Financial Risk Compliance Suite

Version 6.13

REPORTING DATABASE

Tech Doc

Revision A

Publication Date: 2nd April, 2024

Legal Notice:

This document is proprietary & confidential and may only be used by persons who have received it directly from ThetaRay LTD. ("ThetaRay") and may not be transferred to any other party without ThetaRay's express written permission. The document provides preliminary and general information only and is not intended to be comprehensive or to address all the possible issues, applications, exceptions or concerns relating to ThetaRay. All information contained in this document is confidential and shall remain at all times the sole property of ThetaRay. The recipient of this document has no right to disclose any or all of its contents or distribute, transmit, reproduce, publicize or otherwise disseminate this document or copies thereof without the prior written consent of ThetaRay, and shall keep all information contained herein strictly private and confidential. The information is intended to facilitate discussion and is not necessarily meaningful or complete without such supplemental discussion. Please note that the information procedures, practices, policies, and benefits described in this document may be modified or discontinued from time to time by ThetaRay without prior notice. As such, ThetaRay provides the document on an "As- Is" basis and makes no warranties as to the accuracy of the information contained therein. In addition, ThetaRay accepts no responsibility for any consequences whatsoever arising from the use of such information. ThetaRay shall not be required to provide any recipient with access to any additional information or to update this document or to correct any inaccuracies herein which may become apparent.

1. Introduction

1.1. Purpose and Audience

The ThetaRay Reporting Database is a dedicated area within ThetaRay's central relational database (Postgres) focused on serving the needs of queries originating from business intelligence and reporting tools.

This guide details the schema of the Reporting Database, including tables, relationships and individual fields.

The default deployment of the ThetaRay system does not expose Postgres to external clients. The last section of the document details the practice that is required post installation in order to expose the database to the outside world in a secure manner.

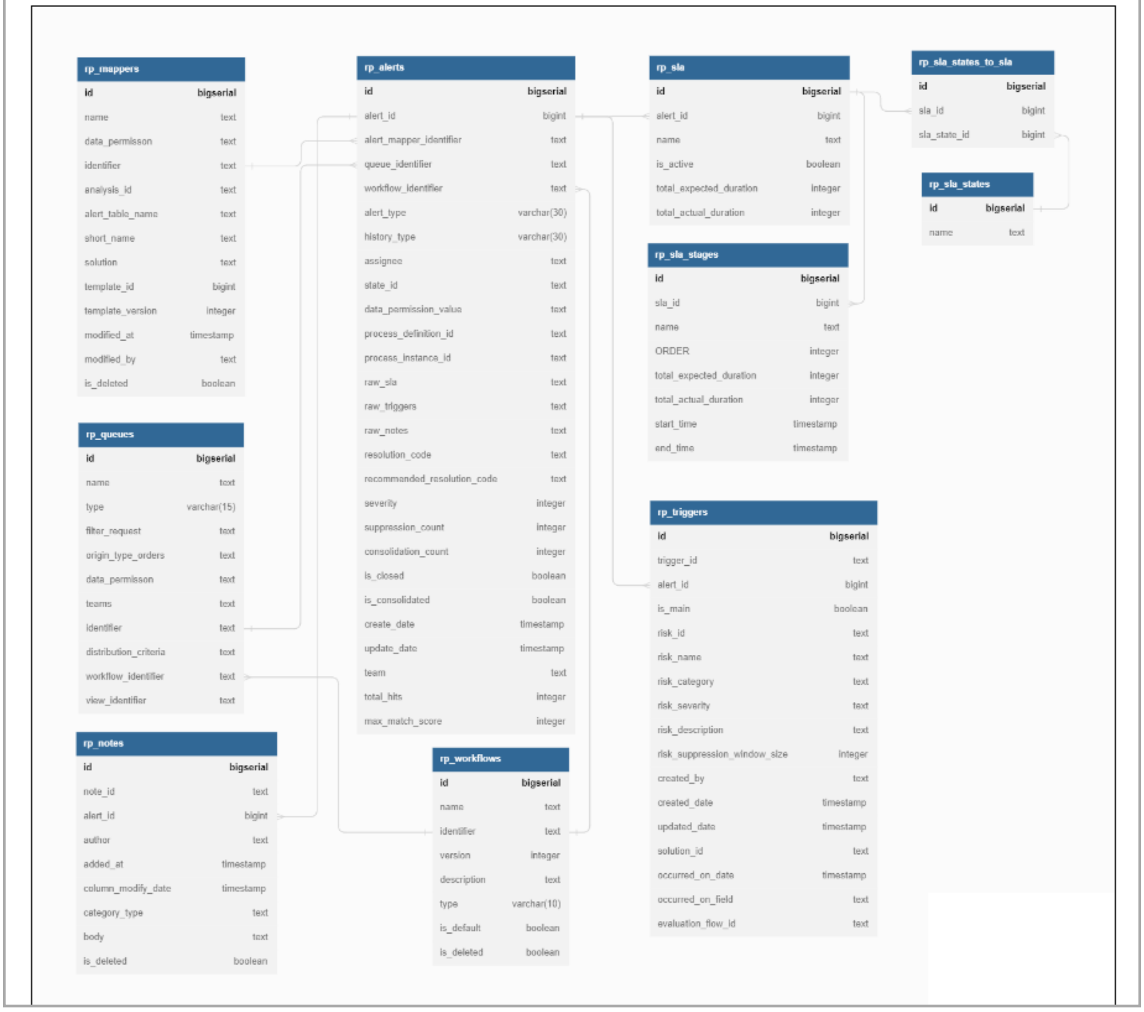
The appendix addresses the process of enabling connectivity from a Windows host to the Reporting Database when using Azure PowerBI Service - including PowerBI desktop and PowerBI On-Premise Gateway.

2. Reporting Database Schema

As the ‘Reporting Database’ is associated with an Investigation Center deployment – each environment may consist of multiple Reporting Database instances, each associated with it’s own database name. The database schema for a given Investigation Center is named - <apps_<ic instance name>.

The Entity Relationship Diagram detailed below presents the Reporting Database schema, including tables, fields and logical foreign key relationships (these are not foreign key constraints at database level).

Schema:



rp_mappers	
id	bigserial
name	text
data_permisson	text
identifier	text
analysis_id	text
alert_table_name	text
short_name	text
solution	text
template_id	bigint
template_version	integer
modified_at	timestamp
modified_by	text
is_deleted	boolean

rp_queues	
id	bigserial
name	text
type	varchar(15)
filter_request	text
origin_type_orders	text
data_permisson	text
teams	text
identifier	text
distribution_criteria	text
workflow_identifier	text
view_identifier	text

rp_notes	
id	bigserial
note_id	text
alert_id	bigint
author	text
added_at	timestamp
column_modify_date	timestamp
category_type	text
body	text
is_deleted	boolean

rp_alerts	
id	bigserial
alert_id	bigint
alert_mapper_identifier	text
queue_identifier	text
workflow_identifier	text
alert_type	varchar(30)
history_type	varchar(30)
assignee	text
state_id	text
data_permission_value	text
process_definition_id	text
process_instance_id	text
raw_sla	text
raw_triggers	text
raw_notes	text
resolution_code	text
recommended_resolution_code	text
severity	integer
suppression_count	integer
consolidation_count	integer
is_closed	boolean
is_consolidated	boolean
create_date	timestamp
update_date	timestamp
team	text
total_hits	integer
max_match_score	integer

rp_workflows	
id	bigserial
name	text
identifier	text
version	integer
description	text
type	varchar(10)
is_default	boolean
is_deleted	boolean

rp_sla	
id	bigserial
alert_id	bigint
name	text
is_active	boolean
total_expected_duration	integer
total_actual_duration	integer

rp_sla_stages	
id	bigserial
sla_id	bigint
name	text
ORDER	integer
total_expected_duration	integer
total_actual_duration	integer
start_time	timestamp
end_time	timestamp

rp_triggers	
id	bigserial
trigger_id	text
alert_id	bigint
is_main	boolean
risk_id	text
risk_name	text
risk_category	text
risk_severity	text
risk_description	text
risk_suppression_window_size	integer
created_by	text
created_date	timestamp
updated_date	timestamp
solution_id	text
occurred_on_date	timestamp
occurred_on_field	text
evaluation_flow_id	text

rp_sla_states_to_sla	
id	bigserial
sla_id	bigint
sla_state_id	bigint

rp_sla_states	
id	bigserial
name	text

With regard to the above schema, it is worth noting that the connections between each schema block are logical connections, and not foreign key connections.

2.1. Schema Tables Details.

2.1.1. rp_alerts

Description: This table lists all alert attributes from all alert tables as reference (historical) data.

The only time changes are made to either historical data rows or new current rows is when changes are made to specific information, such as state, assignee or queue.

Table 1: *rp_alerts*

Field Attribute	Description	Type
ID	Row identifier	big integer
ALERT_MAPPER_IDENTIFIER	Identifier of the mapper	bigint
QUEUE_IDENTIFIER	Identifier of the corresponding queue	varchar (255)
WORKFLOW_IDENTIFIER	Identifier of the corresponding workflow	varchar (255)
ALERT_ID	Identifier of the alert from the corresponding table	text
ALERT_TYPE	type of the alert, for example (SCREENING, CUSTOMER_SCREENING,SYSTEM,EXTERNAL,MANUAL)	varchar (30)
ASSIGNEE	Assignee name	text
STATE_ID	Identifier of the current alert state	text
HISTORY_TYPE	Alert History type, for example: (HISTORY /CURRENT). Only one row will have status CURRENT, other one will have a status HISTORY.	varchar (30)
DATA_PERMISSION	Data permission of the corresponding alert mapper	text
PROCESS_DEFINITION_ID	Identifier of the corresponding process definition	text
PROCESS_INSTANCE_ID	Identifier of the corresponding process instance	text
RESOLUTION_CODE	Alert resolution code	text
RECOMMENDED_RESOLUTION_CODE	Alert recommended resolution code	text
SEVERITY	Alert severity number as a percentage	int
SUPPRESSION_COUNT	Number of suppressed alerts	int
IS_CLOSED	Trigger, if alert is closed	boolean

Table 1: *rp_alerts (continued)*

Field Attribute	Description	Type
CREATE_DATE	Date of alert creation	timestamp
UPDATE_DATE	Date of alert last update	timestamp
TOTAL_HITS	Number of total hits	int
MAX_MATCH_ SCORE	Maximum match score of the alert	int
RAW_ CLASSIFICATION	Json representation for classifications in alert |	json
RAW_TAGS	Json representation for tags in alert |	json

Constraints:

- ID - primary key
- ALERT_MAPPER_IDENTIFIER - not null
- QUEUE_IDENTIFIER - not null
- WORKFLOW_IDENTIFIER - not null

Note: CONSOLIDATION_COUNT, IS_CONSOLIDATION - removed from version 6.10

2.1.2. rp_sla

Description: Table contains sla that references alerts by the ALERT_ID column.

Table 2: *rp_sla*

Field Attribute	Description	Type	From Version
ID	Sla Identifier	big integer	
ALERT_ID	Alert identifier	text	
NAME	SLA name	text	
IS_ACTIVE	Trigger, if SLA is active	boolean	
TOTAL_EXPECTED_ DURATION	Expected duration of all steps in minutes	time	
MODIFIED_AT	Time when row was updated	timestamp	6.10
TOTAL_ACTUAL_ DURATION	Resulting duration of all steps in minutes	time	

Constraints:

- ID - primary key

2.1.3. rp_sla_states

Description: This table lists all *SLA* states.

Table 3: *_sla_states*

Field Attribute	Description	Type
ID	State identifier	bigint
NAME	Name of the sla state	text

- ID - primary key

2.1.4. rp_sla_states_to_sla

Description: This table lists *SLA* to *SLA_STATES* in the many to many mode.

Table 4: *rp_sla_states_to_sla*

Field Attribute	Description	Type
ID	Row identifier	bigint
SLA_ID	Identifier of the SLA	bigint
SLA_STATE_ID	Identifier of the SLA state	bigint

Constraints:

- ID - primary key

2.1.5. rp_sla_stages

Description: This table lists SLA stages. Every SLA has a list of stages, all of them are ordered by ORDER column.

Table 5: *rp_sla_stages*

Field Attribute	Description	Type	From Version
ID	Row identifier	bigint	
SLA_ID	SLA identifier	bigint,not null	
NAME	SLA stage name	text	
STAGE_ ORDER	Order of the stage in the SLA	int	
EXPECTED_ DURATION	Expected duration of this step in minutes	bigint	6.10
ACTUAL_	Resulting duration of this step in minutes	bigint	6.10

Table 5: *rp_sla_stages (continued)*

Field Attribute	Description	Type	From Version
DURATION			
START_TIME	Starting time of this step	timestamp	
END_TIME	Ending time of this step	timestamp	
MODIFIED_AT	Time when row was updated	timestamp	6.10

Note: total_actual_duration and total_expected_duration available prior to 6.10 are no longer available and have been replaced by the expected_duration and actual_duration columns.

Constraints:

- ID - primary key

2.1.6. rp_notes

Description: This table lists note attributes related to alert notes, these are referenced via the ALERT_ID column. Notes can be deleted by applying the *IS_DELETED* attribute.

Table 6: *rp_notes*

Field Attribute	Description	Type	From Version
ID	Note identifier	bigint	
NOTE_ID	Note identifier for the corresponding alert	text	
ALERT_ID	Alert identifier	text	
AUTHOR	The name of the note author	text	
ADDED_AT	Timestamp, when note was added	timestamp	
MODIFY_DATE	Timestamp, when note was created (modified)	timestamp	
CATEGORY_TYPE	Note category type	text	
BODY	Note text body	text	
IS_DELETED	Trigger, if note is deleted	boolean	
MODIFIED_AT	Time when row was updated	timestamp	6.10

Constraints:

- ***ID*** - primary key

2.1.7. rp_triggers

Description: This triggers table is connected to the alerts table by applying the ALERT_ID attribute. Triggers can be deleted by applying the *IS_DELETED* attribute.

Table 7: *rp_triggers*

Field Attribute	Description	Type
ID	Trigger identifier	bigint
TRIGGER_ID	Corresponding trigger id of alert	text
ALERT_ID	Identifier of alert	not-null bigint
IS_MAIN	Boolean if this trigger is main in the alert	boolean
RISK_ID	Risk identifier	text
RISK_NAME	The risk name	text
RISK_CATEGORY	The risk category	text
RISK_SEVERITY	Risk severity, as a percentage	text
RISK_DESCRIPTION	Description of the risk	text
RAW_PRIMARY_KEYS_RAW	Json of the Primary keys	text
RAW_ALERT_FIELDS_RAW	Json of the Alert fields	text
RAW_GROUPING_IDENTIFIERS_RAW	Json of the grouping identifiers	text
CREATED_BY	Trigger author name	text
CREATED_DATE	Date of trigger creation	timestamp
UPDATED_DATE	Date of the triggers last update	timestamp
SOLUTION_ID	Solution identifier	text
OCCURRED_ON_DATE	Date when trigger is occurred	timestamp
OCCURRED_ON_FIELD	Field on which trigger is occurred	text
EVALUATION_FLOW_ID	Evaluation flow identifier	text
RISK_SUPPRESSION_WINDOW_SIZE	*Depricated - no longer in use*	int
SUBTYPE	A subtype of trigger(system/manual)	varchar(10)

Constraints:

- ID - primary key

2.1.8. rp_primary_keys view

Description: A view providing a fielded representation of the primary key fields that are part of an alert trigger.

Note: Each investigated entity field from mappers will be represented in that view, as a different column.

Table 8: *rp_primary_keys view*

Field Attributes	Description	Type
RP_TRIGGER_ID	Corresponding trigger identifier	text
RP_ALERT_ID	Iidentifier of alert	text
PRIMARY_KEY_ NAME	Field per primary key defined within the Investigation Center	text

Constraints:

- ID - primary key

2.1.9. rp_alert_fields view

Description: A view providing a fielded representation of the primary key fields that are part of an alert trigger.

Note: Each alert field from mappers, will be represented in that view by a different column.

Table 9: *rp_alert_fields view*

Field Attributes	Description	Text
RP_TRIGGER_ID	Corresponding trigger identifier	text
RP_ALERT_ID	Identifier of alert	text
PRIMARY_KEY_NAME	Field per primary key defined within the Investigation Center	text

Constraints:

- ***ID***: primary key

2.1.10. rp_grouping_identifiers view

Description: A view providing a fielded representation of the 'group identifiers' fields used as part of the Entity Resolution functionality

Table 10: *rp_grouping_identifiers view*

Field Attributes	Description	Type
RP_TRIGGER_ID	Corresponding trigger identifier	text
RP_ALERT_ID	Identifier of alert	text
GROUPING_IDENTIFIER_NAME	Name of the primary key	text

Constraints:

- ***ID***: primary key

2.1.11. rp_queues

Description: Table consisting of queue details

Table 11: *rp_queues*

Field Attributes	Description	Type	From Version
ID	Row identifier	bigint	
NAME	Queue name	text	
TYPE	Queue request	Varchar(15)	
QUEUE_QUERY	Queue list of orders	text	
ORIGIN_TYPE_ORDERS	Queue list of orders	text	
DATA_PERMISSON	Queue DPV	text	
TEAMS	Queue teams list	text	
IDENTIFIER	Queue identifier	text	
DISTRIBUTION_CRITERIA	Criteria object	text	
WORKFLOW_ID	Id of the corresponding workflow	text	
VIEW_IDENTIFIER	Corresponding view identifier	text	
MODIFIED_AT	Time when row was updated	timestamp	6.10

Constraints

- ID -primary key

2.1.12. rp_mappers

Description: Investigation Center Mappers definitions

Table 12: *rp_mappers*

Field Attributes	Description	Type	From Version
ID	Primary key identifier	bigint	
NAME	Queue name	text	
DATA_PERMISSON	Queue DPV (text)	text	
IDENTIFIER	Mapper text identifier (text)	text	
SOLUTION_EVALUATION_FLOW_UNIT	Evaluation flow reference	text	
ALERT_TABLE_NAME	Queue list of orders (text)	text	
SHORT_NAME	Short name of mapper (text)	text	
TEMPLATE_ID	Id of the corresponding template (text)	bgint	
TEMPLATE_VERSION	Corresponding template version (int)	int	
IS_DELETED	If it is a deleted mapper (boolean)	boolean	
MODIFIED_BY	Author's name	text	
MODIFIED_AT	Timestamp when mapper was updated	timestamp	6.10

Constraints:

- ***ID***: primary key
- ***identifier*** - not null

2.1.13. rp_workflows

Description: Workflow identifiers

Table 13: *rp_workflows*

Field Attribute	Description	Type
ID	Primary key identifier	bigint
NAME	Workflow name	text
IDENTIFIER	Workflow text identifier	text
DESCRIPTION	Workflow description	text
TYPE	Workflow type	varchar(10)
IS_DEFAULT	If it's a default workflow	boolean
IS_DELETED	If its a deleted workflow	timestamp

Constraints:

- ***ID***: Primary key

2.1.14. rp_attachments

Description attachment metadata

Table 14: *rp_attachments*

Field Attribute	Description	Type	From Version
ATTACHMENT_ID	Primary key identifier	Varchar(50)	6.10
ALERT_ID	Alert id	Varchar(50)	6.10
ADDED_BY	User that added attachment	Varchar(100)	6.10
ADDED_ON	Time when attachment was added	timestamp	6.10
CATEGORY	Category of attachment	Varchar(100)	6.10
DESCRIPTION	Description of attachment	text	6.10
STATE	State of attachment	Varchar(100)	6.10
LINK	Path of document in minio	varchar	6.10
SIZE	Size of attachment	bigint	6.10
FILENAME	Filename of attachment in minio	varchar	6.10
MODIFIED_AT	Timestamp when row was updated	timestamp	6.10

Constraints:

- Attachment_id: Primary key

2.1.15. rp_triggers_data

Description triggers data

Table 15: *rp_triggers_data*

Field Attribute	Description	Type	From Version
ID	Primary key	bigint	6.10
ALERT_ID	Alert id	text	6.10
TRIGGER_ID	Id of trigger	text	6.10
DATA_HASH	technical column for internal usage	integer	6.10
TRIGGER_DATA	Contains trigger data	JSON	6.10
MODIFIED_AT	Timestamp when row was updated	timestamp	6.11

Constraints:

- ID: Primary key

2.2. Accessing Encrypted Data

When the Data at Rest Encryption functionality is enabled on the ThetaRay environment, selected alert fields and notes text will appear in encrypted form in the database. The data is encrypted by using the AES algorithm in ECB mode to encrypt the data, base64 encoding the result and prefixing it with an 'ed@@' string to clearly indicate that the content of the data element is encrypted.

Decryption of the data using SQL queries can be performed using the Postgres standard 'pgcrypto' extension, but requires access to the Data Encryption Key (DEK) which should not be publicly exposed and embedded into queries.

To allow access to the DEK, a Postgres stored function named 'get_dek' is automatically registered within the 'encryption' schema of the 'cdd' database in Postgres. Users granted access to this schema (for Postgres users created on behalf of Jupyter end users, these are users which are assigned the DirectSensitiveDataAccess role in Keycloak) may use this function to obtain a DEK and decrypt through SQL in the following manner (this queri is provided as a specific example) -

```
SELECT encode(
    decrypt(
        decode(substring(account_id FROM 5), 'base64'), # The data to decrypt
        decode(encryption.get_dek(), 'base64'),         # The data encryption
key
        'aes-ecb/pad:pkcs'
    ),
    'escape'
) AS account_id
FROM solution_test.transactions
```

Note: It should be noted that granting access to the encryption schema for the 'report_user' or to custom database users provisioned on the environment is not an automated process and should be manually performed by a database administrator.

Appendix A - Accessing Reports DB from Power BI Service

Microsoft Power BI Service is a SaaS service that provides business intelligence services. To allow access to the Reporting Database by the PowerBI Service without requiring the database to be exposed to the global internet, Microsoft provides the 'On-premises Data Gateway',a component that can be deployed on a Windows host and enables PowerBI to access data servers located in (virtual) networks that are not exposed to the internet, but have outbound access to the PowerBI service.

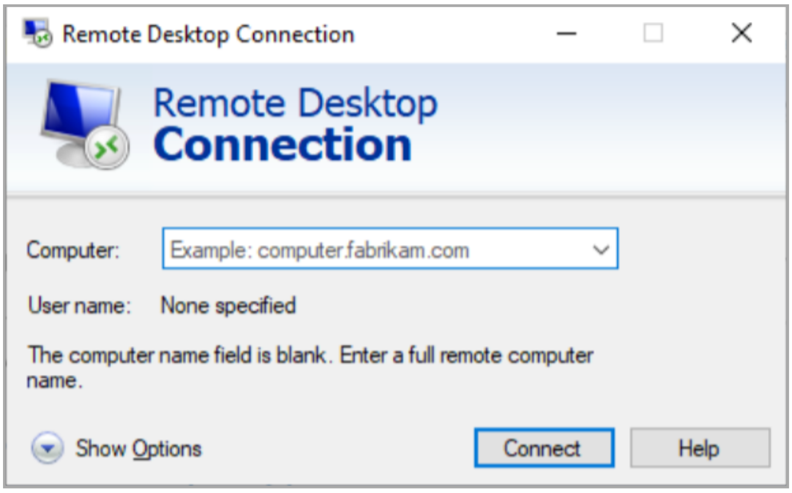
The following sections provides instructions on how to set up the On-Prem DataGateway on a Windows based virtual machine running on Microsoft Azure, enabling connectivity from PowerBI to the Reporting Database.

3.0.1. Enabling RDP Connectivity to Azure from a Local Machine

Your local machine can be any platform capable of running RDP. The method of starting RDP platform dependent. For a Windows system, type **mstsc** on a command line or in **Start>Run.**

Tip:The RDP connect program is located in **C:\Windows\System32\ mstsc.exe**. For repeated use, create a shortcut on the desktop or the taskbar.

Running the program results in the following window:



Computer: Example: computer.fabrikam.com

User name: None specified

The computer name field is blank. Enter a full remote computer name.

Enter the computer name. For this guide we use the VMs IP address, 13.93.93.223. The IP address should be supplied by the creator of the VM. (You may also need a username and password.)

3.0.2. Azure Windows VM Setup

Note: PowerBI On-Prem Gateway and Power BI Desktop run on Windows based virtual machines in Azure. Instructions on how to deploy such VMs on Azure are beyond the scope of this document. The following sections detail the extra steps to be performed on the Windows VMs to enable connectivity to Postgres running on OpenShift.

3.0.2.1. Installing a Trusted Root Certificate

Communication to ThetaRay's Postgres database is secured through TLS. The connection uses a dynamically generated certificate / key signed by a self-signed CA that is part of the ThetaRay environment. Clients running on Windows machines are required to trust ThetaRay's CA to obtain connectivity.

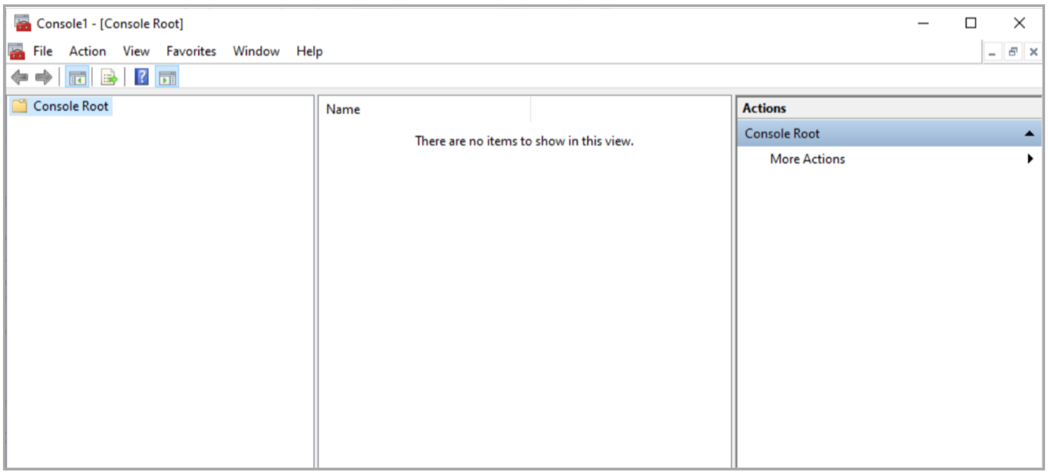
The root CA used by ThetaRay is set as a deployment parameter and can be obtained from the **shared-secrets.yaml.gotmpl** file which is part of the ThetaRay's deployment kit.

You will need to import the received certificate. In our example, it is a small file with a name like **Cert.p12**. To copy it from your local machine to the VM, you can use regular copy/paste.

From this point, the import method is based on Import Certificate:

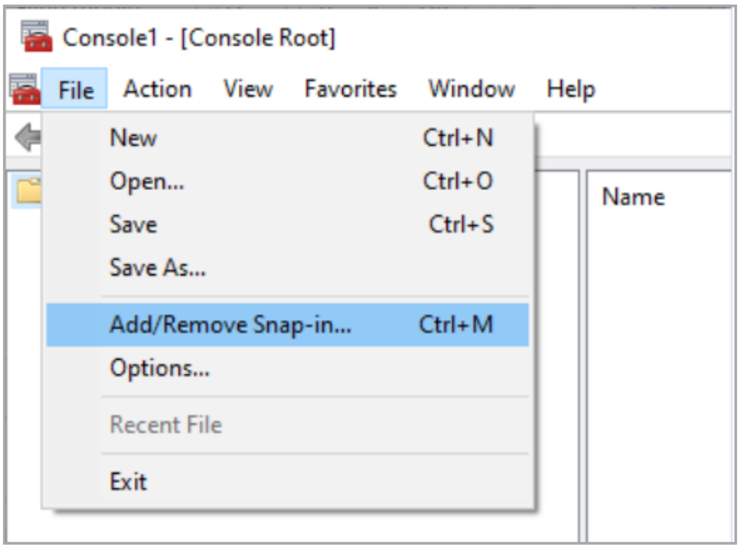
To import the certificate:

1. Right click Widows start and choose **Run**. Enter **mmc**. The **Console** window opens:

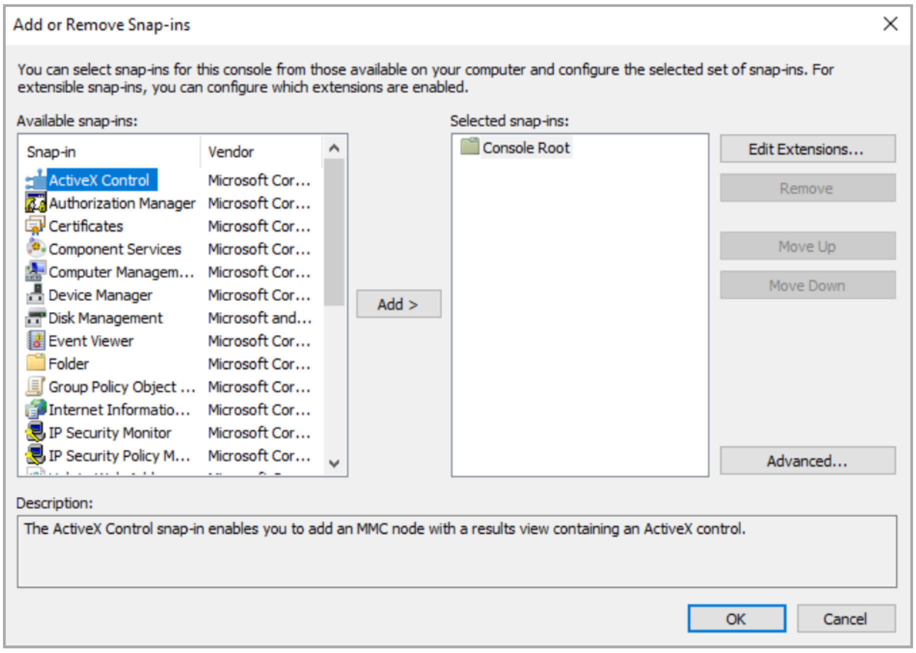


There are no items to show in this view.

2. Open **File>Add/Remove Snap-in**... :



3. The **Add or Remove Snap-ins** window opens:



You can select snap-ins for this console from those available on your computer and configure the selected set of snap-ins. For extensible snap-ins, you can configure which extensions are enabled.

Available snap-ins:

Snap-in	Vendor
ActiveX Control	Microsoft Cor...
Authorization Manager	Microsoft Cor...
Certificates	Microsoft Cor...
Component Services	Microsoft Cor...
Computer Managem...	Microsoft Cor...
Device Manager	Microsoft Cor...
Disk Management	Microsoft and...
Event Viewer	Microsoft Cor...
Folder	Microsoft Cor...
Group Policy Object ...	Microsoft Cor...
Internet Informatio...	Microsoft Cor...
IP Security Monitor	Microsoft Cor...
IP Security Policy M...	Microsoft Cor...

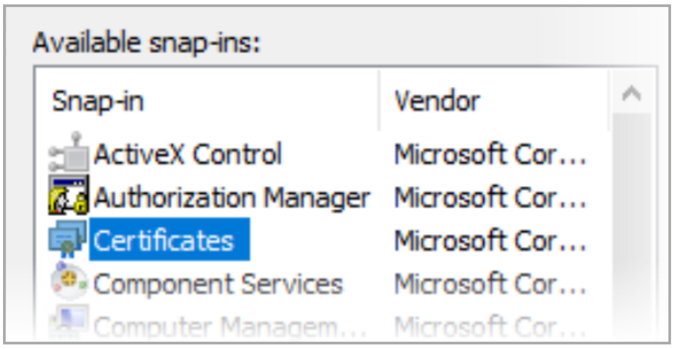
Selected snap-ins:

Console Root

Description:

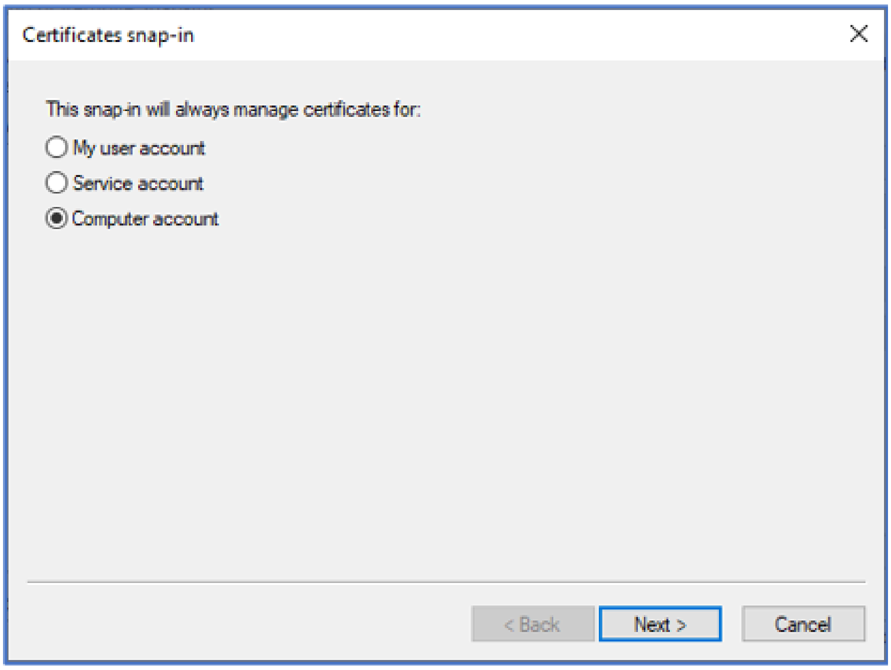
The ActiveX Control snap-in enables you to add an MMC node with a results view containing an ActiveX control.

4. Go to Certificates:


Available snap-ins:

Snap-in	Vendor
ActiveX Control	Microsoft Cor...
Authorization Manager	Microsoft Cor...
Certificates	Microsoft Cor...
Component Services	Microsoft Cor...
Computer Managem...	Microsoft Cor...

5. Click **Add**:

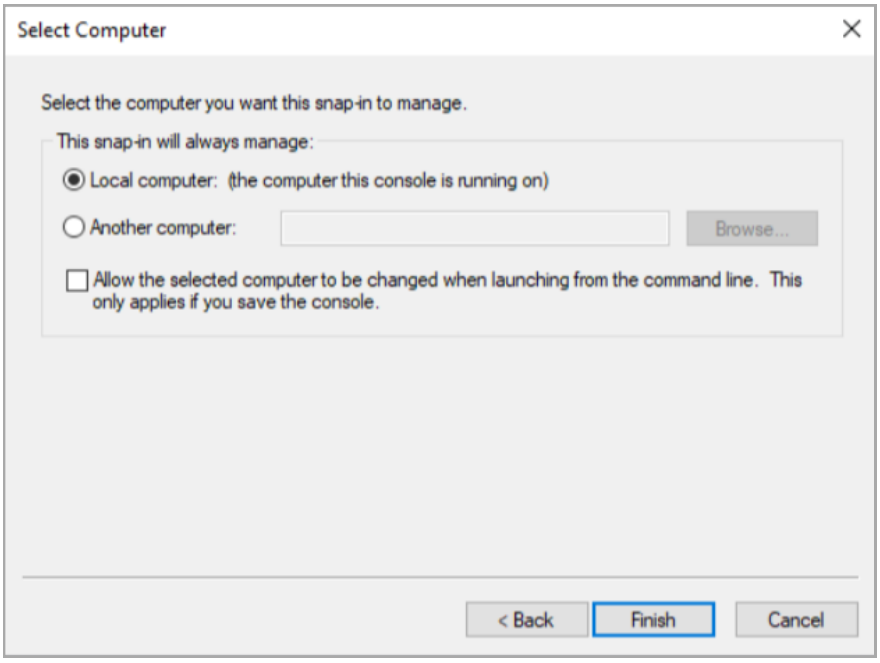


Certificates snap-in

This snap-in will always manage certificates for:

○ My user account
○ Service account
● Computer account

6. Check **Computer account** (as shown above) and click **Next**.



Select Computer

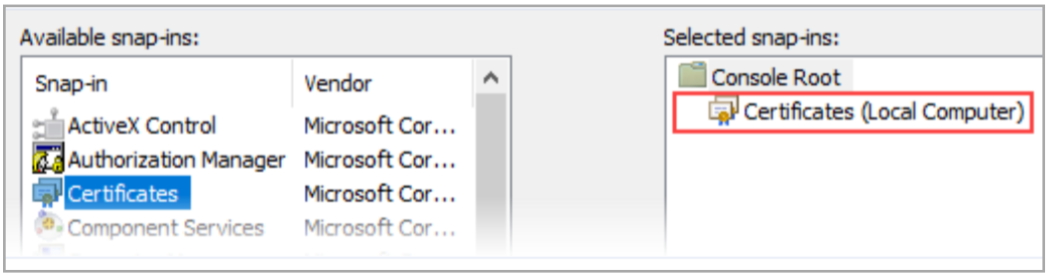
Select the computer you want this snap-in to manage.

This snap-in will always manage:

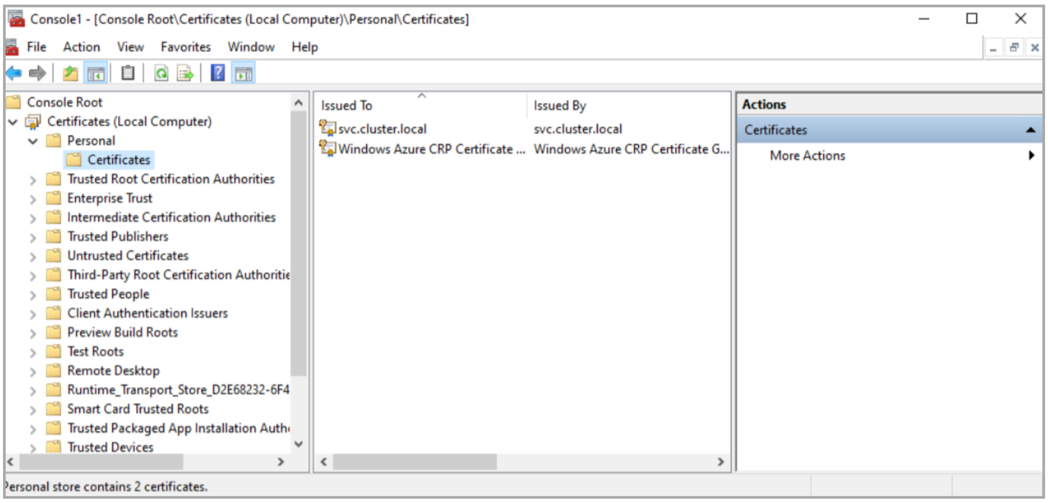
● Local computer: (the computer this console is running on)
○ Another computer:

□ Allow the selected computer to be changed when launching from the command line. This only applies if you save the console.

7. Ensure that **Local computer** is checked and click **Finish**. In the **Add or Remove Snap-ins** window, the **Certificates** item has been added:

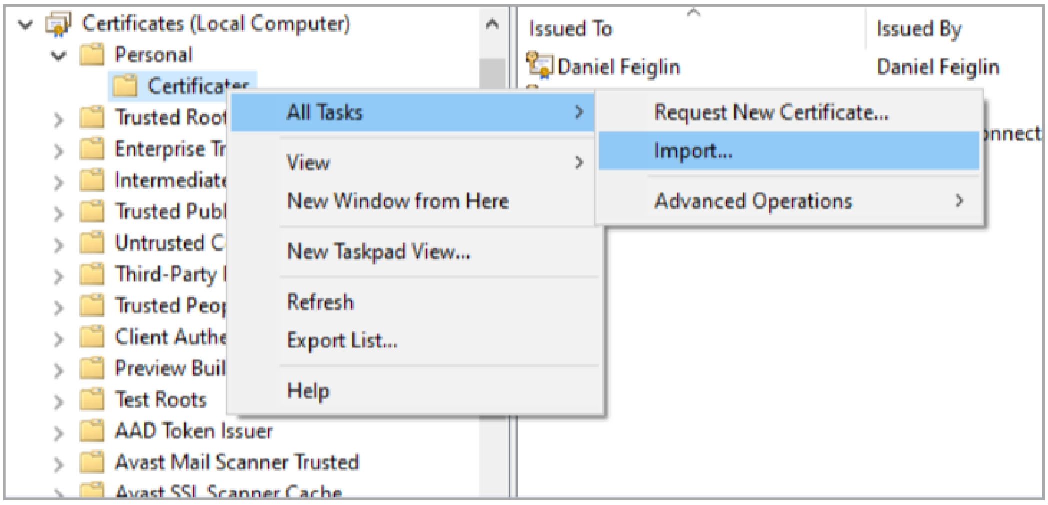


8. Click **OK**. You are returned to the Console window with some additions. Open **Certificates**:

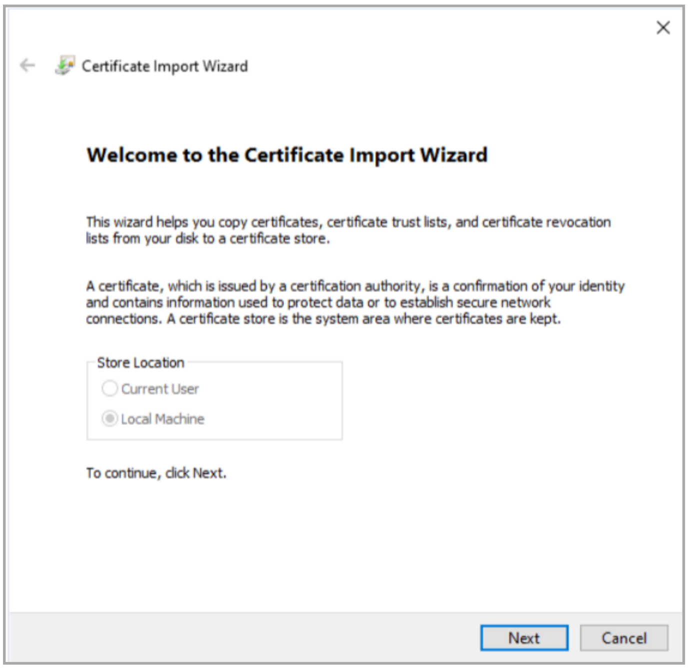

Console1 - [Console Root\Certificates (Local Computer)\Personal\Certificates]
File
Action
View
Favorites
Window
Help
Console Root
Certificates (Local Computer)
Personal
Certificates
Trusted Root Certification Authorities
Enterprise Trust
Intermediate Certification Authorities
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authoritie
Trusted People
Client Authentication Issuers
Preview Build Roots
Test Roots
Remote Desktop
Runtime_Transport_Store_D2E68232-6F4
Smart Card Trusted Roots
Trusted Packaged App Installation Auth
Trusted Devices
Issued To
Issued By
svc.cluster.local
svc.cluster.local
Windows Azure CRP Certificate ...
Windows Azure CRP Certificate G...
Actions
Certificates
More Actions

Personal store contains 2 certificates.

9. Open **Certificates>All Tasks>Import**.



10. The **Certificate Import Wizard** opens:



Certificate Import Wizard

Welcome to the Certificate Import Wizard

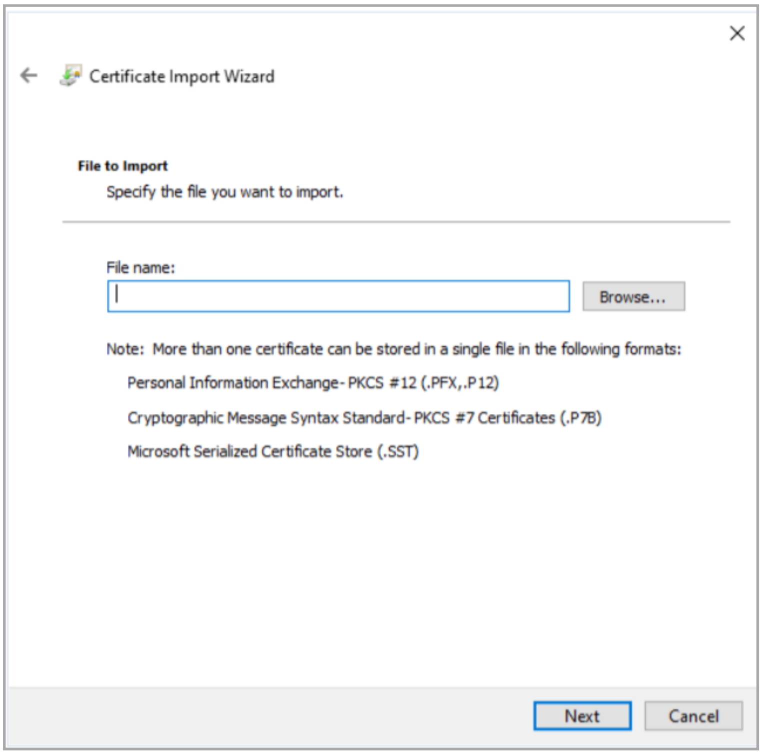
This wizard helps you copy certificates, certificate trust lists, and certificate revocation lists from your disk to a certificate store.

A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.

Store Location
○ Current User
● Local Machine

To continue, click Next.

11. Click **Next**. You are asked for a file to import:



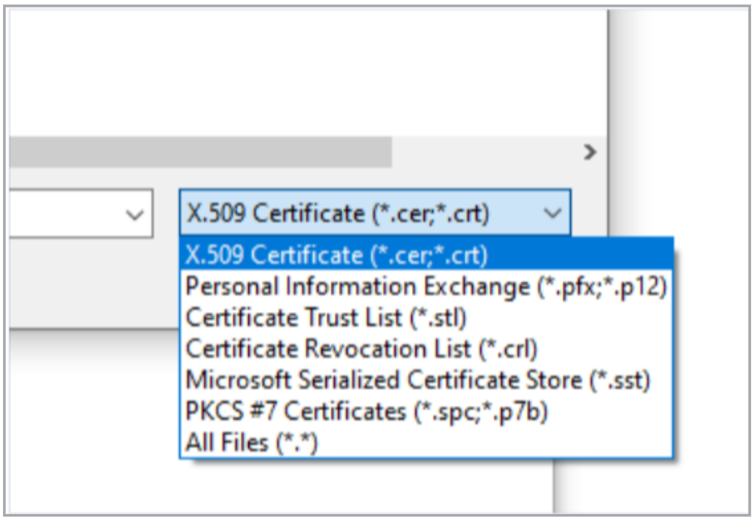
Certificate Import Wizard

File to Import
Specify the file you want to import.

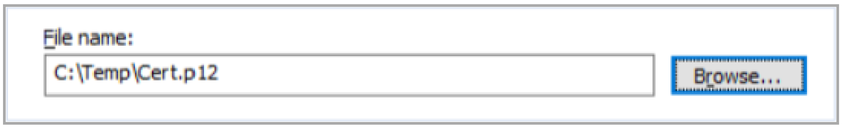
File name:

Note: More than one certificate can be stored in a single file in the following formats:
Personal Information Exchange- PKCS #12 (.PFX,.P12)
Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)
Microsoft Serialized Certificate Store (.SST)

12. Use the Browse button to locate your certificate file. Notice that it allows specific file types:

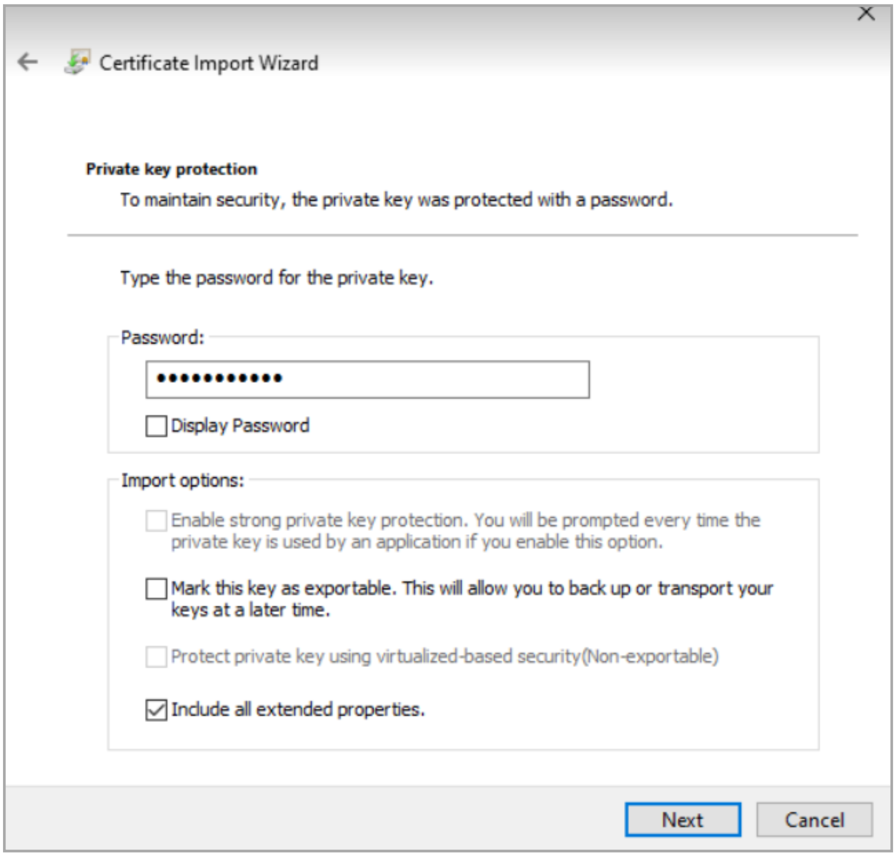

X.509 Certificate (*.cer;*.crt)
X.509 Certificate (*.cer;*.crt)
Personal Information Exchange (*.pfx;*.p12)
Certificate Trust List (*.stl)
Certificate Revocation List (*.crl)
Microsoft Serialized Certificate Store (*.sst)
PKCS #7 Certificates (*.spc;*.p7b)
All Files (*.*)

13. Select your file type: In our example we use a **.p12** file type.



File name:
C:\Temp\Cert.p12

14. Click **Next**. You are required to enter the certificate password:



Private key protection

To maintain security, the private key was protected with a password.

Type the password for the private key.

Password:
••••••••••••

☐ Display Password

Import options:

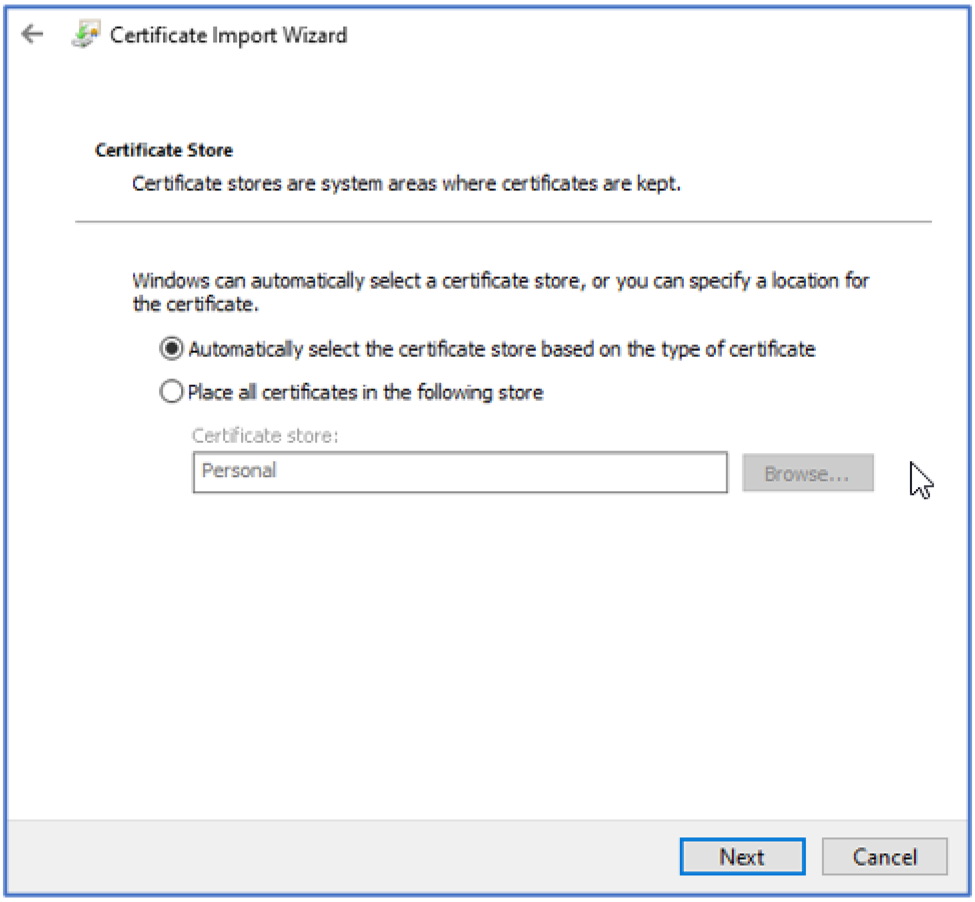
☐ Enable strong private key protection. You will be prompted every time the private key is used by an application if you enable this option.

☐ Mark this key as exportable. This will allow you to back up or transport your keys at a later time.

☐ Protect private key using virtualized-based security(Non-exportable)

☑ Include all extended properties.

15. Click **Next**. Chose where to store the certificate:



Certificate Import Wizard

Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

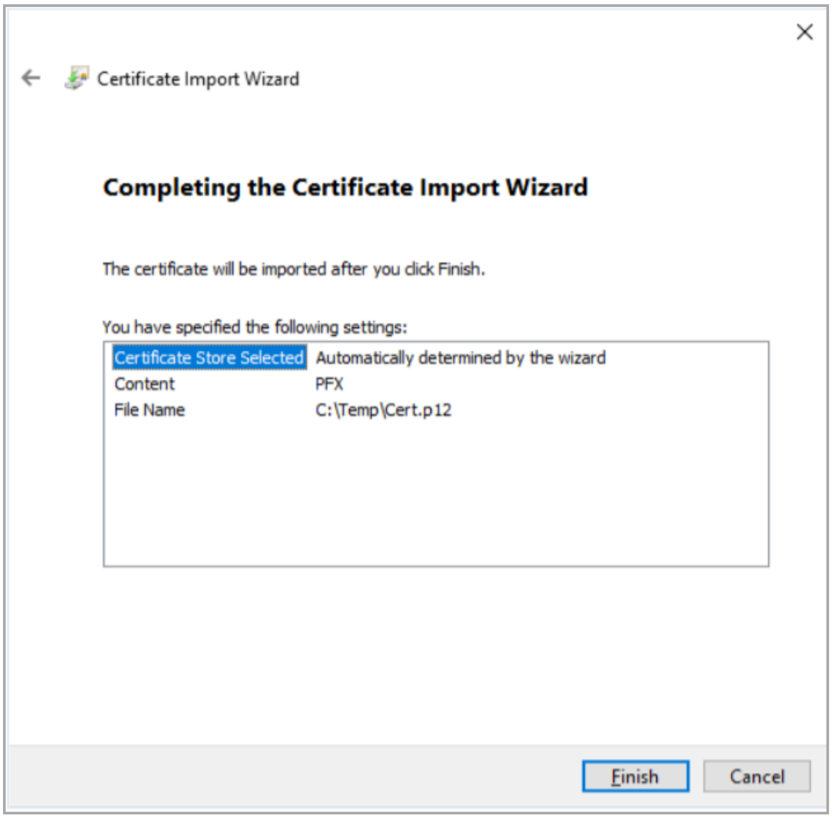
● Automatically select the certificate store based on the type of certificate

○ Place all certificates in the following store

Certificate store:

Personal

16. Click **Next**. The Wizard confirms your import details:



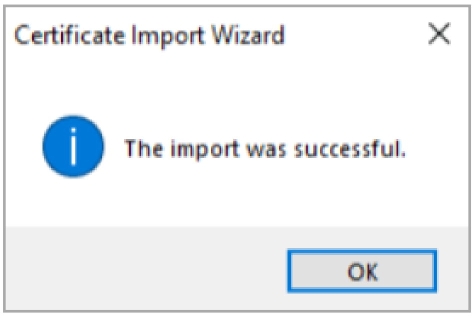
Certificate Import Wizard

Completing the Certificate Import Wizard

The certificate will be imported after you click Finish.

You have specified the following settings:

Certificate Store Selected	Automatically determined by the wizard
Content	PFX
File Name	C:\Temp\Cert.p12



The import was successful.

This completes the certificate import procedure

3.0.2.2. Configuring Windows Machines to Resolve the Postgres Host Name

ThetaRay requires Postgres to be accessed using the internal host name associated with the OpenShift service used to access the database from within the cluster, **postgres.<namespace>.svc.cluster.local**. This requires setting up the windows hosts file to resolve Postgres's host name to the internal load balancer IP address.

For example:

Item	Value Used in this Guide
Internal host name	postgres.dmitryvi-shared-master-1682850588-2015.svc.cluster.local
IP address	80.8.28.39

3.0.2.3. Setting Up a Postgres Load Balancer

A postgress Load balancer has to be installed on a shared environment where postgress db runs, The installation will be provided by customer support team .

However, When the service is run, it provides an external IP address, for example:

Item	Value Used in this Guide
Server URL	*<postgres.dmitryvi-shared-master-1682850588-2015.svc.cluster.local>*
IP address	<80.8.28.39>

These must be added to the hosts file, located here:

```
   c:\Windows\System32\drivers\etc\hosts
```

Here is an extract:

```
 # localhost name resolution is handled within DNS itself.
```

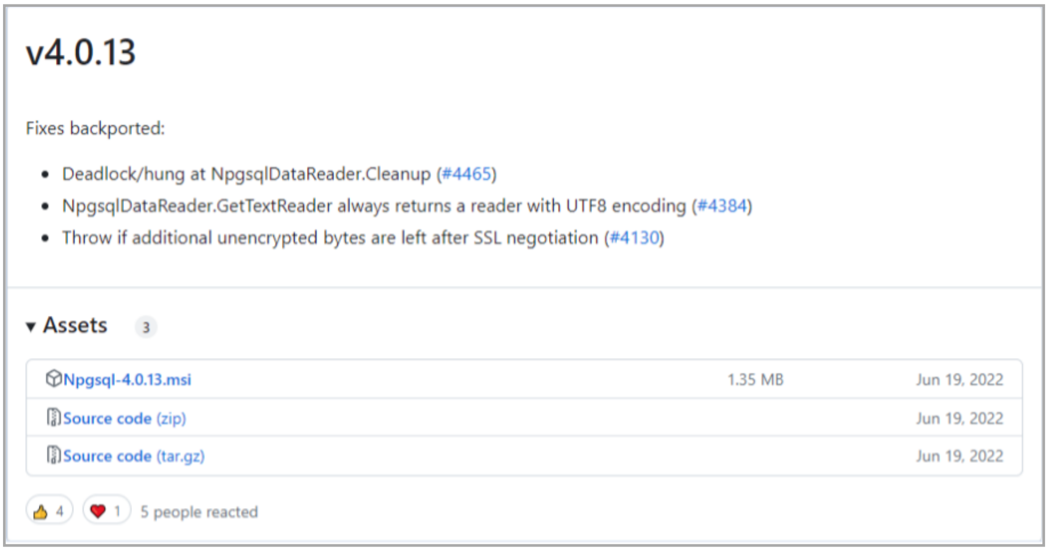
```
#    127.0.0.1       localhost
#    ::1             localhost
    80.8.28.39    postgres.dmitryvi-shared-master-1682850588-
2015.svc.cluster.local
```

Installing Npgssql

Npgsql is the .NET data provider for PostgreSQL.

To install Npgsql:

1. Go to https://github.com/npgsql/npgsql. On the right side of the page, click **Releases**. The most recent 4.0.x release with a Windows msi installer is 4.0.13.
2. Enter it in the top right Find a release search bar or just scroll down to find it. Click Assets:



v4.0.13

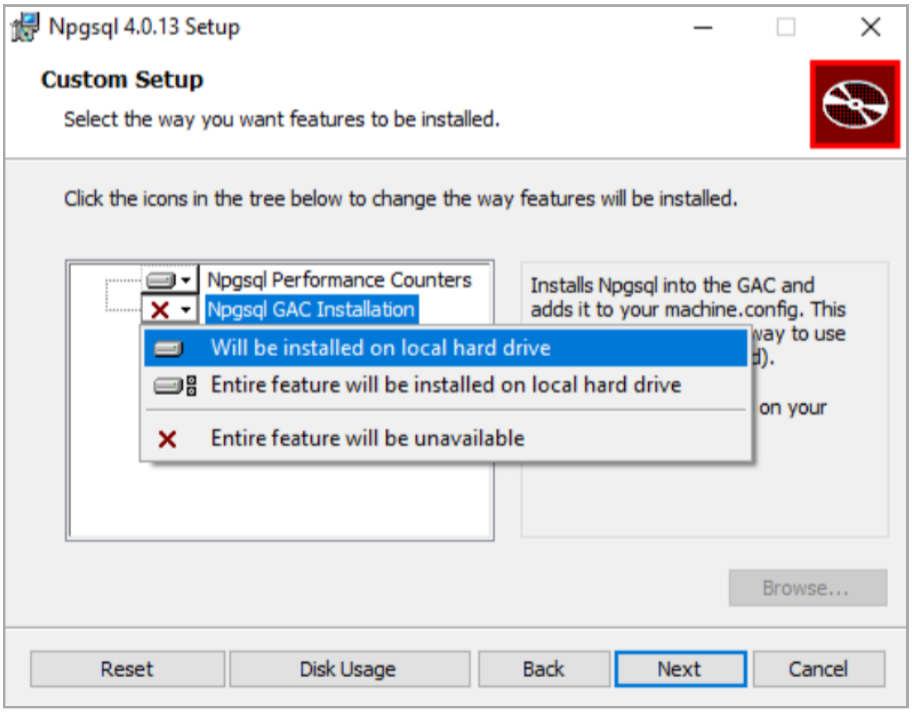
Fixes backported:

- Deadlock/hung at NpgsqlDataReader.Cleanup (#4465)
- NpgsqlDataReader.GetTextReader always returns a reader with UTF8 encoding (#4384)
- Throw if additional unencrypted bytes are left after SSL negotiation (#4130)

▾ Assets 3

Npgsql-4.0.13.msi	1.35 MB	Jun 19, 2022
Source code (zip)		Jun 19, 2022
Source code (tar.gz)		Jun 19, 2022

3. Click the *Npgsql-4.0.13.msi* link to download the installer.
4. Run **Npgsql.exe**. It is a standard Windows installer and we only point out special details.
5. In the **Custom Setup** window, ensure that **Npgsql GAC Installation** is selected as follows:



Npgsql 4.0.13 Setup

Custom Setup

Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

Installs Npgsql into the GAC and adds it to your machine.config. This
way to use
d).
on your

Npgsql Performance Counters
Npgsql GAC Installation
Will be installed on local hard drive
Entire feature will be installed on local hard drive
Entire feature will be unavailable
Browse...
Reset
Disk Usage
Back
Next
Cancel

6. Complete the installation in the usual way.

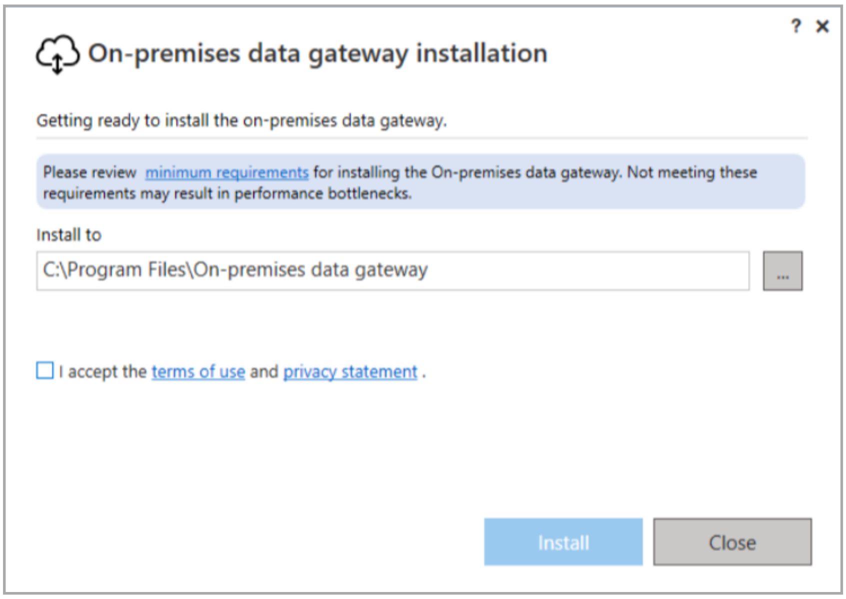
3.0.2.4. Installing GatewayInstall

GatewayInstall acts as a bridge. It provides quick and secure data transfer between on-premises data (data that is **not** in the cloud), and several Microsoft cloud services such as Power BI Pro.

Note: GatewayInstall is only needed for virtual machines accessing the database using the browser (web) interface.

To use GatewayInstall:

1. Go to the Microsoft Download Center, look for and download **Gatewayinstall.exe**.
2. Run **Gatewayinstall.exe**.



On-premises data gateway installation

Getting ready to install the on-premises data gateway.

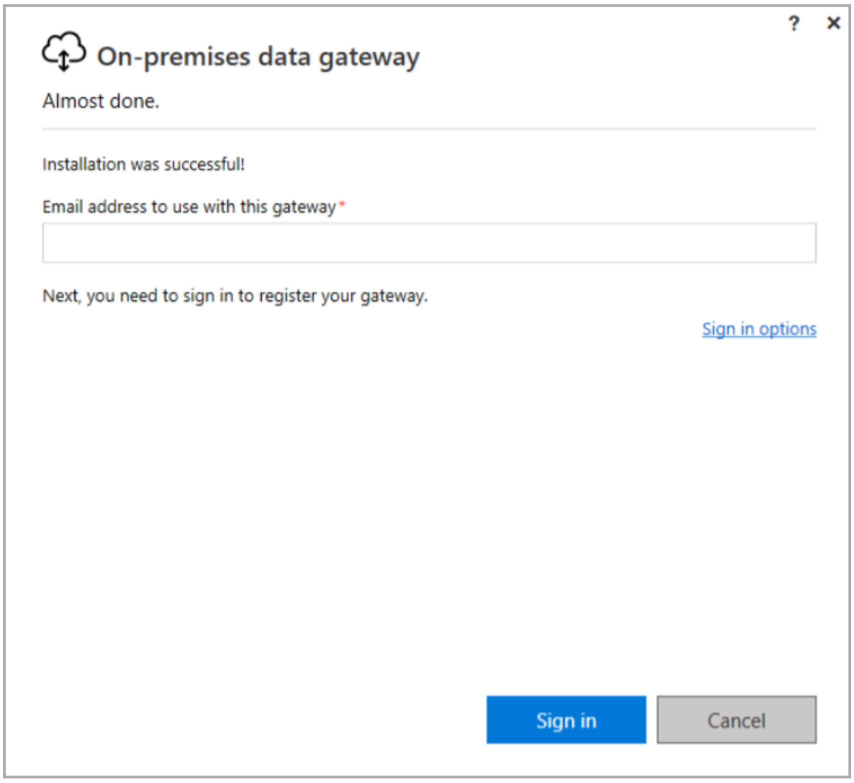
Please review minimum requirements for installing the On-premises data gateway. Not meeting these requirements may result in performance bottlenecks.

Install to

C:\Program Files\On-premises data gateway

□ I accept the terms of use and privacy statement .

3. Accept the terms of use and click **Install**.
4. You will be required to sign in with an email address to be used with this gateway:



On-premises data gateway

Almost done.

Installation was successful!

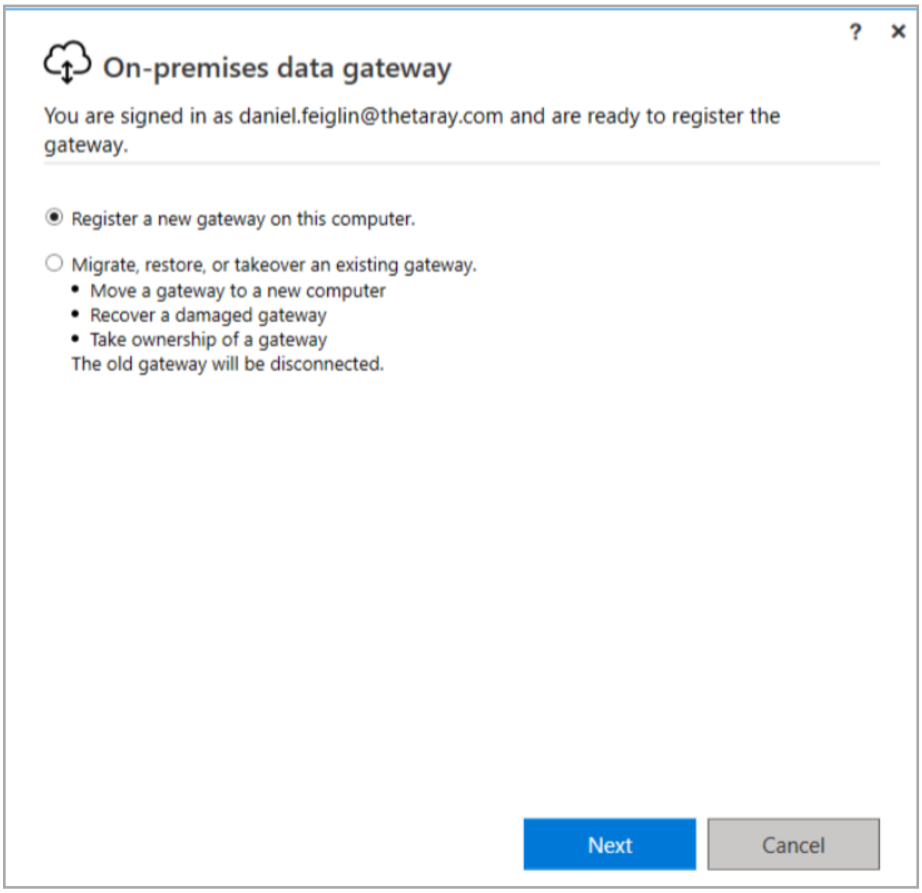
Email address to use with this gateway*

Next, you need to sign in to register your gateway.

The email to enter here is that used for the Power BI Pro license.

You will be redirected to a sign-in to MS Azure.

5. Register the gateway:



On-premises data gateway

You are signed in as daniel.feiglin@thetaray.com and are ready to register the gateway.

● Register a new gateway on this computer.

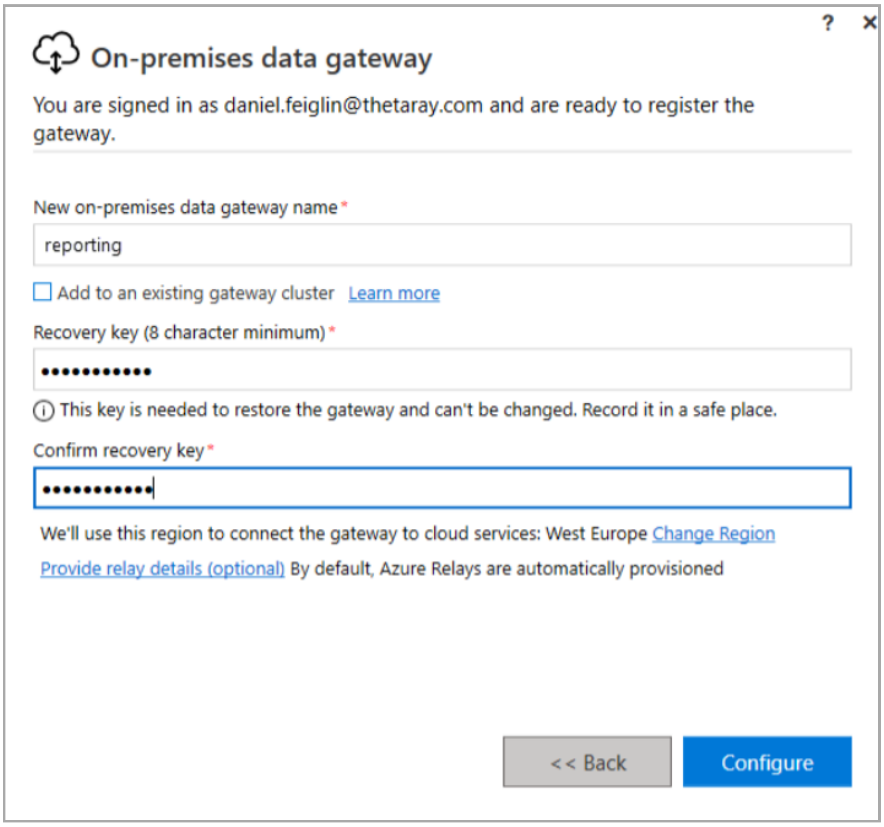
○ Migrate, restore, or takeover an existing gateway.

- Move a gateway to a new computer
- Recover a damaged gateway
- Take ownership of a gateway

The old gateway will be disconnected.

Click **Next**.

6. Provide on-premises gateway details:



On-premises data gateway

You are signed in as daniel.feiglin@thetaray.com and are ready to register the gateway.

New on-premises data gateway name *

reporting

□ Add to an existing gateway cluster Learn more

Recovery key (8 character minimum) *

●●●●●●●●●●●●

ⓘ This key is needed to restore the gateway and can't be changed. Record it in a safe place.

Confirm recovery key *

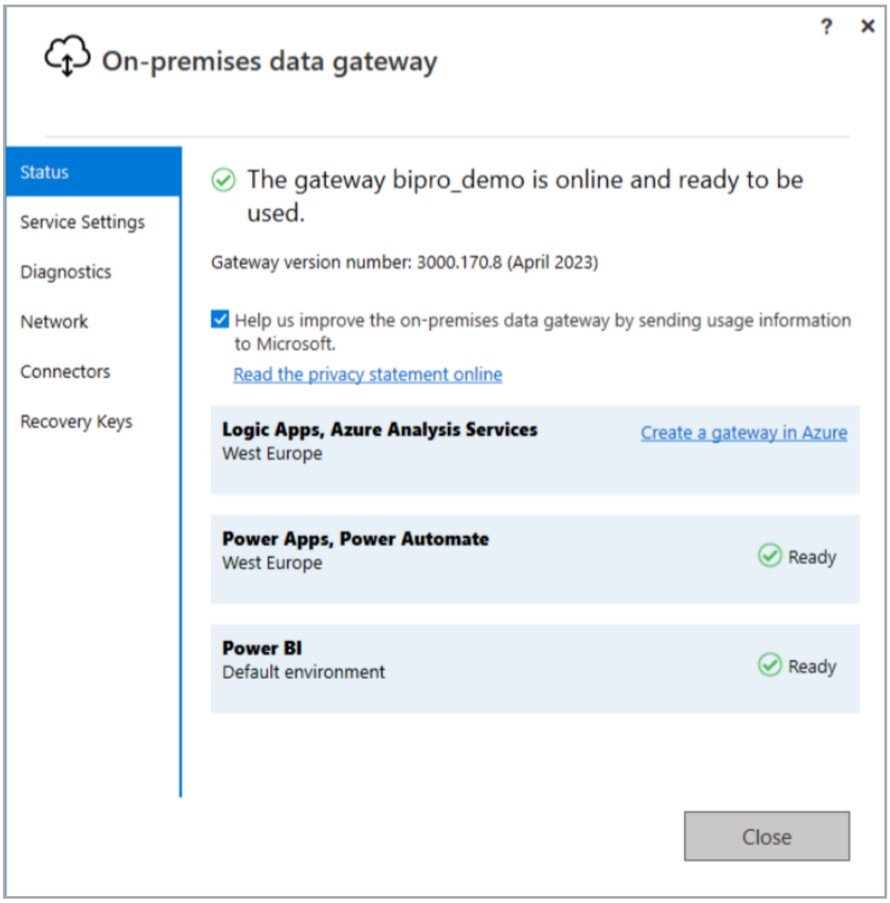
●●●●●●●●●●●●

We'll use this region to connect the gateway to cloud services: West Europe Change Region

Provide relay details (optional) By default, Azure Relays are automatically provisioned

Click **Configure**.

The installation completes with a confirmatory window:



The gateway bipro_demo is online and ready to be used.

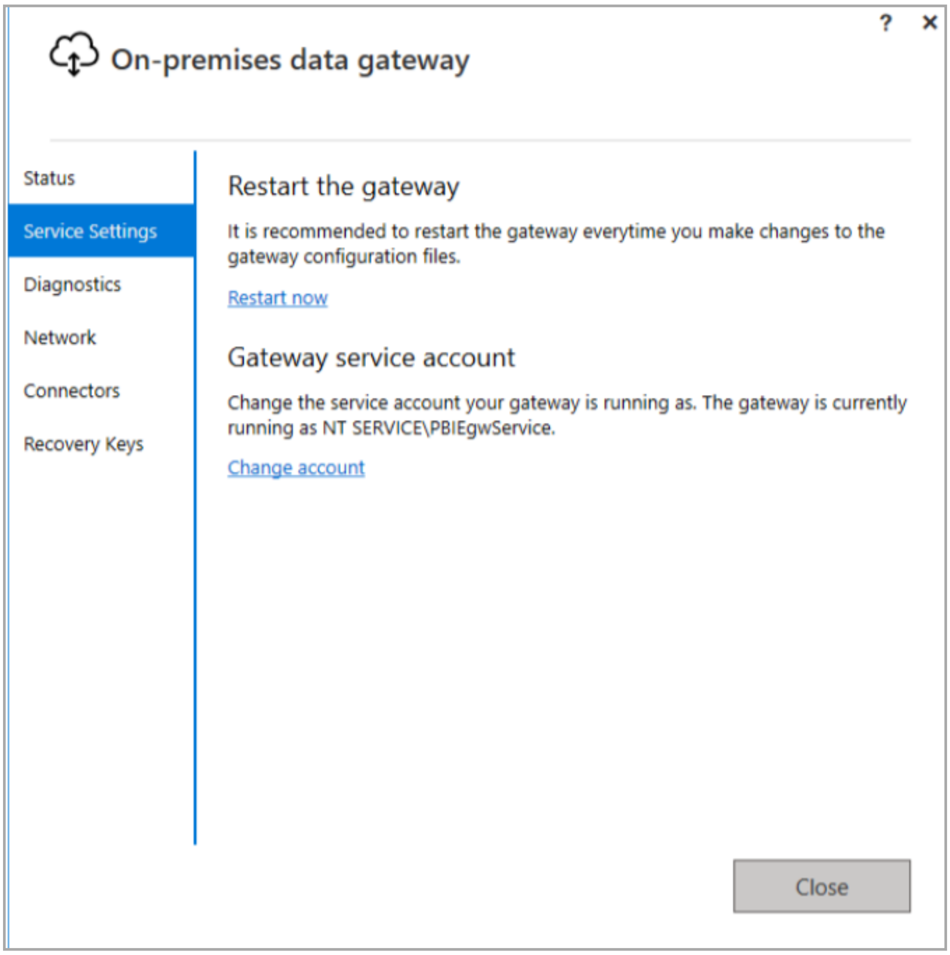
Gateway version number: 3000.170.8 (April 2023)

☑ Help us improve the on-premises data gateway by sending usage information to Microsoft.

Read the privacy statement online

Logic Apps, Azure Analysis Services West Europe	Create a gateway in Azure
Power Apps, Power Automate West Europe	✓ Ready
Power BI Default environment	✓ Ready

7. Have a look at the right-hand menu settings:
   a. Note first item in **Service Settings**:



Restart the gateway

It is recommended to restart the gateway everytime you make changes to the gateway configuration files.

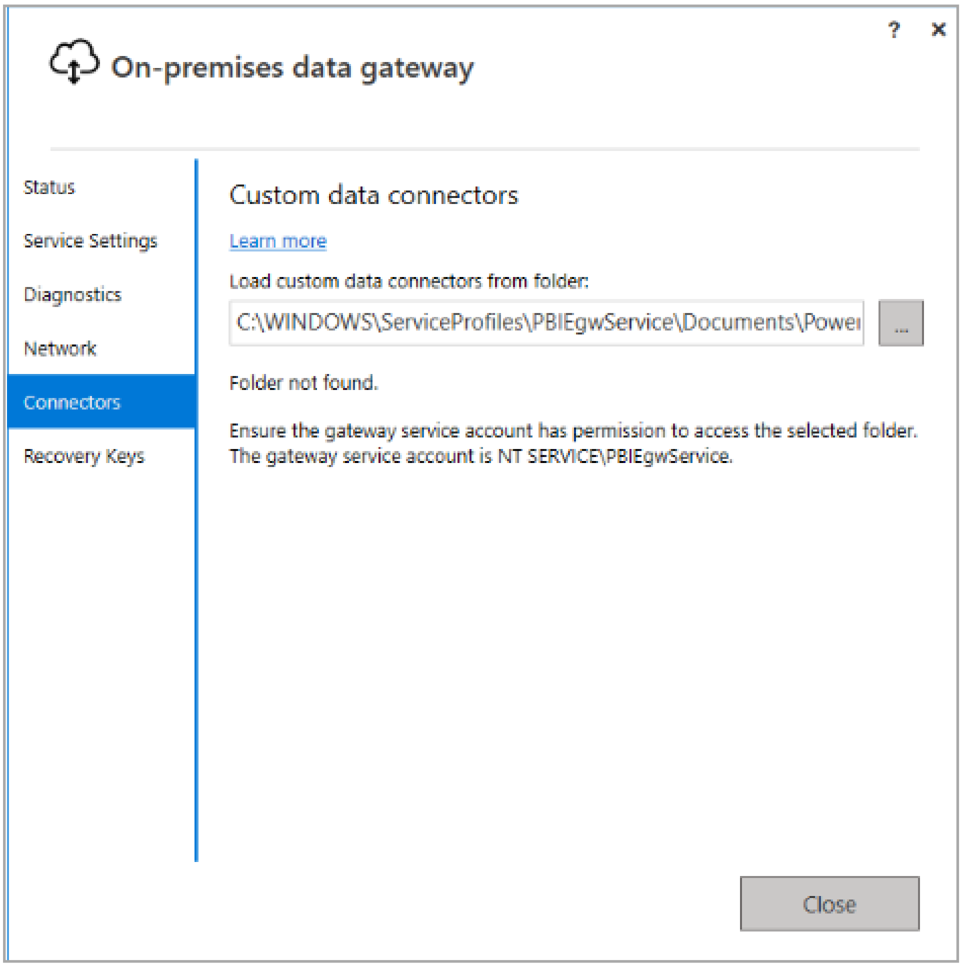
Restart now

Gateway service account

Change the service account your gateway is running as. The gateway is currently running as NT SERVICE\PBIEgwService.

Change account

b. In **Connectors** there is an issue:



Load custom data connectors from folder: C:\WINDOWS\ServiceProfiles\PBIEgwService\Documents\Powei

Folder not found.

Ensure the gateway service account has permission to access the selected folder. The gateway service account is NT SERVICE\PBIEgwService.

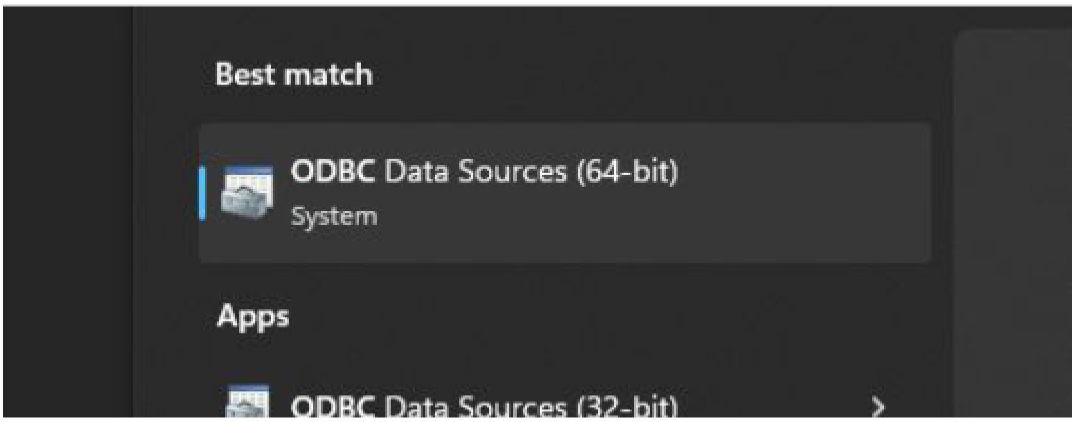
You may ignore this issue for now.

TIP: A desktop icon is not generated but the program is placed in the system Startup Menu programs. If that is not suitable, go to the application in **C:\Program Files\On-premises data gateway**, and create a desktop shortcut for **Microsoft.PowerBI.EnterpriseGateway.exe.**

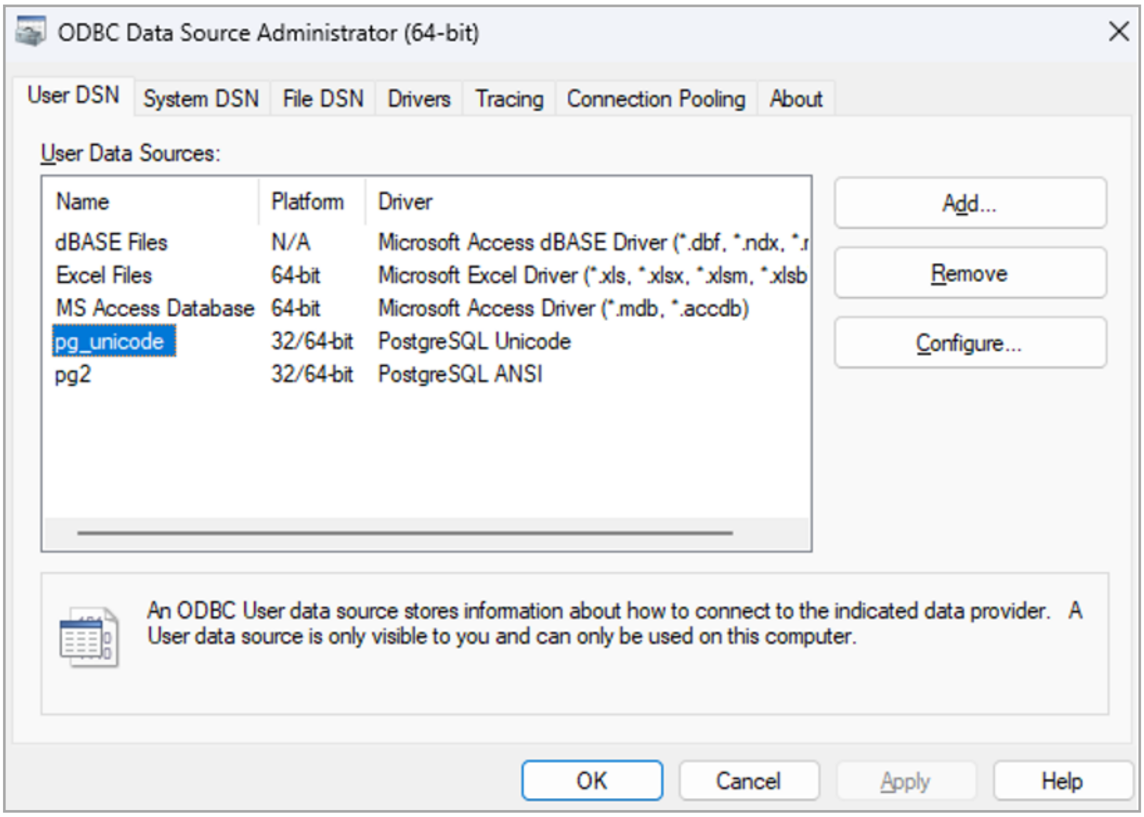
Appendix B - Connecting through Microsoft Excel

Access to the ThetaRay Reporting Database from Microsoft Excel running on a Windows machine, is enabled through the Postgres ODBC Driver. The following steps are required to configure ODBC connectivity to the reporting database:

1. Download and install the latest Postgres ODBC driver from - https://www.postgresql.org/ftp/odbc/versions/msi/
2. Configure an ODBC connection to the reporting database.
3. Launch the ODBC Data Sources Administrator application.



4. Configure a connection using the PostgreSQL Unicode driver.

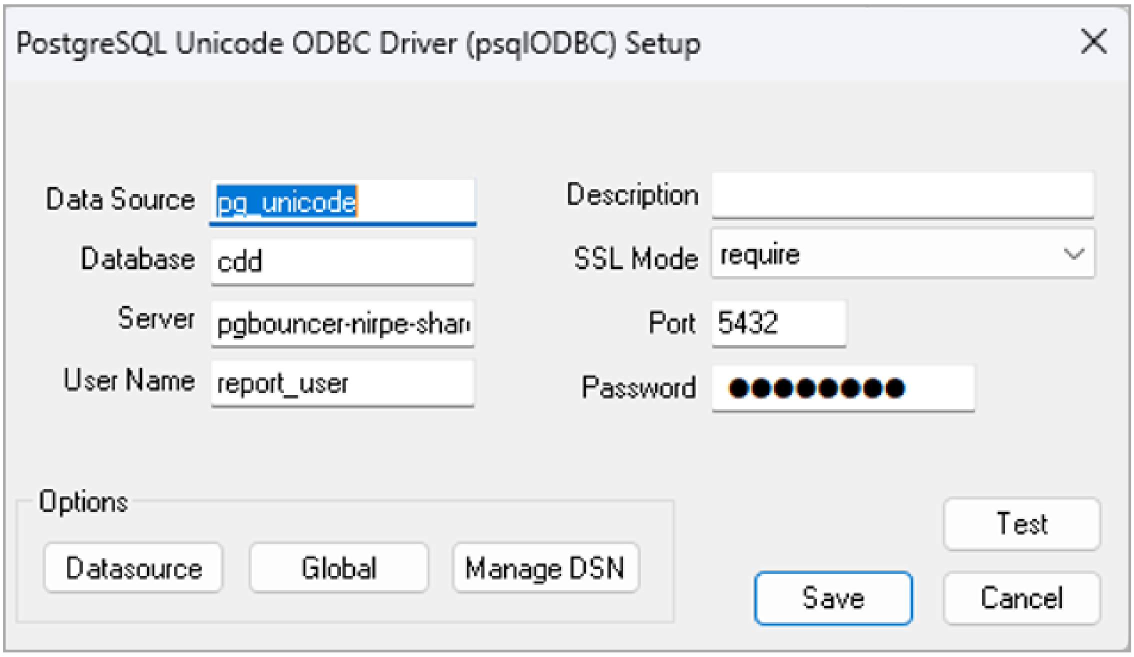


User Data Sources:

Name	Platform	Driver
dBASE Files	N/A	Microsoft Access dBASE Driver (*.dbf, *.ndx, *.r
Excel Files	64-bit	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.xlsb
MS Access Database	64-bit	Microsoft Access Driver (*.mdb, *.accdb)
pg_unicode	32/64-bit	PostgreSQL Unicode
pg2	32/64-bit	PostgreSQL ANSI

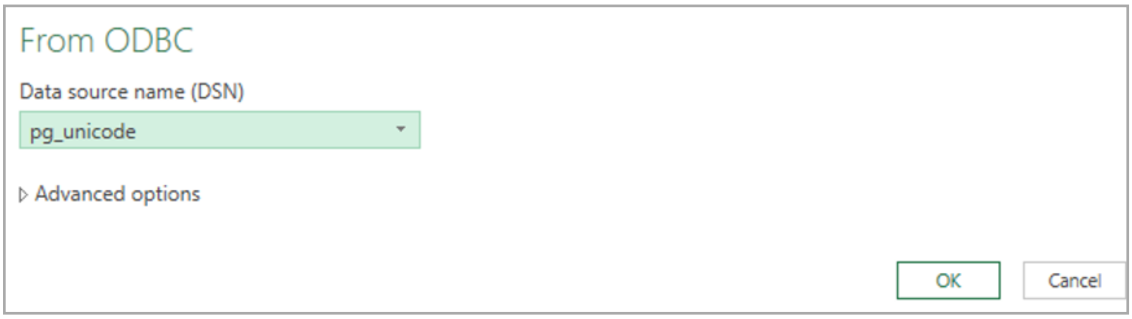
An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you and can only be used on this computer.

5. Click **Add**, and configure a connection based on the hostname and user credentials provided by ThetaRay.



Data Source: pg_unicode
Database: cdd
Server: pgbouncer-nirpe-shar
User Name: report_user
Description:
SSL Mode: require
Port: 5432
Password: ●●●●●●●●

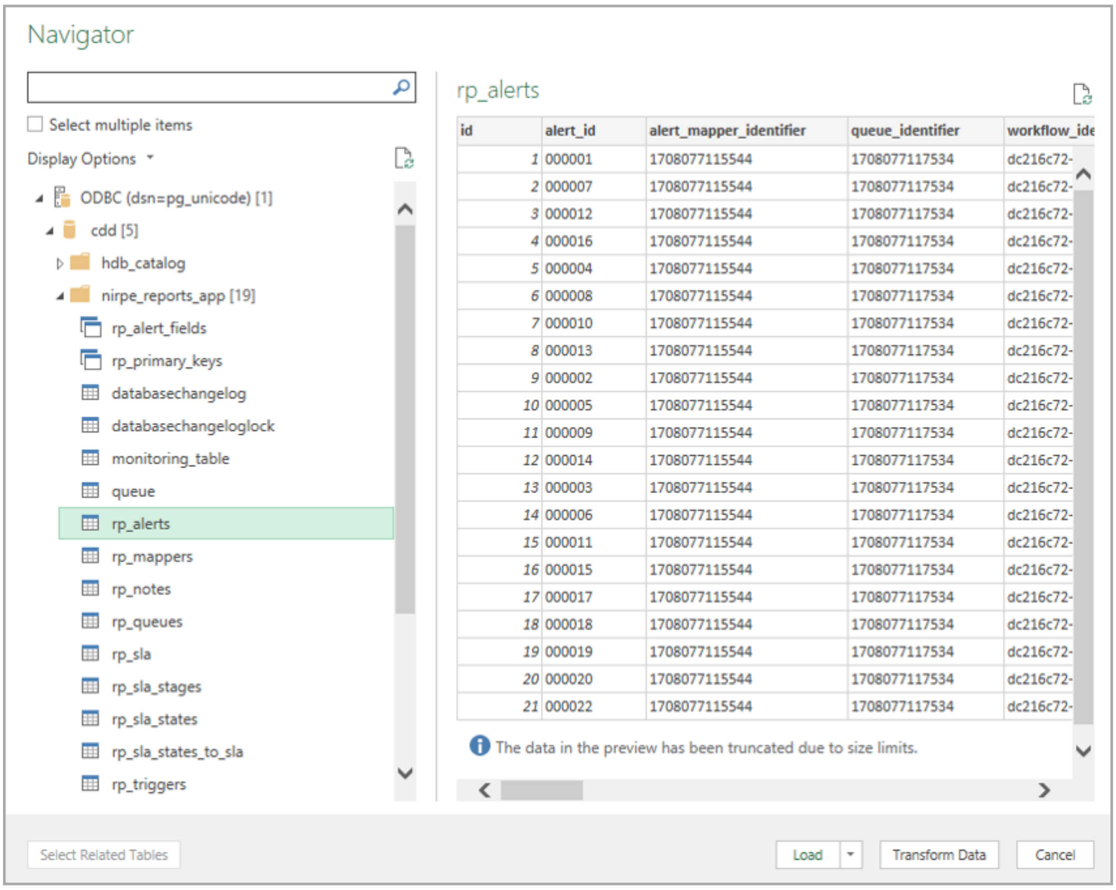
6. From Microsoft Excel, click Data -> Get Data -> From Other Sources -> From ODBC and select the previously configured ODBC data source.



From ODBC

Data source name (DSN): pg_unicode

7. Excel will provide you the ability to populate an Excel sheet, with data queried from the reporting database (custom SQL or visual transformations can be applied through Excel as needed).



Navigator

◢ ODBC (dsn=pg_unicode) [1]
◢ cdd [5]
▷ hdb_catalog
◢ nirpe_reports_app [19]
rp_alert_fields
rp_primary_keys
databasechangelog
databasechangeloglock
monitoring_table
queue
rp_alerts
rp_mappers
rp_notes
rp_queues
rp_sla
rp_sla_stages
rp_sla_states
rp_sla_states_to_sla
rp_triggers

rp_alerts

id	alert_id	alert_mapper_identifier	queue_identifier	workflow_ide
1	000001	1708077115544	1708077117534	dc216c72-
2	000007	1708077115544	1708077117534	dc216c72-
3	000012	1708077115544	1708077117534	dc216c72-
4	000016	1708077115544	1708077117534	dc216c72-
5	000004	1708077115544	1708077117534	dc216c72-
6	000008	1708077115544	1708077117534	dc216c72-
7	000010	1708077115544	1708077117534	dc216c72-
8	000013	1708077115544	1708077117534	dc216c72-
9	000002	1708077115544	1708077117534	dc216c72-
10	000005	1708077115544	1708077117534	dc216c72-
11	000009	1708077115544	1708077117534	dc216c72-
12	000014	1708077115544	1708077117534	dc216c72-
13	000003	1708077115544	1708077117534	dc216c72-
14	000006	1708077115544	1708077117534	dc216c72-
15	000011	1708077115544	1708077117534	dc216c72-
16	000015	1708077115544	1708077117534	dc216c72-
17	000017	1708077115544	1708077117534	dc216c72-
18	000018	1708077115544	1708077117534	dc216c72-
19	000019	1708077115544	1708077117534	dc216c72-
20	000020	1708077115544	1708077117534	dc216c72-
21	000022	1708077115544	1708077117534	dc216c72-

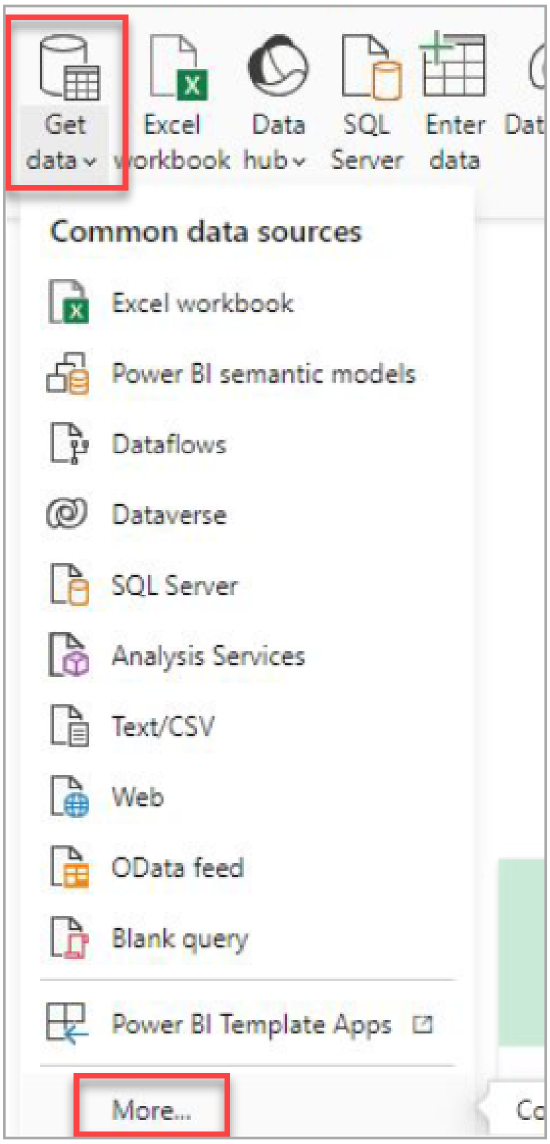
ⓘ The data in the preview has been truncated due to size limits.

Appendix C - Connecting through PowerBI Desktop

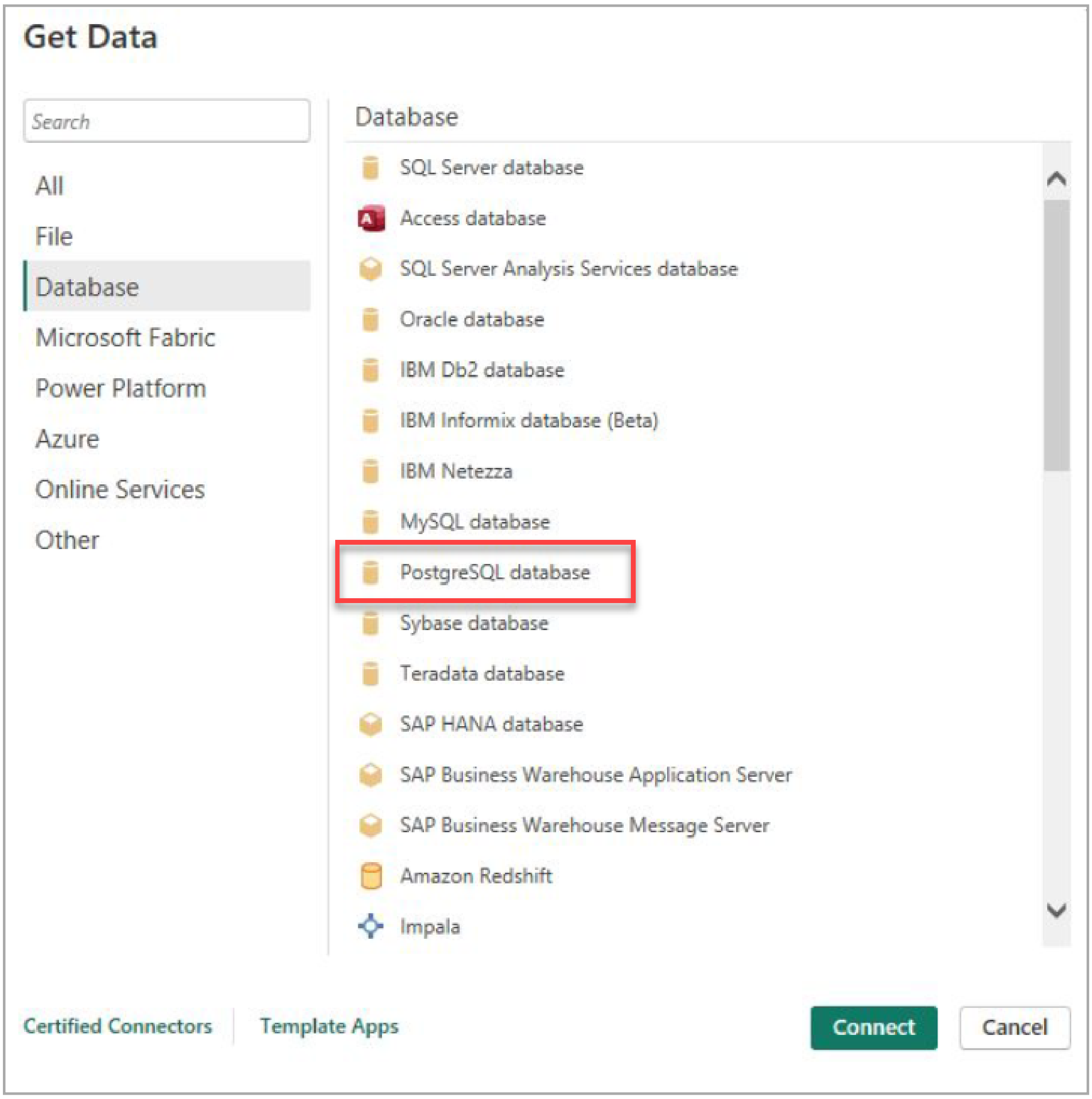
PowerBI Desktop is a Windows desktop application enabling authoring and execution of PowerBI reports in a single user environment.

To connect PowerBI Desktop to the ThetaRay Reporting Database, the following steps should be applied:

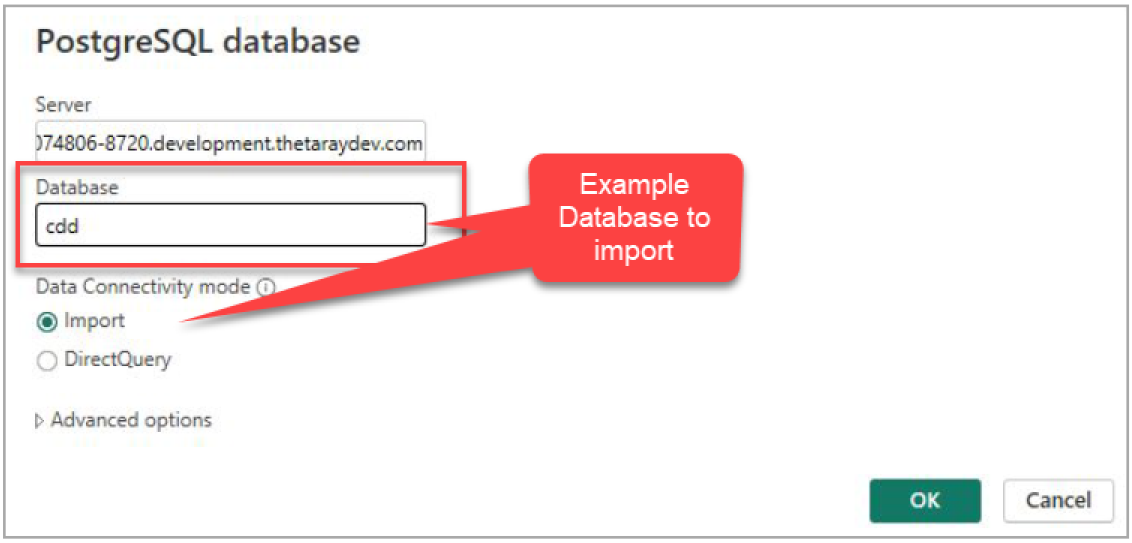
1. Select Get Data -> More from the Toolbar.



2. Select PostgreSQL within the dialog.



3. Provide the databases connection details and select whether to use theImport or Direct Query mode of PowerBI.



PostgreSQL database

Server

)74806-8720.development.thetaraydev.com

Database

cdd

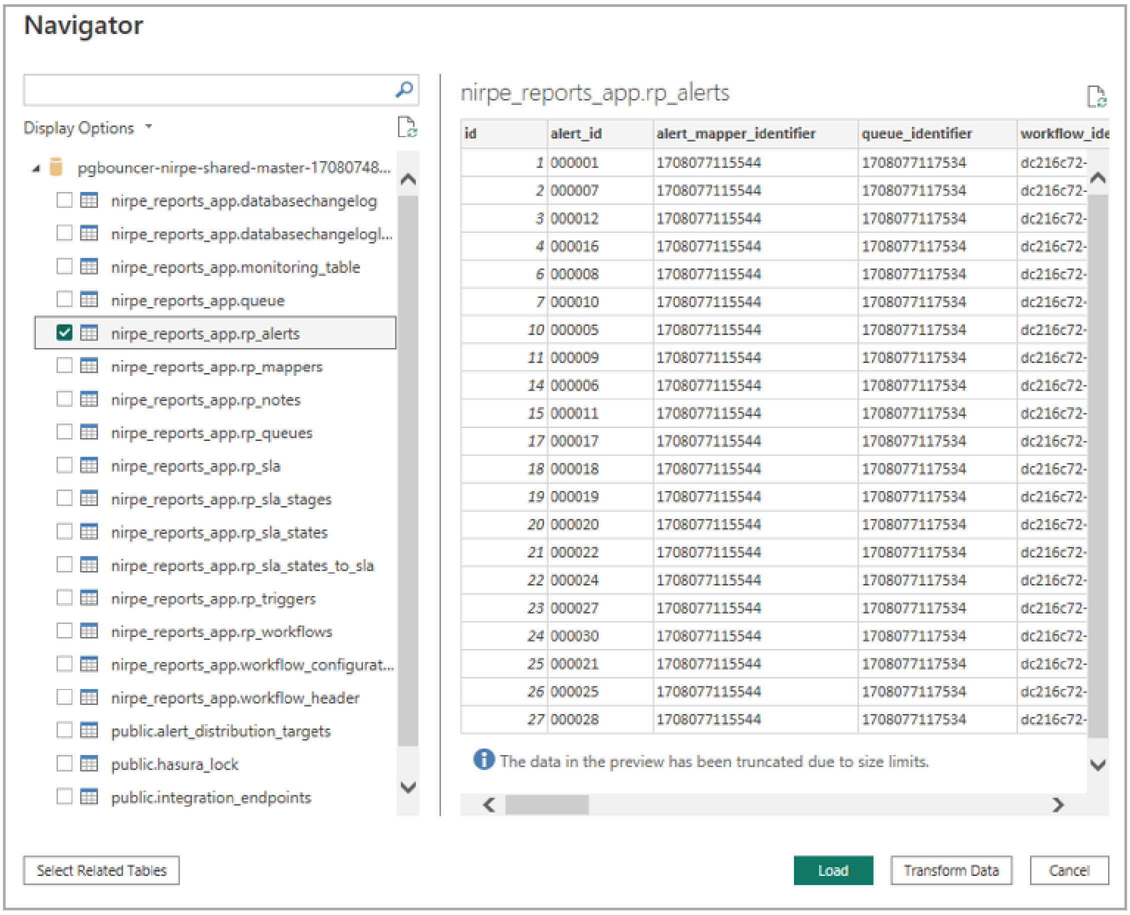
Data Connectivity mode ⓘ

● Import

○ DirectQuery

▷ Advanced options

4. Select the relevant Reporting Database tables to be included in the PowerBI report.



Navigator

Display Options ▾

◢ pgbouncer-nirpe-shared-master-17080748...

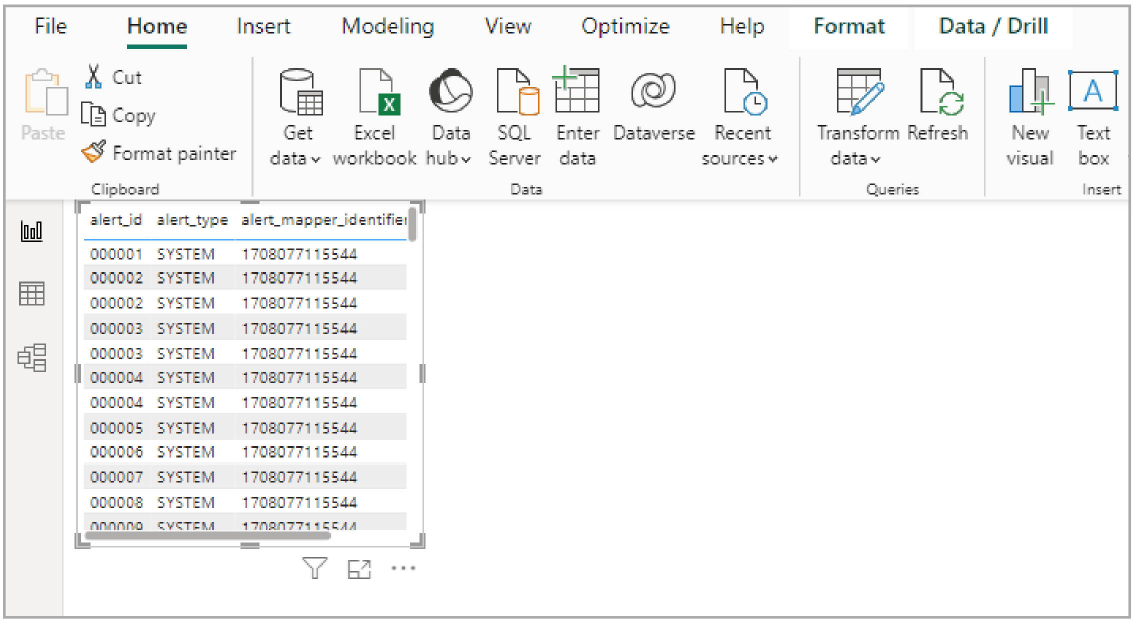
- □ nirpe_reports_app.databasechangelog
- □ nirpe_reports_app.databasechangelogl...
- □ nirpe_reports_app.monitoring_table
- □ nirpe_reports_app.queue
- ☑ nirpe_reports_app.rp_alerts
- □ nirpe_reports_app.rp_mappers
- □ nirpe_reports_app.rp_notes
- □ nirpe_reports_app.rp_queues
- □ nirpe_reports_app.rp_sla
- □ nirpe_reports_app.rp_sla_stages
- □ nirpe_reports_app.rp_sla_states
- □ nirpe_reports_app.rp_sla_states_to_sla
- □ nirpe_reports_app.rp_triggers
- □ nirpe_reports_app.rp_workflows
- □ nirpe_reports_app.workflow_configurat...
- □ nirpe_reports_app.workflow_header
- □ public.alert_distribution_targets
- □ public.hasura_lock
- □ public.integration_endpoints

nirpe_reports_app.rp_alerts

id	alert_id	alert_mapper_identifier	queue_identifier	workflow_ide
1	000001	1708077115544	1708077117534	dc216c72-
2	000007	1708077115544	1708077117534	dc216c72-
3	000012	1708077115544	1708077117534	dc216c72-
4	000016	1708077115544	1708077117534	dc216c72-
6	000008	1708077115544	1708077117534	dc216c72-
7	000010	1708077115544	1708077117534	dc216c72-
10	000005	1708077115544	1708077117534	dc216c72-
11	000009	1708077115544	1708077117534	dc216c72-
14	000006	1708077115544	1708077117534	dc216c72-
15	000011	1708077115544	1708077117534	dc216c72-
17	000017	1708077115544	1708077117534	dc216c72-
18	000018	1708077115544	1708077117534	dc216c72-
19	000019	1708077115544	1708077117534	dc216c72-
20	000020	1708077115544	1708077117534	dc216c72-
21	000022	1708077115544	1708077117534	dc216c72-
22	000024	1708077115544	1708077117534	dc216c72-
23	000027	1708077115544	1708077117534	dc216c72-
24	000030	1708077115544	1708077117534	dc216c72-
25	000021	1708077115544	1708077117534	dc216c72-
26	000025	1708077115544	1708077117534	dc216c72-
27	000028	1708077115544	1708077117534	dc216c72-

ⓘ The data in the preview has been truncated due to size limits.

5. Continue with the report design with the loaded data.



alert_id	alert_type	alert_mapper_identifier
000001	SYSTEM	1708077115544
000002	SYSTEM	1708077115544
000002	SYSTEM	1708077115544
000003	SYSTEM	1708077115544
000003	SYSTEM	1708077115544
000004	SYSTEM	1708077115544
000004	SYSTEM	1708077115544
000005	SYSTEM	1708077115544
000006	SYSTEM	1708077115544
000007	SYSTEM	1708077115544
000008	SYSTEM	1708077115544
000009	SYSTEM	1708077115544