TERADATA®

What would you do if you knew?™

Teradata REST Services

Installation, Configuration, and Upgrade Guide for Customers

Release 15.00
B035-2700-104K
April 2016

TERADATA LABS

Table of Contents

Appendix B: Teradata REST Metrics

Preface

Purpose

This guide provides instructions for the following:

- Installing the REST container package
- Configuring options in order to use the REST API for Teradata Database service
- Configuring options for your REST API services
- Troubleshooting with log files

Audience

This guide is intended for use by:

- Application Developers
- Information Technology Operations
- System Administrators

Revision History

Date	Release	Description
April 2016	15.00	Recorded support for Teradata Database 15.10.
January 2016	15.00	Added content to support the following: • New advanced settings • Installing a PKCS #12 formatted private key and public key certificate chain
October 2014	15.00	Initial release.

Additional Information

Related Documents

Documents are located at http://www.info.teradata.com.

Title	Publication ID
Parallel Upgrade Tool (PUT) Reference Describes how to install application software using PUT.	B035-5713
Electronic Software Distribution Guide	BCD0-0718-0000

Product Safety Information

This document may contain information addressing product safety practices related to data or property damage, identified by the word *Notice*. A notice indicates a situation which, if not avoided, could result in damage to property, such as equipment or data, but not related to personal injury.

Example

Notice: Improper use of the Reconfiguration utility can result in data loss.

CHAPTER 1

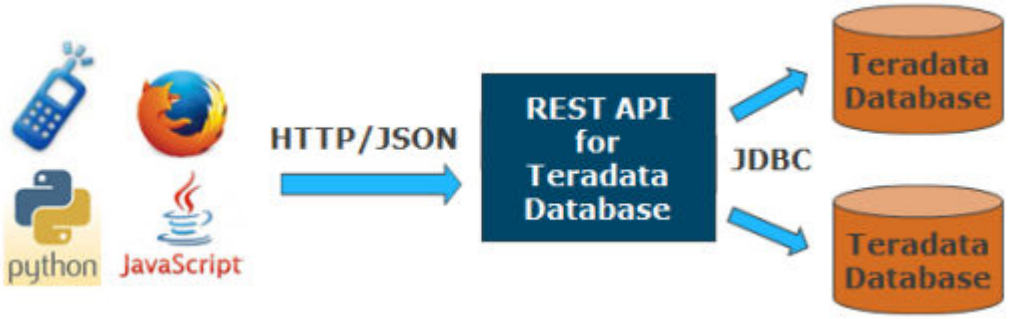
Overview

Teradata REST Services

Teradata REST Services is a container for hosting RESTful web services. The first service deployed in this container is the REST API for Teradata Database.

REST API for Teradata Database

The REST API for Teradata Database service is middleware that provides an HTTP+JSON bridge to the Teradata Database, which allows you to open database sessions, submit SQL queries and access responses, and access metadata. Using the REST API for Teradata Database, you can access a Teradata Database from a web page, a mobile device, or a scripting language of your choice.



Dependencies

Supported Teradata Database Versions

- 15.10
- 15.0
- 14.10
- 14.0
- 13.10
- 13.0

Supported Browsers

- Mozilla Firefox 26
- Internet Explorer 9 or 10
- Google Chrome 32
- Safari 7

Required Software

- Java 7
- SUSE Linux Enterprise Server 10 or 11 *or* Red Hat Enterprise Linux 6

Ports

The ports used by the tdrestd service are:

- 1080 - HTTP
- 1443 - HTTPS

Starting and Stopping the tdrestd Service

The tdrestd service has an `init` script on the server where REST is installed, which is located in the `/etc/init.d` directory. The `init` script supports the following commands.

Command	Description	Example
start	Starts the service. If the service is already running, a new instance is not started.	To start the tdrestd service from the server where REST is installed, type: `/etc/init.d/tdrestd start`
stop	Stops the service. The script completes successfully even when the service is not running.	To stop the tdrestd service from the server where REST is installed, type: `/etc/init.d/tdrestd stop`

CHAPTER 2

Preparing for Software Installation

Downloading Java 7

Java 7, any version, is required to run Teradata REST Services. If you do not already have Java 7, you must download it.

1. Go to Java SE Downloads.
2. Download Java 7.
3. Install according to instructions, prior to installing Teradata REST Services.

   For SLES environments, if the `teradata-jdk7` package is installed Teradata REST Services will use that.

   For RHEL environments or SLES without `teradata-jdk7`,Teradata REST Services will use the value of *$JAVA_HOME* set in `/etc/opt/teradata/rest/setenv.sh` which is initialized to the current value of *$JAVA_HOME* during installation.

   Failing that,Teradata REST Services will attempt to use a version of Java, if found, on `Path` (for example,`/usr/bin/java`).

Downloading the REST Container Package

When you download the Teradata REST package, you can put the package on a local workstation or remote server. If the package is put on a local workstation, it must then be moved to the remote server.

1. On your Windows PC, open a web browser and go to Teradata At Your Service at https://tays.teradata.com and log in.
2. Click **Software** Downloads > Teradata Client.
3. In the left pane, click REST APIs under Teradata REST Services.
4. Select a choice from the API menu.
5. Select a choice from the Version menu.
6. Under Current, select one of the following:

Option	Description
Yes	Select if you want the latest version of the release.

Option	Description
All	Select if you want the initial release and all updates since the release.

7 Click Submit.

8 Select the check box next to the package you want to download.

9 Select a choice for the Download Type.

10 Select whether or not you want the Download Readme Files.

11 Click Submit.

12 Click Download.

CHAPTER 3

Installing Software

Installing the REST Container Package

The service is automatically started after installation and is configured to automatically start on reboot. The service runs as the Linux user, tdrest. If you have installed the latest version of PUT, you can use the Install/Upgrade Software Operation to install Teradata REST Services.

Note: You must be root to install this software.

1. In the directory where the rpm is located, type the following command:

   ```
   rpm -ivh tdrestd-*.rpm
   ```

   rpm installs the package into the default directory `/opt/teradata/rest`, which cannot be changed.

Accessing the Teradata REST Interface

The Teradata REST Services Home view appears when you access the Teradata REST Services interface. You can access to the Service Information view by clicking Overview and to the API Doc view by clicking API Doc.

1. Open a browser.
2. Type the following address:

   `http://`*`hostname`*`:1080`

   where *hostname* is the name of the server where the tdrestd service is installed.

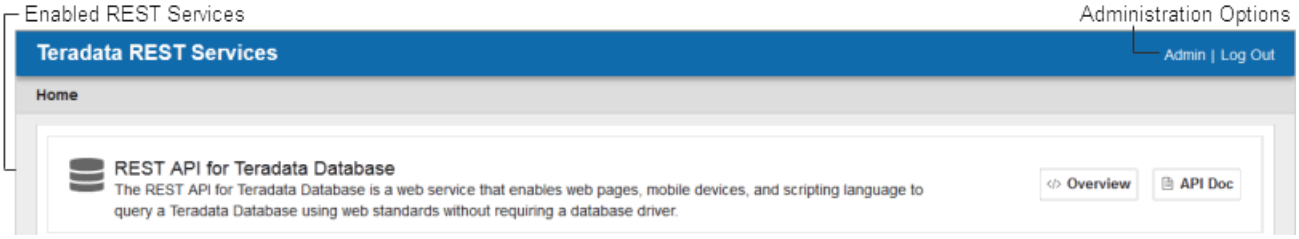
CHAPTER 4

Configuring Administration Options

Home View

The Teradata REST Services Home view displays a list of services that have been enabled.

Following is an example of the Home view.



Administration Options

From the Home view, you can access the administration options that you need to set before you can use the REST API for Teradata Database service. For information on how to log in, see Logging In to the Administration View.

Following is a list of the administration options:

Services

Allows you to enable or disables service and access Service **Configuration** by clicking ⚙ located in the **Configure** column.

Certificate Authority

Allows you to install one or more trusted root certificates that are issued by a trusted certificate authority (CA).

Configure HTTPS

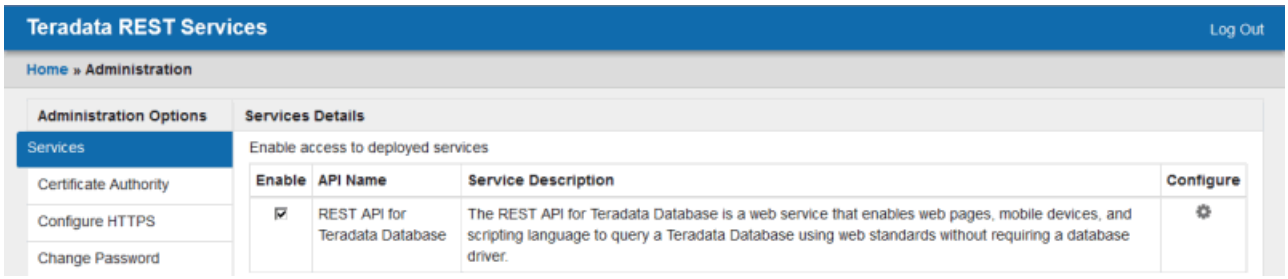
Allows you to do any of the following:

- Create and install a self-signed certificate (SSC).
- Create a certificate signing request (CSR) and install it after it is signed by a CA.
- Install a PKCS #12 file if you want to install a certificate without having to first create a CSR.

Change Password

Allows you to change the admin password.

Following is an example of the Administration view.



Logging In to the Administration View

Teradata REST Services requires a login for the Administration view.

1. In the Teradata REST Services Home view, click Admin in the upper right corner.
2. Enter your password and click Log In.
   The default password is `teradata`.

Enabling or Disabling Access to Deployed Services

Only the REST API for Teradata Database is bundled with the REST service.

1. From the Administration Options list, click Services.
2. Under Services Details, do one of the following:
   - Select the Enable check box in the row of the service you want to enable.
   - Clear the Enable check box in the row of the service you want to disable.
3. Click Apply.

   If enabled, services are added to the list on the Teradata REST Services Home view. If disabled, services are removed from the list on the Home view.

Installing Certificate Authority Certificates

To initiate secure communications with an external server, the Teradata REST service must trust the certificate authority that issued the certificate of the external server. You can install one or more trusted root certificates that are issued by a trusted certificate authority (CA).

1. From the Administration Options list, click **Certificate** Authority.
2. Click Install **Certificate**.
3. Enter an alias for the certificate, up to 30 characters.
4. Browse to select a certificate.

5 Click Install.

The certificate information appears in the Trusted **Certificate** Authorities table.

Deleting Certificate Authority Certificates

1 From the Administration Options list, click **Certificate** Authority.

2 Click ✕ in the Delete column for the certificate authority you want to delete.

A confirmation message appears.

HTTPS Configuration

HTTPS connections are secure connections between a browser and Teradata Database. To enable HTTPS connections, the external server using the browser needs an identifying certificate that provides the public key used to decrypt the information it sends. This certificate may be either self-signed or signed by a Certificate Authority (CA).

To install a new certificate from a CA, you must first generate a Certificate Signing Request and submit it to a CA for signing. Once you receive the signed certificate from the CA you can install it.

Creating a Self-Signed Certificate

A self-signed certificate (SSC) can provide security for an internal site or serve as a temporary measure while a fully authenticated certificate is obtained from a CA. Installing a certificate replaces the existing certificate. After a certificate is replaced, it cannot be reinstalled.

1 From the Administration Options list, click **Configure** HTTPS.

2 [Optional] Select the Require access via HTTPS check box if users will be using a secure connection when accessing the server.

This check box is enabled only when the server is accessed using a secure connection.

3 Under Self-Signed **Certificate**, click Create SSC and enter the certificate information.

Field	Description
Common Name	Domain name
Organizational Unit	Business unit
Organization	Company name
City or Locality	City or locality of the organization
State or Province	State or province of the organization
Country	Country code where the organization is located
Email	[Optional] Email address of the requester
Expiration	Select an expiration period

4 Click Create.

5 Click OK to confirm you want to install this certificate or replace the active certificate.

The certificate information appears in the Installed **Certificate** table.

Creating a Certificate Signing Request

1 From the Administration Options list, click **Configure** HTTPS.

2 Under Signed **Certificate**, click Create CSR and enter the following information:

Field	Description
Common Name	Domain name
Organizational Unit	Business unit
Organization	Company name
City or Locality	City or locality of the organization
State or Province	State or province of the organization
Country	Country code where the organization is located
Email	[Optional] Email address of the requester

3 Click Create.

4 Save the file using the browser options.

The file is saved to your download area or to a location that you specify, depending on the browser settings.

Installing Signed Certificates

After obtaining an authenticated certificate from a trusted CA, you can install the certificate to facilitate secure communications to the Teradata Database.

1 From the Administration Options list, click **Configure** HTTPS.

2 Under Signed **Certificate**, click Install **Certificate**.

3 Click Browse and locate the certificate you want to install.

4 Click Install.

The certificate information appears in the Installed **Certificate** table.

Installing a PKCS #12 File

You can install a PKCS #12 formatted private key and public key certificate chain. The benefit of PKCS #12 is that you can install a certificate without having to first create a CSR. Depending on the particular PKCS #12 file you are installing, a password may or may not be required.

1 From the Administration Options list, click **Configure** HTTPS.

2 Under PKCS #12 File, click Install **Certificate**.

3. Click Browse and locate the certificate you want to install.
4. If required, enter a password.
5. Click Install.
6. Click OK to confirm you want to install the certificate or replace the active certificate.

The certificate information appears in the Installed **Certificate** table.

Deleting Installed Certificates

1. From the Administration Options list, click **Configure** HTTPS.
2. Click ✕ in the Delete column for the certificate you want to delete.

   A confirmation message appears.

Changing the Password

1. From the Administration Options list, click Change Password.
2. Type your old password.
3. Type a new password.
4. Confirm your new password by retyping it.
5. Click Apply.

CHAPTER 5

Setting Up Service Configuration

Service Configuration

The Service **Configuration** view contains configuration options for your REST API services. For information on how to log in, see Logging In to the Service Configuration View.

Sessions

Displays currently running session details.

Queries

Displays currently running query details.

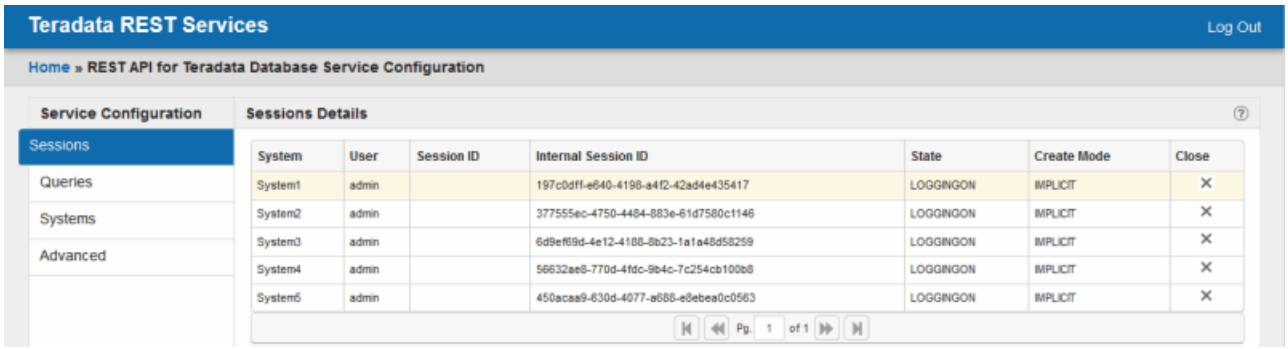
Systems

Allows you to create, modify, or delete systems.

Advanced

Allows you to configure advanced settings and to Export and Import configuration settings.

Following is an example of the Service **Configuration** view.



Logging In to the Service Configuration View

1 Log in to the Service **Configuration** view from any of the following views:

View	Description
Home	a. Click Overview in the row of the service you want to configure.

View	Description
	b. Click Service **Configuration** in the upper right corner.
Administration	a. Click ⚙ located in the **Configure** column in the row of the service you want to configure.

2 Enter your password, and click Log In.

Adding Teradata Systems

1 From the Service **Configuration** view, click Systems, located in the Service **Configuration** list.

2 Click + next to Systems.

3 Enter a name of the system.

4 Enter a name of the host.

5 [Optional] Enter a port number to change the default setting of 1025.
Valid port numbers range from 1 to 65535.

6 [Optional] Click Test to test the connectivity of the system and do the following:

a Enter a username and password.

b Select an authentication mechanism from the list.

c Select a character set from the list.

d Click OK.

If the operation is successful, ✔ appears. If the operation fails, ⚠ appears. Verify the settings are correct and try again.

7 [Optional] Under Limits, change any of the following metrics:

Metric	Description
Max idle time	Amount of time, in minutes, that a session can remain idle before being automatically closed. This is the value used for implicit sessions and the default for explicit sessions. Default value: 10 minutes Valid range: 1-60 minutes
Max implicit sessions	Maximum number of implicit sessions that a single user can have open at a time. Default value: 5 Valid range: 0-100
Max explicit sessions	Maximum number of explicit sessions that a single user can have open at a time.

Metric	Description
	Default value: 5 Valid range: 0-100
Max queued requests	Maximum number of queued requests allowed per user. Default value: 100 Valid range: 0-100000
User restrictions	Use to include or exclude access to the system for specified users. This setting is disabled until the Host field is validated. a. Click . b. In the User Restrictions Authentication dialog box, provide your authentication credentials, and click OK. The User Restrictions dialog box appears. c. If you want to restrict selected users, click Block selected users. d. If you want to allow only selected users, click Allow selected users. e. [Optional] Enter a string to filter the list of users. f. Click the name of a user to select it. To select multiple users, press Ctrl while selecting additional names. g. Click to move the names to the Selected users box. h. To remove names from the Selected users box, select the names and click . i. Click OK.

8 [Optional] Under Session Settings, select any of the following settings:

Setting	Description
Encrypt data	Use encryption when transferring to and from the Teradata Database.
Use X views	Use X views when presenting metadata. X views are more secure, but system performance may be impacted.
Transaction mode	Transaction mode for the connection (DEFAULT, ANSI, or TERA).
Default database	Default database when logged in.
Character set	Default character set for implicit sessions.
Query bands	Key/value pairs to represent the query bands that should be associated with each session.

9 Click Add.

The new system appears in the Systems list.

Configuring Concurrent Queries

You can set the total number of queries that can execute at the same time.

1. From the Service **Configuration** view, click Advanced, located in the Service **Configuration** list.
2. Under Advanced Details, enter a number for Total concurrent queries.

   The number must be between 0 and 500. The default value is 100.
3. Click Apply.

Configuring Asynchronous Queries

When you set queries with an asyncResponse flag, the result sets are saved onto your server until you delete the query or until the value of one or more of the metrics listed below are reached. If your server is being heavily used, you can set limits for these metrics. Setting limits allows you to find a balance so that you do not delete the result sets too soon or over-consume available disk space on the server.

1. From the Service **Configuration** view, click Advanced, located in the Service **Configuration** list.
2. Under Advanced Details, enter a number for any of the following metrics:

Metric	Description
Maximum number of rows to spool per query	Maximum number of rows that are returned by an asynchronous query. Default value: 1,000,000
Maximum number of spooled result sets	Number of result sets to keep. It is recommended you increase this number if you have many short queries or a heavily used TDREST server. After this setting is reached, the oldest spool files are automatically deleted to make room for new result sets until approximately 10% of the maximum result set count is deleted. Default value: 1,000
Maximum retention time for spooled result sets	Number of hours to store result sets for asynchronous queries before they are deleted. The minimum setting is 1 hour. Every hour, all spool files that exceed the maximum retention time are deleted. Default value: 24 hours
Maximum space available for spooled result sets	Total allocated space, in gigabytes, for saving result sets of asynchronous queries before they are deleted. After this setting is reached, the oldest spool files are automatically deleted to make room for new result sets until approximately 10% free space is reached.

Metric	Description
	It is recommended you increase this number for a heavily used TDREST server. Default value: 20 GB

3 [Optional] Enter a new default location of the spool directory that stores the result sets for asynchronous queries.

Current default location is `/var/opt/teradata/rest/daemon/spool`.

4 Click Apply.

Deleting Teradata Systems

1 From the Service **Configuration** view, click Systems, located in the Service **Configuration** list.

2 From the Systems column, click the name of the system you want to delete.

3 Under System Details, click Delete, located at the bottom.

A confirmation message appears.

Closing Sessions

1 From the Service **Configuration** view, click Sessions, located in the Service **Configuration** list.

2 Under Session Details, click ✕ in the Close column for the session you want to close.

Aborting Queries

1 From the Service **Configuration** view, click Queries, located in the Service **Configuration** list.

2 Click ✕ in the Abort column for the query you want to abort.

A confirmation message appears.

Exporting Current System Configuration

1 From the Service **Configuration** view, click Advanced, located in the Service **Configuration** list.

2 Click Export.

3 Save the file using the browser options.

The file is saved to your download area or to a location that you specify, depending on the browser settings.

Importing a Previously Defined System Configuration

If there is a previously defined configuration available, you can import it.

1. From the Service **Configuration** view, click Advanced, located in the Service **Configuration** list.
2. Click Import.

   The imported configuration replaces the current configuration and all currently configured systems are removed.

   A confirmation message appears.
3. Click OK.
4. Click Browse to locate the file you want to import.
5. Click Import.

APPENDIX A

Troubleshooting

Using Log Files to Troubleshoot

Log files for REST API for Teradata Database are located in `/var/opt/teradata/rest/daemon/logs`. Log files rotate out every 100 MB. Up to 10 previous log files are kept, named `*.log1` to `*.log10`.

Name	Description
`tdrest.log`	Contains service log messages reflecting the state of the service
`tdrest.audit.log`	Contains an audit log of the queries and sessions created by users of the service
`catalina.out`	Contains stdout and stderr from the service
`init.out`	Contains stdout and stderr from the service's init script

APPENDIX B

Teradata REST Metrics

Metrics for REST API for Teradata Database Configuration

Sessions Metrics

Metric	Description	
System	Alias of the Teradata Database system in the REST API for Teradata Database configuration.	
User	Name of the Teradata Database user who submitted the query.	
Session ID	Teradata Database session ID or blank if not logged in.	
Internal Session ID	Internal REST API for Teradata Database service ID for the database session.	
State	Current database session state. States can be any of the following:	
	LOGINON	No query. In the process of logging in.
	READY	No query, but logged in and ready to accept work.
	QUEUED	Query on the task queue, waiting to run.
	ACTIVE	Query executing or a query response is spooling.
	ABORTING	Query being aborted.
	LOGGINGOFF	No query. In the process of logging out.
Create Mode	*Implicit* or *Explicit*. Implicit database sessions are those created and used by the REST API for Teradata Database service for requests not specifying an explicit session. An explicit database session is one created by client applications for sole use by that application.	
Close	Close the database session.	

Queries Metrics

Metric	Description
System	Alias of the Teradata Database system in the REST API for Teradata Database configuration.
User	Name of the Teradata Database user who submitted the query.
Client ID	Teradata Database ID specified by the client when the query was submitted.
Int Session ID	REST API for Teradata Database service internal session ID (Implicit or ID #).
State	Query state, which can be PENDING, SUBMITTED, or RESPONDING.
Queue Order	Order of the request in the queue: 0 equals the active request for the database session.
In Queue	Number of milliseconds the query was queued by the REST API for Teradata Database service before being submitted to the database.
Running	Number of milliseconds since the REST API for Teradata Database service submitted the request. Blank if not running yet.
Abort	Abort a query.