

UNCLASSIFIED



**ORACLE DATABASE 12c
SUPPLEMENTAL PROCEDURES**

NONDEFAULT SQLNET.ORA CONFIGURATIONS

Version 2, Release 8

26 July 2023

Developed by Oracle and DISA for the DOD

UNCLASSIFIED

Trademark Information

Names, products, and services referenced within this document may be the trade names, trademarks, or service marks of their respective owners. References to commercial vendors and their products or services are provided strictly as a convenience to our users, and do not constitute or imply endorsement by the Defense Information Systems Agency (DISA) of any nonfederal entity, event, product, service, or enterprise.

Table of Contents

	Page
1. NON-DEFAULT SQLNET.ORA CONFIGURATIONS	1

1. NON-DEFAULT SQLNET.ORA CONFIGURATIONS

In Oracle, the “sqlnet.ora” file contains parameters that can be used to define network settings used by the listener, database, or both. By default, Oracle will look in the \$ORACLE_HOME/network/admin directory for this file. This can be redirected to an alternate location by setting the TNS_ADMIN environment variable.

However, it is possible that what the TNS_ADMIN variable is currently set to is not the same as it was when the listener or database was started. Further, it is also possible within the “listener.ora” to configure the listener to use a different TNS_ADMIN per database.

To ensure that the STIG checks are checking **EVERY** “sqlnet.ora” that is in use, we need a method to determine each possible location. Testing for the most complicated method will also cover the simpler methods.

Adapt the techniques illustrated here to STIG requirements as applicable.

Step 1. Determine all Listeners running on the host.

For Windows hosts, view all Windows services with “TNSListener” embedded in the service name.

- The service name format is:

```
Oracle[ORACLE_HOME_NAME]TNSListener
```

For UNIX hosts, the Oracle Listener process will indicate the TNSLSNR executable.

At a command prompt, issue the command:

```
ps -ef | grep tnslnsr | grep -v grep
```

The alias for the listener follows “tnslnsr” in the command output.

User must be logged on the host system using the account that owns the “tnslnsr” executable (UNIX). If the account is denied local logon, have the system SA assist in this task by adding ‘su’ to the listener account from the root account. On Windows platforms, log on using an account with administrator privileges to complete the check.

From a system command prompt, execute the listener control utility:

```
lsnrctl status [LISTENER NAME]
```

Note the results for the value of “Listener Parameter File”, “Listening Endpoints Summary”, and each “Service”/”Instance” listed under the “Services Summary”.

Repeat the execution of the “lsnrctl” utility for all active listeners.

Step 2. Determine the sqlnet.ora file(s) used by each Service/Instance.

For each “Instance” listed under the “Services Summary” in Step 1, search within the “Listener Parameter File” (listener.ora) for an entry that begins with SID_LIST_[LISTENER NAME]. If this entry exists, look for a SID_DESC subsection containing a GLOBAL_DBNAME entry corresponding to the listed “Service”, and a SID_NAME entry corresponding to the listed “Instance”. If this section exists and contains an ENVS entry specifying TNS_ADMIN, note each distinct location(s) specified.

For instance:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=
(GLOBAL_DBNAME = ORCL)
(ORACLE_HOME = /home/oracle/app/oracle/product/12.1.0/db_1)
(SID_NAME = ORCL)
(ENVS="TNS_ADMIN=/home/oracle/app/oracle/product/12.1.0/db_1"))
```

If no such entry exists, or any do NOT specify a TNS_ADMIN location, or there are any listed “Service”/“Instance” NOT specified in this manner, then we must also determine the location of the “sqlnet.ora” used when the listener and database instance processes were started. The listener process will have used a “sqlnet.ora” file in the same directory as the “listener.ora”, located in Step 1. To determine the location at the time that database was started, log in as SYSDBA and issue the following block:

```
sqlplus / as sysdba
set serveroutput on
declare
  folder varchar2(200);
begin
  sys.dbms_system.get_env('TNS_ADMIN', folder);
  if folder is null then
    sys.dbms_system.get_env('ORACLE_HOME', folder);
    folder:=folder||'/network/admin';
  end if;
  dbms_output.put_line(folder);
end;
/
```

If ANY of the locations determined in this step does NOT contain a “sqlnet.ora” file, then default values will be used and this is a finding.

Step 3. Check the contents of the sqlnet.ora file(s).

If the following entries do not exist in **EVERY** “sqlnet.ora” file identified, then restriction by IP address is not configured and is a finding.

```
tcp.validnode_checking=YES  
tcp.invited_nodes=(IP1, IP2, IP3)
```