

Oracle XE and APEX on CentOS 7

Downloading the software

The first thing to do is download the software from Oracle Technology Network:

- [Database Downloads](#) - you will need *the package for Linux x64* and the *preinstall RPM package*.
- [Developer Tools/Oracle REST Data Services/Downloads](#)
- [Developer Tools/Application Express/Downloads](#)

After the files have been downloaded, transfer them to the server.

Installation of RDBMS

After you checked them, to install the RDBMS, you need to install the *preinstall RPM package* first and then install the database software as following:

```
wget https://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/getPackage/oracle-database-preinstall-18c-1.0-1.el7.x86_64.rpm
yum install oracle-database-preinstall-18c* -y
yum install oracle-database-xe-18c* -y
yum install httpd tomcat -y
```

The user `oracle` and the group `oinstall` are created during the package installation. A default user environment is created during the set up process. You can set a password for this user by invoking `passwd oracle` command. This user is the owner of the `/opt/oracle` directory where the Oracle Database is located and this must stay unchanged.

```
chown oracle:oinstall /opt/oracle
```

When the packages are installed and the user is set up, you need to run the initial database configuration script and answer all of the questions.

```
/etc/init.d/oracle-xe-18c configure
```

After answering the questions it is going to take several minutes to initialize the database.

Setting up environment

Set up Oracle Database environment variables .

```
echo 'ORACLE_SID=XE' >> /etc/profile.d/oraenv.sh
echo 'ORAENV_ASK=NO' >> /etc/profile.d/oraenv.sh
echo '. /opt/oracle/product/18c/dbhomeXE/bin/oraenv -s' >> /etc/profile.d/oraenv.sh
. /etc/profile.d/oraenv.sh
```

Enable Oracle Database XE service for automatic startup:

```
systemctl enable oracle-xe-18c
```

Connecting to database

And we are ready to log into the database.

```
sqlplus /nolog
```

Check if everything is good.

```
-- connect to the database
sqlplus /nolog

-- change role
CONNECT SYS as SYSDBA

-- basic query to check everything came up right
select * from dual;

-- exit the database
exit
```

To make it easier to connect to the pluggable database, edit thetnsnames.ora file and add there a new connection descriptor


```
        privilege => 'resolve');
dbms_network_acl_admin.assign_acl(acl => 'all-network-PUBLIC.xml',
        host => '*');

END;
/
sho err
COMMIT;
/
```

From the APEX directory connect to the pluggable database as `sysdba` and run the installation scripts.

```
cd /opt/oracle/apex
-- connect to the database
sqlplus /nolog

-- change role
CONN sys@pdb1 AS SYSDBA
```

```
-- run the script to install a full development environment
@apexins.sql SYSAUX SYSAUX TEMP /i/

-- create an instance administrator user and set their password
@apxchpwd.sql

-- unlock APEX public user
ALTER USER APEX_PUBLIC_USER ACCOUNT UNLOCK;
ALTER USER APEX_PUBLIC_USER IDENTIFIED BY "S0m3aw3s0m3Pw!";

-- configure REST Data Services
@apex_rest_config.sql

-- run the ACL setup
@apex_acl.sql

-- disconnect from the database
exit
```

Copy APEX static files to the web server directory. They will be used to serve static images from the proxy.

```
mkdir -p /var/www/apex/images
cp -a /opt/oracle/apex/images/. /var/www/apex/images
```

The Application Express installation is complete.

Installation of ORDS

The Oracle Rest Data Services (ORDS) installation consists of unzipping the downloaded archive, running the configuration command, and then deploying the `ords.war` file into the Tomcat webapps folder.

```
cd /root
mkdir -p /opt/oracle/ords
unzip ords-19.*.zip -d /opt/oracle/ords
```

Run the ORDS configuration command with the advanced mode to run the interactive installation process.

```
cd /opt/oracle/ords
java -jar ords.war install advanced
```

When prompted for *ORDS* configuration directory, enter `config`.
Then provide the connection info to your pluggable database `XEPDB1`

Follow the on screen instructions.

After the configuration is completed, the values are saved in `opt/oracle/ords/config/ords/defaults.xml` file. It can be modified there. See more at Oracle Docs.

The `tomcat` user (created as part of *Tomcat* install) must have read-write access to the *ORDS* configuration folder:

```
chown -R tomcat:tomcat /opt/oracle/ords/config
```

Deploy *ORDS* to Tomcat application server. Copy the `ords.war` into the *Tomcat* `webapps` directory for this

```
cp -a /opt/oracle/ords/ords.war /usr/share/tomcat/webapps/
```

Done with ORDS and Tomcat, on to Apache.

Configuration of Apache httpd to map ORDS

The last step is to configure Apache to map HTTP-requests to ORDS and therefore APEX engine.

For this, add a custom `httpd` configuration file. By default, every `.conf` file placed in the `etc/httpd/conf.d/` directory is read by `httpd` as an additional configuration file to the main `/etc/httpd/conf/httpd.conf` config file.

Create the `apex.conf` file in the `etc/httpd/conf.d/` directory with the contents as below:

```
# forward ORDS tomcat
<VirtualHost *:80>
    # uncomment the lines below if you plan to serve different domains
    # on this web server, don't forget to change the domain name
    # ServerName yourdomain.tld
    # ServerAlias www.yourdomain.tld

    # alias for APEX static files
    Alias "/i" "/var/www/apex/images/"

    # uncomment the line below if you want
    # to redirect traffic to ORDS from root path
    # RedirectMatch permanent "^/$" "/ords"

    # proxy ORDS requests to tomcat
    ProxyRequests off
    <Location "/ords">
        ProxyPass "ajp://localhost:8009/ords"
        ProxyPassReverse "ajp://localhost:8009/ords"
    </Location>
</VirtualHost>
```

Tell SELinux (Yes, that should be running) to allow Apache to communicate to tomcat.

```
setsebool httpd_can_network_connect on
```

Now you are ready to save the configuration file and restart the services.

```
systemctl restart httpd  
systemctl restart tomcat
```

Open a few firewall ports. Yes, the the firewall should also be on.

```
firewall-cmd --permanent --add-service={http,https}  
firewall-cmd --reload
```

And finally, access APEX from your web browser using a link like `http://yourdomain.tld/ords` (or `http://yourdomain.tld` in case you switched on force redirection), where `yourdomain.tld` is the domain name or the IP-address of your server.

Revision #6

Created 5 August 2019 19:03:47 by Michael Cleary

Updated 11 February 2024 02:34:11 by Michael Cleary