Trusted Third Party Connectors for Power BI

Instructions for developer

Note: If you need help creating a self-signed certificate to test these instructions, please see the Microsoft Documentation on ‘New-SelfSignedCertificate’ in PowerShell here.

Note: If you need help exporting your certificate as a pfx, please see here.

Extract MakePQX.zip to wherever you want to run it from. To run it, call it in the command-line. Running without any parameters will return the help information.

```
C:\>C:\Users\cpope\Downloads\MakePQX\MakePQX.exe
Usage: MakePQX [options] [command]

Options:
  -?  | -h | --help  Show help information

Commands:
  pack    Create a .pqx file.
  sign    Signs an unsigned pqx, or countersigns if pqx is already signed. Use the --replace option to
replace the existing signature.
  verify  Verify the signature status on a .pqx file. Return value will be non-zero if the signature is
invalid.
```

There are three commands in MakePQX. Use "MakePQX [command] --help" for more information about a command.

Pack

```
C:\Users\cpope\Downloads\MakePQX>MakePQX.exe pack -h


Usage: MakePQX pack [options]

Options:
  -?  | -h | --help    Show help information
  -mz | --mez          Input extension file.
  -c  | --certificate  Certificate (.pfx) used to sign the extension file.
  -p  | --password     Password for the certificate file.
  -t  | --target       Output file name. Defaults to the same name as the input file.
pack --mez <extension.mez> [--target extension.pqx] [--certificate <cert.pfx> [--password <password>]]
```

Example

```
C:\Users\cpope\Downloads\MakePQX>MakePQX.exe pack -mz "C:\Users\cpope\OneDrive\Documents\Power BI
Desktop\Custom Connectors\HelloWorld.mez" -t "C:\Users\cpope\OneDrive\Documents\Power BI Desktop\Custom
Connectors\HelloWorldSigned.pqx"
```

Sign

```
C:\Users\cpope\Downloads\MakePQX>MakePQX.exe sign -h


Usage: MakePQX sign [arguments] [options]

Arguments:
  pqx file  The path to the .pqx file.

Options:
  -c  | --certificate  Certificate (.pfx) used to sign the extension file.
  -p  | --password     Password for the certificate file.
  -r | --replace       Replace existing signature instead of countersigning.
  -?  | -h | --help    Show help information
sign <extension.pqx> --certificate <cert.pfx> [--password <certPassword>] [--replace]
```

Example

```
C:\Users\cpope\Downloads\MakePQX>MakePQX sign "C:\Users\cpope\OneDrive\Documents\Power BI Desktop\Custom
Connectors\HelloWorldSigned.pqx" --certificate ColinPopellTestCertificate.pfx --password password
```

Verify

```
C:\Users\cpope\Downloads\MakePQX>MakePQX.exe verify -h


Usage: MakePQX verify [arguments] [options]

Arguments:
  pqx file  The path to the .pqx file.

Options:
  -q  | --quiet      Hides signature verification output.
  -?  | -h | --help  Show help information
```

Example

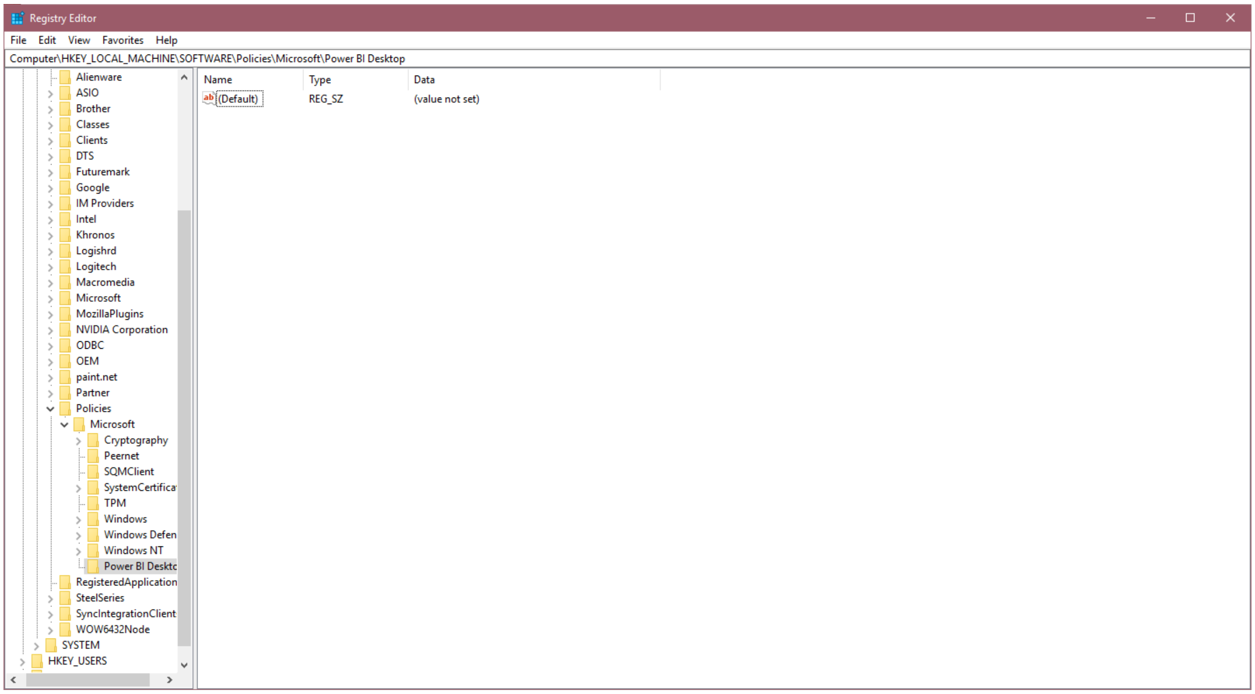
```
C:\Users\cpope\Downloads\MakePQX>MakePQX verify "C:\Users\cpope\OneDrive\Documents\Power BI Desktop\Custom
Connectors\HelloWorldSigned.pqx"
{
  "SignatureStatus": "Success",
  "CertificateStatus": [
    {
      "Issuer": "CN=Colin Popell",
      "Thumbprint": "16AF59E4BE5384CD860E230ED4AED474C2A3BC69",
      "Subject": "CN=Colin Popell",
      "NotBefore": "2019-02-14T22:47:42-08:00",
      "NotAfter": "2020-02-14T23:07:42-08:00",
      "Valid": false,
      "Parent": null,
      "Status": "UntrustedRoot"
    }
  ]
}
```

Once you’ve verified your signature, you can provide the thumbprint to the end-user to list as trusted.

Instructions for user

Trusting third party connectors in Power BI is done by listing the thumbprint of the certificate you want to trust in a specified registry value. If this thumbprint matches the thumbprint of the certificate that the connector you want to load was signed with, you will be able to load it in the ‘Recommended’ security level of Power BI.

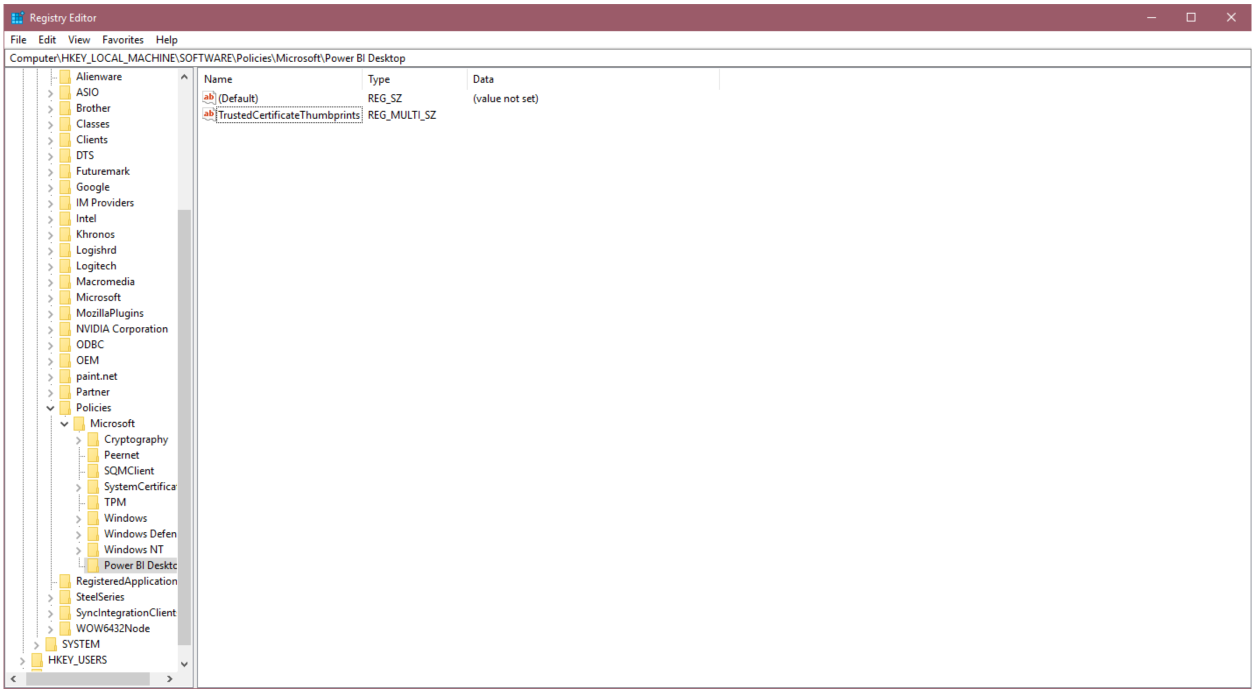
The registry path is `HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Power BI Desktop`. Make sure the path exists, or create it. We chose this location due to it being generally controlled by IT policy, as well as requiring local machine administration access to edit.



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Power BI Desktop

Name	Type	Data
(Default)	REG_SZ	(value not set)

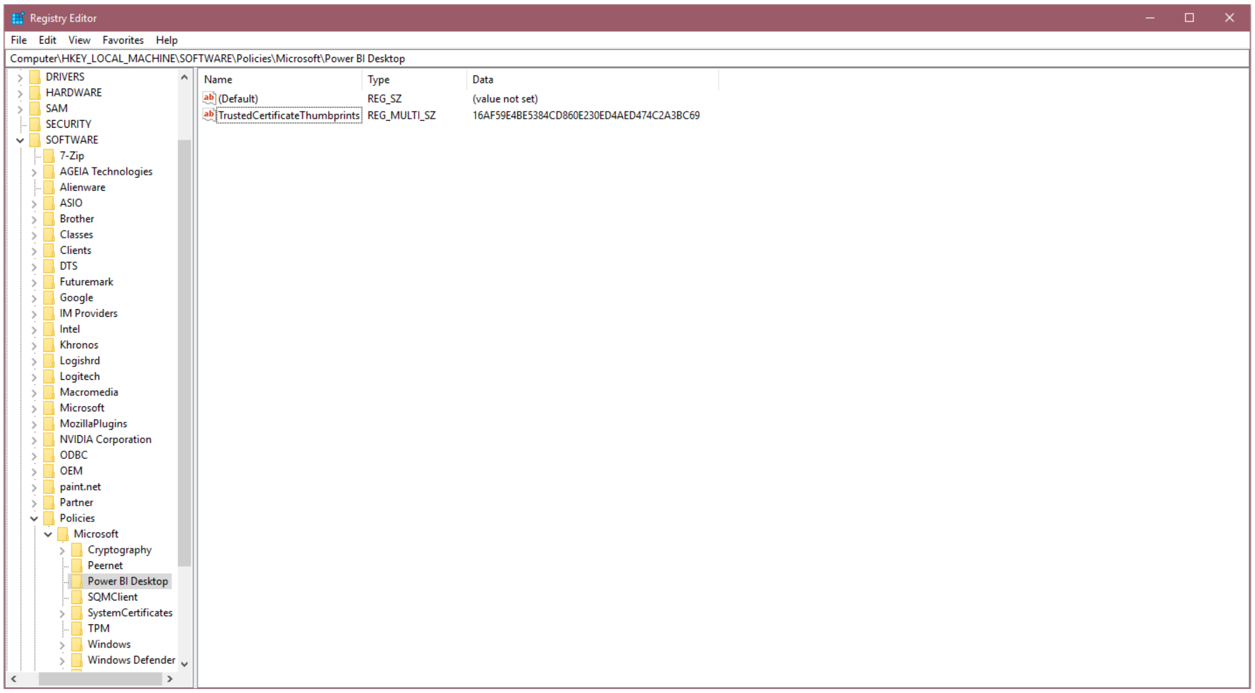
Add a new value under the path specified above. The type should be “Multi-String Value” (REG_MULTI_SZ), and it should be called “TrustedCertificateThumbprints”



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Power BI Desktop

Name	Type	Data
(Default)	REG_SZ	(value not set)
TrustedCertificateThumbprints	REG_MULTI_SZ	

Add the thumbprints of the certificates you want to trust. You can add multiple certificates by using “\0” as a delimiter, or in the registry editor, right click -> modify and put each thumbprint on a new line. Example thumbprint is taken from a self-signed certificate.



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Power BI Desktop

Name	Type	Data
(Default)	REG_SZ	(value not set)
TrustedCertificateThumbprints	REG_MULTI_SZ	16AF59E4BE5384CD860E230ED4AED474C2A3BC69

If you’ve followed the instructions properly, and have been given the proper thumbprint by your developer, you should now be able to securely trust connectors signed with the associated certificate.