGPublicKeyAlgorithm"="RSA"
"CNGHashAlgorithm"="SHA1"
```

Step 8c

```
C:\CACONFIG>certutil -v -getreg ca\csp\hashalgorithm
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration\CONTO
SOROOT-CA\csp:

  HashAlgorithm REG_DWORD = 8004 (32772)
    CALG_SHA1
    Algorithm Class: 0x8000(4) ALG_CLASS_HASH
    Algorithm Type: 0x0(0) ALG_TYPE_ANY
    Algorithm Sub-id: 0x4(4) ALG_SID_SHA1
CertUtil: -getreg command completed successfully.
```

Step 8d– Run CSP.reg

Step 9

Your EncryptionCSP.reg file must contain the information highlighted at the top –

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration\CONTOSOROOT-CA\EncryptionCSP]
"ProviderType"=dword:00000000
"Provider"="Microsoft Software Key Storage Provider"
"CNGPublicKeyAlgorithm"="RSA"
"CNGEncryptionAlgorithm"="3DES"
"MachineKeyset"=dword:00000001
"SymmetricKeySize"=dword:000000a8
```

Step 9c– verification - certutil -v -getreg ca\encryptioncsp\EncryptionAlgorithm

Step 9d– Run EncryptionCsp.reg

Step 10

Change the CA hash algorithm to SHA256

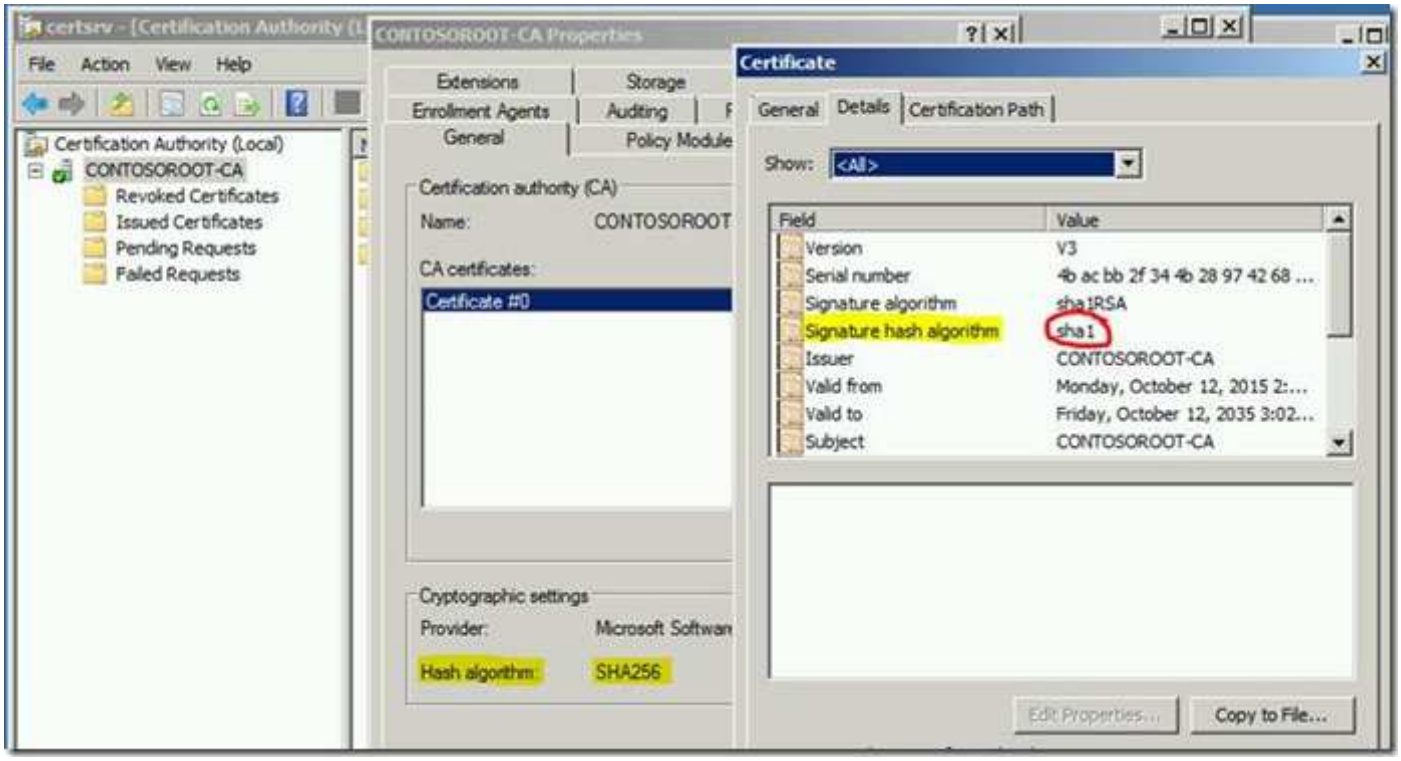
```
C:\CACONFIG>certutil -setreg ca\csp\CNGHashAlgorithm SHA256
SYSTEM\CurrentControlSet\Services\CertSvc\Configuration\CONTOSOROOT-CA\csp:

Old Value:
  CNGHashAlgorithm REG_SZ = SHA1

New Value:
  CNGHashAlgorithm REG_SZ = SHA256
CertUtil: -setreg command completed successfully.
The CertSvc service may need to be restarted for changes to take effect.
```

Start the CA Service

Step 11

For a root CA: You will not see the migration take effect for the CA certificate itself until you complete the migration of the root CA, and then renew the certificate for the root CA.

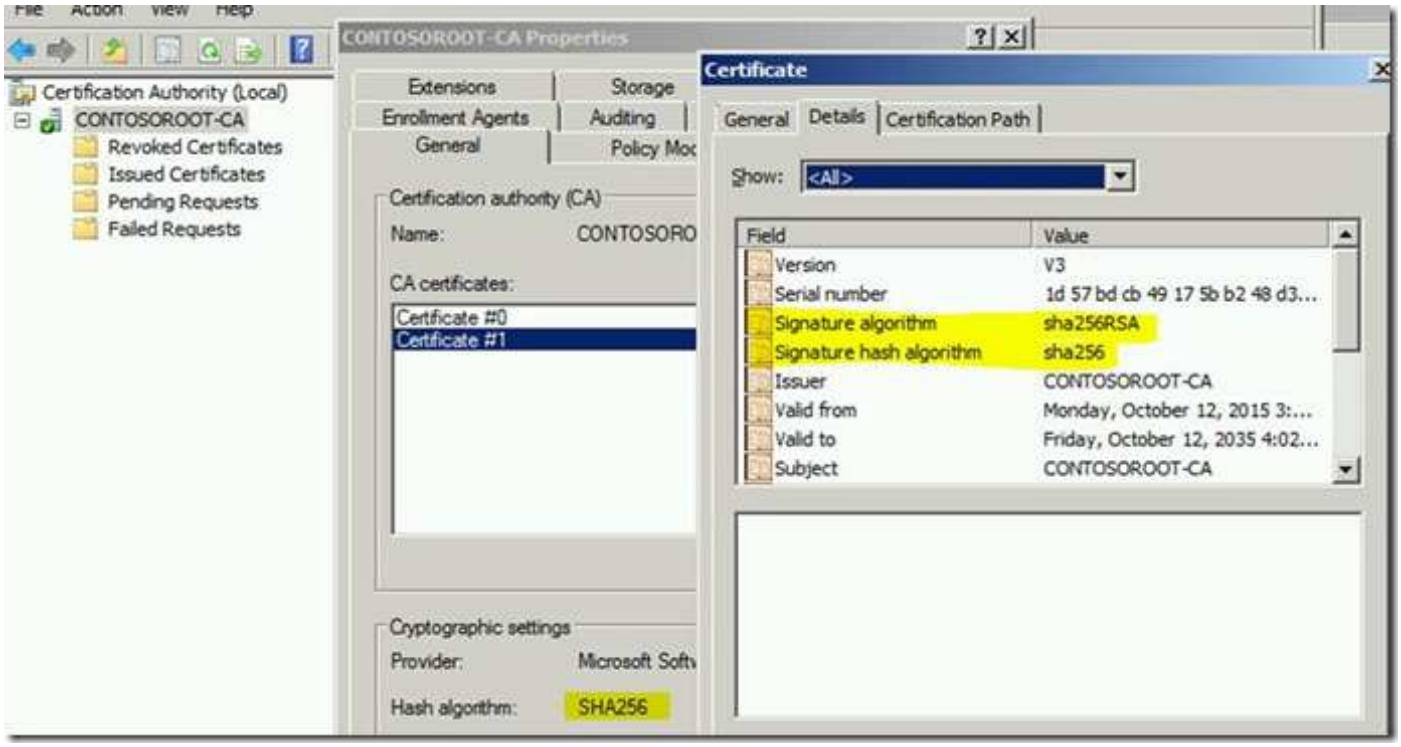
Before we renew the OFFLINE ROOT certificate this is how it looks:



Renewing the CA's own certificate with a new or existing (same) key would depend on the remaining validity of the certificate. If the certificate is at or nearing 50% of its lifetime, it would be a good idea to renew with a new key. See the following for additional information on CA certificate renewal –

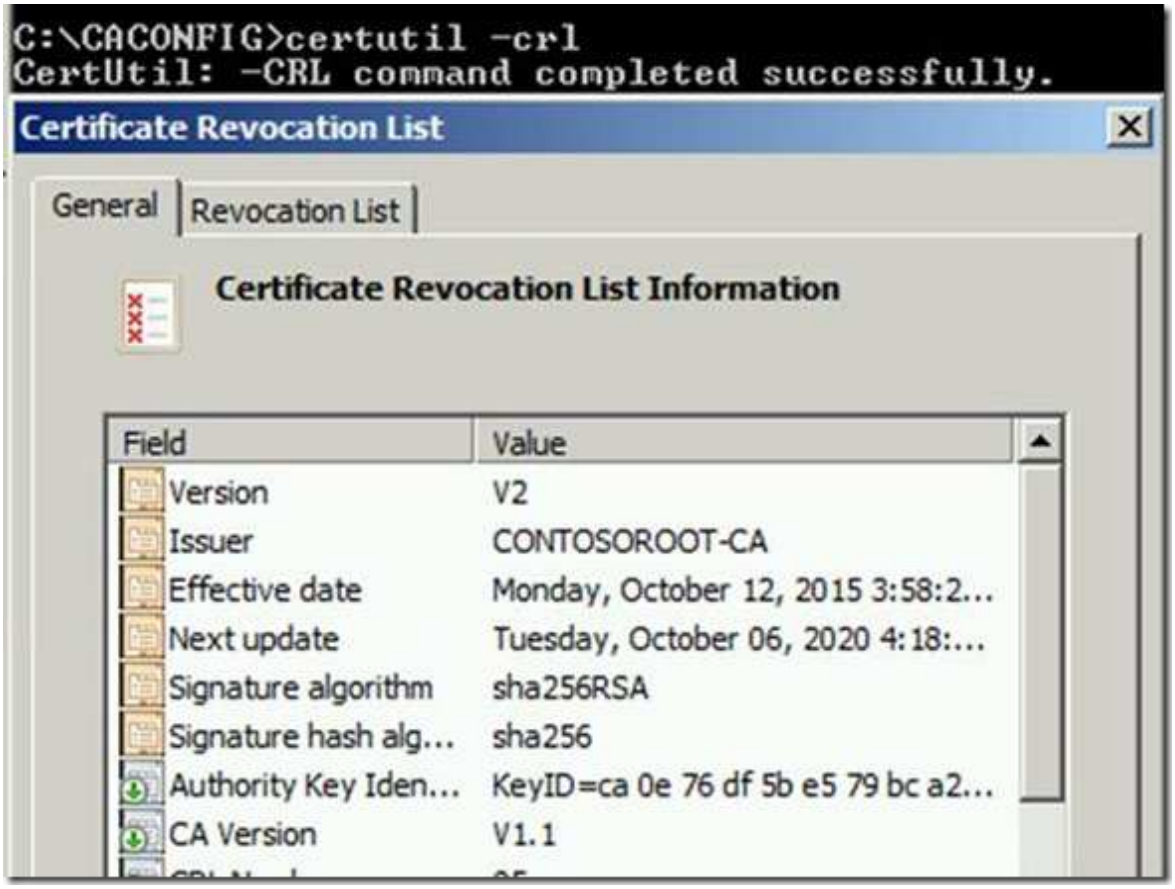
https://technet.microsoft.com/en-us/library/cc730605.aspx

After we renew the OFFLINE ROOT certificate with a new key or the same key, its own Certificate will be signed with the SHA256 signature as indicated in the screenshot below:



Your OFFLINE ROOT CA is now completely configured for SHA256.

Running CERTUTIL –CRL will generate a new CRL file also signed using SHA256



By default, CRT, CRL and delta CRL files are published on the CA in the following location - *%SystemRoot%\System32\CertSrv\CertEnroll*. The format of the CRL file name is the "sanitized name" of the CA plus, in parentheses, the "key id" of the CA (if the CA certificate has been renewed with a new key) and a .CRL extension. See the following for more information on CRL distribution points and the CRL file name - https://technet.microsoft.com/en-us/library/cc782162%28v=ws.10%29.aspx

Copy this new .CRL file to a domain joined computer and publish it to Active Directory while logged on as an Enterprise Administrator from an elevated command prompt.

Do the same for the new SHA256 ROOT CA certificate.

- certutil -f -dspublish ***<.CRT file>*** RootCA
- certutil –f -dspublish ***<.CRL file>***

Now continue with the migration of the Online Issuing Subordinate CA.

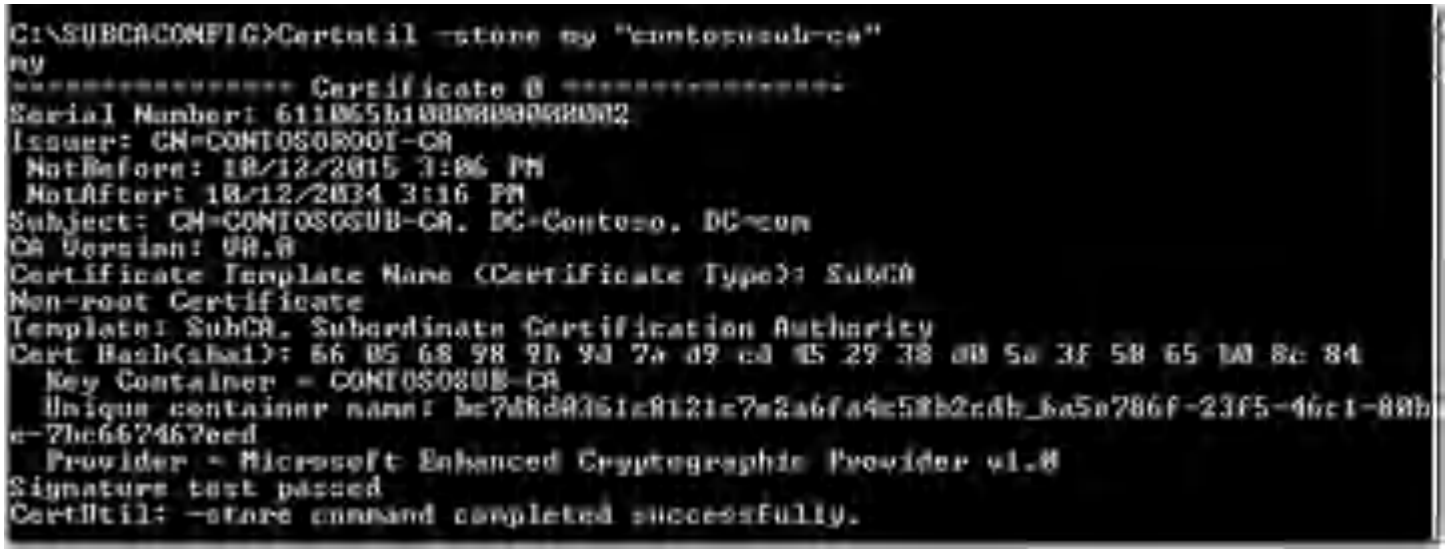
Step 1– Backup the CA database and Private Key.

Backup the CA registry settings

Step 2– Stop the CA Service.

Step 3- Get the details of your CA certificates

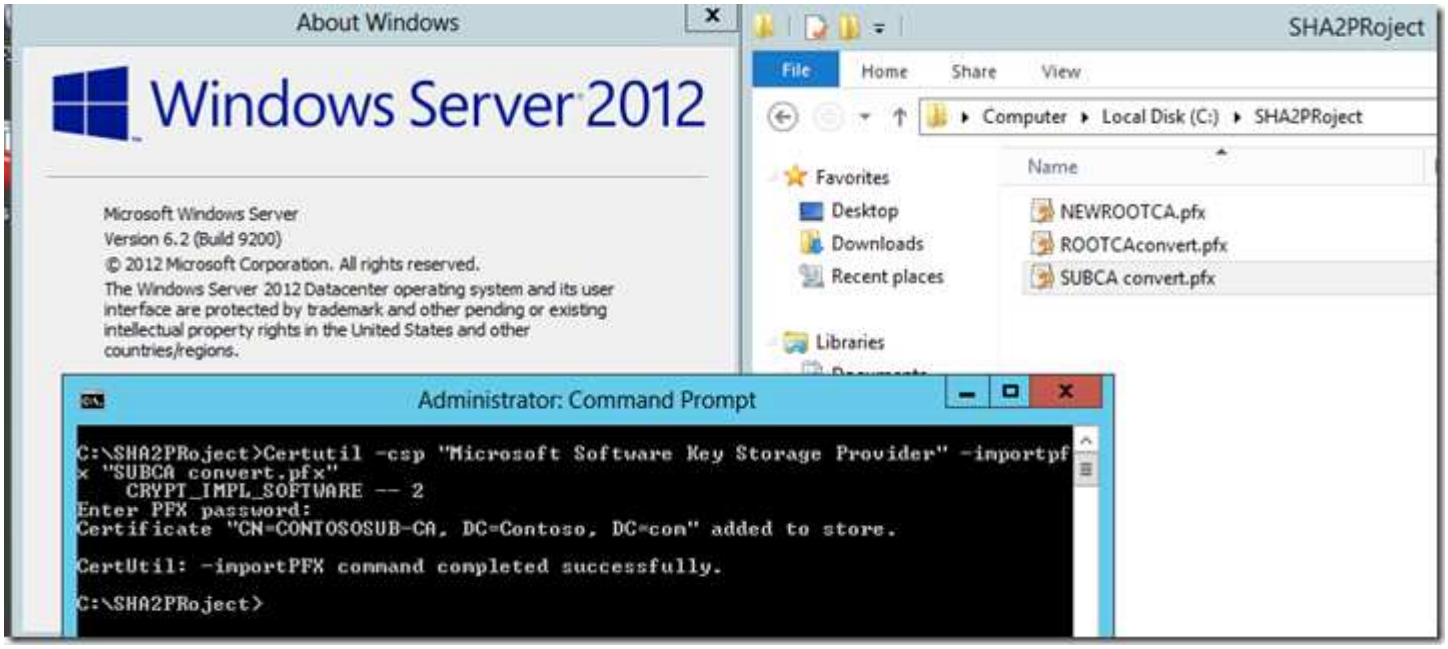
Certutil –store my ***"Your SubCA name"***



I have never renewed the Subordinate CA certificate so there is only one.

Step 4 – 6

As you know from what was previously accomplished with the OFFLINE ROOT, steps 4-6 are done via the MMC and we must do the conversion on a Windows 8 or Windows 2012 or later computer for reasons explained earlier.

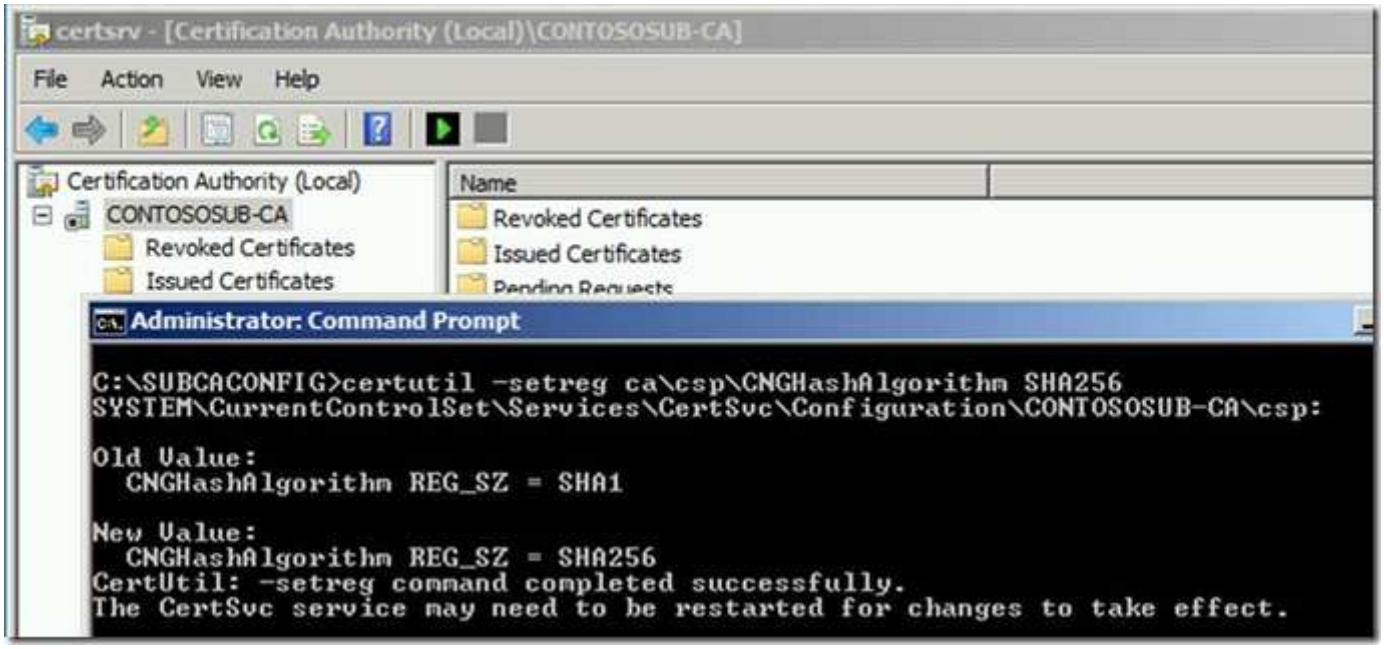


***When you import the converted SUBCA .pfx file via the MMC, you must remember to again Mark this key as exportable.**

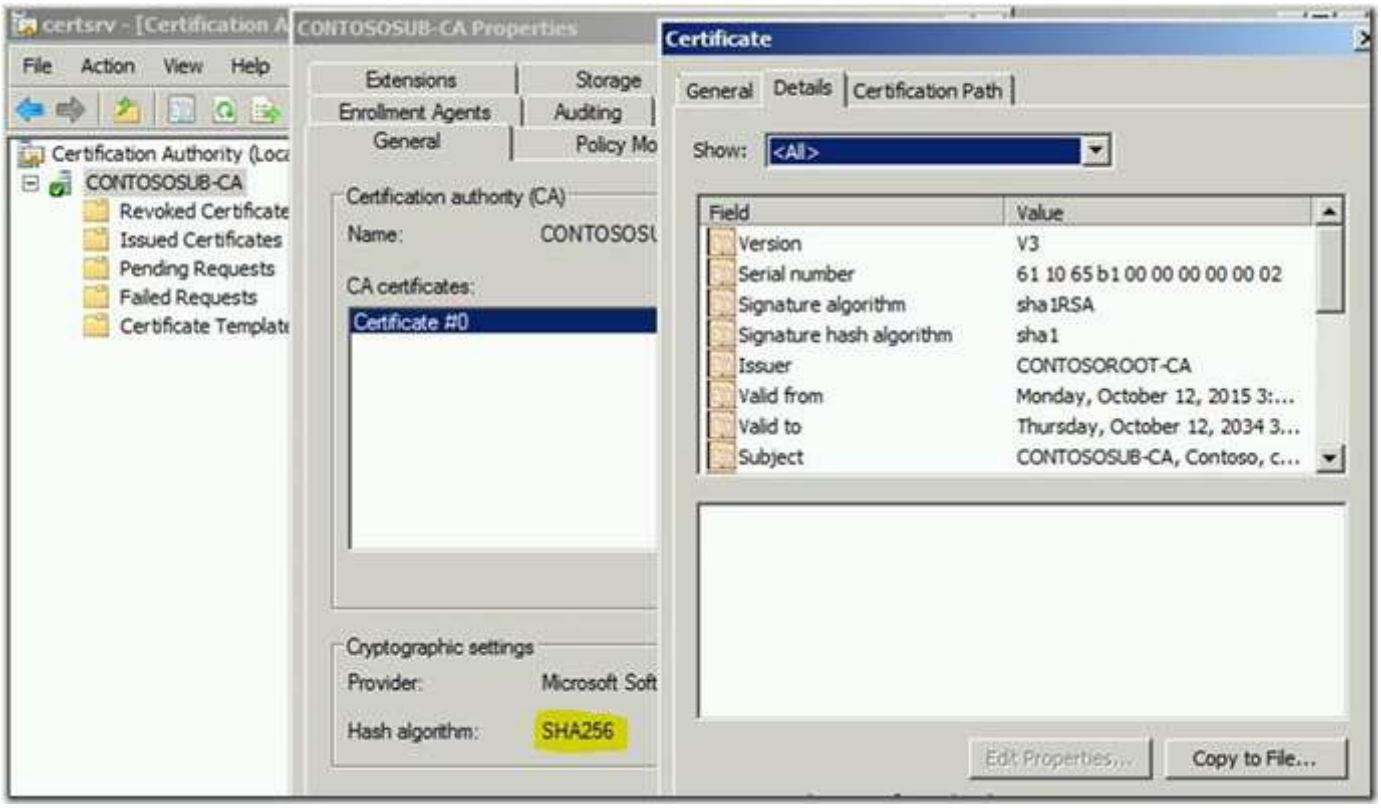
Step 8 – Step 9

Creating and importing the registry files for CSP and CSP Encryption (see above)

Step 10- Change the CA hash algorithm to SHA-2



Now in the screenshot below you can see the Hash Algorithm is SHA256.



The Subordinate CA's own certificate is still SHA1. In order to change this to SHA256 you must renew the Subordinate CA's certificate. When you renew the Subordinate CA's certificate it will be signed with SHA256. This is because we previously changed the hash algorithm on the OFFLINE ROOT to SHA256.

Renew the Subordinate CA's certificate following the proper steps for creating the request and submitting it to the OFFLINE ROOT. Information on whether to renew with a new key or the same key was provided earlier. Then you will copy the resultant .CER file back to the Subordinate CA and install it via the Certification Authority management interface.

If you receive the following error when installing the new CA certificate –

Check the newly procured Subordinate CA certificate via the MMC. On the certification path tab, it will indicate under certificate status that – "The signature of the certificate cannot be verified"

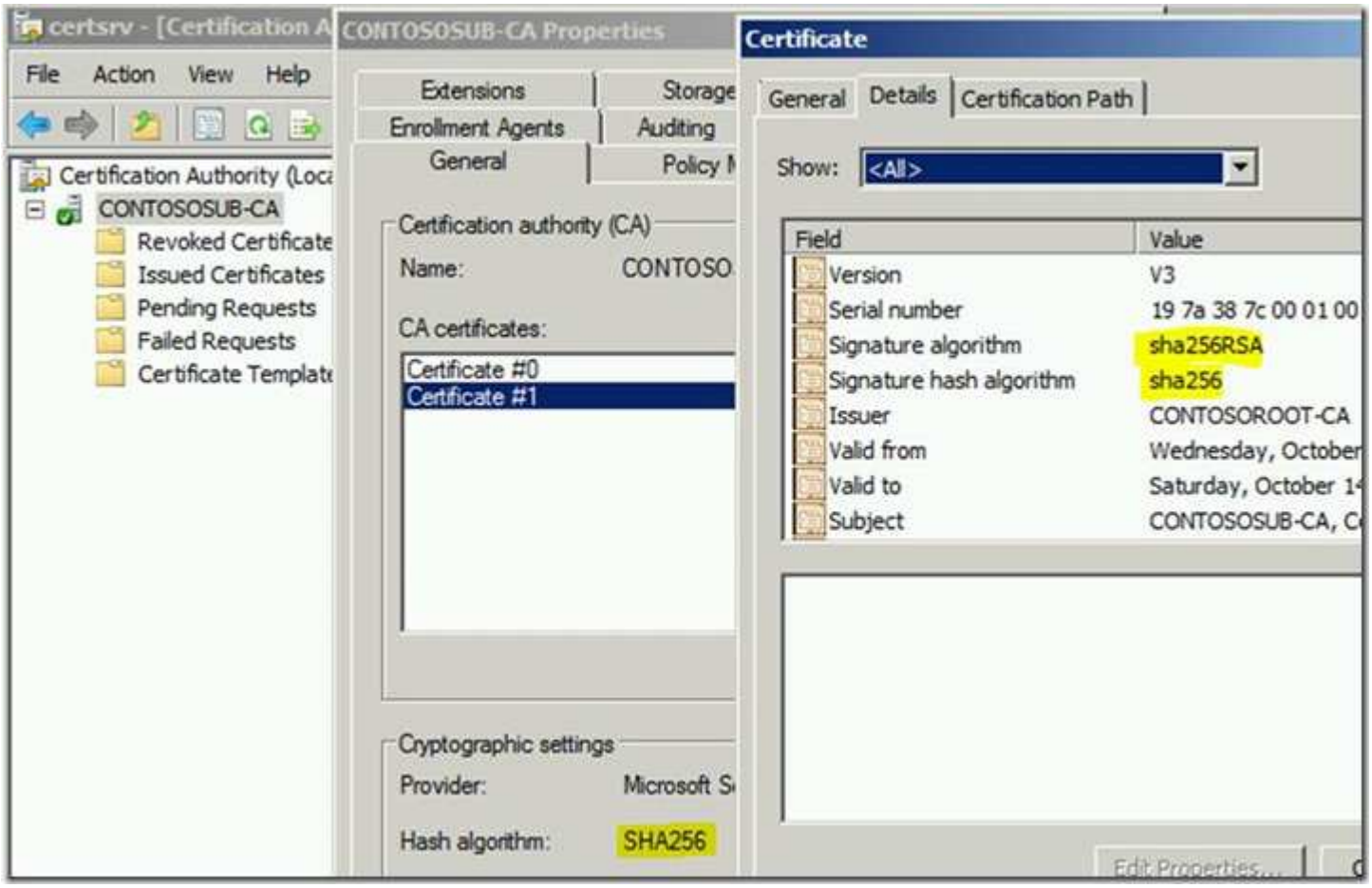
This error could have several causes. You did not –dspublish the new OFFLINE ROOT .CRT file and .CRL file to Active Directory as previously instructed.

```
C:\ROOTCA>certutil -f -dspublish contosoroot_contosoroot-CA(1).crt RootCA
ldap:///CN=CONTOSOROOT-CA,CN=Certification Authorities,CN=Public Key Services,CN
=Services,CN=Configuration,DC=Contoso,DC=com?cACertificate

Certificate added to DS store.

ldap:///CN=CONTOSOROOT-CA,CN=AIA,CN=Public Key Services,CN=Services,CN=Configura
tion,DC=Contoso,DC=com?cACertificate

Certificate added to DS store.

CertUtil: -dsPublish command completed successfully.

C:\ROOTCA>certutil -dspublish contosoroot-CA(1).crl
ldap:///CN=CONTOSOROOT-CA(1),CN=CDP,CN=Public Key Services,CN=Services,CN=Config
uration,DC=Contoso,DC=com?certificateRevocationList?base?objectClass=cRLDistribu
tionPoint?certificateRevocationList

Base CRL added to DS store.

CertUtil: -dsPublish command completed successfully.

C:\ROOTCA>gpupdate /force
Updating Policy...

User Policy update has completed successfully.
Computer Policy update has completed successfully.
```

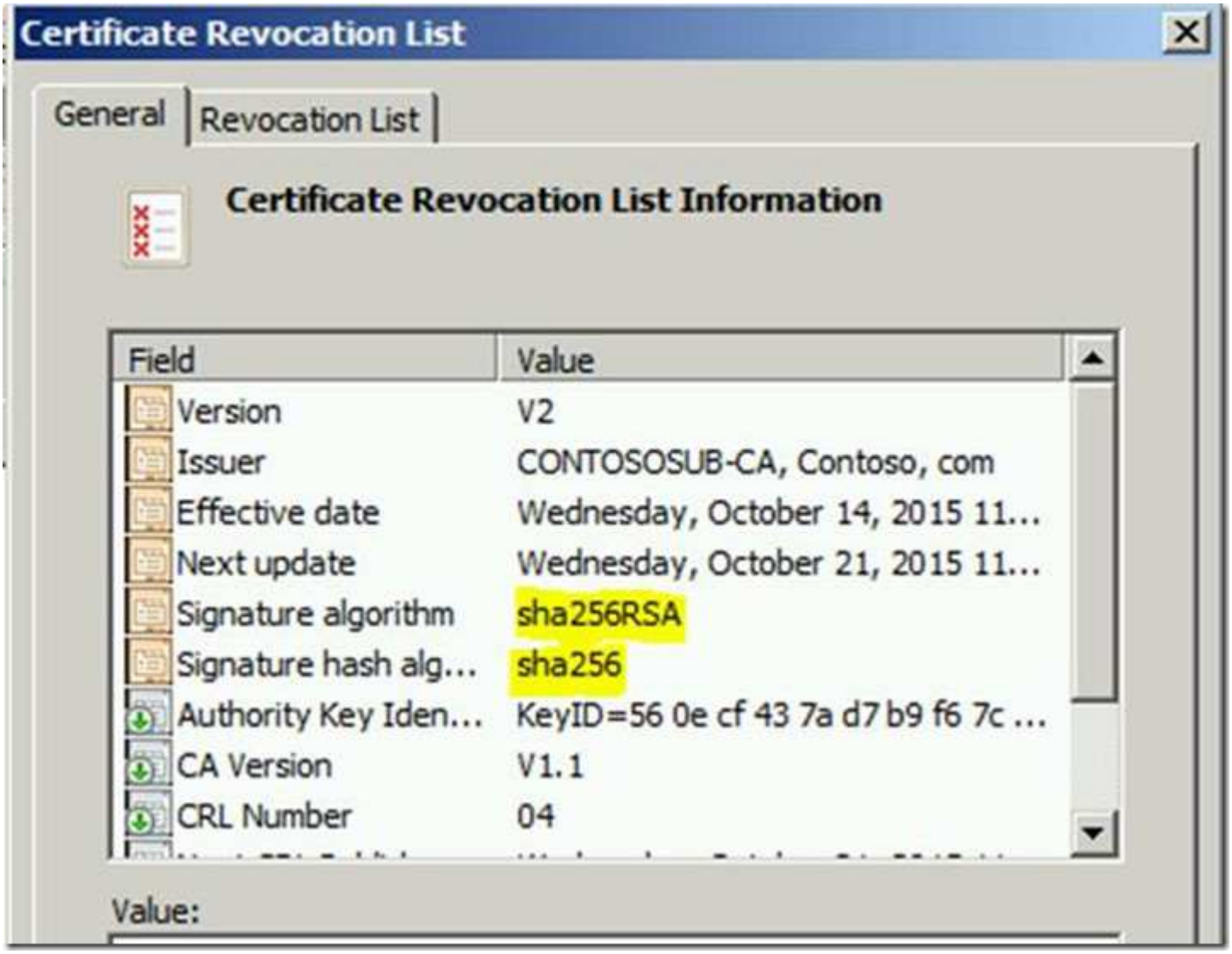
Or you did publish the Root CA certificate but the Subordinate CA has not done Autoenrollment (AE) yet and therefore has not downloaded the "NEW" Root CA certificate via AE methods, or AE may be disabled on the CA all together.

After the files are published to AD and after verification of AE and group policy updates on the Subordinate CA, the install and subsequent starting of Certificate Services will succeed.

Now in addition to the Hash Algorithm being SHA256 on the Subordinate CA, the Signature on its own certificate will also be SHA256.



The Subordinate CA's .CRL files are also now signed with SHA256 –



Your migration to SHA256 on the Subordinate CA is now completed.