Database Setup for IM and Presence Service on Cisco Unified Communications Manager, Release 10.5(1)

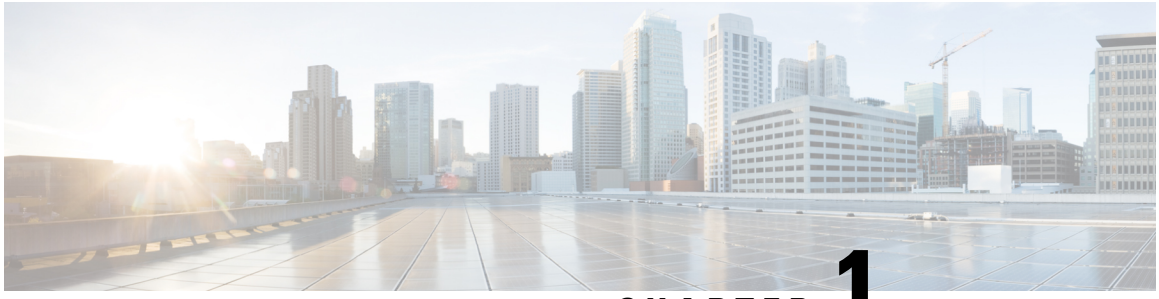
First Published: 2014-01-29

Last Modified: 2014-09-18

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-31663-01

CONTENTS



CHAPTER 1

External Database Requirements

This guide provides information about how to configure an external database for Cisco Unified Communications Manager IM and Presence Service features. The following features require an external database:

- Persistent Group Chat
- Message Archiver (IM Compliance)

How to use this Guide

Refer to the following chapters for instructions on how to configure your external database.

Procedure

	Command or Action	Purpose
Step 1	External Database Requirements, on page 1	Review support information and other requirements for your external database.
Step 2	Install the external database: • Install PostgreSQL, on page 7 • Install Oracle, on page 11	Refer to one of the chapters on the left for installation information.
Step 3	Configure IM and Presence Service for External Database, on page 15	Configure the IM and Presence Service for the external database connection.

What to do next

After setting up the external database, refer to the additional material in this guide for information on administering your external database.

External Database Setup Requirements

General Requirements

Cisco suggests having a certified PostgreSQL or Oracle administrator maintain and retrieve information from the external database.

Hardware Requirements

A remote server on which you install the external database.

Software Requirements

The following table contains general external database support information for the IM and Presence Service. For detailed information specific to IM and Presence features, refer to the subsequent "Feature Requirements" section.

Table 1: Database Support for the IM and Presence Service

Database	Supported Versions
PostgreSQL	Versions 8.3.x through 9.4.x are supported, and in IM and Presence Service Release, 11.0(1) versions: 9.1.9, 9.2.6, 9.3.6, 9.4.1 have been tested. **Note** You can also use Version 8.1.x of the PostgreSQL database, but the configuration of these versions may be different to the PostgreSQL database configuration described in this section. See the PostgreSQL documentation for details on how to configure these PostgreSQL database versions. If you use Version 8.1.x of the PostgreSQL database, the database configuration on IM and Presence Service is the same as described in this section.
Oracle	Versions 9g, 10g, 11g, and 12c are supported, and in IM and Presence Service Release, 11.0(1) versions: 11.2.0.1.0 and 12.1.0.1.0 have been tested.

- You can deploy your database on virtualized or non-virtualized platforms.
- You can deploy your database on Windows or Linux operating systems, where supported. See your database documentation for details on the supported operating systems and platform requirements.
- IPv4 and IPv6 are supported by IM and Presence connections to external databases.

Feature Requirements

External database requirements differ depending on which features you want to deploy on the IM and Presence Service. Refer to the following table for support information for specific IM and Presence features.

Table 2: External Database Requirements for Specific IM and Presence Features

Feature	Requirements
Persistent Group Chat feature	A minimum of one unique logical external database instance (tablespace) is required for the entire IM and Presence Service intercluster. A unique logical external database instance for each IM and Presence Service node or redundancy group in an IM and Presence Service cluster will provide optimum performance and scalability, but is not mandatory. Supports: • Oracle • Postgres SQL
Message Archiver (compliance) feature	We highly recommend that you configure at least one external database for each IM and Presence Service cluster; however you may require more than one external database for a cluster depending on your database server capacity. Supports: • Oracle • Postgres SQL

Note If you deploy the persistent group chat, message archiver (compliance) features on an IM and Presence Service node, the same unique logical external database instance (tablespace) can be shared across the features as each feature uses separate data tables. This is dependent on the capacity of the database instance.

Additional Documentation

This procedure only describes how to configure the external database on the IM and Presence Service. It does not describe how to fully configure the features that require an external database. See the documentation specific to the feature you are deploying for the complete configuration:

- For information on configuring the message archiver (compliance) feature on the IM and Presence Service, see *Instant Messaging Compliance for IM and Presence Service on Cisco Unified Communications Manager*.
- For information on configuring the persistent group chat feature on the IM and Presence Service, see *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager*.

External Database Setup Prerequisites

Before you install and configure the external database on the IM and Presence Service, perform the following tasks:

- Install the IM and Presence Service nodes as described in *Installing Cisco Unified Communications Manager*.
- Configure the IM and Presence Service nodes as described in *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager*.

Caution If the IM and Presence Service connects to an external database server using IPv6, ensure that the enterprise parameter is configured for IPv6 and that Eth0 is set for IPv6 on each node in the deployment; otherwise, the connection to the external database server fails. The message archiver and Cisco XCP Text Conference Manager will be unable to connect to the external database and will fail. For information about configuring IPv6 on the IM and Presence Service, see *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager*.

Hardware and Performance Recommendations

When you configure an external database with the IM and Presence Service, you need to consider the following recommendations:

- We recommend that you use similar hardware for both the external database and the IM and Presence Service nodes.
- We recommend that you maintain the external database according to the best practice guidelines described in the product documentation. If you do not properly maintain the external database, and you allow the external database to fill up, this causes performance problems in the IM and Presence Service cluster.

Note To minimize delay and any negative performance impacts, locate the external database server as close as possible to the IM and Presence Service Database Publisher node.

Note If user login performance is very slow or if logins are hanging, review the database performance information for that server.

Related Topics

PostgreSQL documentation
Oracle documentation

About Security Recommendations

External Database Connection Security

The IM and Presence Service provides a secure TLS/SSL connection to the external database but only when Oracle is chosen as the database type. We recommend that you consider this security limitation when you

plan your IM and Presence Service deployment, and consider the security recommendations we provide in this topic.

Maximum Connection Limit Setup

Note Use this section as a guideline if you wish to limit the number of connections to the external database. This section is optional configuration.

For additional security, you can limit the maximum number of permitted connections to the external database. Use the guideline we provide here to calculate the number of database connections that are appropriate for your deployment.

You must modify this guideline to suit your IM and Presence Service deployment and configuration. The guideline we provide here assumes that:

- You are running both the message archiver and persistent group chat features on the IM and Presence Service.

  and

- You configure the default number of connections to the database for the persistent group chat feature on the **Cisco Unified CM IM and Presence Administration** interface.

PostgreSQL

To limit the number of PostgreSQL database connections, configure the max_connections value in the postgresql.conf file located in the `install_dir/data` directory. We recommend that you set the value of the max_connections parameter equal to this guideline:

max_connections = (N ×10) + Additional Connections

- N is the number of nodes in your IM and Presence Service cluster.
- 10 is the default number of connections to the database on the IM and Presence Service, that is, five connections for the message archiver (compliance) feature and five connections for the persistent group chat feature. You can configure the number of database connections for the persistent group chat feature on the **Cisco Unified CM IM and Presence Administration** interface.
- Additional Connections represents any independent administration or database administrator (DBA) connections to the database server.

  For example, if you have an IM and Presence Service cluster containing six nodes, and you require an additional three DBA connections, using the guideline above, you set the max_connections value to 63.

Oracle

To limit the number of Oracle database connections, configure the QUEUESIZE parameter in the listener.ora file located in the `install_dir/data` directory. We recommend that you set the value of the QUEUESIZE parameter equal to this guideline:

max_connections = (N ×10) + Additional Connections

- N is the number of nodes in your IM and Presence Service cluster.

- 10 is the default number of connections to the database on the IM and Presence Service, that is, five connections for the message archiver feature and five connections for the persistent group chat feature. You can configure the number of database connections for the persistent group chat feature on the **Cisco Unified CM IM and Presence Administration** interface.
- Additional Connections represents any independent administration or database administrator (DBA) connections to the database server.

  For example, if you have an IM and Presence Service cluster containing six nodes, and you require an additional three DBA connections, using the guideline above, you set the QUEUESIZE value to 63.

Related Topics

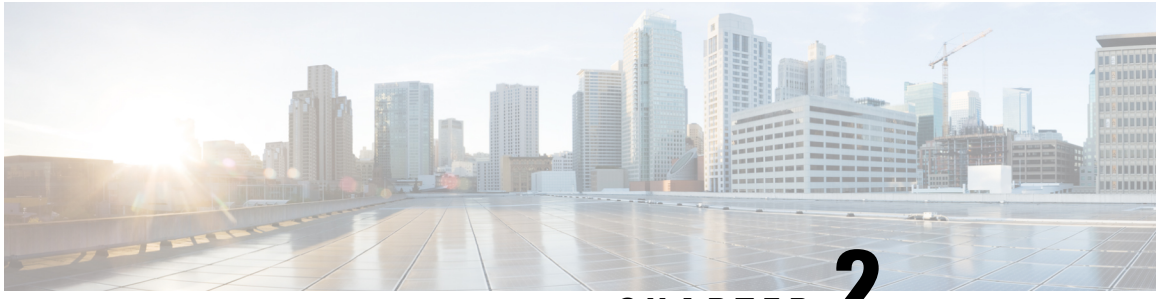
PostgreSQL documentation
Oracle documentation

Default Listener Port Setup

Note This section is an optional configuration.

For additional security, you may choose to change the default listening port on the external database:

- For PostgreSQL, see Set Up PostgreSQL Listening Port, on page 9 for details on how to edit the default listener port.
- For Oracle, you can edit the default listener port by editing the `listener.ora` config file



CHAPTER 2

Install PostgreSQL

This chapter provides information about installing and setting up PostgreSQL.

Install PostgreSQL Database

Before you begin

- Cisco recommends that a PostgreSQL DBA install and maintain the PostgreSQL server.
- Read the security recommendations for the PostgreSQL database in section About Security Recommendations, on page 4.
- For information on supported versions, see External Database Setup Requirements, on page 2.

Procedure

Step 1 Enter these commands to sign in to the database server as a Postgres user:

```
>su - postgres
>psql
```

Step 2 Create a new database user. The example below creates a new database user called *tcuser*:

```
#CREATE ROLE tcuser LOGIN CREATEDB;
```

Note If you deploy PostgresSQL version 8.4.x, you must configure the database user as a superuser at this point in the procedure, for example:

```
#ALTER ROLE tcuser WITH SUPERUSER;
```

Step 3 Create the database. If your database contains ASCII characters only, create the database with SQL_ASCII encoding. If your database contains non-ASCII characters, create the database with UTF8 encoding.

The example below creates an SQL_ASCII database called *tcmadb*.

```
#CREATE DATABASE tcmadb WITH OWNER tcuser ENCODING 'SQL_ASCII';
```

Step 4 Configure user access to the database. Edit the *install_dir*/data/pg_hba.conf file to allow the *postgres* user and the new *tcuser* user to access the database. For example:

# TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
host	tcmadb	tcuser	10.89.99.0/24	password
host	dbinst	mauser	10.89.99.0/24	password

Step 5 Enter these commands to define passwords for the *postgres* and *tcuser* users:

```
#ALTER ROLE postgres WITH PASSWORD 'mypassword';

#ALTER ROLE tcuser WITH PASSWORD 'mypassword';
```

Note You are required to enter a password for the database user when you configure an external database entry on the IM and Presence Service.

Step 6 If you are running the PostgreSQL version 8.3.7 or a later 8.3.x release, change the permission of the *tcuser* to superuser to allow this user access to the database. Enter this command:

```
#ALTER ROLE tcuser WITH SUPERUSER;
```

Step 7 Configure the connections to the database from remote hosts. Edit the listen_addresses parameter in the *install_dir*/data/postgresql.conf file. For example:

```
listen_addresses = '*'
```

Step 8 If you are running PostgreSQL version 9.1.1, or higher, you must set the following values in the postgresql.conf file:

```
escape_string_warning = off

standard_conforming_strings = off
```

Step 9 Stop and restart the PostgreSQL service, for example:

```
/etc/rc.d/init.d/postgresql-8.3 stop

/etc/rc.d/init.d/postgresql-8.3 start
```

Note The commands to stop and start the PostgreSQL service may vary between PostgreSQL releases.

Step 10 Enter these commands to sign in to the new database as the *postgres* user and enable PL/pgSQL:

```
>psql tcmadb -U postgres
```

Note The following example, up to the semicolon, should be entered as one line.

```
#CREATE FUNCTION plpgsql_call_handler () RETURNS LANGUAGE_HANDLER AS '$libdir/plpgsql'
LANGUAGE C;

#CREATE TRUSTED PROCEDURAL LANGUAGE plpgsql HANDLER plpgsql_call_handler;
```

Troubleshooting Tips

Do not turn on the following configuration items in the *install_dir*/data/postgresql.conf file (by default these items are commented out):

```
client_min_messages = log
log_duration = on
```

Related Topics

About Security Recommendations, on page 4

Set Up PostgreSQL Listening Port

Note This section is optional configuration.

By default, the Postgresql database listens on port 5432. If you want to change this port, you must edit the PGPORT environment variable in /etc/rc.d/init.d/postgresql with the new port number.

Note The PGPORT environment variable overrides the 'Port' parameter value in the `/var/lib/pgsql/data/postgresql.conf` file, so you must edit the PGPORT environment variable if you want the Postgresql database to listen on a new port number.

Procedure

Step 1 Edit the PGPORT environment variable in /etc/rc.d/init.d/postgresql with the new port, for example:

```
IE: PGPORT=5555
```

Step 2 Enter these commands to stop and start the PostgreSQL service:

```
# /etc/rc.d/init.d/postgresql start
# /etc/rc.d/init.d/postgresql stop
```

Step 3 Confirm that the Postgresql database is listening on the new port using this command:

```
'lsof -i -n -P | grep postg'
postmaste 5754 postgres 4u IPv4 1692351 TCP *:5555 (LISTEN)
```

Tip For IPv6 servers, enter `postmaste 5754 postgres 4u IPv6 1692351 TCP *:5555 (LISTEN)`

Step 4 To connect to the database after you have changed the port, you must specify the new port number in the command using the -p argument. If you do not include the -p argument in the command, the Postgresql database attempts to use the default port of 5432, and the connection to the database fails.

For example:

```
psql tcmadb -p 5555 -U tcuser
```

User Access Restriction Recommendations

We strongly recommend that you restrict user access to the external database to only the particular user and database instance that the IM and Presence Serivce uses. You can restrict user access to the PostgreSQL database in the pg_hba.conf file located in the <install_dir>/data directory.

Caution Do not configure 'all' for the user and database entries because potentially this could allow any user access to any database.

When you configure user access to the external database, we also recommend that you configure password protection for the database access using the 'password' method.

Note You are required to enter a password for the database user when you configure a database entry on IM and Presence Service.

The following are examples of a secure user access configuration, and a less secure user access configuration, in the pg_hba.conf file.

Example of a secure configuration:

# TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
host	dbinst1	tcuser1	10.89.99.0/24	password
host	dbinst2	mauser1	10.89.99.0/24	password

Example of a less secure configuration:

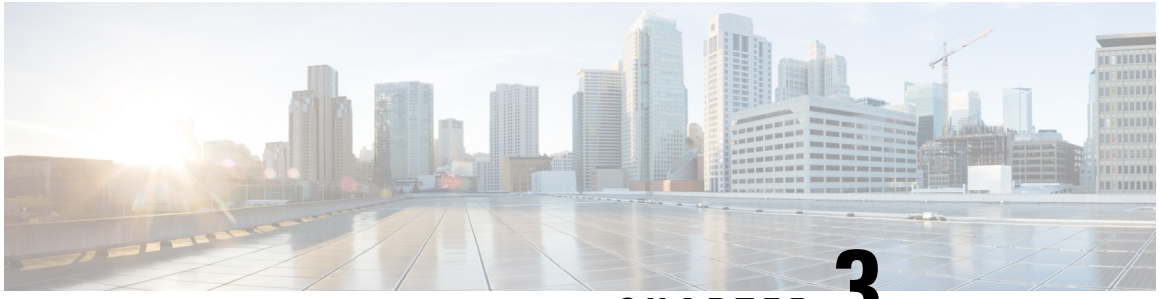
# TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
host	dbinst1	tcuser1	10.89.99.0/24	trust
host	dbinst2	all	10.89.99.0/24	password

Notes on the example of a less secure configuration:

- The first entry contains no password protection for the database.
- The second entry allows any user to access the database “dbinst2”.

Related Topics

Install PostgreSQL Database, on page 7
PostgreSQL documentation



CHAPTER 3

Install Oracle

This chapter provides information about installing and setting up an Oracle database.

Install Oracle Database

Before you begin

- Cisco recommends that an Oracle DBA install the Oracle server.
- You need to update the patch for the known Oracle defect: ORA-22275. If this is not done persistent chat rooms will not work properly.
- Read the security recommendations for the Oracle database in your Oracle documentation.
- For information on supported versions, see External Database Setup Requirements, on page 2.
- To install the Oracle database, refer to your Oracle documentation.

Note In compliance with XMPP specifications, the IM and Presence Service node uses UTF8 character encoding. This allows the node to operate using many languages simultaneously and to display special language characters correctly in the client interface. If you want to use Oracle with the node, you must configure it to support UTF8.

To create tablespace and a database user, connect to the Oracle database as sysdba:

```
sqlplus / as sysdba
```

Procedure

Step 1 Create tablespace.

Note The `DATAFILE` keyword of the `CREATE TABLESPACE` command tells Oracle where to put the tablespace's datafile.

a) Enter the following command:

```
CREATE TABLESPACE tablespace_name DATAFILE
'absolute_path_to_oracle_installation\oradata\database_name\datafile.dbf' SIZE 100M
AUTOEXTEND ON NEXT 1M MAXSIZE UNLIMITED LOGGING EXTENT MANAGEMENT LOCAL SEGMENT SPACE
MANAGEMENT AUTO;
```

- Replace *tablespace_name* with the tablespace name.
- Replace *absolute_path_to_oracle_installation* with the absolute path to where Oracle is installed. The entire path, including *datafile*.dbf, is enclosed in single quotation marks.
- Replace *database_name* with the name of your database folder.
- The *datafile*.dbf must be created in a folder under **\oradata**, in this case the *database_name* folder.
- Replace *datafile*.dbf with the datafile name you want to create.

Step 2 Create a database user.

```
CREATE USER user_name IDENTIFIED BY "new_user's_password" DEFAULT TABLESPACE tablespace_name
TEMPORARY TABLESPACE "TEMP" QUOTA UNLIMITED ON tablespace_name ACCOUNT UNLOCK;
```

- Replace *user_name* with the new user's user name.

  Note The command **CREATE USER user_name** without double quotes will default to upper case and with quotes it will maintain the case

- Replace "*new_user's_password*" with the new user's password.

  Important Enclosing the *new_user's_password* within double quotation marks makes the variable case-sensitive. By default SQL identifiers are not case-sensitive.

- Replace *tablespace_name* with the tablespace name.

Step 3 Grant permissions to the database user.

The following example grants all permissions to a database user:

```
GRANT DBA TO user_name;
```

The following examples grant limited permissions to the database user:

- **GRANT CREATE ANY VIEW TO** *user_name*;
- **GRANT "CONNECT" TO** *user_name*;
- **GRANT "RESOURCE" TO** *user_name*;

Related Topics

Oracle Documentation

Create New Database Instance

Procedure

Step 1 Enter the command **dbca**
The **Database Configuration Assistant** wizard opens.

Step 2 Click **Next**.
The **Operations** window appears.

Step 3 Click the **Create a Database** radio button and then click **Next**.
The **Database Templates** window appears.

Step 4 Click the **General Purpose or Transaction Processing** radio button and then click **Next**.
The **Database Identification** window appears.

Step 5 Enter a unique Global Database Name on this screen and also a unique Oracle System Identifier (SID) for the database and click **Next**.

Note Take note of the SID because it is needed in Step 15.

The **Management Options** window appears.

Step 6 Under the Enterprise Manager tab the required settings are enabled by default but you can configure optional backups and alert notifications. Click **Next**.
The **Database Credentials** window appears.

Step 7 The window has two options to set up password authentication for database users, choose one and click **Next**.
The **Database File Locations** window appears.

Step 8 The Storage Type drop-down list should be the same as your Oracle Installation. Click the **Use Oracle-Managed Files** radio button and click **Next**.

Note This creates the new database instance in the same folder as your other database instances.

The **Recovery Configuration** window appears.

Step 9 Leave the default values and click **Next**.
The **Database Content** window appears.

Step 10 [Optional] Check the check box if you want to enable Sample Schemas and click **Next**.
The **Initialization Parameters** window appears.

Step 11 Under the Memory tab the default value is for a database instance with 4GB of memory. This can be set higher or lower as needed.

Note The amount of memory used should not be configured too high as this starves other database instances of memory.

Step 12 Under the Character Sets tab click the **Use Unicode** radio button and click **Next**.
The **Database Storage** window appears.

Step 13 Leave the default settings as they are and click **Next**.
The **Create Options** window appears.

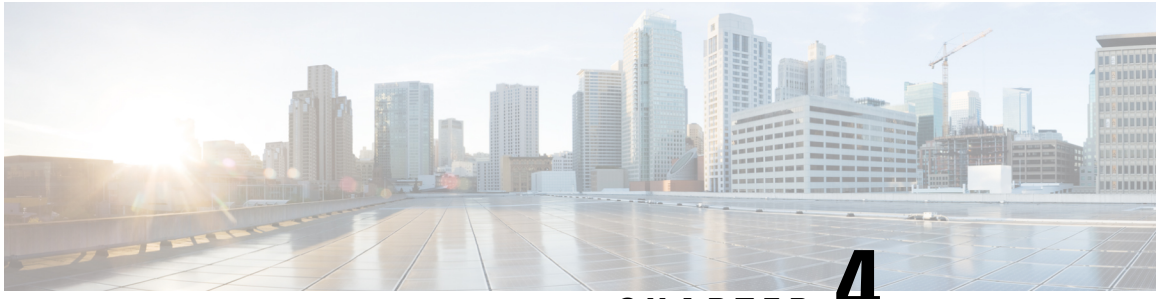
Step 14 Check the Create Database check box and click **Finish**.

Step 15 Once a new database instance is created, you must temporarily change the ORACLE_SID environment variable (from Step 5) on your Unix system by running the command:

```
export ORACLE_SID=new_oracle_db_instance_sid.
```

This will change the SID so when you login using sqlplus, it will use the new instance and not the old one; you can then repeat the steps in Install Oracle Database, on page 11.

Once these steps are completed you can change the ORACLE_SID environment variable by sourcing the bash profile(assuming the old SID is in the bash profile) or by running the export command (Step 15) but changing the SID back to its original value.



CHAPTER 4

Configure IM and Presence Service for External Database

This chapter provides information about configuring the IM and Presence Service for the external database connection.

About External Database Assignment

External Database and Node Assignment

When you configure an external database entry on the IM and Presence Service, you assign the external database to a node, or nodes, in your cluster as follows:

- Message Archiver (compliance) — You require at least one external database per cluster. Depending on your deployment requirements, you can also configure a unique external database per node.
- Persistent Group Chat — You require a unique external database per node. Configure and assign a unique external database for each node in your cluster.
- If you deploy the persistent group chat and message archiver features on an IM and Presence Service node, you can assign the same external database to both of the features.

For more information see:

- Message Archiver — *Instant Messaging Compliance for IM and Presence Service on Cisco Unified Communications Manager*.
- Persistent Group Chat — *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager.*

Related Topics

Set Up External Database Entry on IM and Presence Service, on page 16

External Database Connection, on page 16

External Database Connection

IM and Presence Service does not establish a connection to the external database when you configure an external database entry. The external database has not created the database schema at this point. It is only when you assign an external database entry to a node that IM and Presence Service establishes an ODBC (Open Database Connectivity) connection with the external database. Once IM and Presence Service establishes a connection, the external database creates the database tables for the IM and Presence Service features.

Once you assign an external database entry to a node, you can validate the connection using the System Troubleshooter in the **Cisco Unified CM IM and Presence Service Administration** user interface.

Related Topics

Set Up External Database Entry on IM and Presence Service, on page 16
Verify External Database Connection Status on IM and Presence Service, on page 18

Set Up External Database Entry on IM and Presence Service

Perform this configuration on the IM and Presence Service database publisher node of your cluster.

Caution If your IM and Presence Service node connects to an external database server using IPv6, ensure that the enterprise parameter is configured for IPv6 and that Eth0 is set for IPv6 on each node in the deployment; otherwise, the connection to the external database server fails. The Message Archiver and Cisco XCP Text Conference Manager are unable to connect to the external database and fail. For information about configuring IPv6 on IM and Presence Service, see *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager*.

Before you begin

- Install and configure the external database.
- Obtain the hostname or IP address of the external database.
- If using Oracle, retrieve the tablespace value. To determine the tablespace available for your Oracle database, execute the following query as sysdba:

```
SELECT DEFAULT_TABLESPACE FROM DBA_USERS WHERE USERNAME = 'USER_NAME';
```

Note The user name must be capitalized and in single quotes (a string literal) for this command to succeed, even if you defined the user with lowercase characters.

Procedure

Step 1 Log in to the **Cisco Unified CM IM and Presence Administration** user interface. Choose **Messaging** > **External Server Setup** > **External Databases**.

Step 2 Click **Add New**.

Step 3 Enter the name of the database that you defined at external database installation, for example, **`tcmadb`**.

Step 4 Choose the database type from the drop-down list, Postgres or Oracle.

Step 5 If you chose Oracle as the database type, enter the tablespace value.

Step 6 Enter the username for the database user (owner) that you defined at external database installation, for example, **`tcuser`**.

Step 7 Enter and confirm the password for the database user, for example, **`mypassword`**.

Step 8 Enter the hostname or IP address for the external database.

Step 9 Enter a port number for the external database.

The default port numbers for Postgres (5432), Oracle (1521), and Oracle with SSL enabled (2484) are prepopulated in the **Port Number** field. You can choose to enter a different port number if required.

Step 10 If you chose Oracle as the Database Type the **Enable SSL** check box becomes active. Check the check box to enable SSL.
The **Certificate Name** drop-down list becomes active. Choose a certificate from the drop-down list.

Note
- When the Enable SSL check box or the Certificate drop-down field is modified, a notification to restart the corresponding service assigned to the external database is sent. A message concerning either Cisco XCP Message Archiver or Cisco XCP Text Conference Manager will be generated.
- The certificate you need to enable SSL must be uploaded to the cup-xmpp-trust store. You must upload this certificate before you enable SSL.
- Once the certificate is uploaded to the cup-xmpp-trust store, you must wait 15 minutes for the certificate to propagate to all the nodes of the IM and Presence Service cluster. If you do not wait, the SSL connection on nodes where the certificate has not propagated fails.
- If the certificate is missing or has been deleted from the cup-xmpp-trust store, an alarm XCPExternalDatabaseCertificateNotFound is raised in the Cisco Unified Communications Manager Real Time Monitoring Tool (RTMT).

Step 11 Click **Save**.

Related Topics

Verify External Database Connection, on page 17

Verify External Database Connection

If you make a configuration change in the `install_dir/data/pg_hba.conf` file or the `install_dir/data/postgresql.conf` file after you assign the external database, perform these steps:

Procedure

Step 1 Unassign and reassign the external database to the IM and Presence Service node.

Step 2 Restart the Cisco XCP Router service. Log in to the **Cisco Unified IM and Presence Serviceability** user interface. Choose **Tools** > **Control Center - Network Services** to restart this service.

Related Topics

Install PostgreSQL, on page 7
Install Oracle, on page 11

Verify External Database Connection Status on IM and Presence Service

IM and Presence Service provides the following status information on an external database:

- Database reachability — Verifies that the IM and Presence Service can ping an external database.
- Database connectivity — Verifies that the IM and Presence Service has successfully established an Open Database Connectivity (ODBC) connection with the external database.
- Database schema verification — Verifies that the external database schema is valid.

Caution If your IM and Presence Service node connects to an external database server using IPv6, ensure that the enterprise parameter is configured for IPv6 and that Eth0 is set for IPv6 on each node in the deployment; otherwise, the connection to the external database server fails. The message archiver and Cisco XCP Text Conference Manager is unable to connect to the external database and fails. For information about configuring IPv6 on IM and Presence Service, see *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager*.

Procedure

Step 1 Log in to the **Cisco Unified CM IM and Presence Administration** user interface. Choose **Messaging** > **External Server Setup** > **External Databases**.

Step 2 Click **Find.**

Step 3 Click the link for the external database entry that you want to view.

Step 4 Verify that there are check marks beside each of the result entries for the external database in the External Database Status section.

Tip

- The IM and Presence Service generates an alarm if it loses ODBC to an external database.
- You can also verify the status of the Postgres database connection using the **psql** command. You must sign in to the Linux shell from a remote support account to run this command; it is not accessible through the administrator CLI.

  Run the following command after you install the Postgres database, but before you assign the database to an IM and Presence Service node.

  Important For `psql` to run, you must first set an environment variable by entering:

  ```
  $export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/xcp/lib
  ```

  Enter:

  ```
  $sudo -u xcpuser /usr/local/xcp/bin/psql -U db_user -h db_server db_name
  ```

  For example:

  ```
  $sudo –u xcpuser /usr/local/xcp/bin/psql –U postgres -h postgres_server tcmadb
  ```

- You can verify the status of the Oracle database connection by executing the following commands from the root directory:

  ```
  export ORACLE_HOME=/usr/lib/oracle/client_1/
  export PATH="$ORACLE_HOME/bin:$PATH"
  export LD_LIBRARY_PATH="$ORACLE_HOME/lib:$LD_LIBRARY_PATH"
  sqlplus username/password@dsn
  ```

  Note
  - The *username* and *password* are those of the database administrator.
  - The data source name *(dsn)* value can be obtained from the **$ORACLE_HOME/network/admin/tnsnames.ora** file.

- If you configure the message archiver (compliance) feature, and the Cisco XCP Message Archiver service fails to start, or you configure the persistent group chat feature and the Cisco Text Conference Manager service fails to start, check the External Database Troubleshooter section of the **System Configuration Troubleshooter** window.
  - If the status of the external database connection is not **OK**, verify that you provided the correct connection details and that there are no network issues between the IM and Presence Service node and the external database host.
  - If the status of the external database connection is **OK**, but the schema verification status is not, unassign the external database and the reassign it to the node.

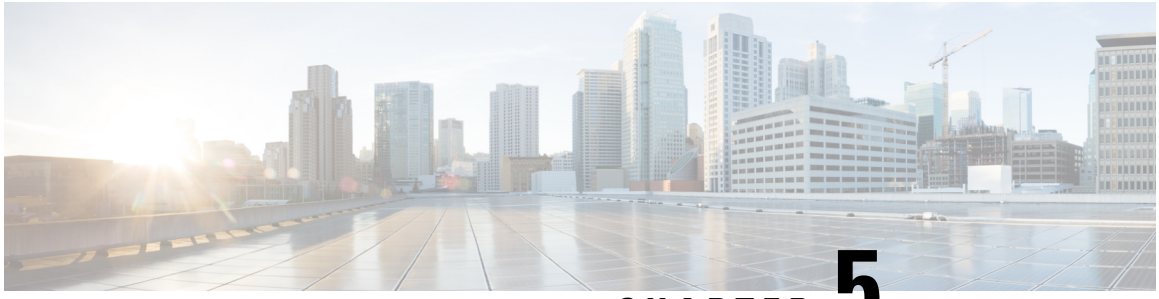
- Once the certificate is uploaded to the cup-xmpp-trust store, you must wait 15 minutes for the certificate to propagate to all the nodes of the IM and Presence Service cluster. If you do not wait, the SSL connection on nodes where the certificate has not propagated fails.
- If the certificate is missing or has been deleted from the cup-xmpp-trust store, an alarm 'XCPExternalDatabaseCertificateNotFound' is raised in the Cisco Unified Communications Manager `Real Time Monitoring Tool` (`RTMT`).

To view the status and verification tests for all database connections, you can also access the System Troubleshooter:

Procedure

Step 1 Log in to the **Cisco Unified CM IM and Presence Administration** user interface, choose **Diagnostics** > **System Troubleshooter**.

Step 2 Verify that there are check marks beside the status of each of the external database connection entries in the External Database Troubleshooter section.



CHAPTER 5

Database Tables

This chapter provides information about the external database tables that are created in your schema to support the IM and Presence Service node.

Note By default, the IM and Presence Service generates 27 tables in the external database but at present it only uses the tables described in this module.

Note If you need to modify any data in the external database, ensure that you restart the Cisco XCP Text Conference Manager service after you have made those changes.

TC_ROOMS Table

The TC_ROOMS table contains information for group chat rooms.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
ROOM_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room.
CREATOR_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the user who created the room.
SUBJECT	VARCHAR (255)	VARCHAR2 (255)	varchar (255)	Yes	The current subject for the room.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
TYPE	VARCHAR (32)	VARCHAR2 (32)	varchar (32)	Yes	The constraint check_type. This value must be either “ad-hoc” or “persistent”.
CONFIG	TEXT	CLOB	text	Yes	The entire packet from the last time the room was configured. This information enables the room to be reconfigured when the room is recreated (for example, at start-up).
SPACKET	TEXT	CLOB	text	Yes	The entire packet from the last time the subject was set for the room. This information enables the room subject to be displayed when the room is recreated.
START_MSG_ID	BIGINT	NUMBER (19)	bigint	Yes	A sequence number that is used to populate the MSG_ID column in the TC_MSGARCHIVE table. Do not modify this value.
NEXT_MSG_ID	BIGINT	NUMBER (19)	bigint	Yes	A sequence number that is used to populate the MSG_ID column in the TC_MSGARCHIVE table. Do not modify this value.

TC_USERS Table

The TC_USERS table contains roles and affiliations, alternate names, and other data associated with group chat room users.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
ROOM_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room.
REAL_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of a user in the room. This value is the actual ID of the user, rather than an alternate name.
ROLE	VARCHAR (32)	VARCHAR2 (32)	varchar (32)	Yes	The role of the user in the room. This value is constrained to one of the following: “none”, “hidden”, “visitor”, “participant”, or “moderator”.
AFFILIATION	VARCHAR (32)	VARCHAR2 (32)	varchar (32)	Yes	The affiliation of the user in the room. This value is constrained to one of the following: “none”, “outcast”, “member”, “admin”, or “owner”.
NICK_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room, plus the alternate name for the user. The format is room@tc-server/nick.
REASON	VARCHAR (255)	VARCHAR2 (255)	varchar (255)	Yes	The reason entered when the user's affiliation was last changed.
INITIATOR_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room in which the configuration change occurred.

TC_MESSAGES Table

The TC_MESSAGES table contains messages that are sent in group chat rooms.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
MSG_ID	BIGINT	NUMBER (19)	bigint	Yes	The ID of the message. The MSG_ID is a unique identifier for each message per chat room; it is not globally unique.
ROOM_JID	VARCHAR (3071)	VARCHAR (3071)	varchar (3071)	Yes	The ID of the room to which the message was sent.
STAMP	TIMESTAMP	TIMESTAMP	datetime	Yes	The date and time the message was sent.
MSG	TEXT	CLOB	text	Yes	The entire message.

TC_TIMELOG Table

The TC_TIMELOG table contains the time that users enter and exit specific group chat rooms. This table may be used in conjunction with the other TC tables to recreate group chat conversations and to determine which users viewed the conversations.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
REAL_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the user who is entering or leaving the room.
NICK_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room, plus the alternate name for the user. The format is room@tc-server/nick.
DIRECTION	VARCHAR (1)	VARCHAR2 (1)	varchar (1)	Yes	Indicates whether the user entered (E) or left (L) the room. Constrained to the values “E” and “L”.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
STAMP	TIMESTAMP	TIMESTAMP	datetime	Yes	The date and time at which the user entered or left the room. UTC format from IMP server.

TC_MSGARCHIVE Table

The TC_MSGARCHIVE table stores messages and associated information for group chat rooms.

Note This table archives all messages if you turn on group chat on IM and Presence Service. Choose the option Archive all room messages on the **Cisco Unified CM IM and Presence Administration** user interface. Choose **Messaging** > **Conferencing and Persistent Chat**. See *Configuration and Administration of IM and Presence Service on Cisco Unified Communications Manager* for information on the group chat feature.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
MSG_ID	BIGINT	NUMBER (19)	bigint	Yes	A unique identifier for the message.
TO_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room that received the message.
FROM_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the user who sent the message.
NICK_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The ID of the room, plus the alternate name of the sender; for example: room@conference.example.com/nick
SENT_DATE	TIMESTAMP	TIMESTAMP	datetime	Yes	The date on which the message was sent.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL Datatype	Not Null	Description
MSG_TYPE	VARCHAR (1)	VARCHAR2 (1)	varchar (1)	Yes	The first character of the type attribute of the message. The possible values are “c” (chat), “n” (normal), “g” (groupchat), “h” (headline), and “e” (error).
BODY_LEN	INT	NUMBER (9)	int	Yes	The length in characters of the message body.
MESSAGE_LEN	INT	NUMBER (9)	int	Yes	The length in characters of the message, including the subject and body.
BODY_STRING	VARCHAR (4000)	VARCHAR2 (4000)	varchar (4000)	Yes	The message body.
MESSAGE_ STRING	VARCHAR (4000)	VARCHAR2 (4000)	varchar (4000)	Yes	The entire raw packet.
BODY_TEXT	TEXT	CLOB	text	Yes	If the message body exceeds 4000 characters, it is stored in this field rather than the BODY_STRING field.
MESSAGE_TEXT	TEXT	CLOB	text	Yes	If the entire raw packet exceeds 4000 characters, it is stored in this column rather than in the MESSAGE_STRING column.
SUBJECT	VARCHAR (255)	VARCHAR2 (255)	varchar (255)	Yes	The current subject of the room.

JM Table

The JM table stores conversations and associated information for the message archiver component. The message archiver component provides the native compliance functionality on the IM and Presence Service.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL datatype	Not Null	Description
TO_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The Jabber ID (JID) of the user who is sending the message being archived.
FROM_JID	VARCHAR (3071)	VARCHAR2 (3071)	varchar (3071)	Yes	The JID of the user who is receiving the message being archived.
SENT_DATE	TIMESTAMP	TIMESTAMP	datetime	Yes	The date the message was sent.
SUBJECT	VARCHAR (128)	VARCHAR2 (128)	varchar (128)	Yes	The subject line of the message that is being archived.
THREAD_ID	VARCHAR (128)	VARCHAR2 (128)	varchar (128)	Yes	The thread ID of the message that is being archived.
MSG_TYPE	VARCHAR (1)	VARCHAR2 (1)	varchar (1)	Yes	The first character of the message's type attribute. The possible values are: • “c” — chat • “n” — normal • “g” — groupchat • “h” — headline • “e” — error
DIRECTION	VARCHAR (1)	VARCHAR2 (1)	varchar (1)	Yes	Indicates whether the message is “O” — outgoing or “I” — incoming. If the message is sent between users on the same server, it is logged twice: once as outgoing and once as incoming.
BODY_LEN	INT	NUMBER (9)	int	Yes	The number of characters in the message body.

Column Name	Postgres Datatype	Oracle Datatype	Microsoft SQL datatype	Not Null	Description
MESSAGE_LEN	INT	NUMBER (9)	int	Yes	The number of characters in the message, including the subject and the body.
BODY_STRING	VARCHAR (4000)	VARCHAR2 (4000)	varchar (4000)	Yes	The message body.
MESSAGE_ STRING	VARCHAR (4000)	VARCHAR2 (4000)	varchar (4000)	Yes	The entire raw packet.
BODY_TEXT	TEXT	CLOB	text	Yes	If the message body exceeds 4000 characters, it is stored in this field rather than the BODY_STRING field.
MESSAGE_TEXT	TEXT	TEXT	text	Yes	If the entire raw packet exceeds 4000 characters, it is stored in this field rather than in the MESSAGE_STRING field.
HISTORY_FLAG	VARCHAR (1)	VARCHAR2 (1)	varchar (1)	Yes	Used when room history messages are sent to new participants (upon entering an existing room). This allows you to distinguish between messages received while actively participating in a room and those received as part of a history push. The latter message type is flagged with HISTORY_FLAG="H" in the database. Otherwise, this column is set to "N."

Sample SQL Queries for the JM Table

This section contains some sample SQL queries that you can run on the JM table to extract specific information. The following queries select all columns from the table but you can be more selective about which information you want to include in your SQL queries.

All Instant Messages Sent by a Specific User

The following SQL query returns all instant messages sent by a specific user:

```
SELECT to_jid, sent_date, subject, thread_id, msg_type, direction, body_len, message_len,
body_string, message_string, body_text, message_text, history_flag
FROM jm
WHERE from_jid like 'bob@cisco.com%';
```

All Instant Messages Received by a Specific User

The following SQL query returns all instant messages received by a specific user:

```
SELECT from_jid, sent_date, subject, thread_id, msg_type, direction, body_len,
message_len, body_string, message_string, body_text, message_text, history_flag
FROM jm
WHERE to_jid like 'bob@cisco.com%';
```

All Instant Messages That Contain a Specific Word

The following SQL query returns all instant messages that contain a specific word:

```
SELECT to_jid, from_jid, sent_date, subject, thread_id, msg_type, direction, body_len,
message_len, body_string, message_string, body_text, message_text, history_flag
FROM jm
WHERE LOWER(body_string) like LOWER('%hello%');
```

All Instant Messages Conversations and Chat Rooms From a Specific Date

The following SQL query returns all instant messages, conversations and chat rooms from a specific date:

```
SELECT to_jid, from_jid, sent_date, subject, thread_id, msg_type, direction, body_len,
message_len, body_string, message_string, body_text, message_text, history_flag
FROM jm
WHERE CAST(sent_date AS Character(32)) like '2011-01-31%';
```