Certificate Issues

IDERA users in environments that have not yet added a certificate signed by a Certification Authority (CA) receive a warning message in their browser each time they attempt to open the SSL version of SQLDM Web Console or Idera Dashboard. To access SQL Diagnostic Manager over HTTPS with a self-signed certificate you may need to enable SSL on the SQL Diagnostic Manager Rest service and add a certificate.

Create a self-signed certificate

To access SQL Diagnostic Manager over HTTPS, you should add a certificate for SQLDM Web Console following the steps below:

1. Launch Windows Powershell as administrator.
2. Create your certificate by running the following command. Leave the PowerShell console session open.

```
$certName = "{certificateName}"         ## Replace {certificateName}
```

> Replace **{certificateName}** with the name that you will use to access the SQLDM Web Console. For example, if you are using the `https://ComputerName:9295` link to access SQLDM Web Console, then use **ComputerName** . In case, you are using the `https://ComputerName.Domain.com:9295` address then use **Computername.Domain.com** .

3. Run the following command to configure your certificate settings.

```
$Params = @{
     "DnsName"           = @($certName,"{Param1}", "{Param2}")   ## If you want to include other
addresses or servers, you must separate each with a comma
     "CertStoreLocation" = "Cert:LocalMachine\My"
     "KeyExportPolicy"   = "Exportable"
     "KeySpec"           = "Signature"
     "KeyUsage"          = @("KeyEncipherment","DigitalSignature")
     "KeyAlgorithm"      = "RSA"
     "KeyLength"         = "2048"
     "HashAlgorithm"     = "SHA256"
     "NotAfter"          = (Get-Date).AddYears(10)
 }

 ## Checks for asterisks in the $certName and replaces it with the underscore character
 If ($certName.Contains("*")) {
     $certName = $certName -replace '\*','_'
 }
```

> Replace **{Param1}** and **{Param2}** with the servers o addresses of your preference. For example `www.mywebsite.com`, `my website`, or `mywebsite.com`
> It is not mandatory to add more than one parameter for the `DnsName`.

> Change the `NotAfter` parameter value to make your certificate valid for a more extended period.

4. Run the command below to create your certificate defined with the parameters above.

```
$cert = New-SelfSignedCertificate @Params
```

Export your certificate private key

Once the certificate is created, you need to export the certificate's private key. To do so, follow the steps below:

1. Export your certificate in .cer format by running the following command.

```
 Export-Certificate -Cert $cert -FilePath "{DesiredPath}\$certname.cer"   ## Replace {DesiredPath}
with the desired location e.g. C:\Users\Public\Documents
```

Once the certificate is created, you should be able to check the certificate specifications.

```
PS C:\Windows\system32> Export-Certificate -Cert $cert -FilePath "C:\Users\Public\Documents\$certname.cer"


    Directory: C:\Users\Public\Documents


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         4/5/2023   6:50 PM            826 windev2302eval.cer
```

2. Create a password for your certificate private key and save it in a variable. Replace `{myPassword}` with the password that you wish to use to protect your certificate's private key.

```
$mypwd = ConvertTo-SecureString -String "{myPassword}" -Force -AsPlainText  ## Replace {myPassword}
```

3. Run the next command to export your private key, use the password you store in the `$mypwd` variable.

```
Export-PfxCertificate -Cert $cert -FilePath "{DesiredPath}\$certname.pfx" -Password $mypwd   ##
Replace {DesiredPath} with your desired location e.g. C:\Users\Public\Documents
```

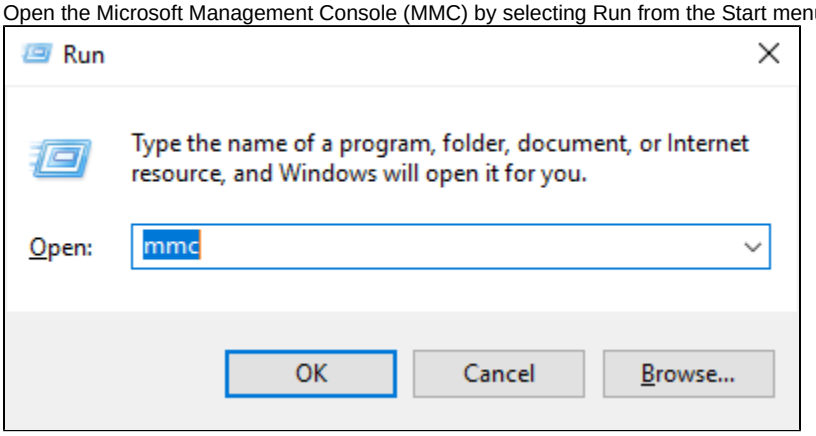
When the private key is exported in a .pfx file, you should be able to check the certificate specifications.

```
PS C:\Windows\system32> $mypwd = ConvertTo-SecureString -String "password" -Force -AsPlainText
PS C:\Windows\system32>  Export-PfxCertificate -Cert $cert -FilePath "C:\Users\Public\Documents\$certname.pfx" -Password $mypwd


    Directory: C:\Users\Public\Documents


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         4/5/2023   6:54 PM           2675 windev2302eval.pfx
```

Import your certificate private key into the Trusted Root Certification Authorities

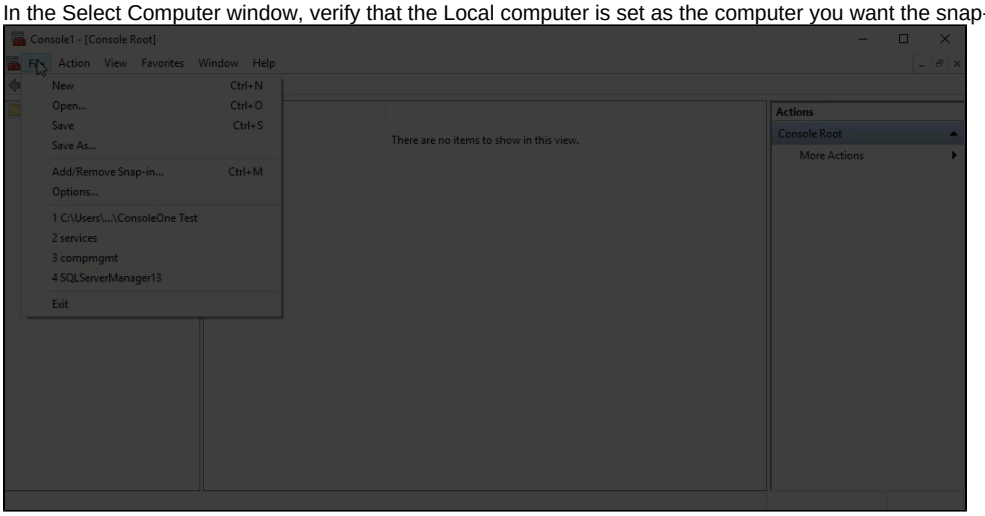
Complete your certificate configuration by adding the `.cer` file to the Trusted Root Certification Authorities folder in the Console Root. To do so, follow the steps below.

1. Open the Microsoft Management Console (MMC) by selecting Run from the Start menu, type "mmc", and click **OK**.

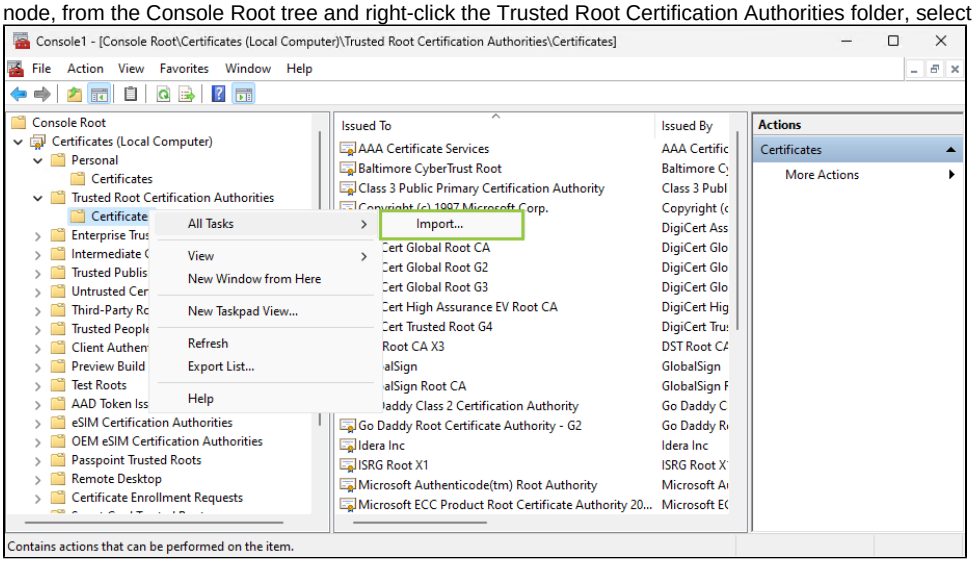


2. When the MMC window opens, click **File** from the menu toolbar, and select **Add/Remove Snap-in...**
3. The Add or Remove Snap-ins window opens, select Certificates from the Available snap-ins options and click **Add >**.
4. In the Certificates snap-in window, select Computer Account, and click **Next**.

5. In the Select Computer window, verify that the Local computer is set as the computer you want the snap-in to manage. Click **Finish**.



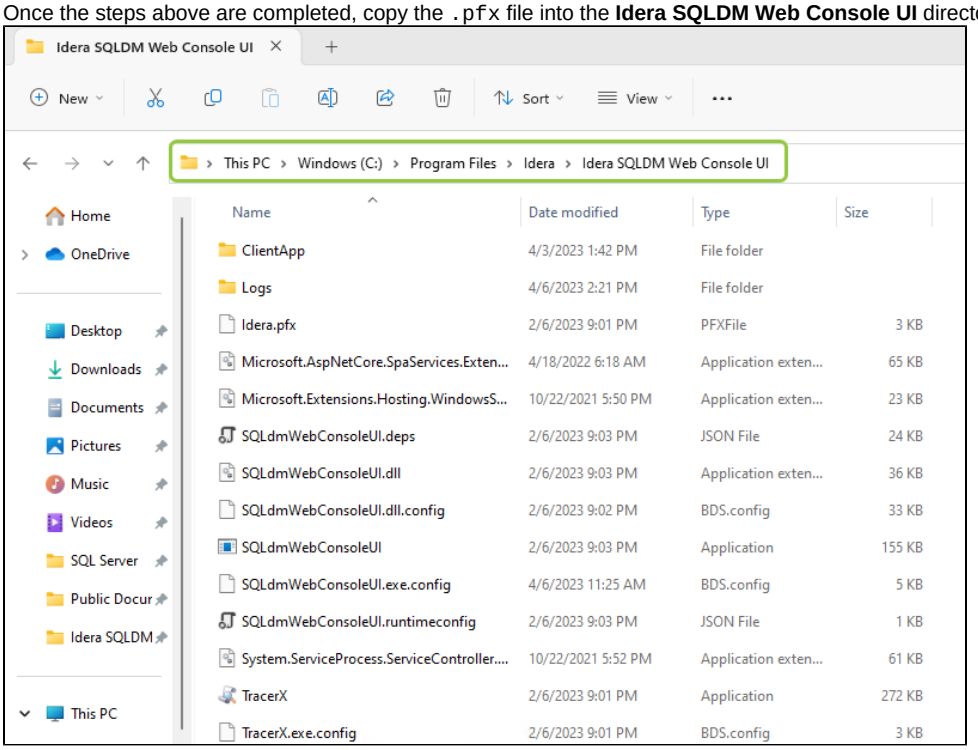
6. Once done, import your certificate (`.cer` file) into the Trusted Root Certification Authorities folder. To do so, expand the Certificates node, from the Console Root tree and right-click the Trusted Root Certification Authorities folder, select **All Tasks**, and click **Import...**



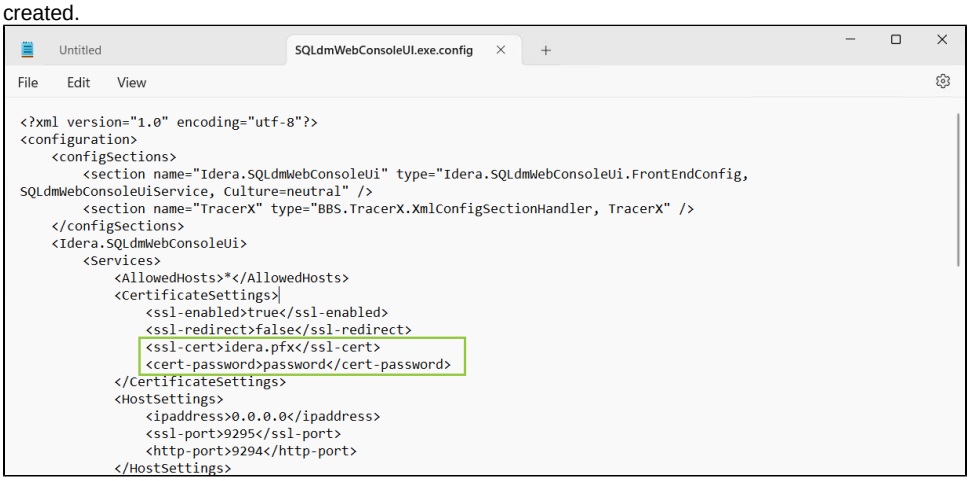
7. When the Certificate Import Wizard opens, follow the instructions to import the `.cer` file previously created.

> When adding your certificate private key using the Certificate Import Wizard, use the password you previously defined in the Export your certificate private key section.

8. Once the steps above are completed, copy the `.pfx` file into the **Idera SQLDM Web Console UI** directory.



9. Open the **SQLdmWebConsoleUI.exe.config** file using any text editor, which should be launched using the **Run as administrator**.
10. Look for the `ssl-cert` and the `cert-password` tags and update them with the name and password of the certificate previously created.



11. Close all the opened browsers.
12. Restart the **Idera SQLDM Web Console UI Service** and you are ready to access your SQLDM Web Console through the following address **`https://<certificateName>:9295/`**

> In case you are working with Idera Dashboard, add a self-signed certificate as you have already completed the steps above, you only have to import the certificate key pair with Key Store Explorer. For more information about it, refer to Resolving the certificate error message.

Binding a certificate to SQL Diagnostic Manager

> ⚠
>
> # Only for IDERA Dashboard
>
> This section is only for users who use the Idera Dashboard Web Console.

After creating a self-signed certificate for Idera Dashboard, follow these instructions to bind a certificate to SQLDM:

- When SQLDM and Dashboard are installed on a local environment Binding a certificate for SQL Diagnostic Manager.
- When SQLDM and Dashboard are installed in different environments Binding certificates for distributed installations.