**Running RGW in a container.**

**Version control**

|  |  |
| --- | --- |
| Created by | Marcos Lara |
| Version | 001 – draft - version |
| When | 21-Mar-2025. |
| Objective | Provide a detailed guide on configuring RGW to run in a container on VirtualBox VM |

**Notes:**

* This document is part of the MTS installation in the VirtualBox VM. If you have already created your Red Hat 8 VM, we can skip some parts of this document.
* This intention is intended for use as a test on your development machine, simplifying your integration tests with MTS.
* This is being created from memory; I will need to run some tests to ensure everything is functioning as intended.

Install a VirtualBox virtual machine with Linux Red Hat 8.

Create a user: mtsadmin

Create a group: mts

Add the user mtsadmin to the group mts

Disable the password for the “Wheel” group

Login in your VM as mtsadmin

Create a directory that will be used as ‘share’ between your redhat8 and the container:

$ mkdir ~/shared

$ cd ~/shared

Create the following scripts in the ‘**shared**’ directory. These will be used to initialise the container. (Note that the same approach will be applied to the MQ container, and the steps will be included in a separate document.)

**[mtsadmin@localhost shared]** $ ls -ltr

total 16

-rw-rw-r-- 1 mtsadmin mts 2747 Mar 21 06:14 startup.mqsc

-rw-rw-r-- 1 mtsadmin mts 241 Mar 21 06:15 startup1.sql

-rw-rw-r-- 1 mtsadmin mts 780 Mar 21 06:16 startup2.sql

-rw-rw-r-- 1 mtsadmin mts 624 Mar 21 06:17 startup3.sql

Create the following structure of directories:

$ cd ~/shared

$ mkdir -p Public/RGW\_6.0/xx/sql

Transfer the installation SQL scripts (\*.sql) from your MTS RGW directory to this location.

You can utilise an MTS working environment on the host where you have already installed MTS.

For example:

$ scp [username@**hostvm**:/usr/intranet/rhel74/mts/**60883**/sql/rgw/oracle/ddl/\*.sql](mailto:username@hostvm:/usr/intranet/rhel74/mts/60883/sql/rgw/oracle/ddl/*.sql) .

Where 60883 refers to a version number of MTS, it is known as a Service Pack (SP).

**Scripts:**

Create the directory scripts in the HOME of the mtsadmin user:

$ mkdir ~/scripts

$ cd ~/scripts

Create the following script:

**[mtsadmin@localhost scripts]$ cat create\_ora.sh**

#!/bin/bash #

# This script is designed to clean the RGW container by deleting

# the existing one and creating a new container. When the new

# container starts, a script runs to configure the users

# "marcoslaranz" and "rgwadmin" for access to the RGW and

# to create the necessary database objects.

# ## Clean all ## contid=`sudo podman ps | grep container | grep -i ora | cut -c 1-3`

if [ "$contid" != "" ];then

echo `date`: Removing old Oracle installation ...

sudo podman stop $contid sudo podman container rm $contid

sudo podman volume rm oradata1 sudo podman volume rm shareora

else contid=`sudo podman ps -a | grep container | grep -i ora | cut -c 1-3`

if [ "$contid" != "" ];then

echo The container ID $contid is not active. This will be removed.

sudo podman container rm $contid sudo podman volume rm oradata1

sudo podman volume rm shareora

fi fi sudo podman volume create oradata1 sudo podman volume create shareora

echo `date`: .. Starting container ..

sudo podman run -d --name OraEX -p 1521:1521 -p 5500:5500 -e ORACLE\_CHARACTERSET=AL32UTF8 -e ORACLE\_PWD=pass01 -v oradata1:/opt/oracle/oradata -v /home/mtsadmin/shareora:/opt/shareora:z -t container-registry.oracle.com/database/express:latest contid=`sudo podman ps | grep container | cut -c 1-3`

echo `date`: Container created with the following ID = $contid

sleep 30 echo `date`: Creating database.

sudo podman exec $contid sqlplus / as sysdba @/opt/shareora/startup1.sql

sleep 5 echo `date`: Grant permissions to rgwadmin user.

sudo podman exec $contid sqlplus sys/pass01@localhost/RGW as sysdba @/opt/shareora/startup2.sql

sleep 5 cd /opt/shareora/Public/RGW\_6.0/xx/sql

echo `date`: Create database s objects, tables, storeprocedures, views, etc..

sudo podman exec $contid sqlplus rgwadmin/pass01@localhost/RGW @/opt/shareora/startup3.sql

echo `date`: Finished !!

Create the following scripts into ~/share directory:

**[mtsadmin@localhost shared]$ cat startup1.sql**

-- File: startup1.sql

-- Create a database in Oracle EX edition --

-- sqlplus / as sysdba

create pluggable database RGW admin user rgwadmin identified by pass01 file\_name\_convert = ('/pdbseed/', '/rgw/');

alter pluggable database RGW open

/

**[mtsadmin@localhost shared]$ cat startup2.sql**

--List file: startup2.sql

grant connect to rgwadmin;

grant create view to rgwadmin;

GRANT DROP ANY view TO rgwadmin;

grant create table to rgwadmin;

GRANT DROP ANY table TO rgwadmin;

GRANT ALTER ANY table TO rgwadmin;

grant create trigger to rgwadmin;

GRANT DROP ANY trigger TO rgwadmin;

GRANT ALTER ANY trigger TO rgwadmin;

grant create procedure to rgwadmin;

GRANT DROP ANY PROCEDURE TO rgwadmin;

GRANT ALTER ANY PROCEDURE TO rgwadmin;

CREATE TABLESPACE MESSAGE DATAFILE 'message\_datafile.dat' SIZE 10M REUSE AUTOEXTEND ON NEXT 10M MAXSIZE 200M;

CREATE TABLESPACE STATIC DATAFILE 'static\_datafile.dat' SIZE 10M REUSE AUTOEXTEND ON NEXT 10M MAXSIZE 200M;

alter user rgwadmin quota unlimited on STATIC;

alter user rgwadmin quota unlimited on MESSAGE;

alter user rgwadmin quota unlimited on SYSTEM;

quit

/

**[mtsadmin@localhost shared]$ cat startup3.sql**

--File: startup3.sql

--sqlplus rgwadmin/pass01@localhost/RGW

@/opt/shareora/Public/RGW\_6.0/xx/sql/install.sql

@/opt/shareora/Public/RGW\_6.0/xx/sql/fix\_rrpt\_40092.sql

@/opt/shareora/Public/RGW\_6.0/xx/sql/load\_rrpt\_reporting\_total\_table.sql

@/opt/shareora/Public/RGW\_6.0/xx/sql/rrpt\_install.sql

select object\_name, object\_type from user\_objects where status = 'INVALID';

@/opt/shareora/Public/RGW\_6.0/xx/sql/create\_rrpt\_process\_account\_info.sql

-- this will create some errors: ---

--@rrpt\_process\_account\_info\_end\_of\_day.sql

--@rrpt\_process\_report\_data\_load.sql

--@create\_rrpt\_process\_repair\_tables.sql

quit

/

Open the database:

The scripts will attempt to open the database; however, if you encounter issues connecting using Oracle SQL Developer, you can execute the following commands:

1. Determine the ID of our Oracle container (alternatively, you can use the container name).

$ sudo podman ps

You can use three first letters of the container: (Example)

$ sudo podman ps

Use the three first letters of the container (Example)

[sudo] password for atsadmin:

CONTAINER ID IMAGE CREATED STATUS PORTS COMMAND NAMES

**die**122c71163 container-registry.oracle.com/database/express:latest /bin/bash-c SORA... 15 minutes ago Up 14 minutes (healthy) 0.0.0.0:1521->1521/5CR 0.0.0.0:5500->5500/tcp OraEX

Then:

$ sudo podman exec -it dle bash

Where **dle** are the initial three letters identifying your container.

Once you have identified your container, run the command:

$ sudo exec -it **dle** bash

$ sqlplus / as sysdba

SQL> alter pluggable database RGW open

SQL>/

SQL> exit;

Connect using Oracle SQL Developer:

Create the following connection in Oracle SQL Developer:

Note: use your machine's IP address and set it as a STATIC IP.

The username, service name, and password are automatically set up by the script you ran earlier.

Press the **Test** button before saving; this should display ‘Success'.

(image)

Open a session, then check if you can access some RGW tables:

(image)

Next steps:

Exporting data from MTS database

In the MTS host machine, run these commands:

$ cd ./output

$ by\_logs\_dmp -hist -dest -rtext -rgw -date 05-Jan-2021 -i22

This will create a bunch of extracted files from the MTS database

Then run these next commands to convert those files to the loadable:

rgw\_message today o7

rgw\_msg\_que today o7

rgw\_adt today o7

rgw\_alg today o7

rgw\_aux today o7

rgw\_bnk today o7

rgw\_cfg today o7

rgw\_csp today o7

rgw\_cur today o7

rgw\_dfm today o7

rgw\_fee today o7

rgw\_hol today o7

rgw\_mon today o7

rgw\_opr today o7

rgw\_pin today o7

rgw\_psr today o7

rgw\_rel today o7

rgw\_rmt today o7

rgw\_rpt today o7

rgw\_sat today o7

rgw\_sto today o7

rgw\_stp today o7

rgw\_tpl today o7

rgw\_opr\_act today o7

Note: These usually are run from the ./config/rgw.cfg, but you can run it one-by-one or all together.

After finishing, you will find the LOAD files in the directory ./output.

**Loading files into RGW database**

In the Virtual VM where you are running the Oracle RGW container:

$ sudo su – mtsadmin

Check what the ID of your container

$ sudo podman ps

Use the first three letters of the container ID

Example:

$ cd /home/mtsamin/share/Public/RGW\_6.0/xx/dat

$ scp [username@192.168.26.201:/usr/local/intranet/areas/<AREANAME>/output/\*.LOAD](mailto:username@192.168.26.201:/usr/local/intranet/areas/%3cAREANAME%3e/output/*.LOAD) .

Note: the IP: 192.168.26.201 is for the host where your MTS is running.

Then run the following script:

$ ./load.sh

This script will load all extracted files into the RGW database.

**Next Steps:**

* Setting up RGW Streaming in the MTS