Resolving the certificate error message

There are multiple ways for you to create a self-signed certificate. The steps in this topic include KeyStore Explorer, a free third-party utility. This product is not supported by IDERA and is only an example.

IDERA Dashboard must be installed prior to performing this task.

In environments that have not yet added a certificate signed by a Certification Authority (CA), IDERA users receive a warning message in their browser each time they attempt to open the SSL version of IDERA Dashboard. Note that the default certificate provided with an IDERA product **is not signed by any well-known CA and is intended only for use in testing purposes ONLY**.

You can resolve this issue by adding a signed CA using the steps provided in Run IDERA Dashboard over TLS (HTTPS), or you can use the following steps to resolve this issue at no certificate cost.

Adding a self-signed certificate

Create a self-signed certificate

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

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

Replace **{certificateName}** with the name that you will use to access the IDERA Dashboard. For example, if you are using the `https://ComputerName:9291` link to access IDERA Dashboard, then use **ComputerName** . In case, you are using the `https://ComputerName.Domain.com:9291` address then use **Computername.Domain.com** . In this example we are using **localhost** .

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

```
$Params = @{
     "DnsName"           = @($certName,"{certificateName}")   ## 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 **{certificateName}** with the certificate name you previously defined in Step 2.

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
```

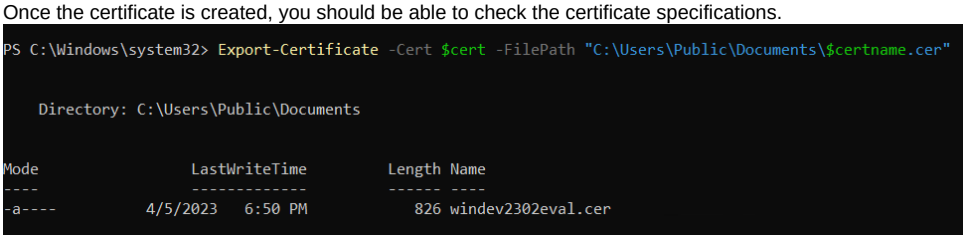
Export your certificate private key

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.



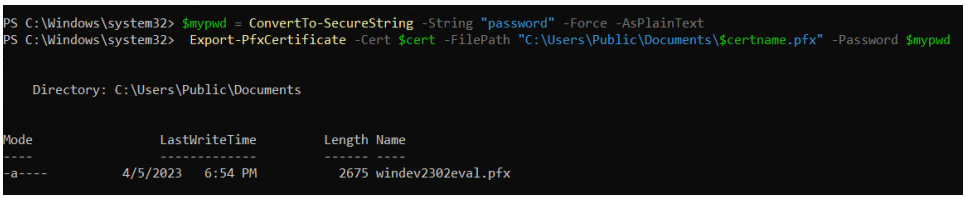
2. Create a password for your certificate private key and save it in a variable. Replace `{myPassword}` with the password that you wish to use to protect your certificate's private key.

```
 $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
```

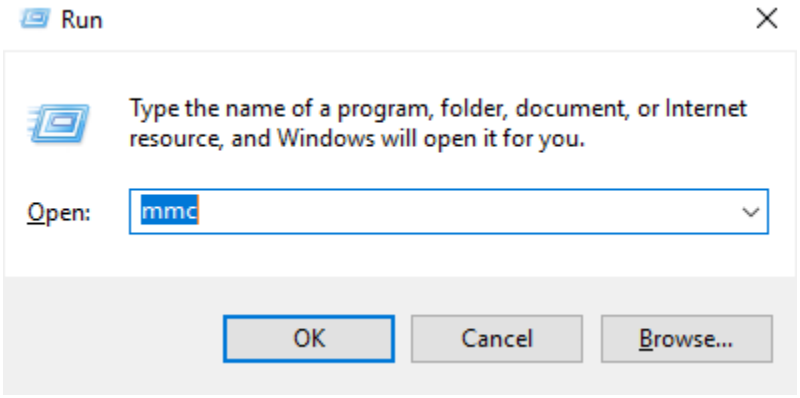
   When the private key is exported in a `.pfx` file, you should be able to check the certificate specifications.



Import your private key into the Trusted Root Certification Authorities

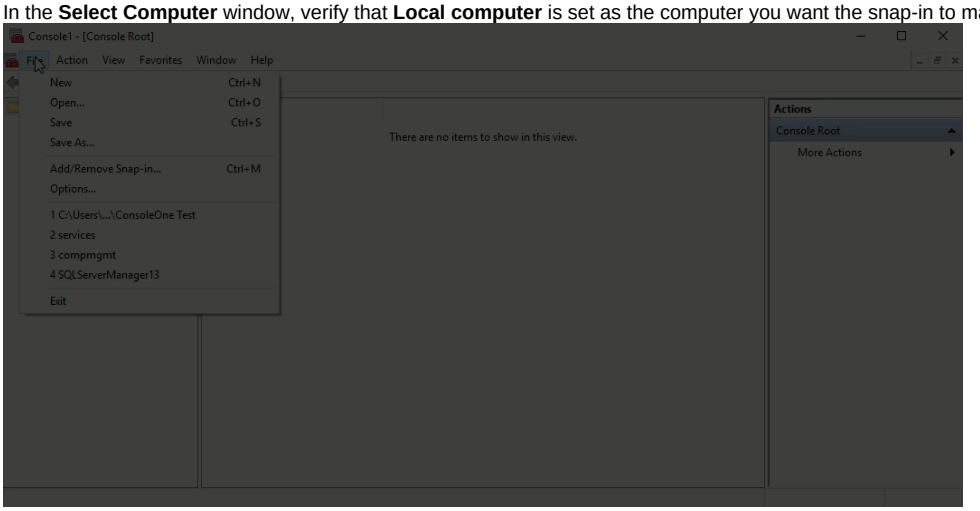
Complete your certificate configuration by adding the `.cer` file to the **Trusted Root Certification Authorities** folder in the **Console Root**. To do so, follow the steps below:

1. Open the **Microsoft Management Console (MMC)** by selecting **Start > Run** and typing **mmc.** Click **OK**.

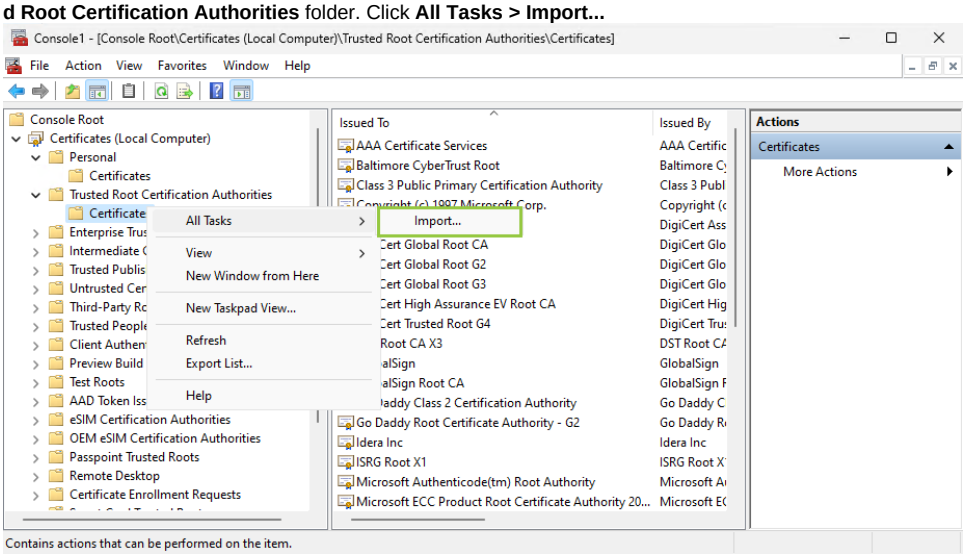


2. When the MCC window opens, click **File** from the menu toolbar, and select **Add/Remove Snap-in...**
3. Select **Certificates** from the **Available snap-ins** options and click **Add >**.
4. In the **Certificates snap-in** window, select **Computer Account**, and click **Next**.

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



6. Import your certificate (`.cer` file) into the **Trusted Root Certification Authorities** folder. To do so, expand **Certificates** and right-click the **Trusted Root Certification Authorities** folder. Click **All Tasks > Import...**



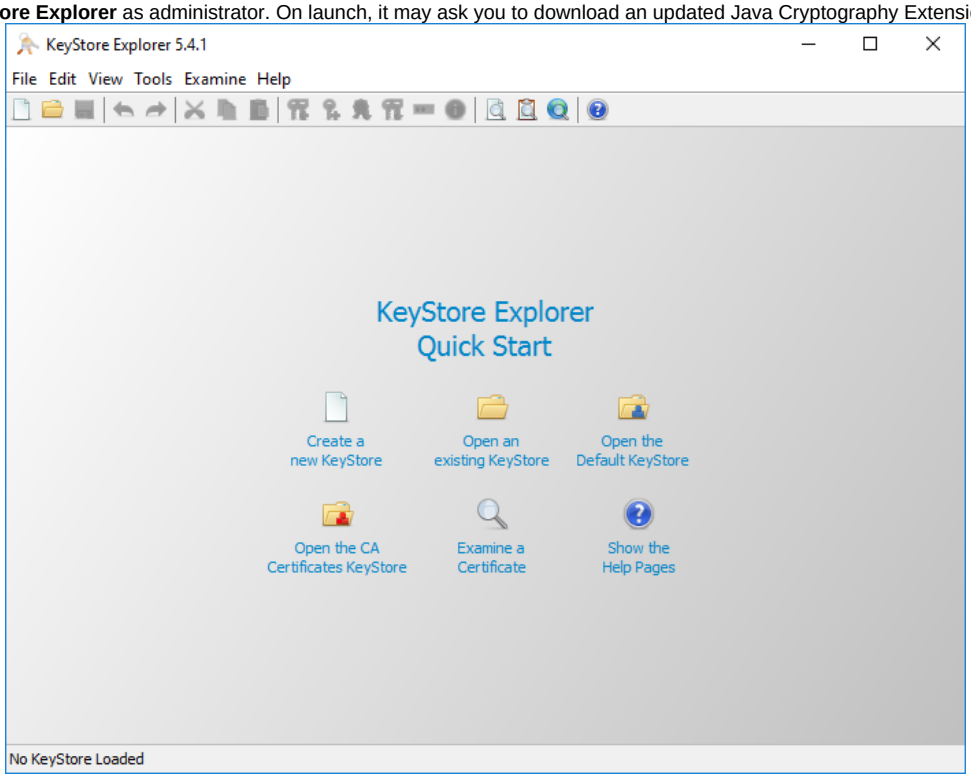
7. Follow the **Certificate Import Wizard** instructions to import the `.cer` file previously created.

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

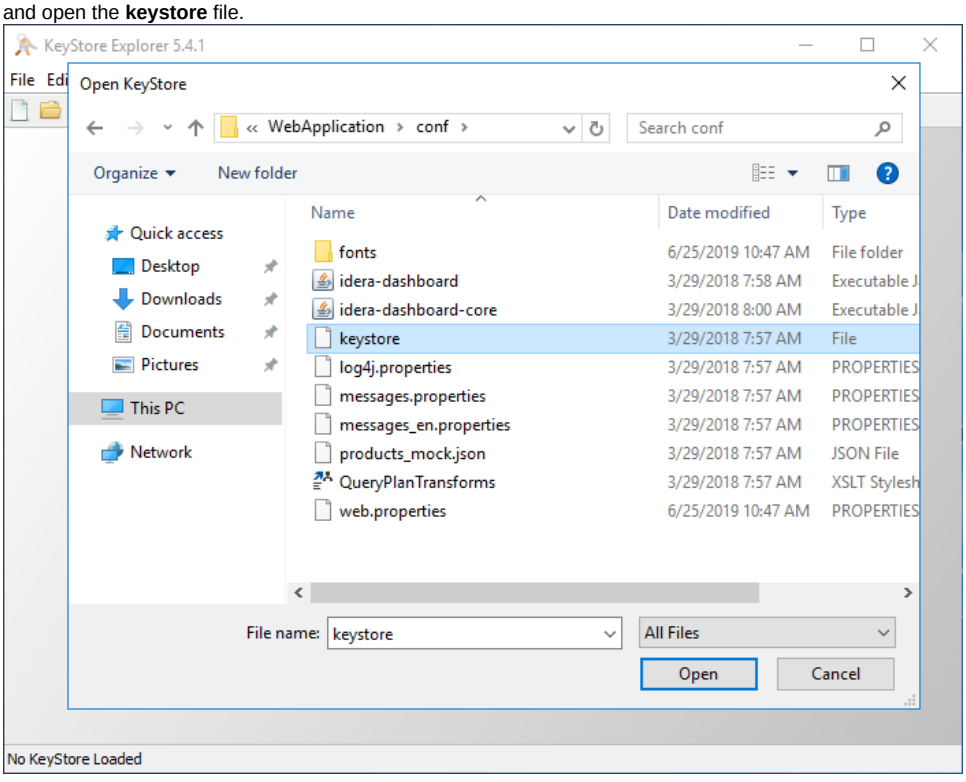
Import Key Pair

1. Download the free **KeyStore Explorer** utility from http://keystore-explorer.org/ and install it.

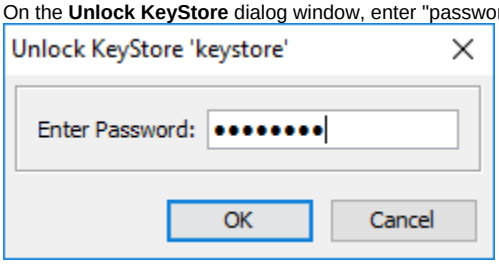
2. Open **KeyStore Explorer** as administrator. On launch, it may ask you to download an updated Java Cryptography Extension (JCE) Unlimited Strength file.



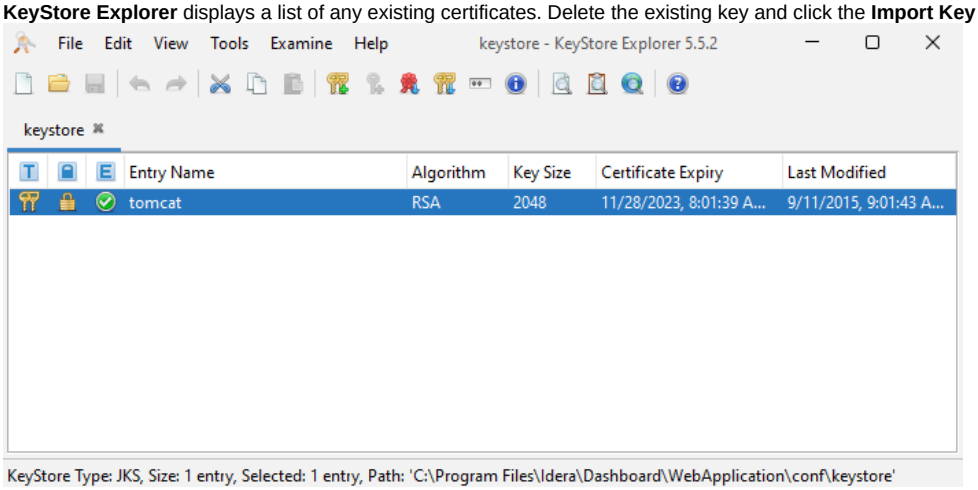
3. Click **Open an existing KeyStore**.
4. Browse to the **IDERA Dashboard conf directory** (the default path is `C:\Program Files\Idera\Dashboard\WebApplication\conf`), and open the **keystore** file.



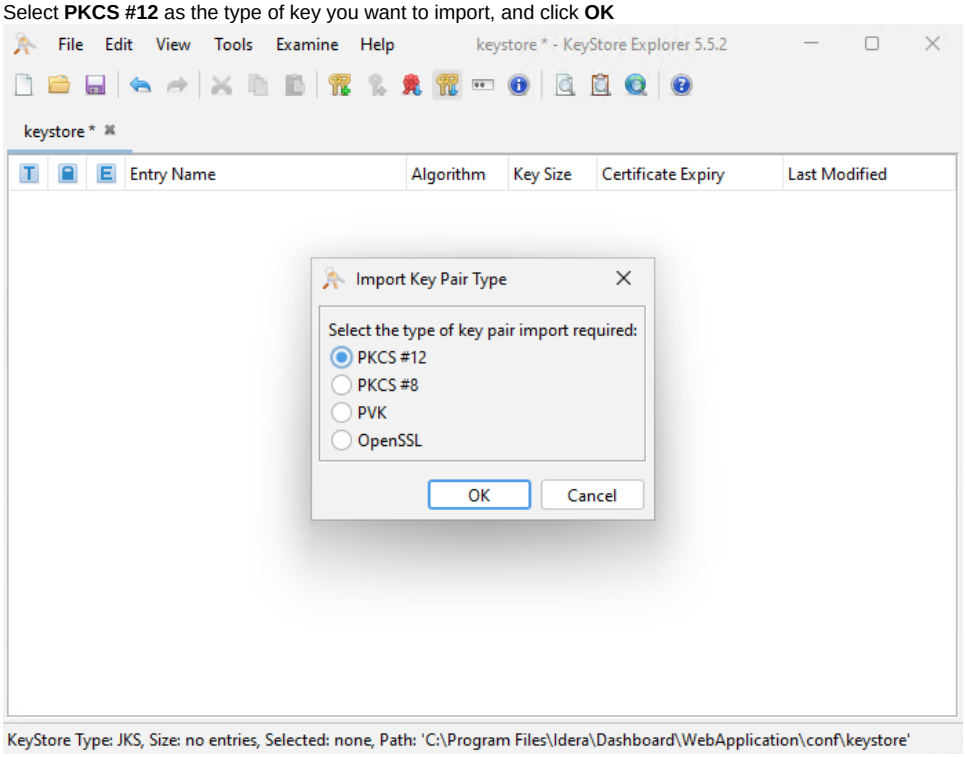
5. On the **Unlock KeyStore** dialog window, enter "password" and then click **OK**.



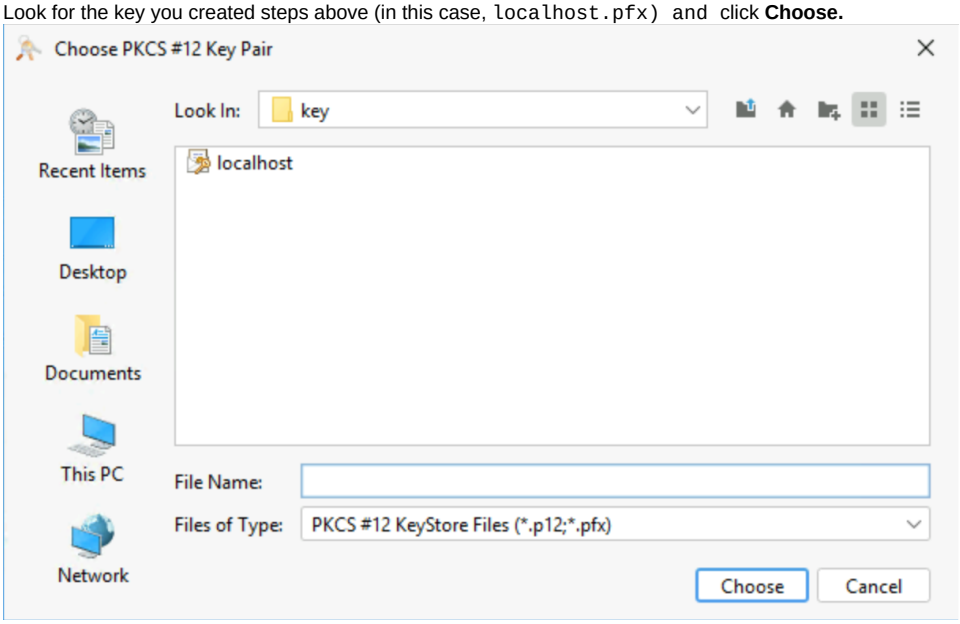
6. **KeyStore Explorer** displays a list of any existing certificates. Delete the existing key and click the **Import Key Pair** button.



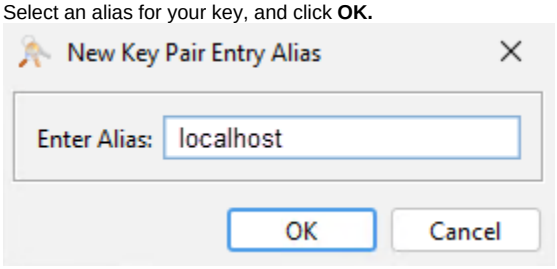
7. Select **PKCS #12** as the type of key you want to import, and click **OK**



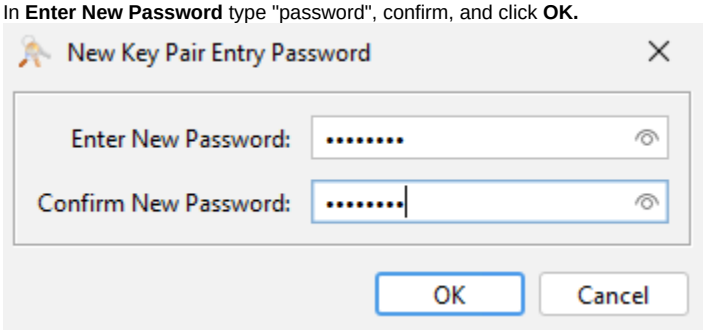
8. Look for the key you created steps above (in this case, `localhost.pfx) and` click **Choose.**



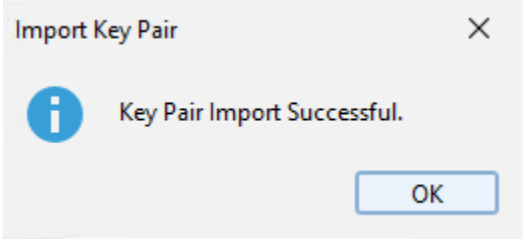
9. Select an alias for your key, and click **OK.**



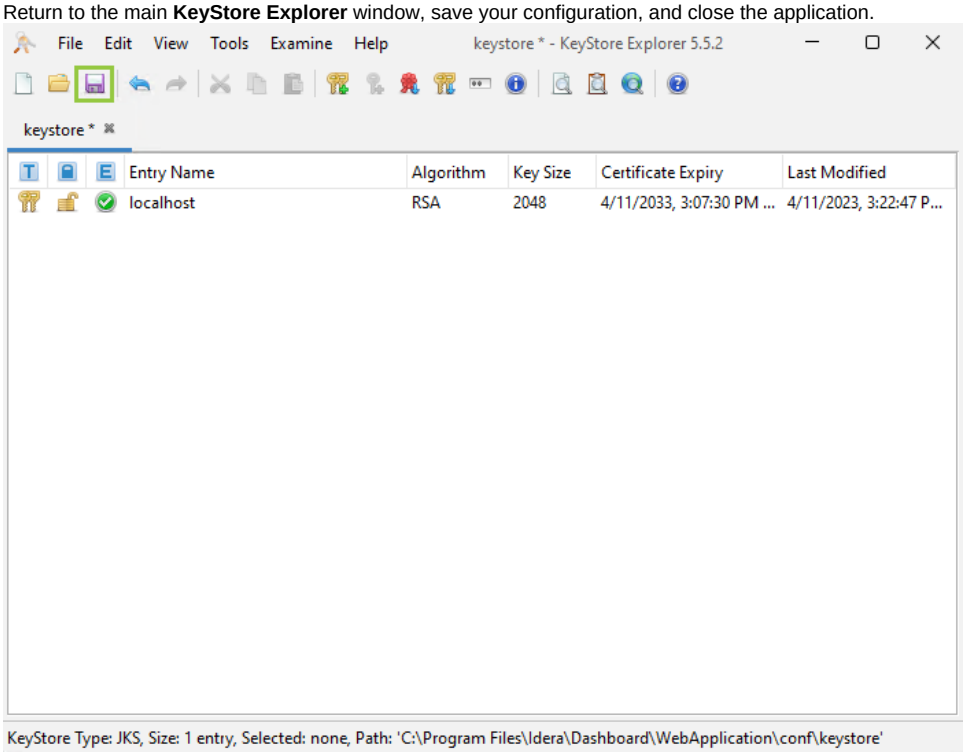
10. In **Enter New Password** type "password", confirm, and click **OK.**



11. When the configuration is completed the following message pops up, click **OK.**



12. Return to the main **KeyStore Explorer** window, save your configuration, and close the application.



13. Close all the opened browsers.
14. Restart **Idera Dashboard Core Service** and **Idera Dashboard Web Application Service**.
15. Access to **IDERA Dashboard** with the following link `https://<{certificateName}>:9291`.