Lab 9.1: Exploring the Default Listener

Objective:

To explore the configuration for the default listener, `LISTENER`, and understand dynamic service registration in your Oracle database environment. This lab uses the `CDBLAB` and its PDBs created in previous labs.

Steps:

1. **Open a New Terminal and Set the Environment Variables for `CDBLAB`**

   Open a new terminal window and set the environment variables for the `CDBLAB` database:

   ```
   . oraenv
   ORACLE_SID = [oracle] ? CDBLAB
   The Oracle base has been set to /u01/app/oracle
   ```

2. **Start SQL*Plus and Log in as the SYS User with SYSDBA Privilege**

   Start SQL*Plus and log in as the SYS user with the SYSDBA privilege:

   ```
   sqlplus / as sysdba
   ```

3. **View Initialization Parameters for Dynamic Service Registration**

   a. **INSTANCE_NAME**: This parameter identifies the database instance name.

   ```
   SHOW PARAMETER INSTANCE_NAME;
   ```

   Expected Output:

   ```
   NAME            TYPE        VALUE
   --------------- ----------- ----------
   instance_name   string      CDBLAB
   ```

   b. **SERVICE_NAMES**: This parameter identifies the service names that users can use in their connection strings to connect to the database instance.

   ```
   SHOW PARAMETER SERVICE_NAMES;
   ```

   Expected Output:

   ```
   NAME            TYPE        VALUE
   --------------- ----------- -----------
   service_names   string      CDBLAB
   ```

   c. **LOCAL_LISTENER**: This parameter specifies the alias names for local listeners that resolve to addresses in the `tnsnames.ora` file.

   ```
   SHOW PARAMETER LOCAL_LISTENER;
   ```

   Expected Output:

   ```
   NAME            TYPE        VALUE
   --------------- ----------- ---------------
   local_listener  string      LISTENER_CDBLAB
   ```

d. **REMOTE_LISTENER**: This parameter specifies the alias names for remote listeners.

```
SHOW PARAMETER REMOTE_LISTENER;
```

Expected Output:

```
NAME            TYPE        VALUE
--------------- ----------- -------
remote_listener string
```

4. **Exit SQL*Plus**

```
exit
```

5. **View the `tnsnames.ora` File and Locate the Entry for `LOCAL_LISTENER`**

a. Change directories to `$ORACLE_HOME/network/admin`:

```
cd $ORACLE_HOME/network/admin
```

b. List the files in this directory:

```
ls -l
```

Expected Output:

```
-rw-r--r-- 1 oracle oinstall 287 Jun 27 2019 listener.ora
drwxr-xr-x 2 oracle oinstall 4096 Apr 17 2019 samples
-rw-r--r-- 1 oracle oinstall 1536 Feb 14 2018 shrept.lst
-rw-r----- 1 oracle oinstall 1870 Oct 16 05:06 tnsnames.ora
-rw-r----- 1 oracle oinstall 1870 Oct 16 22:05 tnsnames.old
```

c. View the `tnsnames.ora` file using the `less` command:

```
less tnsnames.ora
```

Expected Content:

```
LISTENER_CDBLAB =
  (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT =
1521))

CDBLAB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT =
1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = CDBLAB)
    )
  )
```

6. **View the `listener.ora` File**

   View the `listener.ora` file using the `cat` command:

```
cat listener.ora
```

   Expected Content:

```
LISTENER =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT =
1521))
  )

ADR_BASE_LISTENER = /u01/app/oracle
```

7. **Start the Listener Control Utility**

   Start the Listener Control utility with the `lsnrctl` command:

```
lsnrctl
```

   Expected Output:

```
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 27-OCT-2020 13:56:52
Copyright (c) 1991, 2019, Oracle. All rights reserved.
Welcome to LSNRCTL, type "help" for information.
LSNRCTL>
```

8. **View Information About the Default Listener**

   a. View the available operations:

```
LSNRCTL> help
```

   b. View the name of the current listener:

```
LSNRCTL> show current_listener
```

   Expected Output:

```
Current Listener is LISTENER
```

   c. View the status of `LISTENER`:

```
LSNRCTL> status
```

   Expected Output:

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
(HOST=your_fully_qualified_hostname)(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
```

```
Version                   TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date                27-OCT-2020 13:56:52
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
/u01/app/oracle/product/19.3.0/dbhome_1/network/admin/listener.ora
Listener Log File
/u01/app/oracle/diag/tnslsnr/your_fully_qualified_hostname/listener/alert/log.xml

Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=your_fully_qualified_hostname)
(PORT=1521)))
Services Summary...
Service "CDBLAB" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
The command completed successfully
```

d. View additional details about the registered services:

```
LSNRCTL> services
```

Expected Output:

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
(HOST=your_fully_qualified_hostname)(PORT=1521)))
Services Summary...
Service "CDBLAB" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB1" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB2" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB3" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
The command completed successfully
```

e. Show the log status:

```
LSNRCTL> show log_status
```

Expected Output:

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
(HOST=your_fully_qualified_hostname)(PORT=1521)))
LISTENER parameter "log_status" set to ON
The command completed successfully
```

9. **Exit the Listener Control Utility**

```
LSNRCTL> exit
```

Part B: Creating a Second Listener

Overview:

In this practice, you create a listener named `LISTENER2` that listens on the non-default port 1561 for all database services. Configure the listener to use dynamic service registration, similar to the default listener, `LISTENER`.

Steps:

1. **Open the `tnsnames.ora` File and Create an Entry for `LISTENER2`**

   a. Set your environment variables using `oraenv` to `CDBLAB`:

   ```
   . oraenv
   ORACLE_SID = [orclcdb] ? CDBLAB
   The Oracle base remains unchanged with value /u01/app/oracle
   ```

   b. Obtain your host name and domain:

   ```
   hostname -f
   ```

   Expected Output:

   ```
   your_fully_qualified_hostname
   ```

   c. Navigate to `$ORACLE_HOME/network/admin`:

   ```
   cd $ORACLE_HOME/network/admin
   ```

   d. Copy the `tnsnames.ora` file and open it in a text editor:

   ```
   cp tnsnames.ora tnsnames.ora.3-2
   vi tnsnames.ora
   ```

```
e. Add an entry for `LISTENER2`:
```

LISTENER2 = (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT = 1561))

```
f. Save the file and exit the editor.

2. **Modify the `LOCAL_LISTENER` Initialization Parameter**

a. Open a new terminal window and set the environment variables for `CDBLAB`:
```sh
. oraenv
ORACLE_SID = [oracle] ? CDBLAB
The Oracle base has been set to /u01/app/oracle
```

b. Start SQL*Plus and log in as the SYS user with the SYSDBA privilege:
```

```
sqlplus / as sysdba
```

c. View the `LOCAL_LISTENER` initialization parameter:

```
SHOW PARAMETER local_listener;
```

Expected Output:

```
NAME            TYPE        VALUE
--------------- ----------- -------------
local_listener  string      LISTENER_CDBLAB
```

d. Check if the `LOCAL_LISTENER` parameter is a static or dynamic parameter:

```
SELECT isses_modifiable, issys_modifiable FROM v$parameter WHERE
name='local_listener';
```

Expected Output:

```
ISSES ISSYS_MOD
----- ---------
FALSE IMMEDIATE
```

e. Set the `LOCAL_LISTENER` parameter to include both `LISTENER_CDBLAB` and `LISTENER2`:

```
ALTER SYSTEM SET local_listener='LISTENER_CDBLAB,LISTENER2';
```

Expected Output:

```
System altered.
```

f. Confirm the new value of the `LOCAL_LISTENER` parameter:

```
SHOW PARAMETER local_listener;
```

Expected Output:

```
NAME            TYPE        VALUE
--------------- ----------- ----------------------
local_listener  string      LISTENER_CDBLAB,LISTENER2
```

g. Exit SQL*Plus:

```
exit
```

3. **Add an Entry for `LISTENER2` in the `listener.ora` File**

   a. Make a copy of the `listener.ora` file:

   ```
   cd $ORACLE_HOME/network/admin
   cp listener.ora listener.old
   ```

   b. Start Oracle Net Manager:

```
netmgr
```

In Oracle Net Manager:

- Expand `Local` and then `Listeners`.
- Click the green plus sign to add a new listener.
- Enter `LISTENER2` as the listener name and click `OK`.
- Click `Add Address` and configure the address with the host name and port 1561.
- Save the network configuration and exit Oracle Net Manager.

c. Verify the new entry in the `listener.ora` file:

```
cat listener.ora
```

Expected Content:

```
LISTENER2 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT =
1561))
  )

LISTENER =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = your_fully_qualified_hostname)(PORT =
1521))
  )

ADR_BASE_LISTENER = /u01/app/oracle
ADR_BASE_LISTENER2 = /u01/app/oracle
```

4. **Start and Verify the New Listener (`LISTENER2`)**

a. Start the Listener Control utility:

```
lsnrctl
```

b. Check the status of `LISTENER2`:

```
LSNRCTL> status LISTENER2
```

Expected Output:

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
(HOST=your_fully_qualified_hostname)(PORT=1561)))
TNS-12541: TNS:no listener
TNS-12560: TNS:protocol adapter error
TNS-00511: No listener
Linux Error: 111: Connection refused
```

c. Start `LISTENER2`:

```
LSNRCTL> start LISTENER2
```

Expected Output:

```
Starting /u01/app/oracle/product/19.3.0/dbhome_1/bin/tnslsnr: please wait...
TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is
/u01/app/oracle/product/19.3.0/dbhome_1/network/admin/listener.ora
Log messages written to
/u01/app/oracle/diag/tnslsnr/your_fully_qualified_hostname/listener2/alert/log.xm

Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)
(HOST=your_fully_qualified_hostname)(PORT=1561)))
The command completed successfully
```

d. Check the status of `LISTENER2` again after waiting for about 60 seconds:

```
LSNRCTL> status LISTENER2
```

Expected Output:

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
(HOST=your_fully_qualified_hostname)(PORT=1561)))
STATUS of the LISTENER
------------------------
Alias listener2
Version TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date 16-OCT-2020 23:27:54
Uptime 0 days 0 hr. 0 min. 55 sec
Trace Level off
Security ON: Local OS Authentication
SNMP OFF
Listener Parameter File
/u01/app/oracle/product/19.3.0/dbhome_1/network/admin/listener.ora
Listener Log File
/u01/app/oracle/diag/tnslsnr/your_fully_qualified_hostname/listener2/alert/log.xm

Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=your_fully_qualified_hostname)
(PORT=1561)))
Services Summary...
Service "CDBLAB" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB1" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB2" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
Service "PDB3" has 1 instance(s).
  Instance "CDBLAB", status READY, has 1 handler(s) for this service...
The command completed successfully
```

5. **Exit the Listener Control Utility**

```
LSNRCTL> exit
```

Summary

In this lab, you explored the configuration for the default listener, `LISTENER`, and dynamic service registration. You then created a new listener named `LISTENER2` that listens on the non-default port 1561, configured it for dynamic service registration, and verified its operation. This practice ensures you understand the configuration and management of multiple listeners in an Oracle database environment.

Part C: Connecting to a Database Service Using the New Listener

Overview:

Now that you have `LISTENER2` configured, test it by making a connection to one of its supported database services, for example, `CDBLAB`.

Steps:

1. **Using Easy Connect Syntax, Start SQL*Plus and Connect to the CDB Using `LISTENER2`**

   Open a terminal and connect to the `CDBLAB` database using `LISTENER2` on port 1561:

   ```
   sqlplus system/password@localhost:1561/CDBLAB
   ```

   Expected Output:

   ```
   SQL*Plus: Release 19.0.0.0.0 - Production on Tue Oct 27 13:56:52 2020
   Version 19.0.0.0.0

   Copyright (c) 1982, 2020, Oracle. All rights reserved.

   Enter password: ********
   Connected to:
   Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
   Version 19.0.0.0.0

   SQL>
   ```

2. **Exit SQL*Plus and Close the Terminal Window**

   Exit SQL*Plus:

   ```
   exit
   ```

   Close the terminal window:

   ```
   exit
   ```

Summary

In this part, you tested the newly configured `LISTENER2` by making a connection to the `CDBLAB` database using Easy Connect syntax. This verifies that `LISTENER2` is correctly configured and can handle connections to the specified database services.