

Oracle® Database
Net Services Reference
11g Release 2 (11.2)
E10835-09

October 2012

Oracle Database Net Services Reference, 11g Release 2 (11.2)

E10835-09

Copyright © 2002, 2012, Oracle and/or its affiliates. All rights reserved.

Contributors: Robert Achacoso, Lance Ashdown, Abhishek Dadhich, Santanu Datta, Steve Ding, Caroline Johnston, Feroz Khan, Bhaskar Mathur, Scot McKinley, Ed Miner, Sweta Mogra, Srinivas Pamu, Kant Patel, Murali Purayathu, Karthik Rajan, Saravanakumar Ramasubramanian, Sudeep Reguna, Richard Strohm, Norman Woo

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	xi
Audience	xi
Documentation Accessibility	xi
Related Documents	xi
Conventions	xii

Part I Control Utilities

1 Listener Control Utility

Listener Control Utility Overview	1-1
SET and SHOW Commands of the Listener Control utility	1-2
Distributed Operations	1-2
Oracle Net Listener Security	1-3
Listener Control Utility Commands	1-3
EXIT	1-4
HELP	1-4
QUIT	1-5
RELOAD	1-5
SAVE_CONFIG	1-6
SERVICES	1-7
SET	1-8
SET CURRENT_LISTENER	1-8
SET DISPLAYMODE	1-9
SET INBOUND_CONNECT_TIMEOUT	1-9
SET LOG_DIRECTORY	1-10
SET LOG_FILE	1-11
SET LOG_STATUS	1-11
SET SAVE_CONFIG_ON_STOP	1-12
SET TRC_DIRECTORY	1-12
SET TRC_FILE	1-13
SET TRC_LEVEL	1-13
SHOW	1-14
SPAWN	1-15
START	1-15
STATUS	1-17

STOP.....	1-18
TRACE.....	1-18
VERSION.....	1-19

2 Oracle Connection Manager Control Utility

Oracle Connection Manager Control Utility Overview.....	2-1
Command Modes and Syntax.....	2-1
Oracle Connection Manager Control Utility Commands.....	2-2
ADMINISTER	2-3
CLOSE CONNECTIONS	2-4
EXIT.....	2-5
HELP	2-6
QUIT.....	2-7
RELOAD	2-7
RESUME GATEWAYS	2-8
SAVE_PASSWD	2-8
SET.....	2-9
SET ASO_AUTHENTICATION_FILTER.....	2-9
SET CONNECTION_STATISTICS	2-10
SET EVENT.....	2-10
SET IDLE_TIMEOUT.....	2-11
SET INBOUND_CONNECT_TIMEOUT.....	2-12
SET LOG_DIRECTORY.....	2-12
SET LOG_LEVEL	2-13
SET OUTBOUND_CONNECT_TIMEOUT.....	2-14
SET PASSWORD.....	2-14
SET SESSION_TIMEOUT	2-15
SET TRACE_DIRECTORY.....	2-15
SET TRACE_LEVEL.....	2-16
SHOW	2-17
SHOW ALL.....	2-17
SHOW CONNECTIONS.....	2-18
SHOW DEFAULTS.....	2-20
SHOW EVENTS	2-20
SHOW GATEWAYS	2-21
SHOW PARAMETERS.....	2-21
SHOW RULES	2-22
SHOW SERVICES	2-23
SHOW STATUS.....	2-24
SHOW VERSION	2-25
SHUTDOWN	2-25
STARTUP	2-26
SUSPEND GATEWAY	2-27

Part II Configuration Parameters

3 Syntax Rules for Configuration Files

Configuration File Syntax Overview.....	3-1
Syntax Rules for Configuration Files.....	3-2
Network Character Set for Keywords.....	3-2
Character Set for Listener and Net Service Names.....	3-3

4 Protocol Address Configuration

Protocol Addresses.....	4-1
ADDRESS.....	4-1
ADDRESS_LIST.....	4-2
Protocol Parameters.....	4-2
Recommended Port Numbers.....	4-3
Port Number Limitations.....	4-3

5 Parameters for the sqlnet.ora File

Overview of Profile Configuration File.....	5-1
sqlnet.ora Profile Parameters.....	5-1
BEQUEATH_DETACH.....	5-3
DEFAULT_SDU_SIZE.....	5-4
DISABLE_OOB.....	5-4
NAMES.DEFAULT_DOMAIN.....	5-4
NAMES.DIRECTORY_PATH.....	5-5
NAMES.LDAP_AUTHENTICATE_BIND.....	5-5
NAMES.LDAP_CONN_TIMEOUT.....	5-6
NAMES.LDAP_PERSISTENT_SESSION.....	5-6
RECV_BUF_SIZE.....	5-7
SDP.PF_INET_SDP.....	5-7
SEC_USER_AUDIT_ACTION_BANNER.....	5-7
SEC_USER_UNAUTHORIZED_ACCESS_BANNER.....	5-8
SEND_BUF_SIZE.....	5-8
SQLNET.ALLOWED_LOGON_VERSION.....	5-9
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE.....	5-11
SQLNET.AUTHENTICATION_SERVICES.....	5-11
SQLNET.CLIENT_REGISTRATION.....	5-12
SQLNET.CRYPTO_CHECKSUM_CLIENT.....	5-12
SQLNET.CRYPTO_CHECKSUM_SERVER.....	5-12
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT.....	5-13
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER.....	5-13
SQLNET.ENCRYPTION_CLIENT.....	5-13
SQLNET.ENCRYPTION_SERVER.....	5-14
SQLNET.ENCRYPTION_TYPES_CLIENT.....	5-14
SQLNET.ENCRYPTION_TYPES_SERVER.....	5-15
SQLNET.EXPIRE_TIME.....	5-15
SQLNET.INBOUND_CONNECT_TIMEOUT.....	5-16
SQLNET.KERBEROS5_CC_NAME.....	5-17
SQLNET.KERBEROS5_CLOCKSKEW.....	5-17

SQLNET.KERBEROS5_CONF	5-17
SQLNET.KERBEROS5_KEYTAB.....	5-18
SQLNET.KERBEROS5_REALMS	5-18
SQLNET.OUTBOUND_CONNECT_TIMEOUT.....	5-18
SQLNET.RADIUS_ALTERNATE.....	5-19
SQLNET.RADIUS_ALTERNATE_PORT	5-19
SQLNET.RADIUS_ALTERNATE_RETRIES.....	5-19
SQLNET.RADIUS_AUTHENTICATION.....	5-20
SQLNET.RADIUS_AUTHENTICATION_INTERFACE.....	5-20
SQLNET.RADIUS_AUTHENTICATION_PORT.....	5-20
SQLNET.RADIUS_AUTHENTICATION_RETRIES	5-20
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT.....	5-21
SQLNET.RADIUS_CHALLENGE_RESPONSE	5-21
SQLNET.RADIUS_SECRET	5-21
SQLNET.RADIUS_SEND_ACCOUNTING.....	5-22
SQLNET.RECV_TIMEOUT	5-22
SQLNET.SEND_TIMEOUT	5-22
SSL_CERT_REVOCATION	5-23
SSL_CERT_FILE	5-23
SSL_CERT_PATH	5-24
SSL_CIPHER_SUITES	5-24
SSL_CLIENT_AUTHENTICATION	5-25
SSL_SERVER_DN_MATCH.....	5-25
SSL_VERSION	5-26
TCP.CONNECT_TIMEOUT.....	5-26
TCP.EXCLUDED_NODES.....	5-26
TCP.INVITED_NODES.....	5-27
TCP.NODELAY.....	5-27
TCP.VALIDNODE_CHECKING.....	5-27
TNSPING.TRACE_DIRECTORY.....	5-28
TNSPING.TRACE_LEVEL	5-28
USE_CMAN.....	5-28
USE_DEDICATED_SERVER.....	5-29
WALLET_LOCATION	5-29
WALLET_OVERRIDE.....	5-31
ADR Diagnostic Parameters in sqlnet.ora.....	5-32
ADR_BASE.....	5-33
DIAG_ADR_ENABLED.....	5-33
TRACE_LEVEL_CLIENT.....	5-33
TRACE_LEVEL_SERVER	5-34
TRACE_TIMESTAMP_CLIENT	5-34
TRACE_TIMESTAMP_SERVER.....	5-34
Non-ADR Diagnostic Parameters in sqlnet.ora	5-35
LOG_DIRECTORY_CLIENT.....	5-35
LOG_DIRECTORY_SERVER	5-36
LOG_FILE_CLIENT.....	5-36
LOG_FILE_SERVER	5-36

TRACE_DIRECTORY_CLIENT	5-36
TRACE_DIRECTORY_SERVER.....	5-37
TRACE_FILE_CLIENT.....	5-37
TRACE_FILE_SERVER	5-37
TRACE_FILELEN_CLIENT.....	5-38
TRACE_FILELEN_SERVER	5-38
TRACE_FILENO_CLIENT	5-38
TRACE_FILENO_SERVER.....	5-38
TRACE_UNIQUE_CLIENT.....	5-39

6 Local Naming Parameters (tnsnames.ora)

Overview of Local Naming Parameters	6-1
General Syntax of tnsnames.ora	6-2
Multiple Descriptions in tnsnames.ora	6-2
Multiple Address Lists in tnsnames.ora	6-3
Connect-Time Failover and Client Load Balancing with Oracle Connection Managers	6-3
Connect Descriptor Descriptions	6-4
DESCRIPTION	6-5
DESCRIPTION_LIST	6-5
Protocol Address Section	6-5
ADDRESS	6-5
ADDRESS_LIST.....	6-6
Optional Parameters for Address Lists	6-6
ENABLE	6-6
FAILOVER	6-7
LOAD_BALANCE	6-7
RECV_BUF_SIZE	6-8
SDU	6-9
SEND_BUF_SIZE	6-9
SOURCE_ROUTE.....	6-10
TYPE_OF_SERVICE.....	6-11
Connection Data Section	6-11
CONNECT_DATA.....	6-11
FAILOVER_MODE.....	6-12
GLOBAL_NAME	6-13
HS	6-13
INSTANCE_NAME	6-14
RDB_DATABASE.....	6-14
SERVER	6-14
SERVICE_NAME	6-15
Security Section	6-15
SECURITY	6-16
SSL_SERVER_CERT_DN	6-16
Timeout Parameters	6-16
CONNECT_TIMEOUT.....	6-17
RETRY_COUNT.....	6-17
TRANSPORT_CONNECT_TIMEOUT	6-18

7 Oracle Net Listener Parameters (listener.ora)

Overview of Oracle Net Listener Configuration File	7-1
Protocol Address Parameters	7-2
ADDRESS.....	7-2
DESCRIPTION	7-3
IP.....	7-3
QUEUESIZE.....	7-4
RECV_BUF_SIZE	7-4
SEND_BUF_SIZE	7-5
Connection Rate Limiter Parameters	7-5
CONNECTION_RATE_listener_name	7-6
RATE_LIMIT.....	7-6
Control Parameters	7-7
ADMIN_RESTRICTIONS_listener_name	7-7
CRS_NOTIFICATION_listener_name	7-7
DEFAULT_SERVICE_listener_name	7-8
INBOUND_CONNECT_TIMEOUT_listener_name	7-8
SAVE_CONFIG_ON_STOP_listener_name	7-9
SSL_CLIENT_AUTHENTICATION	7-9
SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name	7-10
WALLET_LOCATION	7-10
ADR Diagnostic Parameters for Oracle Net Listener	7-12
ADR_BASE_listener_name	7-12
DIAG_ADR_ENABLED_listener_name	7-13
LOGGING_listener_name.....	7-13
TRACE_LEVEL_listener_name.....	7-13
TRACE_TIMESTAMP_listener_name.....	7-14
Non-ADR Diagnostic Parameters for Oracle Net Listener	7-14
LOG_DIRECTORY_listener_name	7-14
LOG_FILE_listener_name	7-15
TRACE_DIRECTORY_listener_name	7-15
TRACE_FILE_listener_name	7-15
TRACE_FILELEN_listener_name.....	7-15
TRACE_FILENO_listener_name.....	7-16
Class of Secure Transports Parameters	7-16
DYNAMIC_REGISTRATION_listener_name.....	7-16
SECURE_CONTROL_listener_name	7-17
SECURE_REGISTER_listener_name	7-17
SECURE_PROTOCOL_listener_name	7-18
Using COST Parameters in Combination.....	7-19

8 Oracle Connection Manager Parameters (cman.ora)

Overview of Oracle Connection Manager Configuration File	8-1
Oracle Connection Manager Parameters	8-3
ADDRESS.....	8-4
ASO_AUTHENTICATION_FILTER.....	8-4
CONNECTION_STATISTICS	8-5

EVENT_GROUP.....	8-5
IDLE_TIMEOUT.....	8-5
INBOUND_CONNECT_TIMEOUT.....	8-6
LOG_DIRECTORY.....	8-6
LOG_LEVEL.....	8-6
MAX_CMCTL_SESSIONS.....	8-6
MAX_CONNECTIONS.....	8-6
MAX_GATEWAY_PROCESSES.....	8-7
MIN_GATEWAY_PROCESSES.....	8-7
OUTBOUND_CONNECT_TIMEOUT.....	8-7
PASSWORD_ <i>instance_name</i>	8-7
RULE.....	8-7
SESSION_TIMEOUT.....	8-9
TRACE_FILE.....	8-9
TRACE_FILELEN.....	8-9
TRACE_FILENO.....	8-9
TRACE_LEVEL.....	8-9
TRACE_TIMESTAMP.....	8-10
ADR Diagnostic Parameters for Oracle Connection Manager.....	8-10
ADR_BASE.....	8-10
DIAG_ADR_ENABLED.....	8-11
LOG_LEVEL.....	8-11
TRACE_LEVEL.....	8-11
TRACE_TIMESTAMP.....	8-12
Non-ADR Diagnostic Parameters for Oracle Connection Manager.....	8-12
LOG_DIRECTORY.....	8-13
TRACE_DIRECTORY.....	8-13
TRACE_FILELEN.....	8-13
TRACE_FILENO.....	8-14

9 Directory Usage Parameters (ldap.ora)

Overview of Directory Server Usage File.....	9-1
Directory Usage Parameters.....	9-1
DIRECTORY_SERVERS.....	9-1
DIRECTORY_SERVER_TYPE.....	9-2
DEFAULT_ADMIN_CONTEXT.....	9-2

Part III Appendixes

A Features Not Supported in this Release

Overview of Unsupported Features.....	A-1
Oracle Names.....	A-1
Unsupported Parameters.....	A-1
Unsupported Control Utility Commands.....	A-1

B Upgrade Considerations for Oracle Net Services

Anonymous Access to Oracle Internet Directory	B-1
Migrating a Non-default Listener to Oracle Database 11g Release 2 (11.2)	B-1
Removing the Listener Password	B-1
Client and Database Coexistence Issues	B-2
Oracle9i Database Connections	B-2
Oracle Names	B-3

C LDAP Schema for Oracle Net Services

Structural Object Classes	C-1
Attributes	C-2

Glossary

Index

Preface

The *Oracle Database Net Services Reference* contains a complete listing and description of the control utility commands and configuration file parameters available for managing components of Oracle Net Services.

This document describes the features of Oracle Database 11g software that apply to the Microsoft Windows and UNIX operating systems.

This preface contains the following topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)

Audience

Oracle Database Net Services Reference is intended for network administrators who are responsible for configuring and administering network components.

To use this document, you should be familiar with the networking concepts and configuration tasks described in *Oracle Database Net Services Administrator's Guide*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For additional information, see the following Oracle resources:

- *Oracle Database Net Services Administrator's Guide*

- Online Help for Oracle Net Services tools and utilities
- Oracle Database documentation set

A glossary of Net Services terms is available in the Oracle Net Services Administrator's Guide.

Many books in the documentation set use the sample schemas of the seed database, which is installed by default when you install Oracle. Refer to *Oracle Database Sample Schemas* for additional information about how these schemas were created and how you can use them yourself.

To download free release notes, installation documentation, white papers, or other collateral, visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

<http://www.oracle.com/technetwork/index.html>

If you have a user name and password for OTN, then you can go directly to the documentation section of the OTN Web site at

<http://www.oracle.com/technetwork/indexes/documentation/index.html>

Conventions

The examples for directories in the book are for Linux. Unless otherwise noted, Microsoft Windows directory paths are the same except that they use a backslash (\) instead of the slash (/).

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Part I

Control Utilities

Oracle Net Services provides control utilities to administer listeners, Oracle Names servers, and Oracle Connection Manager. Part 1 lists the commands that are available with each utility, including any applicable prerequisites, passwords, syntax or argument rules, and usage notes or examples to help you use them.

This part contains the following chapters:

- [Chapter 1, "Listener Control Utility"](#)
- [Chapter 2, "Oracle Connection Manager Control Utility"](#)

Listener Control Utility

This chapter describes the commands and associated syntax of the **Listener Control utility**.

This chapter contains the following topics:

- [Listener Control Utility Overview](#)
- [SET and SHOW Commands of the Listener Control utility](#)
- [Distributed Operations](#)
- [Oracle Net Listener Security](#)
- [Listener Control Utility Commands](#)

Listener Control Utility Overview

The Listener Control utility enables you to administer **listeners**. You can use its commands to perform basic management functions on one or more listeners. Additionally, you can view and change parameter settings.

The basic syntax of Listener Control utility commands is as follows:

```
lsnrctl command listener_name
```

In the preceding command, *listener_name* is the name of the listener to be administered. If no name is specified, then the default name, LISTENER, is assumed.

You can also issue Listener Control utility commands at the LSNRCTL> program prompt. To obtain the prompt, enter `lsnrctl` with no arguments at the operating system command line. When you run `lsnrctl`, the program is started. You can then enter the necessary commands from the program prompt. The basic syntax of issuing commands from LSNRCTL> program prompt is as follows:

```
lsnrctl  
LSNRCTL> command listener_name
```

Note: You can combine commands in a standard text file, and then run them as a sequence of commands. To execute in batch mode, use the format:

```
lsnrctl @file_name
```

You can use either `REM` or `#` to identify comments in the batch script; all other lines are considered commands. Any commands that would typically require confirmation do not require confirmation during batch execution.

For most commands, the Listener Control utility establishes an Oracle Net connection with the listener that is used to transmit the command. To initiate an Oracle Net connection to the listener, the Listener Control utility must obtain the protocol addresses for the named listener or a listener named `LISTENER`. This is done by resolving the listener name with one of the following mechanisms:

- `listener.ora` file in the directory specified by the `TNS_ADMIN` environment variable
- `listener.ora` file in the `ORACLE_HOME/network/admin` directory.
- Naming method, for example, a `tnsnames.ora` file

If none of the preceding mechanisms resolve the listener name, then the Listener Control utility uses the default listener name `LISTENER`, resolves the host name IP address and uses port 1521.

The Listener Control utility supports several types of commands:

- Operational commands, such as [START](#), and [STOP](#).
- Modifier commands, such as [SET TRC_LEVEL](#).
- Informational commands, such as [STATUS](#) and `SHOW LOG_FILE`.

SET and SHOW Commands of the Listener Control utility

You can use the [SET](#) command to alter parameter values for a specified listener. You set the name of the listener you want to administer with the [SET CURRENT_LISTENER](#) command. Parameter values remain in effect until the listener is shut down. If you want these settings to persist, then use the [SAVE_CONFIG](#) command to save changes to the `listener.ora`.

You can use the [SHOW](#) command to display the current value of a configuration setting.

Distributed Operations

The Listener Control utility can perform operations on a local or a remote listener.

To set up a computer to remotely administer a listener, do the following:

1. Ensure that the Listener Control utility (`lsnrctl`) executable is installed in the `ORACLE_HOME/bin` directory.
2. Ensure that the name of the listener you want to administer can be resolved through a `listener.ora` file or a naming method, as described in "[Listener Control Utility Overview](#)" on page 1-1.

All commands except **START** can be issued when a listener is administered remotely. The Listener Control utility can only start the listener on the same computer from where the utility is running.

When issuing commands, specify the listener name as an argument. For example:

```
LSNRCTL> SERVICES lsnr
```

If the name is omitted, then listener name set with the **SET CURRENT_LISTENER** command is used, or the default name, **LISTENER** is assumed.

Oracle Net Listener Security

Local listener administration is secure through local operating system authentication, which restricts listener administration to the user who started the listener or to the super user. By default, remote listener administration is disabled.

Oracle recommends that you perform listener administration in the default mode, and access the system remotely using a remote login. When you administer the listener remotely, use Oracle Enterprise Manager or Secure Shell (SSH) to access the remote host.

Listener Control Utility Commands

This section describes the following Listener Control utility commands:

- **EXIT**
- **HELP**
- **QUIT**
- **RELOAD**
- **SAVE_CONFIG**
- **SERVICES**
- **SET**
- **SET CURRENT_LISTENER**
- **SET DISPLAYMODE**
- **SET INBOUND_CONNECT_TIMEOUT**
- **SET LOG_DIRECTORY**
- **SET LOG_FILE**
- **SET LOG_STATUS**
- **SET SAVE_CONFIG_ON_STOP**
- **SET TRC_DIRECTORY**
- **SET TRC_FILE**
- **SET TRC_LEVEL**
- **SHOW**
- **SPAWN**
- **START**
- **STATUS**

- [STOP](#)
- [TRACE](#)
- [VERSION](#)

EXIT

Purpose

To exit from the Listener Control utility, and return to the operating system prompt.

Prerequisites

None

Syntax

From the Listener Control utility:

```
LSNRCTL> EXIT
```

Arguments

None

Usage Notes

This command is identical to the [QUIT](#) command.

Example

```
LSNRCTL> EXIT
```

HELP

Purpose

To provide a list of all the Listener Control utility commands or provide syntax help for a particular Listener Control utility command.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl HELP command
```

From the Listener Control utility:

```
LSNRCTL> HELP command
```

Arguments

command: The Listener Control utility command. Commands are shown in the following example output.

When you enter a command as an argument to `HELP`, the Listener Control utility displays information about how to use the command. When you enter `HELP` without an argument, the Listener Control utility displays a list of all the commands.

Example

```
LSNRCTL> HELP
The following operations are available
An asterisk (*) denotes a modifier or extended command:
exit
quit
reload
services
set*
show*
spawn
start
status
stop
trace
version
```

QUIT**Purpose**

To exit the Listener Control utility and return to the operating system prompt.

Prerequisites

None

Syntax

From the Listener Control utility

```
LSNRCTL> QUIT
```

Arguments

None

Usage Notes

This command is identical to the [EXIT](#) command.

Example

```
LSNRCTL> QUIT
```

RELOAD**Purpose**

To reread the `listener.ora` file. This command enables you to add or change statically configured services without actually stopping the listener.

In addition, the database services, instances, service handlers, and listening endpoints that were dynamically registered with the listener are unregistered and subsequently registered again.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl RELOAD listener_name
```

From the Listener Control utility:

```
LSNRCTL> RELOAD listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Example

```
LSNRCTL> RELOAD  
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=sales-server) (PORT=1521)))  
The command completed successfully
```

SAVE_CONFIG**Purpose**

To save the current configuration state of the listener, including trace level, trace file, trace directory, and logging to the `listener.ora` file. Any changes are stored in `listener.ora`, preserving formatting, comments, and case as much as possible. Before modification of the `listener.ora` file, a backup of the file, called `listener.bak`, is created.

Syntax

From the operating system:

```
lsnrctl SAVE_CONFIG listener_name
```

From the Listener Control utility:

```
LSNRCTL> SAVE_CONFIG listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Usage Notes

This command enables you to save all run-time configuration changes to the `listener.ora` file.

Example

```
LSNRCTL> SAVE_CONFIG listener  
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=sales-server) (PORT=1521)))  
Saved LISTENER configuration parameters.  
Listener Parameter File /oracle/network/admin/listener.ora  
Old Parameter File /oracle/network/admin/listener.bak  
The command completed successfully
```

SERVICES

Purpose

To obtain detailed information about the database services, instances, and service handlers (dispatchers and dedicated servers) to which the listener forwards client connection requests.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SERVICES listener_name
```

From the Listener Control utility:

```
LSNRCTL> SERVICES listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Usage Notes

The [SET DISPLAYMODE](#) command changes the format and the detail level of the output.

See Also: *Oracle Database Net Services Administrator's Guide* for a complete description of SERVICES output

Example

This example shows SERVICES output in the default display mode. The output shows the following:

- An instance named sales belonging to two services, sales1.us.example.com and sales2.us.example.com, with a total of three service handlers.
- Service sales1.us.example.com is handled by one dispatcher only.
- Service sales2.us.example.com is handled by one dispatcher and one dedicated server, as specified by in the following output.

```
LSNRCTL> SERVICES
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=net)))
Services Summary...
Service "sales1.us.example.com" has 1 instance(s).
  Instance "sales", status READY, has 1 handler(s) for this service...
    Handler(s):
      "D000" established:0 refused:0 current:0 max:10000 state:ready
        DISPATCHER <machine: sales-server, pid: 5696>
        (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=53411))
Service "sales2.us.example.com" has 1 instance(s).
  Instance "sales", status READY, has 2 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
      "D001" established:0 refused:0 current:0 max:10000 state:ready
        DISPATCHER <machine: sales-server, pid: 5698>
        (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=52618))
```

The command completed successfully

SET

Purpose

To alter the parameter values for the listener. Parameter value changes remain in effect until the listener is shut down. To make the changes permanent, use the [SAVE_CONFIG](#) command to save changes to the `listener.ora` file.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET parameter
```

From the Listener Control utility:

```
LSNRCTL> SET parameter
```

Arguments

parameter: A SET parameter to modify its configuration setting. Parameters are shown in the example output.

When you enter SET without an argument, the Listener Control utility displays a list of all the parameters.

Usage Notes

If you are using the SET commands to alter the configuration of a listener other than the default LISTENER listener, then use the [SET CURRENT_LISTENER](#) command to set the name of the listener you want to administer.

Example

```
LSNRCTL> SET
The following operations are available with set.
An asterick (*) denotes a modifier or extended command.
current_listener
displaymode
inbound_connect_timeout
log_file
log_directory
log_status
rawmode
save_config_on_stop
trc_file
trc_directory
trc_level
```

SET CURRENT_LISTENER

Purpose

To set the name of the listener to administer. Subsequent commands that would normally require *listener_name* can be issued without it.

Syntax

From the Listener Control utility

```
LSNRCTL> SET CURRENT_LISTENER listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Usage Notes

When SET CURRENT_LISTENER is set, the Listener Control utility commands act on the listener you set. You do not have to specify the name of the listener.

Example

```
LSNRCTL> SET CURRENT_LISTENER lsnr
Current Listener is lsnr
```

SET DISPLAYMODE**Purpose**

To change the format and level of detail for the [SERVICES](#) and [STATUS](#) commands.

Syntax

From the Listener Control utility:

```
LSNRCTL> SET DISPLAYMODE {compat | normal | verbose | raw}
```

Arguments

Specify one of the following modes:

compat: Output that is compatible with older versions of the listener.

normal: Output that is formatted and descriptive. Oracle recommends this mode.

verbose: All data received from the listener in a formatted and descriptive output.

raw: All data received from the listener without any formatting. This argument should be used only if recommended by Oracle Support Services.

Example

```
LSNRCTL> SET DISPLAYMODE normal
Service display mode is NORMAL
```

SET INBOUND_CONNECT_TIMEOUT**Purpose**

To specify the time, in seconds, for the client to complete its connect request to the listener after establishing the network connection.

If the listener does not receive the client request in the time specified, then it terminates the connection. In addition, the listener logs the IP address of the client and an ORA-12525:TNS: listener has not received client's request in time allowed error message to the listener.log file.

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about specifying the time out for client connections

Syntax

From the Listener Control utility:

```
LSNRCTL> SET INBOUND_CONNECT_TIMEOUT time
```

Arguments

time: The time, in seconds. Default setting is 60 seconds.

Example

```
LSNRCTL> SET INBOUND_CONNECT_TIMEOUT 2
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
LISTENER parameter "inbound_connect_timeout" set to 2
The command completed successfully.
```

SET LOG_DIRECTORY

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled, and the log directory is ORACLE_HOME/log/diag/*product_type*.

Purpose

To set destination directory where the listener log file is written. By default, the log file is written to the ORACLE_HOME/network/log directory.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET LOG_DIRECTORY directory
```

From the Listener Control utility:

```
LSNRCTL> SET LOG_DIRECTORY directory
```

Arguments

directory: The directory path of the listener log file.

Example

```
LSNRCTL> SET LOG_DIRECTORY /usr/oracle/admin
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
LISTENER parameter "log_directory" set to /usr/oracle/admin
The command completed successfully
```

SET LOG_FILE

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled, and the log directory is `ORACLE_HOME/log/diag/product_type`.

Purpose

To set the name for the listener log file. By default, the log file name is `listener.log`.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET LOG_FILE file_name
```

From the Listener Control utility:

```
LSNRCTL> SET LOG_FILE file_name
```

Arguments

file_name: The file name of the listener log.

Example

```
LSNRCTL> SET LOG_FILE list.log
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
LISTENER parameter "log_file" set to list.log
The command completed successfully
```

SET LOG_STATUS

Purpose

To turn listener logging on or off.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET LOG_STATUS {on | off}
```

From the Listener Control utility:

```
LSNRCTL> SET LOG_STATUS {on | off}
```

Arguments

`on`: To turn logging on.

`off`: To turn logging off.

Example

```
LSNRCTL> SET LOG_STATUS on
Connecting to (ADDRESS=(PROTOCOL=TCP) (HOST=sales-server) (PORT=1521))
LISTENER parameter "log_status" set to ON
The command completed successfully
```

SET SAVE_CONFIG_ON_STOP

Purpose

To specify whether changes made to the parameter values for the listener by the [SET](#) commands are to be saved to the `listener.ora` file at the time the listener is stopped with the [STOP](#) command.

When changes are saved, the Listener Control utility tries to preserve formatting, comments, and letter case. Before modification of the `listener.ora` file, a back up of the file, called `listener.bak`, is created.

To have all parameters saved immediately, use the [SAVE_CONFIG](#) command.

Syntax

From the operating system:

```
lsnrctl SET SAVE_CONFIG_ON_STOP {on | off}
```

From the Listener Control utility:

```
LSNRCTL> SET SAVE_CONFIG_ON_STOP {on | off}
```

Arguments

`on`: To save configuration to `listener.ora`.

`off`: To not save configuration to `listener.ora`.

Example

```
LSNRCTL> SET SAVE_CONFIG_ON_STOP on
LISTENER parameter "save_config_on_stop" set to ON
The command completed successfully
```

SET TRC_DIRECTORY

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled, and the log directory is `ORACLE_HOME/log/diag/product_type`.

Purpose

To set the destination directory where the listener trace files are written. By default, the trace file are written to the `ORACLE_HOME/network/trace` directory.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET TRC_DIRECTORY directory
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_DIRECTORY directory
```

Arguments

directory: The directory path of the listener trace files.

Example

```
LSNRCTL> SET TRC_DIRECTORY /usr/oracle/admin
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
LISTENER parameter "trc_directory" set to /usr/oracle/admin
The command completed successfully
```

SET TRC_FILE

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled, and the log directory is `ORACLE_HOME/log/diag/product_type`.

Purpose

To set the name of the listener trace file. By default, the trace file name is `listener.trc`.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET TRC_FILE file_name
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_FILE file_name
```

Arguments

file_name: The file name of the listener trace.

Example

```
LSNRCTL> SET TRC_FILE list.trc
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
LISTENER parameter "trc_file" set to list.trc
The command completed successfully
```

SET TRC_LEVEL

Purpose

To set a specific level of tracing for the listener.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SET TRC_LEVEL level
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_LEVEL level
```

Arguments

level: One of the following trace levels:

- `off` for no trace output
- `user` for user trace information
- `admin` for administration trace information
- `support` for Oracle Support Services trace information

Usage Notes

This command has the same functionality as the [TRACE](#) command.

Example

```
LSNRCTL> SET TRC_LEVEL admin
Connecting to (ADDRESS=(PROTOCOL=TCP) (HOST=sales-server) (PORT=1521))
LISTENER parameter "trc_level" set to admin
The command completed successfully
```

SHOW**Purpose**

To view the current parameter values for the listener.

All of the [SET](#) parameters have equivalent `SHOW` parameters.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SHOW parameter
```

From the Listener Control utility:

```
LSNRCTL> SHOW parameter
```

Arguments

parameter: A `SHOW` parameter to view its configuration settings. Parameters are shown in the example output.

When you enter `SHOW` without an argument, the Listener Control utility displays a list of all the parameters.

Example

```
LSNRCTL> SHOW
The following properties are available with SHOW:
An asterisk (*) denotes a modifier or extended command:
current_listener
displaymode
inbound_connect_timeout
log_file
log_directory
log_status
rawmode
save_config_on_stop
trc_file
trc_directory
trc_level
```

SPAWN**Purpose**

To start a program stored on the computer on which the listener is running, and which is listed with an alias in the `listener.ora` file.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl SPAWN listener_name alias (arguments='arg1,arg2,...')
```

From the Listener Control utility

```
LSNRCTL> SPAWN listener_name alias (arguments='arg1,arg2,...')
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

alias: The alias of the program to be spawned off is specified by a `listener.ora` file entry, similar to the following:

```
alias = (PROGRAM=(NAME=) (ARGS=) (ENVS=))
```

For example:

```
nstest = (PROGRAM=(NAME=nstest) (ARGS=test1) (ENVS='ORACLE_HOME=/usr/oracle'))
```

Example

This program can then be spawned off using the following command:

```
lsnrctl SPAWN listener_name nstest
```

START**Purpose**

To start the named listener.

Prerequisites

Listener must not be running.

Syntax

From the operating system:

```
lsnrctl START listener_name
```

From the Listener Control utility:

```
LSNRCTL> START listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Usage Notes

To start a listener configured in the `listener.ora` file with a name other than LISTENER, include that name.

For example, if the listener name is `tcp_lsnr`, enter:

```
lsnrctl START tcp_lsnr
```

From the Listener Control utility:

```
LSNRCTL> START tcp_lsnr
```

Example

```
LSNRCTL> START
```

```
Starting /private/sales_group/sales/bin/tnslsnr: please wait...
```

```
TNSLSNR for Linux: Version 11.2.0.0.0
```

```
System parameter file is /oracle/network/admin/listener.ora
```

```
Log messages written to /oracle/network/log/listener.log
```

```
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)))
```

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
```

```
STATUS of the LISTENER
```

```
-----
```

```
Alias                LISTENER
Version              TNSLSNR for Linux: Version 11.2.0.0.0
Start Date           15-NOV-2008 18:02:25
Uptime               0 days 0 hr. 0 min. 0 sec
Trace Level          off
Security             OFF
SNMP                 OFF
Listener Parameter File /oracle/network/admin/listener.ora
Listener Log File    /oracle/network/log/listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)))
The listener supports no services
The command completed successfully
```

STATUS

Purpose

To display basic status information about a listener, including a summary of listener configuration settings, listening protocol addresses, and a summary of services registered with the listener.

Note: You can also obtain the status of the listener through the Oracle Enterprise Manager Console. See the *Oracle Database 2 Day DBA* for additional information.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl STATUS listener_name
```

From the Listener Control utility:

```
LSNRCTL> STATUS listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Usage Notes

The [SET DISPLAYMODE](#) command changes the format and level of the detail of the output.

See Also: *Oracle Database Net Services Administrator's Guide* for a complete description of STATUS output

Example

The following example shows STATUS output in the default display mode. The output contains:

- Listener configuration settings
- Listening endpoints summary
- Services summary, which is an abbreviated version of the [SERVICES](#) command output

```
LSNRCTL> STATUS
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=net)))
STATUS of the LISTENER
-----
Alias                LISTENER
Version              TNSLSNR for Linux: Version 11.2.0.0.0 -
Beta
Start Date           15-JAN-2008 12:02:00
Uptime                0 days 0 hr. 5 min. 29 sec
Trace Level           support
Security              OFF
SNMP                  OFF
```

```
Listener Parameter File  /oracle/network/admin/listener.ora
Listener Log File       /oracle/network/log/listener.log
Listener Trace File     /oracle/network/trace/listener.trc

Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=net)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=2484)))

Services Summary...
Service "sales1.us.example.com" has 1 instance(s).
  Instance "sales", status READY, has 1 handler(s) for this service...
Service "sales2.us.example.com" has 1 instance(s).
  Instance "sales", status READY, has 2 handler(s) for this service...
The command completed successfully
```

STOP

Purpose

To stop the named listener.

Prerequisites

The listener must be running.

Syntax

From the operating system:

```
lsnrctl STOP listener_name
```

From the Listener Control utility:

```
LSNRCTL> STOP listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Example

```
LSNRCTL> STOP
Connecting to (ADDRESS=(PROTOCOL=TCP) (HOST=sales-server) (PORT=1521))
The command completed successfully
```

TRACE

Purpose

To turn on tracing for the listener.

Syntax

From the operating system:

```
lsnrctl trace level listener_name
```

From the Listener Control utility:

```
LSNRCTL> trace level listener_name
```

Arguments

level: One of the following trace levels:

- `off` for no trace output
- `user` for user trace information
- `admin` for administration trace information
- `support` for Oracle Support Services trace information

listener_name: Specify the listener name, if the default name of LISTENER is not used.

Usage Notes

This command has the same functionality as `SET TRC_LEVEL` command.

Example

```
LSNRCTL> TRACE ADMIN lsnr
Connecting to (ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
Opened trace file: /oracle/network/trace/listener.trc
The command completed successfully
```

VERSION**Purpose**

To display the current version of Listener Control utility.

Prerequisites

None

Syntax

From the operating system:

```
lsnrctl VERSION listener_name
```

From the Listener Control utility:

```
LSNRCTL> VERSION listener_name
```

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.

Example

```
LSNRCTL> VERSION listener1
Connecting to ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)
TNSLSNR for Linux: Version 11.2.0.0.0
  TNS for Linux: Version 11.2.0.0.0
  Oracle Bequeath NT Protocol Adapter for Linux: Version 11.2.0.0.0
  Unix Domain Socket IPC NT Protocol Adaptor for Linux: Version 11.2.0.0.0
  TCP/IP NT Protocol Adapter for Linux: Version 11.2.0.0.0
The command completed successfully
```

Oracle Connection Manager Control Utility

This chapter describes the commands and associated syntax of the [Oracle Connection Manager Control utility](#).

This chapter contains the following topics:

- [Oracle Connection Manager Control Utility Overview](#)
- [Command Modes and Syntax](#)
- [Oracle Connection Manager Control Utility Commands](#)

Oracle Connection Manager Control Utility Overview

The Oracle Connection Manager Control utility enables you to administer [Oracle Connection Managers](#). You can use its commands to perform basic management functions on one or more Oracle Connection Managers. Additionally, you can view and change parameter settings.

Command Modes and Syntax

The basic syntax of the Oracle Connection Manager Control utility is as follows:

```
cmctl command [argument]
```

You can use Oracle Connection Manager Control utility in command mode, or batch mode.

- Using the command mode:

- From the Oracle Connection Manager Control utility:

Enter `cmctl` at the command line to obtain the program prompt; then issue the command:

```
cmctl  
CMCTL> command
```

- From the operating system:

Enter the entire command from the operating system command prompt:

```
cmctl [command] [argument1 . . . argumentN] [-c instance_name]
```

Each command issued in this way can have the name of an Oracle Connection Manager appended as an argument. If an Oracle Connection Manager instance name is not provided, then the default instance name is assumed. The default name is `cman_hostname`. You may be prompted for a password if one was

set in a previous CMCTL session. Issuing commands from an Oracle Connection Manager Control utility session of Oracle Connection Manager requires that a password be entered once, at the beginning of the session, if one has been set.

Caution: There is an option to specify the password on the command line. However, this exposes the password on the screen, and is a potential security risk. Oracle recommends not using the password option (-p) on the command line.

See Also: *Oracle Database Net Services Administrator's Guide* for an overview of the Oracle Connection Manager processes

- Using the batch mode:

You can combine commands in a standard text file and then run them as a sequence of commands. To run in batch mode, use the following syntax:

```
cmctl @input_file
```

The Oracle Connection Manager Control utility supports four types of commands:

- Initialization and termination commands such as [STARTUP](#) and [SHUTDOWN](#)
- Alter commands such as [SET LOG_LEVEL](#) and [SET EVENT](#)
- Display commands, such as [SHOW STATUS](#) and [SHOW RULES](#)
- Gateway commands such as [SHOW GATEWAYS](#) and [RESUME GATEWAYS](#)

Note: You can use [SET](#) commands to dynamically alter configuration parameters, the changes only remain in effect until the Oracle Connection Manager is shut down. You cannot save them to the `cman.ora` file. The one exception is the Oracle Connection Manager password, which you can save using the command [SAVE_PASSWD](#).

Oracle Connection Manager Control Utility Commands

This section lists and describes the following commands for the Oracle Connection Manager Control utility:

- [ADMINISTER](#)
- [CLOSE CONNECTIONS](#)
- [EXIT](#)
- [HELP](#)
- [QUIT](#)
- [RELOAD](#)
- [RESUME GATEWAYS](#)
- [SAVE_PASSWD](#)
- [SET](#)
- [SET ASO_AUTHENTICATION_FILTER](#)

- SET CONNECTION_STATISTICS
- SET EVENT
- SET IDLE_TIMEOUT
- SET INBOUND_CONNECT_TIMEOUT
- SET LOG_DIRECTORY
- SET LOG_LEVEL
- SET OUTBOUND_CONNECT_TIMEOUT
- SET PASSWORD
- SET SESSION_TIMEOUT
- SET TRACE_DIRECTORY
- SET TRACE_LEVEL
- SHOW
- SHOW ALL
- SHOW CONNECTIONS
- SHOW DEFAULTS
- SHOW EVENTS
- SHOW GATEWAYS
- SHOW PARAMETERS
- SHOW RULES
- SHOW SERVICES
- SHOW STATUS
- SHOW VERSION
- SHUTDOWN
- STARTUP
- SUSPEND_GATEWAY

ADMINISTER

Purpose

To select an Oracle Connection Manager instance.

Prerequisites

None

Syntax

From the Oracle Connection Manager Control utility:

```
CMCTL> ADMINISTER [-c] instance_name
```

Arguments

instance_name: The instance of Oracle Connection Manager that you would like to administer. Instances are defined in the `cman.ora` file.

Usage Notes

You can issue the `ADMINISTER` command only within the utility. You cannot issue the command from the operating system.

`ADMINISTER` enables you to choose an Oracle Connection Manager to administer. To start this Oracle Connection Manager, you must issue `STARTUP`.

When you omit the instance name from the command, the instance administered defaults to the local instance.

Use the `-c` option when you want to administer an instance that is not the local instance.

A password is required only if one was provided at installation time or during a previous session of the Oracle Connection Manager.

Example

```
CMCTL> ADMINISTER cman_indl040ad
Enter CMAN password: password
Current instance cman_indl040ad is already started
Connections refer to (address=(protocol=TCP) (host=indl040ad) (port=1560)).
The command completed successfully
```

CLOSE CONNECTIONS

Purpose

To terminate connections, using specific qualifiers to select connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl CLOSE CONNECTIONS [in state] [gt time] [from source] [to destination]
[for service] [using gateway_process_id] [connect_identifier_list]
[-c cman_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> CLOSE CONNECTIONS [in state] [gt time] [from source] [to destination]
[for service] [using gateway_process_id] [connect_identifier_list]
```

Arguments

state: One of the following values to specify the connection state:

- `idle`: Connections that are inactive in the established state
- `connecting`: Connections that are in the process of connecting
- `established`: Connections that are connected and are transferring data
- `terminating`: Connections that are disconnecting

If no state is specified, then `CLOSE CONNECTIONS` defaults to all possible states. If the time qualifier is included under these conditions, then the time specified is the amount of time that has elapsed since a client initiated a connection.

time: The time format. Use the following format to specify connections greater than the time indicated:

`gt [hh:mm:]ss`

source: The source address. Use one of the following formats to specify the source address:

- from *IP*
- from *hostname*
- from *subnet*

destination: The destination address. Use one of the following formats to specify the destination address:

- to *IP*
- to *hostname*
- to *subnet*

service: The service name. Use the `service_name` parameter to specify the service, such as `sales.us.example.com`.

gateway_process_id: The gateway process identifier is a number. Use this number to specify connections that are proxied by the gateway process indicated. To determine the gateway process identifier, use the Oracle Connection Manager control utility `show gateways` command.

connect_identifier_list: The connection identifiers. Use a space between multiple connection identifiers in a list.

Usage Notes

Because the `CLOSE CONNECTIONS` command terminates connections, it might generate error messages on both client and server sides.

The `IDLE` state qualifier always requires a time qualifier.

Issuing `CLOSE CONNECTIONS` without an argument closes all connections.

Examples

The following example shuts down connections in any state. The elapsed time of the connection must be greater than 1 hour and 30 minutes. The connection source is the specified subnet, and the destination is the specified host name.

```
CMCTL>
CLOSE CONNECTIONS gt 1:30:00 from 192.168.2.32/27 to host1
```

The following example shuts down those connections proxied by gateway process 0 that have been in the idle state more than 30 minutes:

```
CMCTL> CLOSE idle CONNECTIONS gt 30:00 using 0
```

The following example shuts down connections that are connected to the service `sales.us.example.com`:

```
CMCTL> CLOSE established CONNECTIONS for sales.us.example.com
```

EXIT

Purpose

To exit from the Oracle Connection Manager Control utility.

Prerequisites

None

Syntax

From the operating system:

```
cmctl EXIT [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> EXIT
```

Usage Notes

This command is identical to the [QUIT](#) command.

Example

```
CMCTL> EXIT
```

HELP**Purpose**

To provide a list of all commands for the Oracle Connection Manager Control utility or to provide help with the syntax of a particular command.

Prerequisites

None

Syntax

From the operating system:

```
cmctl HELP [command] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> HELP [command]
```

Arguments

command: Specify a HELP command. Commands are shown in the following example output.

When you enter a command as an argument to HELP, the Oracle Connection Manager Control utility displays information about how to use the command. When you enter HELP without an argument, the Oracle Connection Manager Control utility displays a list of all the commands.

Example

```
CMCTL> HELP
```

The following operations are available

An asterisk (*) denotes a modifier or extended command:

administer	close*	exit	reload
resume*	save_passwd	set*	show*
shutdown	sleep	startup	suspend*
show_version	quit		

QUIT

Purpose

To exit the Oracle Connection Manager Control utility and return to the operating system prompt.

Prerequisites

None

Syntax

From the operating system:

```
cmctl QUIT
```

From the Oracle Connection Manager Control utility:

```
CMCTL> QUIT
```

Usage Notes

This command is identical to the [EXIT](#) command.

Example

```
CMCTL> QUIT
```

RELOAD

Purpose

To dynamically reread parameters and rules.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl RELOAD [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> RELOAD
```

Usage Notes

Configuration information modified using the RELOAD command applies only to new connections. Existing connections are unaffected. The SET RELOAD command restores configurations set in `cmcn.ora`, and override the SET command.

RELOAD reregisters gateways with the Oracle Connection Manager listener during which some new connections might be refused until the registration process is complete.

Example

```
CMCTL> RELOAD  
The command completed successfully
```

RESUME GATEWAYS

Purpose

To resume gateway processes that have been suspended.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl RESUME GATEWAYS [gateway_process_id] [cman_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> RESUME GATEWAYS [gateway_process_id]
```

Arguments

gateway_process_id: One or more gateway processes to reopen. Separate multiple gateway processes using a space between the process identifiers.

Usage Notes

Issuing `RESUME GATEWAYS` without an argument reopens all gateway processes that have been closed.

Example

```
CMCTL> RESUME GATEWAYS 1  
The command completed successfully
```

SAVE_PASSWD

Purpose

To save the current password to the `cman.ora` file, the configuration file for Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SAVE_PASSWD [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SAVE_PASSWD
```

Usage Notes

If you run this command, then the next session of Oracle Connection Manager will use this password. The password is stored in an encrypted format in the `cman.ora` file.

Example

```
CMCTL> SAVE_PASSWD
```

SET

Purpose

To display a list of parameters that can be modified using this command.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SET
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET
```

Example

```
CMCTL> SET
```

The following operations are available after set

An asterisk (*) denotes a modifier or extended command:

aso_authentication_filter	outbound_connect_timeout
connection_statistics	password
event	session_timeout
idle_timeout	trace_directory
inbound_connect_timeout	trace_level
log_directory	
log_level	

SET ASO_AUTHENTICATION_FILTER

Purpose

To indicate whether the client must use Oracle Advanced Security to authenticate.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET ASO_AUTHENTICATION_FILTER {on | off}[-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET ASO_AUTHENTICATION_FILTER {on | off}
```

Arguments

on: To reject connections that are not using Secure Network Service (SNS) to perform client authentication. SNS is part of Oracle Advanced Security.

off: To specify whether no authentication is required for client connections. This is the default.

Example

```
CMCTL> set aso_authentication_filter ON
CMAN_user.us.example.com parameter aso_authentication_filter set to ON
The command completed successfully
```

SET CONNECTION_STATISTICS**Purpose**

To specify whether gateway processes collect connection statistics.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET CONNECTION_STATISTICS {yes | no}[-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET CONNECTION_STATISTICS {yes | no}
```

Arguments

yes: To have gateway processes collect connection statistics.

no: To not have gateway processes collect connection statistics. This is the default.

Usage Notes

If `SET CONNECTION_STATISTICS` is set to `yes`, then you can obtain statistics by issuing the [SHOW CONNECTIONS](#) command.

Example

```
CMCTL> set connection_statistics ON
CMAN_user.us.example.com parameter connection_statistics set to ON
The command completed successfully
```

SET EVENT**Purpose**

To log information for a particular event.

Syntax

From the operating system:

```
cmctl SET EVENT event_group [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET EVENT event_group {on | off}
```

Arguments

event_group: Specify one of the following event groups:

- `init_and_term`: Initialization and termination event group.

- `memory_ops`: Memory operations event group.
- `conn_hdlg`: Connection handling event group.
- `proc_mgmt`: Process management event group.
- `reg_and_load`: Registration and load update event group.
- `wake_up`: Events related to CMADMIN wakeup queue event group.
- `timer`: Gateway timeouts event group.
- `cmd_proc`: Command processing event group.
- `relay`: Events associated with connection control blocks event group.

`on`: To turn an event group on.

`off`: To turn an event group off.

Usage Notes

The `SET EVENT` command accepts only one argument. To log multiple events, you must issue the command for each event separately.

Example

```
CMCTL> set event memory_ops off
cman11 event memory_ops set to OFF.
The command completed successfully
```

SET IDLE_TIMEOUT

Purpose

To specify the amount of time a client can be idle without transmitting data.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET IDLE_TIMEOUT [time] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET IDLE_TIMEOUT [time]
```

Arguments

time: Specify the idle timeout in seconds. The default is 0, which disables this feature.

Example

```
CMCTL> SET IDLE_TIMEOUT 30
CMAN_user.us.example.com parameter idle_timeout set to 30
The command completed successfully
```

SET INBOUND_CONNECT_TIMEOUT

Purpose

To specify the maximum amount of time the Oracle Connection Manager listener waits for a valid connection request from the client before timing out.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET INBOUND_CONNECT_TIMEOUT [time] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET INBOUND_CONNECT_TIMEOUT [time]
```

Arguments

time: The inbound connect timeout in seconds. The default is 0, which disables this feature.

Example

```
CMCTL> SET INBOUND_CONNECT_TIMEOUT 30
CMAN_user.us.example.com parameter inbound_connect_timeout set to 30
The command completed successfully
```

SET LOG_DIRECTORY

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled, and the log directory is ORACLE_HOME/log.

Purpose

To designate where the log files for an Oracle Connection Manager are written.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET LOG_DIRECTORY [directory_path] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET LOG_DIRECTORY [directory_path]
```

Arguments

directory_path: The location of the log directory. The default path is as follows:

- Linux and UNIX:
ORACLE_HOME/network/log directory

- Microsoft Windows:
ORACLE_HOME\network\log directory

Usage Notes

Use the [SHOW PARAMETERS](#) command to determine the location of the log files.

Example

```
CMCTL>
SET LOG_DIRECTORY /disk1/user_cman_test/oracle/network/admin

CMAN_user.us.example.com parameter log_directory set to /disk1/user
_cman_test/oracle/network/admin

The command completed successfully
```

SET LOG_LEVEL

Purpose

To set the log level for an Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET LOG_LEVEL [level] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET LOG_LEVEL [level]
```

Arguments

level: Specify one of the following log levels:

- off: No logging.
- user: User log information.
- admin: Administrative log information.
- support: Oracle Support Services log information. This is the default.

Usage Notes

Specify `off` to capture the minimum amount of log information. Specify `support` to capture the maximum amount.

Example

```
CMCTL> SET LOG_LEVEL SUPPORT
CMAN_user.us.example.com parameter log_level set to support
The command completed successfully
```

SET OUTBOUND_CONNECT_TIMEOUT

Purpose

To specify the maximum amount of time the Oracle Connection Manager instance waits for a valid connection with the server before timing out.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET OUTBOUND_CONNECT_TIMEOUT [time] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET OUTBOUND_CONNECT_TIMEOUT [time]
```

Arguments

time: The outbound connect timeout in seconds. The default is 0.

Example

```
CMCTL> SET OUTBOUND_CONNECT_TIMEOUT 30
CMAN_user.us.example.com parameter outbound_connect_timeout set to 30
The command completed successfully
```

SET PASSWORD

Purpose

To assign a password to the Oracle Connection Manager instance.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET PASSWORD
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET PASSWORD
```

Arguments

None.

Usage Notes

This command may be used either to set a password for the first time or to change an existing one.

This command does not save the password to `cmn.ora`. As a result the password is valid only for the current session. To save the password after you have set it, run the [SAVE_PASSWD](#) command.

Example

```
CMCTL> SET PASSWORD

Enter Old password: old_password
Enter New password: new_password
Reenter New password: new_password

The command completed successfully
```

SET SESSION_TIMEOUT**Purpose**

To specify the maximum amount of time for a session of Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET SESSION_TIMEOUT [time] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET SESSION_TIMEOUT [time]
```

Arguments

time: The session timeout in seconds. The default is 0, which disables this feature.

Example

```
CMCTL> SET SESSION_TIMEOUT 60
CMAN_user.us.example.com parameter session_timeout set to 60
The command completed successfully
```

SET TRACE_DIRECTORY

Note: This command works only if Automatic Diagnostic Repository (ADR) is not enabled. The default is for ADR to be enabled.

Purpose

To designate where the trace files for an Oracle Connection Manager are written.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET TRACE_DIRECTORY [directory_path] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET TRACE_DIRECTORY [directory_path]
```

Arguments

directory_path: The location of the trace directory. The default path is ORACLE_HOME/network/trace.

Usage Notes

Use the [SHOW PARAMETERS](#) command to determine the location of the trace files.

Example

```
CMCTL>SET TRACE_DIRECTORY /disk1/mpurayat_newtest/oracle/network/trace
cman1 parameter trace_directory set to /disk1/mpurayat_newtest/oracle/network
/trace
The command completed successfully
```

SET TRACE_LEVEL**Purpose**

To set the trace level for an Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET TRACE_LEVEL [level] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET TRACE_LEVEL [level]
```

Arguments

level: Specify one of the following log levels:

- off: No tracing. This is the default.
- user: User trace information.
- admin: Administrative trace information.
- support: Oracle Support Services trace information.

Usage Notes

Specify `off` to capture the minimum amount of trace information. Specify `support` to capture the maximum amount.

Use the [SHOW PARAMETERS](#) command to determine the current trace level.

Example

```
CMCTL> SET TRACE_LEVEL SUPPORT
CMAN_user.us.example.com parameter trace_level set to user
The command completed successfully
```

SHOW

Purpose

To display a list of parameters that may be used as arguments for this command. Entering one of these parameters with the command displays the parameter value or values.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHOW [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW
```

Example

```
CMCTL> SHOW
```

The following operations are available after show

An asterisk (*) denotes a modifier or extended command:

all	gateways	status
connections	parameters	version
defaults	rules	
events	services	

SHOW ALL

Purpose

To combine and display output from the SHOW PARAMETERS and SHOW RULES commands.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW ALL [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW ALL
```

Example

```
CMCTL> SHOW ALL
```

```
listener_address |
(address=(protocol=tcp) (host=users.us.example.com) (port=1630))
aso_authentication_filter | OFF
connection_statistics | OFF
event_group | OFF
log_directory | /disk1/user_cman_test/oracle/network/log/
```

```

log_level          | SUPPORT
max_connections    | 256
idle_timeout       | 0
inbound_connect_timeout | 0
session_timeout    | 0
outbound_connect_timeout | 0
max_gateway_processes | 16
min_gateway_processes | 2
max_cmctl_sessions | 4
password           | OFF
trace_directory    | /disk1/user_cman_test/oracle/network/trace/
trace_level        | OFF
trace_timestamp    | OFF
trace_filelen      | 0
trace_fileno       | 0
(rule_list=
(rule=
(src=*)
(dst=*)
(srv=*)
(act=accept)
)
)
)
The command completed successfully

```

SHOW CONNECTIONS

Purpose

To display information about specific connections or all connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```

cmctl SHOW CONNECTIONS [information] [in state] [gt time] [from source]
to destination] [for service] [using gateway_process_id] [connect_identifier_list]
[-c instance_name]

```

From the Oracle Connection Manager Control utility:

```

CMCTL> SHOW CONNECTIONS [information][in state] [gt time] [from source]
[to destination] [for service] [using gateway_process_id] [connect_identifier_
list]

```

Arguments

information: Specify one of the following values to display information about connections. Information categories include connection ID, source, destination, service, current state, total idle time, and total elapsed time.

- **count**: The total number of connections that meet the criteria specified by the other qualifiers. This is the default.
- **detail**: All information about connections specified by the other qualifiers.

state: Specify one of the following values to specify the connection state:

- **idle**: Connections that are inactive in the established state.

- `connecting`: Connections that are in the process of connecting.
- `established`: Connections that are connected and are transferring data.
- `terminating`: Connections that are disconnecting.

If no state is specified, then `SHOW CONNECTIONS` defaults to all possible states. If the time qualifier is included under these conditions, then the time specified is the amount of time that has elapsed since a client initiated a connection.

time: Use the following format to specify connections greater than the time indicated:

`gt [hh:mm:]ss`

source: Specify one of the following formats to specify the source address:

- `from IP`
- `from hostname`
- `from subnet`

destination: Specify one of the following formats to specify the destination address:

- `to IP`
- `to hostname`
- `to subnet`

service: Use the `service_name` format to request a service:

gateway_process_id: Use the following format to specify connections that are proxied by the gateway process indicated:

`using gateway_process_id`

connect_identifier_list: Separate multiple connection identifiers using a space.

Usage Notes

Connections are sorted by gateway process ID and connection identifier, in ascending order.

Issuing `SHOW CONNECTIONS` without an argument displays all connections.

Examples

The following displays a detailed description of connections in any state. The elapsed time of the connection must be greater than 1 hour and 30 minutes. The connection source is the specified subnet, and the destination the specified host name.

```
CMCTL> SHOW CONNECTIONS gt 1:30:00 from 192.168.2.32/27 to host1
```

The following displays the number of connections proxied by Oracle Connection Manager using the gateway process identifier 0 that have been in the idle state more than 30 minutes:

```
CMCTL> SHOW idle CONNECTIONS count gt 30:00 using 0
```

The following displays a detailed description of connections that are connected to the service `sales.us.example.com`:

```
CMCTL> SHOW established CONNECTIONS detail for sales.us.example.com
```

SHOW DEFAULTS

Purpose

To display default parameter settings.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW DEFAULTS [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW DEFAULTS
```

Example

```
CMCTL> SHOW DEFAULTS
listener_address          |
(ASSOCIATED_INSTANCE_NAME=) (ADDRESS=(PROTOCOL=TCP) (HOST=users.us.example.com) (PORT=1521))
aso_authentication_filter | OFF
connection_statistics     | OFF
event_group               | OFF
log_directory             | /disk1/user_cman_test/oracle/network/log/
log_level                  | SUPPORT
max_connections           | 256
idle_timeout              | 0
inbound_connect_timeout  | 0
session_timeout           | 0
outbound_connect_timeout | 0
max_gateway_processes    | 16
min_gateway_processes    | 2
max_cmctl_sessions       | 4
password                  | OFF
trace_directory           | /disk1/user_cman_test/oracle/network/trace/
trace_level                | OFF
trace_timestamp           | OFF
trace_filelen             | 0
trace_fileno              | 0
The command completed successfully
```

SHOW EVENTS

Purpose

To display the events that are in operation.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW EVENTS [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW EVENTS
```

Example

```
CMCTL> SHOW EVENTS
Event Groups:
memory_ops
The command completed successfully
```

SHOW GATEWAYS

Purpose

To display the current status of a specific gateway process or processes. Statistics displayed include number of active connections, number of peak active connections, total number of connections handled, and number of connections refused.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW GATEWAYS [gateway] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW GATEWAYS [gateway]
```

Arguments

gateway: The identifier of the gateway or gateways whose status you want to display.

Issuing `SHOW GATEWAYS` without an argument displays the status of all gateway processes.

Usage Notes

To display multiple gateways, then use a space to separate the identifiers when entering the command.

Example

```
CMCTL> SHOW GATEWAYS 1
Gateway ID                1
Gateway state             READY
Number of active connections 0
Peak active connections   0
Total connections         0
Total connections refused 0
The command completed successfully
```

SHOW PARAMETERS

Purpose

To display current parameter settings for an instance.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW PARAMETERS [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW PARAMETERS
```

Usage Notes

Several configuration parameters can be dynamically modified using the [SET](#) command. Therefore, the information that SHOW PARAMETERS displays might be different from what appears in the `cmctl.ora` file.

Example

```
CMCTL> SHOW PARAMETERS
listener_address          |
(address=(protocol=tcp)(host=users.us.example.com)(port=1630))
aso_authentication_filter |      ON
connection_statistics    |      ON
event_group              | (memory_ops)
log_directory            | /disk1/user_cman_test/oracle/network/log/
log_level                | SUPPORT
max_connections          |     256
idle_timeout             |         0
inbound_connect_timeout |         0
session_timeout         |         0
outbound_connect_timeout |         0
max_gateway_processes   |      16
min_gateway_processes   |         2
max_cmctl_sessions      |         4
password                |      OFF
trace_directory         | /disk1/user_cman_test/oracle/network/trace/
trace_level             | SUPPORT
trace_timestamp         |      OFF
trace_filelen           |         0
trace_fileno            |         0
The command completed successfully
```

SHOW RULES

Purpose

To display the access control list currently used by the instance.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW RULES [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW RULES
```

Usage Notes

You can update the rules list by issuing the [RELOAD](#) command.

Example

```

CMCTL> SHOW RULES
Number of filtering rules currently in effect: 4
(rule_list=
  (rule=
    (src=usunnae12)
    (dst=usunnae13)
    (srv=*)
    (act=accept)
    (action_list=(mit=120) (mct=1800) (conn_stats=on) (aut=off))
  )
  (rule=
    (src=usunnae12)
    (dst=usunnae14)
    (srv=service2)
    (act=accept)
  )
  (rule=
    (src=*)
    (dst=usunnae15)
    (srv=*)
    (act=accept)
    (action_list=(mit=120) (mct=3000) (moct=200) (aut=on))
  )

  (rule=
    (src=*)
    (dst=usunnae16)
    (srv=*)
    (act=reject)
    (action_list=(moct=20) (aut=on))
  )

  (rule=
    (src=users.us.example.com)
    (dst=users.us.example.com)
    (srv=cmon)
    (act=accept)
    (action_list=(mit=100) (mct=1130) (moct=200) (aut=on))
  )
)

```

SHOW SERVICES

Purpose

To display comprehensive information about the Oracle Connection Manager instance. The information displayed includes number of handlers for gateway and CMADMIN processes, listening ports of handlers, and number of connections, both refused and current.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW SERVICES [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW SERVICES
```

Example

```
CMCTL> SHOW SERVICES
Services Summary...
Proxy service "cmgw" has 1 instance(s).
  Instance "cman", status READY, has 2 handler(s) for this service...
    Handler(s):
      "cmgw001" established:0 refused:0 current:0 max:256 state:ready
        <machine: user-sun, pid: 29190>
        (ADDRESS=(PROTOCOL=tcp) (HOST=user-sun) (PORT=33175))
      "cmgw000" established:0 refused:0 current:0 max:256 state:ready
        <machine: user-sun, pid: 29188>
        (ADDRESS=(PROTOCOL=tcp) (HOST=user-sun) (PORT=33174))
Service "cmon" has 1 instance(s).
  Instance "cman", status READY, has 1 handler(s) for this service...
    Handler(s):
      "cmon" established:0 refused:0 current:0 max:4 state:ready
        <machine: user-sun, pid: 29184>
        (ADDRESS=(PROTOCOL=tcp) (HOST=users) (PORT=33168))
The command completed successfully
```

SHOW STATUS

Purpose

To display basic information about the instance, including version, start time, and current statistics.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW STATUS
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW STATUS
```

Example

```
CMCTL> SHOW STATUS
Status of the Instance
-----
Instance name          CMAN_user.us.example.com
Version                CMAN for Linux: Version 11.2.0.0.0
Start date             20-JAN-2008 14:50:35
Uptime                 0 days 1 hr. 25 min. 24 sec
Num of gateways started 2
Average Load level     0
```

```

Log Level                SUPPORT
Trace Level              OFF
Instance Config file     /disk1/user_cman_test/oracle/network/admin/cman.ora
Instance Log directory   /disk1/user_cman_test/oracle/network/log/
Instance Trace directory /disk1/user_cman_test/oracle/network/trace/
The command completed successfully

```

SHOW VERSION

Purpose

To display the current version and name of the Oracle Connection Manager Control utility.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHOW VERSION [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW VERSION
```

Examples

```

CMCTL> SHOW VERSION
CMAN for Linux: Version 11.2.0.0.0
The command completed successfully

```

SHUTDOWN

Purpose

To shut down specific gateway processes or the entire Oracle Connection Manager instance.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHUTDOWN [gateways gateway] [normal | abort] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHUTDOWN [gateways gateway] {normal | abort}
```

Arguments

gateways: To shut down a specific gateway.

normal: To reject new connections and terminate after existing connections close. This is the default.

abort: To shut down Oracle Connection Manager immediately, and close all open connections.

To specify more than one gateway, separate gateways using a space.

Usage Notes

Issuing SHUTDOWN without an argument shuts down all gateways.

Example

```
CMCTL> SHUTDOWN GATEWAYS 0
The command completed successfully
```

STARTUP

Purpose

To start Oracle Connection Manager.

Prerequisites

Another Oracle Connection Manager configured with the same protocol address must not be running.

Syntax

From the operating system:

```
cmctl STARTUP [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> STARTUP
```

Usage Notes

Before issuing this command, you must use the [ADMINISTER](#) command to select an instance to start.

Issuing this command starts all instance components, which are the listener, CMADMIN, and the gateway processes. The command fails if any one of these components is already running.

Example

```
CMCTL> STARTUP
Starting CMAN instance: CMAN_user.us.example.com, please wait...
TNS-04090: *** CMCTL WARNING: No password set in the CMAN instance ***
CMAN for Linux: Version 11.2.0.0.0
Status of the Instance
-----
Instance name          CMAN_user.us.example.com
Version                CMAN for Linux: Version 11.2.0.0.0
Start date             20-JAN-2008 19:04:25
Uptime                 0 days 0 hr. 0 min. 3 sec
Num of gateways started 2
Average Load level     0
Log Level              SUPPORT
Trace Level            OFF
Instance Config file   /disk1/user_cman_test/oracle/network/admin/cman.ora
Instance Log directory /disk1/user_cman_test/oracle/network/log/
Instance Trace directory /disk1/user_cman_test/oracle/network/trace/
```

The command completed successfully

SUSPEND GATEWAY

Purpose

To specify which gateway processes that will no longer accept new client connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SUSPEND GATEWAY [gateway_process_id] [-c instance_name]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SUSPEND GATEWAY [gateway_process_id]
```

Arguments

gateway_process_id: The gateway process that will no longer accept new connections. Specify multiple gateway processes by putting a space between entries.

Issuing `SUSPEND GATEWAY` without an argument suspends all gateway processes.

Usage Notes

Use the [RESUME GATEWAYS](#) command to enable gateway processes to accept new connections.

Example

```
CMCTL> SUSPEND GATEWAY 1  
The command completed successfully
```


Part II

Configuration Parameters

Part II describes how to configure listening protocol addresses and Oracle Net Services configuration parameters.

This part contains the following chapters:

- [Chapter 3, "Syntax Rules for Configuration Files"](#)
- [Chapter 4, "Protocol Address Configuration"](#)
- [Chapter 5, "Parameters for the sqlnet.ora File"](#)
- [Chapter 6, "Local Naming Parameters \(tnsnames.ora\)"](#)
- [Chapter 7, "Oracle Net Listener Parameters \(listener.ora\)"](#)
- [Chapter 8, "Oracle Connection Manager Parameters \(cman.ora\)"](#)
- [Chapter 9, "Directory Usage Parameters \(ldap.ora\)"](#)

Syntax Rules for Configuration Files

This chapter describes the syntax rules for Oracle Net Services configuration files.

This chapter contains the following topics:

- [Configuration File Syntax Overview](#)
- [Syntax Rules for Configuration Files](#)
- [Network Character Set for Keywords](#)
- [Character Set for Listener and Net Service Names](#)

Configuration File Syntax Overview

The Oracle Net Services configuration files consist of parameters which include keyword-value pairs. Keyword-value pairs are surrounded by parentheses:

```
parameter=(keyword=value)
```

Some keywords have other keyword-value pairs as their values:

```
(keyword=  
  (keyword=value)  
  (keyword=value))
```

For example, the address portion of a local naming configuration file (`tnsnames.ora`) might include the following lines:

```
(ADDRESS=  
  (PROTOCOL=tcp)  
  (HOST=sales-server)  
  (PORT=1521))
```

Set up configuration files so that indentation reflects what keyword is the parent or owner of other keyword-value pairs. If you do not choose to indent your files in this way, you must still indent a wrapped line by at least one space, or it will be misread as a new parameter. The following syntax is acceptable:

```
(ADDRESS=(PROTOCOL=tcp)  
  (HOST=sales-server) (PORT=1521))
```

The following syntax is not acceptable:

```
(ADDRESS=(PROTOCOL=tcp)  
(HOST=sales-server) (PORT=1521))
```

Syntax Rules for Configuration Files

The following rules apply to the syntax of configuration files:

- Any keyword in a configuration file that begins a parameter that includes one or more keyword-value pairs must be in the far left column of a line. If it is indented by one or more spaces, then it is interpreted as a continuation of the previous line.
- All characters must belong to the network character set.
- Keywords are not case sensitive. However, values may be case sensitive, depending on the operating system and protocol.
- Spaces around the equal sign (=) are optional in keyword-value pairs.
- There is a hierarchy of keywords such that some keywords are always followed by others. At any level of the hierarchy, keywords can be listed in any order. For example, the following entries are equally valid:

```
(ADDRESS=
  (PROTOCOL=TCP)
  (HOST=sales-server)
  (PORT=1521))
(ADDRESS=
  (PROTOCOL=tcp)
  (PORT=1521)
  (HOST=sales-server))
```

- Keywords cannot contain spaces.
- Values must not contain spaces unless enclosed within double quotation marks (") or single quotation marks (').
- If the keyword-value pair consists of a single word or a concatenation of words on either side of the equal sign, then no parentheses are needed.
- The maximum length of a connect descriptor is 4 KB
- Comments can be included using the number sign (#) at the beginning of a line. Anything following the number sign to the end of the line is considered a comment.

Network Character Set for Keywords

The network character set for keyword values consists of the following characters. Connect descriptors must be made up of single-byte characters.

```
A-Z, a-z
0-9
( ) < > / \
, . : ; ' " = - _
$ + * # & ! % ? @
```

Within this character set, the following symbols are reserved:

```
( ) = \ " ' #
```

Reserved symbols are used as delimiters, not as part of a keyword or a value unless the keyword or value has quotation marks. Either single or double quotation marks can be used to enclose a value containing reserved symbols. To include a quotation marks within a value that is surrounded by quotation marks, use different quotation marks. The backslash (\) is used as an escape character.

The following characters may be used within a connect descriptor, but not in a keyword or value:

- Space
- Tab
- Carriage return
- Newline

Character Set for Listener and Net Service Names

The listener name and net service name are limited to the following character set:

[a...z] [A...Z] [0...9] _

The first character must be an alphanumeric character. In general, up to 64 characters is acceptable. A database service name must match the global database name defined by the database administrator, which consists of a database name, and the database domain. Net service names and global database names are not case sensitive.

Protocol Address Configuration

A network object is identified by a [protocol address](#). When a connection is made, the client and the receiver of the request ([listener](#) or [Oracle Connection Manager](#)) are configured with identical protocol addresses.

The client uses this address to send the connection request to a particular network object location, and the recipient "listens" for requests on this address, and grants a connection based on its address information matching the client information.

This chapter contains the following topics:

- [Protocol Addresses](#)
- [Protocol Parameters](#)
- [Recommended Port Numbers](#)
- [Port Number Limitations](#)

Protocol Addresses

The protocol address is comprised of [ADDRESS](#) and [ADDRESS_LIST](#) elements.

ADDRESS

Purpose

To define a protocol address.

Usage Notes

Put this parameter under an [ADDRESS_LIST](#) or [DESCRIPTION](#) parameter. A [DESCRIPTION](#) is used in a `tnsnames.ora` or a `listener.ora` file.

Example

```
(ADDRESS=  
  (PROTOCOL=tcp)  
  (HOST=sales-server)  
  (PORT=1521))
```

See Also: "[Protocol Parameters](#)" on page 4-2 for each protocol's required parameters

ADDRESS_LIST

Purpose

To define a list of protocol addresses that share common characteristics.

Example

```
(ADDRESS_LIST=
 (LOAD_BALANCE=on)
 (ADDRESS=
 (PROTOCOL=tcp)
 (HOST=sales-server)
 (PORT=1521))
 (ADDRESS=
 (PROTOCOL=tcp)
 (HOST=hr-server)
 (PORT=1521)))
```

Protocol Parameters

The listener and Oracle Connection Manager are identified by protocol addresses. [Table 4–1, "Protocol-Specific Parameters"](#) lists the parameters used by the Oracle protocol support.

Table 4–1 Protocol-Specific Parameters

Protocol	Parameter	Description
IPC	PROTOCOL	Specify <code>ipc</code> as the value.
	KEY	Specify a unique name for the service. Oracle recommends using the service name or the Oracle System Identifier (SID) of the service. Example: <code>(PROTOCOL=ipc) (KEY=sales)</code>
Named Pipes	PROTOCOL	Specify <code>nmp</code> as the value.
	SERVER	Specify the name of the Oracle server.
	PIPE	Specify the pipe name used to connect to the database server. This is the same <code>PIPE</code> keyword specified on server with Named Pipes. This name can be any name. Example: <code>(PROTOCOL=nmp) (SERVER=sales) (PIPE=dbpipe0)</code>
SDP	PROTOCOL	Specify <code>sdp</code> as the value.
	HOST	Specify the host name or IP address of the computer.
	PORT	Specify the listening port number. Example: <code>(PROTOCOL=sdp) (HOST=sales-server) (PORT=1521)</code> <code>(PROTOCOL=sdp) (HOST=192.168.2.204) (PORT=1521)</code>
TCP/IP	PROTOCOL	Specify <code>tcp</code> as the value.
	HOST	Specify the host name or IP address of the computer.

Table 4–1 (Cont.) Protocol-Specific Parameters

Protocol	Parameter	Description
	PORT	Specify the listening port number. Example: (PROTOCOL=tcp) (HOST=sales-server) (PORT=1521) (PROTOCOL=tcp) (HOST=192.168.2.204) (PORT=1521)
TCP/IP with SSL	PROTOCOL	Specify <code>tcps</code> as the value.
	HOST	Specify the host name or IP address of the computer.
	PORT	Specify the listening port number. Example: (PROTOCOL=tcps) (HOST=sales-server) (PORT=2484) (PROTOCOL=tcps) (HOST=192.168.2.204) (PORT=2484)

Recommended Port Numbers

Table 4–2, "Recommended Port Numbers" lists the recommends the port numbers.

Table 4–2 Recommended Port Numbers

Port	Description
1521	Default listening port for client connections to the listener. This port number may change to the officially registered port number of 2483 for TCP/IP and 2484 for TCP/IP with SSL.
1521	Default and officially registered listening port for client connections to Oracle Connection Manager.
1830	Default and officially registered listening port for administrative commands to Oracle Connection Manager.

Port Number Limitations

Oracle allows port numbers from 1 to 65535. However, many operating systems reserve port numbers less than 1024. For example, on certain operating systems, only privileged processes can listen for TCP connections on ports less than 1024.

If you need to configure listener to listen on a port number less than 1024, then do the following:

Note: Your operating system may require a different procedure.

1. Use Oracle Net Configuration Assistant or Oracle Net Manager to configure the listener with protocol addresses and other configuration parameters.

See Also: *Oracle Database Net Services Administrator's Guide*

2. Log in as the `root` user on the machine that has the listener.
3. Set file ownership and access permissions for the listener executable (`tnslsnr`) and the dependent shared libraries so that these files can be modified only by the `root` user.

4. Ensure that the permissions of the individual directories found in the path names to these files, starting with the root directory have the same ownership and access permissions.
5. Start the listener as the `root` user.
6. Enter the following command at the system prompt:

```
tnslsnr listener_name -user user -group group
```

In the preceding command, the following options are used:

Table 4–3 *tnslsnr Utility Options*

Options	Description
<i>listener_name</i>	Specify the name of the listener. If omitted, then the default name <code>LISTENER</code> is used.
<i>user</i>	Specify the user whose privileges the listener will use when super user (<code>root</code>) privileges are not needed. After performing the privileged operations, the listener will give up <code>root</code> privileges irreversibly.
<i>group</i>	Specify the group whose privileges the listener will use when super user (<code>root</code>) group privileges are not needed. After performing the privileged operations, the listener will give up <code>root</code> group privileges irreversibly.

During this step, the listener switches to the specified user and group. All operations are done with the specified user and group privileges, except the system calls necessary to listen on configured endpoints. The listener reverts to the `root` user to listen on reserved addresses, such as TCP ports less than 1024.

After the listener starts listening on all of its endpoints configured in `listener.ora`, it switches to the specified user and group irreversibly. Therefore, the listener will give up the `root` privilege that it initially had. The `-user` and `-group` command line arguments only accept user and group identifiers specified in numeric form.

For example, to run a listener with `root` privileges called `mylsnr` and have it use privileges of a user identified as 37555 with a group identifier of 16, enter the following at the operating system command prompt:

```
tnslsnr mylsnr -user 37555 -group 16
```

In the preceding example, 37555 could be the identifier for the `oracle` user, and 16 could be the identifier for the `dba` group.

7. After the listener has been started, you can administer it with the Listener Control utility.

Important Notes:

- Oracle recommends that the user which the listener process runs be the `oracle` user, or a user that the listener process normally runs on the operating system.
- Do not leave the listener process running as the `root` user because doing so is a security vulnerability.

Parameters for the sqlnet.ora File

This chapter provides complete listing of the `sqlnet.ora` file configuration parameters.

This chapter includes the following topics:

- [Overview of Profile Configuration File](#)
- [sqlnet.ora Profile Parameters](#)
- [ADR Diagnostic Parameters in sqlnet.ora](#)
- [Non-ADR Diagnostic Parameters in sqlnet.ora](#)

Overview of Profile Configuration File

The `sqlnet.ora` file is the profile configuration file. It resides on the client machines and the database server. Profiles are stored and implemented using this file. The database server can be configured with access control parameters in the `sqlnet.ora` file. These parameters specify whether clients are allowed or denied access based on the protocol.

The `sqlnet.ora` file enables you to do the following:

- Specify the client domain to append to unqualified names
- Prioritize **naming methods**
- Enable logging and tracing features
- Route connections through specific processes
- Configure parameters for **external naming**
- Configure **Oracle Advanced Security**
- Use protocol-specific parameters to restrict access to the database

By default, the `sqlnet.ora` file is located in the `ORACLE_HOME/network/admin` directory. The `sqlnet.ora` file can also be stored in the directory specified by the `TNS_ADMIN` environment variable.

sqlnet.ora Profile Parameters

This section lists and describes the following `sqlnet.ora` file parameters:

- `BEQUEATH_DETACH`
- `DEFAULT_SDU_SIZE`

- DISABLE_OOB
- NAMES.DEFAULT_DOMAIN
- NAMES.DIRECTORY_PATH
- NAMES.LDAP_AUTHENTICATE_BIND
- NAMES.LDAP_CONN_TIMEOUT
- NAMES.LDAP_PERSISTENT_SESSION
- RECV_BUF_SIZE
- SDP.PF_INET_SDP
- SEC_USER_AUDIT_ACTION_BANNER
- SEC_USER_UNAUTHORIZED_ACCESS_BANNER
- SEND_BUF_SIZE
- SQLNET.ALLOWED_LOGON_VERSION
- SQLNET.AUTHENTICATION_KERBEROS5_SERVICE
- SQLNET.AUTHENTICATION_SERVICES
- SQLNET.CLIENT_REGISTRATION
- SQLNET.CRYPTO_CHECKSUM_CLIENT
- SQLNET.CRYPTO_CHECKSUM_SERVER
- SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT
- SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER
- SQLNET.ENCRYPTION_CLIENT
- SQLNET.ENCRYPTION_SERVER
- SQLNET.ENCRYPTION_TYPES_CLIENT
- SQLNET.ENCRYPTION_TYPES_SERVER
- SQLNET.EXPIRE_TIME
- SQLNET.INBOUND_CONNECT_TIMEOUT
- SQLNET.KERBEROS5_CC_NAME
- SQLNET.KERBEROS5_CLOCKSKEW
- SQLNET.KERBEROS5_CONF
- SQLNET.KERBEROS5_KEYTAB
- SQLNET.KERBEROS5_REALMS
- SQLNET.OUTBOUND_CONNECT_TIMEOUT
- SQLNET.RADIUS_ALTERNATE
- SQLNET.RADIUS_ALTERNATE_PORT
- SQLNET.RADIUS_ALTERNATE_RETRIES
- SQLNET.RADIUS_AUTHENTICATION
- SQLNET.RADIUS_AUTHENTICATION_INTERFACE
- SQLNET.RADIUS_AUTHENTICATION_PORT

- SQLNET.RADIUS_AUTHENTICATION_RETRIES
- SQLNET.RADIUS_AUTHENTICATION_TIMEOUT
- SQLNET.RADIUS_CHALLENGE_RESPONSE
- SQLNET.RADIUS_SECRET
- SQLNET.RADIUS_SEND_ACCOUNTING
- SQLNET.RECV_TIMEOUT
- SQLNET.SEND_TIMEOUT
- SSL_CERT_REVOCACTION
- SSL_CERT_FILE
- SSL_CERT_PATH
- SSL_CIPHER_SUITES
- SSL_CLIENT_AUTHENTICATION
- SSL_SERVER_DN_MATCH
- SSL_VERSION
- TCP.CONNECT_TIMEOUT
- TCP.EXCLUDED_NODES
- TCP.INVITED_NODES
- TCP.NODELAY
- TCP.VALIDNODE_CHECKING
- TNSPING.TRACE_DIRECTORY
- TNSPING.TRACE_LEVEL
- USE_CMAN
- USE_DEDICATED_SERVER
- WALLET_LOCATION
- WALLET_OVERRIDE

BEQUEATH_DETACH

Purpose

To turn signal handling on or off for Linux and UNIX systems.

Default

NO

Values

- yes to turn signal handling off
- no to leave signal handling on

Example

BEQUEATH_DETACH=yes

DEFAULT_SDU_SIZE

Purpose

To specify the **session data unit (SDU)** size, in bytes to connections.

Usage

Oracle recommends setting this parameter in both the client-side and server-side `sqlnet.ora` file to ensure the same SDU size is used throughout a connection. When the configured values of client and database server do not match for a session, the lower of the two values is used.

You can override this parameter for a particular client connection by specifying the **SDU** parameter in the connect descriptor for a client.

See Also: *Oracle Database Net Services Administrator's Guide* for complete SDU usage and configuration information

Default

8192 bytes (8 KB)

Values

512 to 65535 bytes

Example

```
DEFAULT_SDU_SIZE=4096
```

DISABLE_OOB

Purpose

To enable or disable Oracle Net to send or receive out-of-band break messages using urgent data provided by the underlying protocol.

If turned `off`, then the parameter enables Oracle Net to send and receive break messages. If turned `on`, then the parameter disables the ability to send and receive break messages. Once enabled, this feature applies to all protocols used by this client.

Default

OFF

Example

```
DISABLE_OOB=on
```

See Also: Operating system-specific documentation to determine if the protocols you are using support urgent data requests. TCP/IP is an example of a protocol that supports this feature.

NAMES.DEFAULT_DOMAIN

Purpose

To set the domain from which the client most often looks up names resolution requests. When this parameter is set, the default domain name is automatically appended to any unqualified net service name or service name.

For example, if the default domain is set to `us.example.com`, then the connect string `CONNECT scott@sales` gets searched as `sales.us.example.com`. If the connect string includes the domain extension, such as `CONNECT scott@sales.us.example.com`, then the domain is not appended to the string.

Default

None

Example

```
NAMES.DEFAULT_DOMAIN=example.com
```

NAMES.DIRECTORY_PATH**Purpose**

To specify the order of the naming methods used for client name resolution lookups.

Default

```
NAMES.DIRECTORY_PATH=(tnsnames, ldap, ezconnect)
```

Values

Table 5–1 NAMES.DIRECTORY_PATH Values

Naming Method Value	Description
<code>tnsnames</code> (local naming method)	Set to resolve a net service name through the <code>tnsnames.ora</code> file on the client.
<code>ldap</code> (directory naming method)	Set to resolve a database service name, net service name, or net service alias through a directory server .
<code>ezconnect</code> or <code>hostname</code> (Easy Connect naming or host naming method)	Select to enable clients to use a TCP/IP connect identifier, consisting of a host name and optional port and service name.
<code>nis</code> (Network Information Service (NIS) external naming method)	Set to resolve service information through an existing NIS.

Example

```
NAMES.DIRECTORY_PATH=(tnsnames)
```

NAMES.LDAP_AUTHENTICATE_BIND**Purpose**

To specify whether the LDAP naming adapter should attempt to authenticate using a specified wallet when it connects to the LDAP directory to resolve the name in the connect string.

Usage

The parameter value is Boolean.

If the parameter is set to `TRUE`, then the LDAP connection is authenticated using a wallet whose location must be specified in the `WALLET_LOCATION` parameter.

If the parameter is set to `FALSE`, then the LDAP connection is established using an anonymous bind.

Default

`FALSE`

Example

```
NAMES.LDAP_AUTHENTICATE_BIND=TRUE
```

NAMES.LDAP_CONN_TIMEOUT

Purpose

To specify number of seconds for a non-blocking connect timeout to the LDAP server.

Usage

The parameter value `-1` is for infinite timeout.

Default

15 seconds

Values

Values are in seconds. The range is `-1` to the number of seconds acceptable for your environment. There is no upper limit.

Example

```
names.ldap_conn_timeout = -1
```

NAMES.LDAP_PERSISTENT_SESSION

Purpose

To specify whether the LDAP naming adapter should leave the session with the LDAP server open after name lookup is complete.

Usage

The parameter value is Boolean.

If the parameter is set to `TRUE`, then the connection to the LDAP server is left open after the name lookup is complete; the connection will effectively stay open for the duration of the process. If the connection is lost, then it will be re-established as needed.

If the parameter is set to `FALSE`, then the LDAP connection is terminated as soon as the name lookup completes. Every subsequent lookup opens the connection, performs the lookup, and closes the connection. This option prevents the LDAP server from having a large number of clients connected to it at any one time.

Default

`FALSE`

Example

```
NAMES.LDAP_PERSISTENT_SESSION=TRUE
```

RECV_BUF_SIZE**Purpose**

To specify the buffer space limit for receive operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note: Additional protocols might support this parameter on certain operating systems. Refer to the operating system-specific documentation for additional information about additional protocols that support this parameter.

See Also: *Oracle Net Services Administrator's Guide* for additional information about configuring this parameter

Default

The default value for this parameter is operating system-specific.

Usage

You can override this parameter for a particular client connection by specifying the [RECV_BUF_SIZE](#) parameter in the connect descriptor for a client.

Example

```
RECV_BUF_SIZE=11784
```

SDP.PF_INET_SDP**Purpose**

To specify the protocol family or address family constant for the SDP protocol on your system.

Default

27

Values

Any positive integer

Example

```
SDP.PF_INET_SDP=30
```

SEC_USER_AUDIT_ACTION_BANNER**Purpose**

To specify a text file containing the banner contents that warn the user about possible user action auditing. The complete path of the text file must be specified in the `sqlnet.ora` file on the server. Oracle Call Interface (OCI) applications can make use

of OCI features to retrieve this banner and display it to the user. The text file has a maximum limit of 512 bytes.

Default

None

Values

Name of the file for which the database owner has read permissions.

Example

```
SEC_USER_AUDIT_ACTION_BANNER=/opt/oracle/admin/data/auditwarning.txt
```

SEC_USER_UNAUTHORIZED_ACCESS_BANNER

Purpose

To specify a text file containing the banner contents that warn the user about unauthorized access to the database. The complete path of the text file must be specified in the `sqlnet.ora` file on the server. OCI applications can make use of OCI features to retrieve this banner and display it to the user. The text file has a maximum limit of 512 bytes.

Default

None

Values

Name of the file for which the database owner has read permissions.

Example

```
SEC_USER_UNAUTHORIZED_ACCESS_BANNER=/opt/oracle/admin/data/unauthwarning.txt
```

SEND_BUF_SIZE

Purpose

To specify the buffer space limit for send operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note: Additional protocols might support this parameter on certain operating systems. Refer to the operating system-specific documentation for additional information about additional protocols that support this parameter.

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring this parameter

Default

The default value for this parameter is operating system-specific.

Usage

You can override this parameter for a particular client connection by specifying the `SEND_BUF_SIZE` parameter in the connect descriptor for a client.

Example

```
SEND_BUF_SIZE=11784
```

SQLNET.ALLOWED_LOGON_VERSION**Purpose**

To set the minimum authentication protocol allowed when connecting to Oracle Database instances. The term `VERSION` in the parameter name refers to the version of the authentication protocol, not the Oracle Database release.

If the client release does not meet or exceed the value defined by this parameter, then authentication fails with an `ORA-28040: No matching authentication protocol error` or an `ORA-03134: Connections to this server version are no longer supported error`.

Usage Notes

A setting of 8 permits most password versions, and allows any combination of the `DBA_USERS.PASSWORD_VERSIONS` values 10G, and 11G.

A greater value means the server is less compatible in terms of the protocol that clients must understand in order to authenticate. The server is also more restrictive in terms of the password version that must exist to authenticate any specific account. The ability for a client to authenticate depends on the `DBA_USERS.PASSWORD_VERSIONS` value on the server for that account.

Note the following implications of setting the value to 12:

- To take advantage of the password protections introduced in Oracle Database 11g, users must change their passwords. The new passwords are case sensitive. When an account password is changed, the earlier 10G case-insensitive password version is automatically removed.
- Releases of OCI clients before Oracle Database 10g and all versions of JDBC thin clients cannot authenticate to the Oracle database using password-based authentication.
- If the client uses Oracle9i Database, then the client will receive an `ORA-03134` error message. To allow the connection, remove the `SQLNET.ALLOWED_LOGON_VERSION` setting to return to the default. Ensure the `DBA_USERS.PASSWORD_VERSIONS` value for the account contains the value 10G. It may be necessary to reset the password for that account.

The client must support certain abilities of an authentication protocol before the server will authenticate. If the client does not support a specified authentication ability, then the server rejects the connection with an `ORA-28040: No matching authentication protocol error` message.

The following is the list of all client abilities. Some clients do not have all abilities. Clients that are more recent have all the capabilities of the older clients, but older clients tend to have less abilities than more recent clients.

- `O5L_NP`: The ability to perform the Oracle Database 10g authentication protocol using the 11G password version, and generating a session key encrypted for critical patch update CPUOct2012.

- 05L: The ability to perform the Oracle Database 10g authentication protocol using the 10G password version.
- 04L: The ability to perform the Oracle9i database authentication protocol using the 10G password version.
- 03L: The ability to perform the Oracle8i database authentication protocol using the 10G password version.

A higher ability value is more recent and secure than a lower ability value. Clients that are more recent have all the capabilities of the older clients.

The following table describes the allowed values, password versions, and descriptions:

Value of the ALLOWED_LOGON_VERSION Parameter	Generated Password Version	Ability Required of the Client	Meaning for Clients
12 ¹	11G	05L_NP	Only clients which have applied critical patch update CPUOct2012 or later, or release 11.2.0.3 clients with an equivalent update can connect to the server.
11	10G, 11G	05L	Clients using Oracle Database 10g and later can connect to the server. Clients using releases earlier than Oracle Database release 11.2.0.3 that have not applied critical patch update CPUOct2012 or later patches must use the 10G password version.
10	10G, 11G	05L	Clients using Oracle Database 10g and later can connect to the server. Clients using releases earlier than Oracle Database release 11.2.0.3 that have not applied critical patch update CPUOct2012 or later patches must use the 10G password version.
9	10G, 11G	04L	Oracle9i Database or later clients can connect to the server.
8	10G, 11G	03L	Oracle8i Database and later clients can connect to the server.

¹ This is considered "Exclusive Mode" because it excludes the use of the 10G password version.

Allowed Values

- 12 for the critical patch updates CPUOct2012 and later Oracle Database 11g authentication protocols (recommended)
- 11 for Oracle Database 11g authentication protocols
- 10 for Oracle Database 10g authentication protocols
- 9 for Oracle9i Database authentication protocols
- 8 for Oracle8i Database authentication protocols (default)

Default

8

Example

If both Oracle Database 11g and Oracle Database 10g are present, then set the parameter as follows:

```
SQLNET.ALLOWED_LOGON_VERSION=10
```

SQLNET.AUTHENTICATION_KERBEROS5_SERVICE**Purpose**

To define the name of the service used to obtain a Kerberos service ticket.

Default

None

Example

```
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE=oracle
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.AUTHENTICATION_SERVICES**Purpose**

To enable one or more authentication services. If authentication has been installed, then it is recommended that this parameter be set to either `none` or to one of the authentication methods.

Default

None

Note: When installing the database with Database Configuration Assistant (DBCA), this parameter may be set to `nts` in the `sqlnet.ora` file.

Values

- Authentication Methods Available with Oracle Net Services:
 - `none` for no authentication methods, including Microsoft Windows native operating system authentication. When `SQLNET.AUTHENTICATION_SERVICES` is set to `none`, a valid user name and password can be used to access the database.
 - `all` for all authentication methods.
 - `nts` for Microsoft Windows native operating system authentication.
- Authentication Methods Available with Oracle Advanced Security:
 - `kerberos5` for Kerberos authentication.
 - `radius` for RADIUS authentication.
 - `tcps` for SSL authentication.

Example

```
SQLNET.AUTHENTICATION_SERVICES=(kerberos5)
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.CLIENT_REGISTRATION

Purpose

To set a unique identifier for the client computer. This identifier is passed to the listener with any connection request and is included in the Audit Trail. The identifier can be any alphanumeric string up to 128 characters long.

Default

None

Example

```
SQLNET.CLIENT_REGISTRATION=1432
```

SQLNET.CRYPTO_CHECKSUM_CLIENT

Purpose

To specify the checksum behavior for the client.

See Also: *Oracle Database Advanced Security Administrator's Guide*

Default

accepted

Values

- `accepted` to enable the security service if required or requested by the other side.
- `rejected` to disable the security service, even if the required by the other side.
- `requested` to enable the security service if the other side allows it.
- `required` to enable the security service and disallow the connection if the other side is not enabled for the security service.

Example

```
SQLNET.CRYPTO_CHECKSUM_CLIENT=accepted
```

SQLNET.CRYPTO_CHECKSUM_SERVER

Purpose

To specify the checksum behavior for the database server.

Default

accepted

Values

- `accepted` to enable the security service if required or requested by the other side.
- `rejected` to disable the security service, even if the required by the other side.
- `requested` to enable the security service if the other side allows it.

- required to enable the security service and disallow the connection if the other side is not enabled for the security service.

Example

```
SQLNET.CRYPTO_CHECKSUM_SERVER=accepted
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT**Purpose**

To specify a list of crypto-checksum algorithms for the client to use.

Default

All available algorithms

Values

- md5 for the RSA Data Security MD5 algorithm.
- sha1 for the Secure Hash algorithm.

Example

```
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT=(MD5)
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER**Purpose**

To specify a list of crypto-checksum algorithms for the database server to use.

Default

All available algorithms

Values

- md5 for the RSA Data Security's MD5 algorithm
- sha1 for the Secure Hash algorithm

Example

```
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER=(md5)
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.ENCRYPTION_CLIENT**Purpose**

To turn encryption on for the client.

Default

accepted

Values

- `accepted` to enable the security service if required or requested by the other side.
- `rejected` to disable the security service, even if the required by the other side.
- `requested` to enable the security service if the other side allows it.
- `required` to enable the security service and disallow the connection if the other side is not enabled for the security service.

Example

```
SQLNET.ENCRYPTION_CLIENT=accepted
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.ENCRYPTION_SERVER**Purpose**

To turn encryption on for the database server.

Default

`accepted`

Values

- `accepted` to enable the security service if required or requested by the other side.
- `rejected` to disable the security service, even if the required by the other side.
- `requested` to enable the security service if the other side allows it.
- `required` to enable the security service and disallow the connection if the other side is not enabled for the security service.

Example

```
SQLNET.ENCRYPTION_SERVER=accepted
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.ENCRYPTION_TYPES_CLIENT**Purpose**

To specify a list of encryption algorithms for the client to use.

Default

All available algorithms.

Values

One or more of the following:

- `3des112` for triple DES with a two-key (112-bit) option
- `3des168` for triple DES with a three-key (168-bit) option
- `des` for standard 56-bit key size
- `des40` for 40-bit key size

- rc4_40 for 40-bit key size
- rc4_56 for 56-bit key size
- rc4_128 for 128-bit key size
- rc4_256 for 256-bit key size

Example

```
SQLNET.ENCRYPTION_TYPES_CLIENT=(rc4_56)
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.ENCRYPTION_TYPES_SERVER**Purpose**

To specify a list of encryption algorithms for the database server to use.

Default

All available algorithms.

Values

One or more of the following:

- 3des112 for triple DES with a two-key (112-bit) option
- 3des168 for triple DES with a three-key (168-bit) option
- des for standard 56-bit key size
- des40 for 40-bit key size
- rc4_40 for 40-bit key size
- rc4_56 for 56-bit key size
- rc4_128 for 128-bit key size
- rc4_256 for 256-bit key size

Example

```
SQLNET.ENCRYPTION_TYPES_SERVER=(rc4_56, des, ...)
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.EXPIRE_TIME**Purpose**

To specify a time interval, in minutes, to send a check to verify that client/server connections are active. The following usage notes apply to this parameter:

- Setting a value greater than 0 ensures that connections are not left open indefinitely, due to an abnormal client termination.
- If the probe finds a terminated connection, or a connection that is no longer in use, then it returns an error, causing the server process to exit.
- This parameter is primarily intended for the database server, which typically handles multiple connections at any one time.

- Limitations on using this terminated connection detection feature are:
 - It is not allowed on bequeathed connections.
 - Though very small, a probe packet generates additional traffic that may downgrade network performance.
 - Depending on which operating system is in use, the server may need to perform additional processing to distinguish the connection probing event from other events that occur. This can also result in degraded network performance.

Default

0

Minimum Value

0

Recommended Value

10

Example

```
SQLNET.EXPIRE_TIME=10
```

SQLNET.INBOUND_CONNECT_TIMEOUT

Purpose

To specify the time, in seconds, for a client to connect with the database server and provide the necessary authentication information.

If the client fails to establish a connection and complete authentication in the time specified, then the database server terminates the connection. In addition, the database server logs the IP address of the client and an `ORA-12170: TNS:Connect timeout occurred` error message to the `sqlnet.log` file. The client receives either an `ORA-12547: TNS:lost contact` or an `ORA-12637: Packet receive failed` error message.

The default value of this parameter is appropriate for typical usage scenarios. However, if you need to explicitly set a different value, then Oracle recommends setting this parameter in combination with the [INBOUND_CONNECT_TIMEOUT_listener_name](#) parameter in the `listener.ora` file. When specifying the values for these parameters, note the following recommendations:

- Set both parameters to an initial low value.
- Set the value of the `INBOUND_CONNECT_TIMEOUT_listener_name` parameter to a lower value than the `SQLNET.INBOUND_CONNECT_TIMEOUT` parameter.

For example, you can set `INBOUND_CONNECT_TIMEOUT_listener_name` to 2 seconds and `SQLNET.INBOUND_CONNECT_TIMEOUT` parameter to 3 seconds. If clients are unable to complete connections within the specified time due to system or network delays that are normal for the particular environment, then increment the time as needed.

Default

60 seconds

Example

```
SQLNET.INBOUND_CONNECT_TIMEOUT=3
```

See Also:

- ["Control Parameters"](#) on page 7-7 for additional information about `INBOUND_CONNECT_TIMEOUT_listener_name`
- *Oracle Net Services Administrator's Guide* for additional information about configuring these parameters

SQLNET.KERBEROS5_CC_NAME**Purpose**

To specify the complete path name to the Kerberos credentials cache file.

Default

`/usr/tmp/krb5cache` on Linux and UNIX operating systems, and
`c:\tmp\krb5cache` on Microsoft Windows operating systems

Example

```
SQLNET.KERBEROS5_CC_NAME=/usr/tmp/krb5cache
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.KERBEROS5_CLOCKSKEW**Purpose**

To specify how many seconds can pass before a Kerberos credential is considered out of date.

Default

300

Example

```
SQLNET.KERBEROS5_CLOCKSKEW=1200
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.KERBEROS5_CONF**Purpose**

To specify the complete path name to the Kerberos configuration file, which contains the realm for the default Key Distribution Center (KDC) and maps realms to KDC hosts. The KDC maintains a list of user principals and is contacted through the `kinit` program for the user's initial ticket.

Default

`/krb5/krb.conf` on Linux and UNIX operating systems and `c:\krb5\krb.conf` on Microsoft Windows operating systems

Example

```
SQLNET.KERBEROS5_CONF=/krb5/krb.conf
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.KERBEROS5_KEYTAB**Purpose**

To specify the complete path name to the Kerberos principal/secret key mapping file, which is used to extract keys and decrypt incoming authentication information.

Default

/etc/v5srvtab on Linux and UNIX operating systems and c:\krb5\v5srvtab on Microsoft Windows operating systems

Example

```
SQLNET.KERBEROS5_KEYTAB=/etc/v5srvtab
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.KERBEROS5_REALMS**Purpose**

To specify the complete path name to the Kerberos realm translation file, which provides a mapping from a host name or domain name to a realm.

Default

/krb5/krb.realms on Linux and UNIX operating systems and c:\krb5\krb.realms on Microsoft Windows operating systems

Example

```
SQLNET.KERBEROS5_REALMS=/krb5/krb.realms
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.OUTBOUND_CONNECT_TIMEOUT**Purpose**

To specify the time, in seconds, for a client to establish an Oracle Net connection to the database instance.

If an Oracle Net connection is not established in the time specified, then the connect attempt is terminated. The client receives an `ORA-12170: TNS:Connect timeout occurred error`.

The outbound connect timeout interval is a superset of the TCP connect timeout interval, which specifies a limit on the time taken to establish a TCP connection. Additionally, the outbound connect timeout interval includes the time taken to be connected to an Oracle instance providing the requested service.

Without this parameter, a client connection request to the database server may block for the default TCP connect timeout duration (60 seconds) when the database server host system is unreachable.

The outbound connect timeout interval is only applicable for TCP, TCP with SSL, and IPC transport connections.

Default

None

Usage Notes

This parameter is overridden by the [CONNECT_TIMEOUT](#) parameter in the address description.

Example

```
SQLNET.OUTBOUND_CONNECT_TIMEOUT=10
```

SQLNET.RADIUS_ALTERNATE

Purpose

To specify an alternate RADIUS server to use in case the primary server is unavailable. The value can be either the IP address or host name of the server.

Default

None

Example

```
SQLNET.RADIUS_ALTERNATE=radius2
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_ALTERNATE_PORT

Purpose

To specify the listening port of the alternate RADIUS server.

Default

1645

Example

```
SQLNET.RADIUS_ALTERNATE_PORT=1667
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_ALTERNATE_RETRIES

Purpose

To specify the number of times the database server should resend messages to the alternate RADIUS server.

Default

3

Example

```
SQLNET.RADIUS_ALTERNATE_RETRIES=4
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_AUTHENTICATION**Purpose**

To specify the location of the primary RADIUS server, either by its host name or IP address.

Default

Local host

Example

```
SQLNET.RADIUS_AUTHENTICATION=officeacct
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_AUTHENTICATION_INTERFACE**Purpose**

To specify the class containing the user interface used to interact with the user.

Default

DefaultRadiusInterface

Example

```
SQLNET.RADIUS_AUTHENTICATION_INTERFACE=DefaultRadiusInterface
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_AUTHENTICATION_PORT**Purpose**

Use the parameter `SQLNET.RADIUS_AUTHENTICATION_PORT` to specify the listening port of the primary RADIUS server.

Default

1645

Example

```
SQLNET.RADIUS_AUTHENTICATION_PORT= 1667
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_AUTHENTICATION_RETRIES**Purpose**

To specify the number of times the database server should resend messages to the primary RADIUS server.

Default

3

Example

```
SQLNET.RADIUS_AUTHENTICATION_RETRIES=4
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_AUTHENTICATION_TIMEOUT**Purpose**

To specify the time, in seconds, that the database server should wait for a response from the primary RADIUS server.

Default

5

Example

```
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT=10
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_CHALLENGE_RESPONSE**Purpose**

To turn challenge response on or off.

Default

off

Values

on | off

Example

```
SQLNET.RADIUS_CHALLENGE_RESPONSE=on
```

SQLNET.RADIUS_SECRET**Purpose:**

To specify the location of the RADIUS secret key.

Default

The ORACLE_HOME/network/security/radius.key file.

Example

```
SQLNET.RADIUS_SECRET=oracle/bin/admin/radiuskey
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RADIUS_SEND_ACCOUNTING

Purpose

To turn accounting `on` and `off`. If enabled, then packets are sent to the active RADIUS server at listening port plus one. The default port is 1646.

Default

`off`

Values

`on` | `off`

Example

```
SQLNET.RADIUS_SEND_ACCOUNTING=on
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SQLNET.RECV_TIMEOUT

Purpose

To specify the time, in seconds, for a database server to wait for client data after establishing a connection. A client must send some data within the time interval.

For environments in which clients shut down on occasion or abnormally, setting this parameter is recommended. If a client does not send any data in time specified, then the database server logs `ORA-12535: TNS:operation timed out` and `ORA-12609: TNS: Receive timeout occurred` messages to the `sqlnet.log` file. Without this parameter, the database server may continue to wait for data from clients that may be down or are experiencing difficulties.

You can also set this parameter on the client-side to specify the time, in seconds, for a client to wait for response data from the database server after connection establishment. Without this parameter, the client may wait a long period of time for a response from a database server saturated with requests. If you choose to set the value, then set the value to an initial low value and adjust according to system and network capacity. If necessary, use this parameter with the [SQLNET.SEND_TIMEOUT](#) parameter.

Default

None

Example

```
SQLNET.RECV_TIMEOUT=3
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring these parameters

SQLNET.SEND_TIMEOUT

Purpose

To specify the time, in seconds, for a database server to complete a send operation to clients after establishing a connection. Setting this parameter is recommended for environments in which clients shut down occasionally or abnormally.

If the database server cannot complete a send operation in the time specified, then it logs `ORA-12535: TNS:operation timed out` and `ORA-12608: TNS: Send timeout occurred` messages to the `sqlnet.log` file. Without this parameter, the database server may continue to send responses to clients that are unable to receive data due to a downed computer or a busy state.

You can also set this parameter on the client-side to specify the time, in seconds, for a client to complete send operations to the database server after connection establishment. Without this parameter, the client may continue to send requests to a database server already saturated with requests. If you choose to set the value, then set the value to an initial low value and adjust according to system and network capacity. If necessary, use this parameter with the [SQLNET.RECV_TIMEOUT](#) parameter.

Default

None

Example

```
SQLNET.SEND_TIMEOUT=3
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring these parameters

SSL_CERT_REVOCATION**Purpose**

To configure a revocation check for a certificate.

See Also: *Oracle Database Advanced Security Administrator's Guide*

Default

None

Values

- `none` to turn off certificate revocation checking. This is the default.
- `requested` to perform certificate revocation in case a Certificate Revocation List (CRL) is available. Reject SSL connection if the certificate is revoked. If no appropriate CRL is found to determine the revocation status of the certificate and the certificate is not revoked, then accept the SSL connection.
- `required` to perform certificate revocation when a certificate is available. If a certificate is revoked and no appropriate CRL is found, then reject the SSL connection. If no appropriate CRL is found to ascertain the revocation status of the certificate and the certificate is not revoked, then accept the SSL connection.

Example

```
SSL_CERT_REVOCATION=required
```

SSL_CERT_FILE**Purpose**

To specify the name of the file where you can assemble the certificate revocation list (CRL) for client authentication.

This file contains the PEM-encoded CRL files, in order of preference. You can use this file alternatively or in addition to the [SSL_CERT_PATH](#) parameter. This parameter is only valid if [SSL_CERT_REVOCATION](#) is set to either `requested` or `required`.

Default

None

Example

```
SSL_CERT_FILE=
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SSL_CERT_PATH

Purpose

To specify the destination directory of the CRL of CA. The files in this directory are hashed symbolic links created by Oracle Wallet Manager. This parameter is only valid if [SSL_CERT_REVOCATION](#) is set to either `requested` or `required`.

See Also: *Oracle Database Security Guide*

Default

None

Example

```
SSL_CERT_PATH=
```

SSL_CIPHER_SUITES

Purpose

To control which combination of encryption and data integrity is used by the [Secure Sockets Layer \(SSL\)](#). Cipher suites that use Advanced Encryption Standard (AES) only work with Transport Layer Security (TLS 1.0).

Default

None

Values

- SSL_RSA_WITH_3DES_EDE_CBC_SHA
- SSL_RSA_WITH_RC4_128_SHA
- SSL_RSA_WITH_RC4_128_MD5
- SSL_RSA_WITH_DES_CBC_SHA
- SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
- SSL_DH_anon_WITH_RC4_128_MD5
- SSL_DH_anon_WITH_DES_CBC_SHA
- SSL_RSA_EXPORT_WITH_RC4_40_MD5
- SSL_RSA_EXPORT_WITH_DES40_CBC_SHA

- SSL_RSA_WITH_AES_128_CBC_SHA
- SSL_RSA_WITH_AES_256_CBC_SHA

Example

```
SSL_CIPHER_SUITES=(ssl_rsa_with_rc4_138_md5)
```

See Also: *Oracle Database Security Guide* for additional information about cipher suite values

SSL_CLIENT_AUTHENTICATION**Purpose**

To specify whether a client, in addition to the database server, is authenticated using SSL.

Default

true

Values

true | false

Example

```
SSL_CLIENT_AUTHENTICATION=true
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SSL_SERVER_DN_MATCH**Purpose**

To enforce that the **distinguished name (DN)** for the database server matches its service name. If you enforce the match verifications, then SSL ensures that the certificate is from the server. If you select to not enforce the match verification, then SSL performs the check but allows the connection, regardless if there is a match. Not enforcing the match allows the server to potentially fake its identify.

Default

no

Values

- yes | on | true to specify to enforce a match. If the DN matches the service name, then the connection succeeds. If the DN does not match the service name, then the connection fails.
- no | off | false to specify not to enforce a match. If the DN does not match the service name, then the connection is successful, but an error is logged to the `sqlnet.log` file.

Usage Notes

In addition to the `sqlnet.ora` file, configure the `tnsnames.ora` parameter [SSL_SERVER_CERT_DN](#) to enable server DN matching.

Example

```
SSL_SERVER_DN_MATCH=yes
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SSL_VERSION**Purpose**

To force the version of the SSL connection. Clients and database servers must use a compatible version.

Default

undetermined

Values

undetermined | 2.0 | 3.0

Example

```
SSL_VERSION=2.0
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

TCP.CONNECT_TIMEOUT**Purpose**

To specify the time, in seconds, for a client to establish a TCP connection (`PROTOCOL=tcp` in the TNS connect address) to the database server. If a TCP connection to the database host is not established in the time specified, then the connection attempt is terminated. The client receives an `ORA-12170: TNS:Connect timeout occurred` error.

The timeout applies to each IP address to which a host name resolves. For example, if a host name resolves to an IPv6 and an IPv4 address, and if the host is not reachable through the network, then the connection request times out twice the `TCP.CONNECT_TIMEOUT` setting because there are two IP addresses. In this example, the default timeout setting of 60 would cause a timeout in 120 seconds.

Default

60

Example

```
TCP.CONNECT_TIMEOUT=10
```

TCP.EXCLUDED_NODES**Purpose**

To specify which clients are denied access to the database.

Syntax

```
TCP.EXCLUDED_NODES=(hostname | ip_address, hostname | ip_address, ...)
```

Usage Notes

- This parameter is only valid when the `TCP.VALIDNODE_CHECKING` parameter is set to `yes`.
- This parameter can use wildcards for IPv4 addresses and CIDR (Classless Inter-Domain Routing) notation for IPv4 and IPv6 addresses.

Example

```
TCP.EXCLUDED_NODES=(finance.us.example.com, mktg.us.example.com, 192.168.2.25,
172.30.*, 2001:DB8:200C:417A/32)
```

TCP.INVITED_NODES**Purpose**

To specify which clients are allowed access to the database. This list takes precedence over the `TCP.EXCLUDED_NODES` parameter if both lists are present.

Syntax

```
TCP.INVITED_NODES=(hostname | ip_address, hostname | ip_address, ...)
```

Usage Notes

- This parameter is only valid when the `TCP.VALIDNODE_CHECKING` parameter is set to `yes`.
- This parameter can use wildcards for IPv4 addresses and CIDR notation for IPv4 and IPv6 addresses.

Example

```
TCP.INVITED_NODES=(sales.us.example.com, hr.us.example.com, 192.168.*,
2001:DB8:200C:433B/32)
```

TCP.NODELAY**Purpose**

To preempt delays in buffer flushing within the TCP/IP protocol stack.

Default

yes

Values

yes | no

Example

```
TCP.NODELAY=yes
```

TCP.VALIDNODE_CHECKING**Purpose**

To enable and disable valid node checking for incoming connections. If this parameter is set to `yes`, then incoming connections are allowed only if they originate from a node

that conforms to list specified by [TCP.INVITED_NODES](#) or [TCP.EXCLUDED_NODES](#) parameters.

Default

no

Values

yes | no

Usage Notes

The [TCP.INVITED_NODES](#) and [TCP.EXCLUDED_NODES](#) parameters are valid only when the [TCP.VALIDNODE_CHECKING](#) parameter is set to `yes`.

Example

```
TCP.VALIDNODE_CHECKING=yes
```

TNSPING.TRACE_DIRECTORY

Purpose

To specify the destination directory for the TNSPING utility trace file, `tnsping.trc`.

Default

The `ORACLE_HOME/network/trace` directory.

Example

```
TNSPING.TRACE_DIRECTORY=/oracle/traces
```

TNSPING.TRACE_LEVEL

Purpose

To turn TNSPING utility tracing on at a specified level or to turn it off.

Default

off

Values

- `off` for no trace output
- `user` for user trace information
- `admin` for administration trace information
- `support` for Oracle Support Services trace information

Example

```
TNSPING.TRACE_LEVEL=admin
```

USE_CMAN

Purpose

To specify client routing to Oracle Connection Manager.

If set to `true`, then the parameter routes the client to a protocol address for an Oracle Connection Manager.

If set to `false`, then the client picks one of the address lists at random and fails over to the other address list if the chosen `ADDRESS_LIST` fails. With `USE_CMAN=true`, the client always uses the first address list.

If no Oracle Connection Manager addresses are available, then connections are routed through any available listener address.

Default

`false`

Values

`true` | `false`

Example

`USE_CMAN=true`

USE_DEDICATED_SERVER**Purpose**

To append (`SERVER=dedicated`) to the `CONNECT_DATA` section of the connect descriptor used by the client. It overrides the current value of the `SERVER` parameter in the `tnsnames.ora` file.

If set to `on`, then the parameter `USE_DEDICATED_SERVER` automatically appends (`SERVER=dedicated`) to the connect data for a connect descriptor. This way connections from this client use a **dedicated server** process, even if **shared server** is configured.

Default

`off`

Values

- `on` to append (`SERVER=dedicated`)
- `off` to send requests to existing server processes

Example

`USE_DEDICATED_SERVER=on`

See Also: *Oracle Database Net Services Administrator's Guide* for complete configuration information

WALLET_LOCATION**Purpose**

To specify the location of wallets. Wallets are certificates, keys, and trustpoints processed by SSL.

Syntax

The syntax depends on the wallet, as follows:

- Oracle wallets on the file system:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=file)  
    (METHOD_DATA=  
      (DIRECTORY=directory)  
      [ (PKCS11=TRUE/FALSE) ] ) )
```

- Microsoft certificate store:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=mcs) )
```

- Oracle wallets in the Microsoft Windows registry:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=reg)  
    (METHOD_DATA=  
      (KEY=registry_key) ) )
```

- Entrust wallets:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=entr)  
    (METHOD_DATA=  
      (PROFILE=file.epf)  
      (INIFILE=file.ini) ) )
```

Additional Parameters

WALLET_LOCATION supports the following parameters:

- SOURCE: The type of storage for wallets and storage location.
- METHOD: The type of storage.
- METHOD_DATA: The storage location.
- DIRECTORY: The location of Oracle wallets on file system.
- KEY: The wallet type and location in the Microsoft Windows registry.
- PROFILE: The Entrust profile file (.epf).
- INIFILE: The Entrust initialization file (.ini).

Default

None

Usage Notes

- The key/value pair for Microsoft certificate store (MCS) omits the METHOD_DATA parameter because MCS does not use wallets. Instead, Oracle PKI (public key infrastructure) applications obtain certificates, trustpoints and private keys directly from the user's profile.
- If an Oracle wallet is stored in the Microsoft Windows registry and the wallet's key (KEY) is SALESAPP, then the storage location of the encrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12. The

storage location of the decrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO.

Values

true | false

Examples

Oracle wallets on file system:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=file)
(METHOD_DATA=
(DIRECTORY=/etc/oracle/wallets/databases)))
```

Microsoft certificate store:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=mcs))
```

Oracle Wallets in the Microsoft Windows registry:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=REG)
(METHOD_DATA=
(KEY=SALESAPP)))
```

Entrust Wallets:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=entr)
(METHOD_DATA=
(PROFILE=/etc/oracle/wallets/test.epf)
(INIFILE=/etc/oracle/wallets/test.ini)))
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

WALLET_OVERRIDE

Purpose

To determine whether the client should override the strong authentication credential with the password credential in the stored wallet to log in to the database.

Usage Notes

- When wallets are used for authentication, the database credentials for user name and password are securely stored in an Oracle wallet. The auto-login feature of the wallet is turned on so the database does not need a password to open the wallet. From the wallet, the database gets the credentials to access the database for the user.
- Wallet usage can simplify large-scale deployments that rely on password credentials for connecting to databases. When this feature is configured, application code, batch jobs, and scripts do not need embedded user names and passwords. Risk is reduced because such passwords are no longer exposed in the

clear, and password management policies are more easily enforced without changing application code whenever user names or passwords change.

- Users connect using the `connect /@database_name` command instead of specifying a user name and password explicitly. This simplifies the maintenance of the scripts and secures the password management for the applications.
- Middle-tier applications create an Oracle Applications wallet at installation time to store the application's specific identity. The password may be randomly generated rather than hardcoded. When an Oracle application accesses the database, it sets appropriate values for `SQLNET.AUTHENTICATION_SERVICES` and `WALLET_LOCATION`. The new wallet-based password authentication code uses the password credential in the Oracle Applications wallet to log on to the database.

Values

TRUE | FALSE

Examples

```
WALLET_OVERRIDE=TRUE
```

See Also: In order to use wallets, a wallet must be configured on the client. Refer to *Oracle Database Security Guide* for additional information about configuring the clients.

ADR Diagnostic Parameters in sqlnet.ora

Beginning with Oracle Database 11g, Oracle Database includes an advanced fault diagnosability infrastructure for preventing, detecting, diagnosing, and resolving problems. The problems are critical errors such as those caused by database code bugs, metadata corruption, and customer data corruption.

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error, such as traces and dumps, is immediately captured and tagged with the incident number. The data is then stored in the **Automatic Diagnostic Repository (ADR)**, a file-based repository outside the database.

This section describes the parameters used when ADR is enabled. "[Non-ADR Diagnostic Parameters in sqlnet.ora](#)" on page 5-35 describes the parameters used when ADR is disabled. Non-ADR parameters listed in the `sqlnet.ora` file are ignored when ADR is enabled. ADR is enabled by default.

This section lists the parameters used when ADR is enabled (when `DIAG_ADR_ENABLED` is set to on):

- [ADR_BASE](#)
- [DIAG_ADR_ENABLED](#)
- [TRACE_LEVEL_CLIENT](#)
- [TRACE_LEVEL_SERVER](#)
- [TRACE_TIMESTAMP_CLIENT](#)
- [TRACE_TIMESTAMP_SERVER](#)

ADR_BASE

Purpose

To specify the base directory into which tracing and logging incidents are stored when ADR is enabled.

Default

The default on the server side is `ORACLE_BASE`, or `ORACLE_HOME/log`, if `ORACLE_BASE` is not defined.

Values

Any valid directory path to a directory with write permission.

Example

```
ADR_BASE=/oracle/network/trace
```

See Also: *Oracle Call Interface Programmer's Guide* for the default on the client side

DIAG_ADR_ENABLED

Purpose

To specify whether ADR tracing is enabled.

Usage

If the `DIAG_ADR_ENABLED` parameter is set to `OFF`, then non-ADR file tracing is used.

Default

on

Values

on | off

Example

```
DIAG_ADR_ENABLED=on
```

TRACE_LEVEL_CLIENT

Purpose

To turn client tracing on at a specified level or to turn it off. This parameter is also applicable when non-ADR tracing is used.

Default

off or 0

Values

- off or 0 for no trace output
- user or 4 for user trace information
- admin or 10 for administration trace information

- `support` or `16` for Oracle Support Services trace information

Example

```
TRACE_LEVEL_CLIENT=user
```

TRACE_LEVEL_SERVER**Purpose**

To turn server tracing on at a specified level or to turn it off. This parameter is also applicable when non-ADR tracing is used.

Default

off or 0

Values

- `off` or `0` for no trace output
- `user` or `4` for user trace information
- `admin` or `10` for administration trace information
- `support` or `16` for Oracle Support Services trace information

Example

```
TRACE_LEVEL_SERVER=admin
```

TRACE_TIMESTAMP_CLIENT**Purpose**

To add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi` to every trace event in the client trace file, which has a default name of `sqlnet.trc`. This parameter is also applicable when non-ADR tracing is used.

Default

on

Values

on or true | off or false

Example

```
TRACE_TIMESTAMP_CLIENT=true
```

TRACE_TIMESTAMP_SERVER**Purpose**

To add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi` to every trace event in the database server trace file, which has a default name of `svr_pid.trc`. This parameter is also applicable when non-ADR tracing is used.

Default

on

Values

on or true | off or false

Example

```
TRACE_TIMESTAMP_SERVER=true
```

Non-ADR Diagnostic Parameters in sqlnet.ora

This section lists the parameters used when ADR is disabled.

Notes: The default value of [DIAG_ADR_ENABLED](#) is on. Therefore, the [DIAG_ADR_ENABLED](#) parameter must explicitly be set to off in order for non-ADR tracing to be used.

- [LOG_DIRECTORY_CLIENT](#)
- [LOG_DIRECTORY_SERVER](#)
- [LOG_FILE_CLIENT](#)
- [LOG_FILE_SERVER](#)
- [TRACE_DIRECTORY_CLIENT](#)
- [TRACE_DIRECTORY_SERVER](#)
- [TRACE_FILE_CLIENT](#)
- [TRACE_FILE_SERVER](#)
- [TRACE_FILELEN_CLIENT](#)
- [TRACE_FILELEN_SERVER](#)
- [TRACE_FILENO_CLIENT](#)
- [TRACE_FILENO_SERVER](#)
- [TRACE_UNIQUE_CLIENT](#)

LOG_DIRECTORY_CLIENT**Purpose**

To specify the destination directory for the client log file. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/log

Values

Any valid directory path.

Example

```
LOG_DIRECTORY_CLIENT=/oracle/network/log
```

LOG_DIRECTORY_SERVER

Purpose

To specify the destination directory for the database server log file. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/trace

Values

Any valid directory path to a directory with write permission.

Example

```
LOG_DIRECTORY_SERVER=/oracle/network/trace
```

LOG_FILE_CLIENT

Purpose

To specify the name of the log file for the client. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/log/sqlnet.log

Values

The default value cannot be changed.

LOG_FILE_SERVER

Purpose

To specify the name of the log file for the database server. Use this parameter when ADR is not enabled.

Default

sqlnet.log

Example

```
LOG_FILE_SERVER=svr.log
```

TRACE_DIRECTORY_CLIENT

Purpose

To specify the destination directory for the client trace file. Use this parameter when ADR is not enabled.

Default

The current working directory.

Values

Any valid directory path to a directory with write permission.

Example

```
TRACE_DIRECTORY_CLIENT=/oracle/traces
```

TRACE_DIRECTORY_SERVER**Purpose**

To specify the destination directory for the database server trace file. Use this parameter when ADR is not enabled.

Default

```
ORACLE_HOME/network/trace
```

Values

Any valid directory path to a directory with write permission.

Example

```
TRACE_DIRECTORY_SERVER=/oracle/traces
```

TRACE_FILE_CLIENT**Purpose**

To specify the name of the client trace file. Use this parameter when ADR is not enabled.

Values

Any valid file name.

Default

```
ORACLE_HOME/network/trace/cli.trc
```

Example

```
TRACE_FILE_CLIENT=clientsqlnet.trc
```

TRACE_FILE_SERVER**Purpose**

To specify the name of the file to which the execution trace of the server program is written. Use this parameter when ADR is not enabled.

Default

```
ORACLE_HOME/network/trace/svr_pid.trc
```

Values

Any valid file name. The pid is appended to the name automatically.

Example

```
TRACE_FILE_SERVER=svrsqlnet.trc
```

TRACE_FILELEN_CLIENT**Purpose**

To specify the size of the client trace files in kilobytes (KB). When the size is met, the trace information is written to the next file. The number of files is specified with the [TRACE_FILENO_CLIENT](#) parameter. Use this parameter when ADR is not enabled.

Example

```
TRACE_FILELEN_CLIENT=100
```

TRACE_FILELEN_SERVER**Purpose**

To specify the size of the database server trace files in kilobytes (KB). When the size is met, the trace information is written to the next file. The number of files is specified with the [TRACE_FILENO_SERVER](#) parameter. Use this parameter when ADR is not enabled.

Example

```
TRACE_FILELEN_SERVER=100
```

TRACE_FILENO_CLIENT**Purpose**

To specify the number of trace files for client tracing. When this parameter is set with the [TRACE_FILELEN_CLIENT](#) parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is re-used, and so on.

The trace file names are distinguished from one another by their sequence number. For example, if the default trace file of `sqlnet.trc` is used, and this parameter is set to 3, then the trace files would be named `sqlnet1.trc`, `sqlnet2.trc` and `sqlnet3.trc`.

In addition, trace events in the trace files are preceded by the sequence number of the file. Use this parameter when ADR is not enabled.

Default

None

Example

```
TRACE_FILENO_CLIENT=3
```

TRACE_FILENO_SERVER**Purpose**

To specify the number of trace files for database server tracing. When this parameter is set with the [TRACE_FILELEN_SERVER](#) parameter, trace files are used in a cyclical

fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is re-used, and so on.

The trace file names are distinguished from one another by their sequence number. For example, if the default trace file of *svr_pid.trc* is used, and this parameter is set to 3, then the trace files would be named *svr1_pid.trc*, *svr2_pid.trc* and *svr3_pid.trc*.

In addition, trace events in the trace files are preceded by the sequence number of the file. Use this parameter when ADR is not enabled.

Default

None

Example

```
TRACE_FILENO_SERVER=3
```

TRACE_UNIQUE_CLIENT

Purpose

To specify whether a unique trace file is created for each client trace session. When the value is set to *on*, a process identifier is appended to the name of each trace file, enabling several files to coexist. For example, trace files named *sqlnetpid.trc* are created if default trace file name *sqlnet.trc* is used. When the value is set to *off*, data from a new client trace session overwrites the existing file. Use this parameter when ADR is not enabled.

Default

on

Values

on or off

Example

```
TRACE_UNIQUE_CLIENT=on
```

Local Naming Parameters (tnsnames.ora)

This chapter provides a complete listing of the `tnsnames.ora` file configuration parameters.

This chapter contains the following topics:

- [Overview of Local Naming Parameters](#)
- [General Syntax of tnsnames.ora](#)
- [Multiple Descriptions in tnsnames.ora](#)
- [Multiple Address Lists in tnsnames.ora](#)
- [Connect-Time Failover and Client Load Balancing with Oracle Connection Managers](#)
- [Connect Descriptor Descriptions](#)
- [Protocol Address Section](#)
- [Optional Parameters for Address Lists](#)
- [Connection Data Section](#)
- [Security Section](#)
- [Timeout Parameters](#)

Overview of Local Naming Parameters

This `tnsnames.ora` file is a configuration file that contains **net service names** mapped to **connect descriptors** for the **local naming** method, or net service names mapped to listener protocol addresses.

A net service name is an alias mapped to a database network address contained in a connect descriptor. A connect descriptor contains the location of the listener through a protocol address and the service name of the database to which to connect. Clients and database servers (that are clients of other database servers) use the net service name when making a connection with an application.

By default, the `tnsnames.ora` file is located in the `ORACLE_HOME/network/admin` directory. Oracle Net will check the other directories for the configuration files. For example, the order checking the `tnsnames.ora` file is as follows:

1. The directory specified by the `TNS_ADMIN` environment variable. If the file is not found in the directory specified, then it is assumed that the file does not exist.
2. If the `TNS_ADMIN` environment variable is not set, then Oracle Net will check the `ORACLE_HOME/network/admin` directory.

Note: On Microsoft Windows, the `TNS_ADMIN` environment variable is used if it is set in the environment of the process. If the `TNS_ADMIN` environment variable is not defined in the environment, or the process is a service which does not have an environment, then Microsoft Windows scans the registry for a `TNS_ADMIN` parameter.

See Also: Oracle operating system-specific documentation

General Syntax of tnsnames.ora

The basic syntax for a `tnsnames.ora` file is shown in [Example 6-1](#).

Example 6-1 Basic Format of tnsnames.ora File

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS=(protocol_address_information))
    (CONNECT_DATA=
      (SERVICE_NAME=service_name)))
```

In the preceding example, `DESCRIPTION` contains the connect descriptor, `ADDRESS` contains the protocol address, and `CONNECT_DATA` contains the database service identification information.

Multiple Descriptions in tnsnames.ora

A `tnsnames.ora` file can contain net service names with one or more connect descriptors. Each connect descriptor can contain one or more protocol addresses. [Example 6-2](#) shows two connect descriptors with multiple addresses. `DESCRIPTION_LIST` defines a list of connect descriptors.

Example 6-2 Net Service Name with Multiple Connect Descriptors in tnsnames.ora

```
net_service_name=
  (DESCRIPTION_LIST=
    (DESCRIPTION=
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information))
      (CONNECT_DATA=
        (SERVICE_NAME=service_name)))
    (DESCRIPTION=
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information))
      (CONNECT_DATA=
        (SERVICE_NAME=service_name))))
```

Note: Oracle Net Manager does not support the creation of multiple connect descriptors for a net service name when using Oracle Connection Manager.

Multiple Address Lists in tnsnames.ora

The `tnsnames.ora` file also supports connect descriptors with multiple lists of addresses, each with its own characteristics. In [Example 6–3](#), two address lists are presented. The first address list features **client load balancing** and no **connect-time failover**, affecting only those protocol addresses within the `ADDRESS_LIST`. The second protocol address list features connect-time failover and no client load balancing, affecting only those protocol addresses within the `ADDRESS_LIST`. The client first tries the first or second protocol address at random, then tries protocol addresses three and four sequentially.

Example 6–3 Multiple Address Lists in tnsnames.ora

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (LOAD_BALANCE=on)
      (FAILOVER=off)
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information)))
    (ADDRESS_LIST=
      (LOAD_BALANCE=off)
      (FAILOVER=on)
      (ADDRESS=(protocol_address_information))
      (ADDRESS=(protocol_address_information)))
    (CONNECT_DATA=
      (SERVICE_NAME=service_name)))
```

Note: Oracle Net Manager supports only the creation of one protocol address list for a connect descriptor.

Connect-Time Failover and Client Load Balancing with Oracle Connection Managers

When a connect descriptor in a `tnsnames.ora` file contains at least two protocol addresses for **Oracle Connection Manager**, parameters for connect-time failover and load balancing can be included in the file.

[Example 6–4](#) illustrates failover of multiple Oracle Connection Manager protocol addresses.

Example 6–4 Multiple Oracle Connection Manager Addresses in tnsnames.ora

```
sample1=
  (DESCRIPTION=
    (SOURCE_ROUTE=yes)
    (ADDRESS=(PROTOCOL=tcp)(HOST=host1)(PORT=1630)) # 1
    (ADDRESS_LIST=
      (FAILOVER=on)
      (LOAD_BALANCE=off) # 2
      (ADDRESS=(PROTOCOL=tcp)(HOST=host2a)(PORT=1630))
      (ADDRESS=(PROTOCOL=tcp)(HOST=host2b)(PORT=1630)))
    (ADDRESS=(PROTOCOL=tcp)(HOST=host3)(PORT=1521)) # 3
    (CONNECT_DATA=(SERVICE_NAME=Sales.us.example.com)))
```

In [Example 6–4](#), the syntax does the following:

1. The client is instructed to connect to an protocol address of the first Oracle Connection Manager, as indicated by:

```
(ADDRESS=(PROTOCOL=tcp)(HOST=host1)(PORT=1630))
```

2. The first Oracle Connection Manager is instructed to connect to the first protocol address of another Oracle Connection Manager. If the first protocol address fails, then it tries the second protocol address. This sequence is specified with the following configuration:

```
(ADDRESS_LIST=
  (FAILOVER=ON)
  (LOAD_BALANCE=off)
  (ADDRESS=(PROTOCOL=tcp)(HOST=host2a)(PORT=1630))
  (ADDRESS=(PROTOCOL=tcp)(HOST=host2b)(PORT=1630))
```

3. The Oracle Connection Manager connects to the database service using the following protocol address:

```
(ADDRESS=(PROTOCOL=tcp)(HOST=host3)(PORT=1521))
```

Example 6–5 illustrates client load balancing among two Oracle Connection Managers and two protocol addresses:

Example 6–5 Client Load Balancing in tnsnames.ora

```
sample2=
  (DESCRIPTION=
    (LOAD_BALANCE=on) # 1
    (FAILOVER=on)
    (ADDRESS_LIST=
      (SOURCE_ROUTE=yes)
      (ADDRESS=(PROTOCOL=tcp)(HOST=host1)(PORT=1630)) # 2
      (ADDRESS=(PROTOCOL=tcp)(HOST=host2)(PORT=1521)))
    (ADDRESS_LIST=
      (SOURCE_ROUTE=yes)
      (ADDRESS=(PROTOCOL=tcp)(HOST=host3)(port=1630))
      (ADDRESS=(PROTOCOL=tcp)(HOST=host4)(port=1521)))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com)) # 3
```

In **Example 6–5**, the syntax does the following:

1. The client is instructed to pick an ADDRESS_LIST at random and to failover to the other if the chosen ADDRESS_LIST fails. This is indicated by the LOAD_BALANCE and FAILOVER parameters being set to on.
2. When an ADDRESS_LIST is chosen, the client first connects to the Oracle Connection Manager, using the Oracle Connection Manager protocol address that uses port 1630 indicated for the ADDRESS_LIST.
3. The Oracle Connection Manager then connects to the database service, using the protocol address indicated for the ADDRESS_LIST.

Connect Descriptor Descriptions

Each connect descriptor is contained within the **DESCRIPTION** parameter. Multiple connect descriptors are characterized by the **DESCRIPTION_LIST** parameter. These parameters are described in this section.

DESCRIPTION

Purpose

To specify a container for a connect descriptor. Put this parameter under the `DESCRIPTION_LIST` parameter.

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS=...))
(CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))
```

DESCRIPTION_LIST

Purpose

To define a list of connect descriptors for a particular net service name.

Example

```
net_service_name=
(DESCRIPTION_LIST=
  (DESCRIPTION=
    (ADDRESS=...))
  (CONNECT_DATA=(SERVICE_NAME=sales.example.com)))
  (DESCRIPTION=
    (ADDRESS=...))
  (CONNECT_DATA=(SERVICE_NAME=sales2.us.example.com)))
```

Protocol Address Section

The protocol address section of the `tnsnames.ora` file specifies the protocol addresses of the listener. If there is only one listener protocol address, then use the [ADDRESS](#) parameter. If there is more than one address, then use the [ADDRESS_LIST](#) parameter.

ADDRESS

Purpose

To define a single listener protocol address. Put this parameter under either the `ADDRESS_LIST` parameter or the `DESCRIPTION` parameter.

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales-svr)(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com)))
```

See Also: [Chapter 4, "Protocol Address Configuration"](#) for descriptions of the correct parameters to use for each protocol

ADDRESS_LIST

Purpose

To define a list of protocol addresses. If there is only one listener protocol address, then ADDRESS_LIST is not necessary. Put this parameter under either the DESCRIPTION parameter or the DESCRIPTION_LIST parameter.

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com)))
```

Optional Parameters for Address Lists

For multiple addresses, the following parameters are available:

- [ENABLE](#)
- [FAILOVER](#)
- [LOAD_BALANCE](#)
- [RECV_BUF_SIZE](#)
- [SDU](#)
- [SEND_BUF_SIZE](#)
- [SOURCE_ROUTE](#)
- [TYPE_OF_SERVICE](#)

ENABLE

Purpose

The keepalive feature on the supported TCP transports can be enabled for a net service client by putting (ENABLE=broken) under the [DESCRIPTION](#) parameter in the connect string. The keepalive feature allows the caller to detect a terminated remote server, although typically it takes 2 hours or more to notice. On the client side, the default for tcp_keepalive is off. Operating system TCP configurables, which vary by platform, define the actual keepalive timing details.

Values

BROKEN

Example

```
net_service_name=
  (DESCRIPTION=
    (ENABLE=broken)
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
  (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))
```

FAILOVER

Purpose

To enable or disable connect-time failover for multiple protocol addresses.

When you set the parameter to *on*, *yes*, or *true*, Oracle Net, at connect time, fails over to a different address if the first protocol address fails. When you set the parameter to *off*, *no*, or *false*, Oracle Net tries one protocol address.

Put this parameter under the `DESCRIPTION_LIST` parameter, the `DESCRIPTION` parameter, or the `ADDRESS_LIST` parameter.

Important: Do not set the `GLOBAL_DBNAME` parameter in the `SID_LIST_listener_name` section of the `listener.ora`. A statically configured global database name disables connect-time failover.

Default

on for `DESCRIPTION_LIST`, `DESCRIPTION`, and `ADDRESS_LIST`.

Values

- *yes* | *on* | *true*
- *no* | *off* | *false*

Example

```
net_service_name=
  (DESCRIPTION=
    (FAILOVER=on)
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))
```

LOAD_BALANCE

Purpose

To enable or disable client load balancing for multiple protocol addresses.

When you set the parameter to *on*, *yes*, or *true*, Oracle Net progresses the list of addresses in a random sequence, balancing the load on the various listener or Oracle Connection Manager protocol addresses. When you set the parameter to *off*, *no*, or *false*, Oracle Net tries the first address in the address list. If the connection fails and the failover parameter is enabled, then Oracle Net tries the addresses sequentially until one succeeds.

Put this parameter under either the `DESCRIPTION_LIST` parameter, the `DESCRIPTION` parameter, or the `ADDRESS_LIST` parameter.

Default

on for `DESCRIPTION_LIST`

Values

- *yes* | *on* | *true*
- *no* | *off* | *false*

Example

```
net_service_name=
  (DESCRIPTION=
    (LOAD_BALANCE=on)
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
  (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))
```

RECV_BUF_SIZE**Purpose**

To specify, in bytes, the buffer space for receive operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Put this parameter under the `DESCRIPTION` parameter or at the end of the protocol address.

Note: Additional protocols might support this parameter on certain operating systems. Refer to the operating system-specific documentation for additional information about additional protocols.

Default

The default value for this parameter is specific to the operating system.

Usage

Setting this parameter in the connect descriptor for a client overrides the [RECV_BUF_SIZE](#) parameter at the client-side `sqlnet.ora` file.

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-server)(PORT=1521)
        (RECV_BUF_SIZE=11784))
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-server)(PORT=1521)
        (RECV_BUF_SIZE=11784))
    )
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)))
net_service_name=
  (DESCRIPTION=
    (RECV_BUF_SIZE=11784)
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=hr1-server)(PORT=1521)
        (ADDRESS=(PROTOCOL=tcp)(HOST=hr2-server)(PORT=1521)))
    )
    (CONNECT_DATA=
      (SERVICE_NAME=hr.us.example.com)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring this parameter

SDU

Purpose

To instruct Oracle Net to optimize the transfer rate of data packets being sent across the network with a specified **session data unit (SDU)** size.

Put this parameter under the `DESCRIPTION` parameter.

Default

8192 bytes (8 KB).

Values

512 to 65535 bytes.

Usage

Setting this parameter in the connect descriptor for a client overrides the `DEFAULT_SDU_SIZE` parameter at client-side `sqlnet.ora` file.

Example

```
net_service_name=
(DESCRIPTION=
(SDU=8192)
 (ADDRESS_LIST=
 (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-server) (PORT=1521))
 (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-server) (PORT=1521)))
(CONNECT_DATA=
 (SERVER_NAME=sales.us.example.com))
```

See Also: *Oracle Database Net Services Administrator's Guide* for complete SDU usage and configuration information

SEND_BUF_SIZE

Purpose

To specify, in bytes, the buffer space for send operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note: Additional protocols might support this parameter on certain operating systems. Refer to the operating system-specific documentation for additional information about additional protocols.

Put this parameter under the `DESCRIPTION` parameter or at the end of the protocol address.

Default

The default value for this parameter is operating system-specific.

Usage

Setting this parameter in the connect descriptor for a client overrides the `SEND_BUF_SIZE` parameter at the client-side `sqlnet.ora` file.

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-server) (PORT=1521)
      (SEND_BUF_SIZE=11784))
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-server) (PORT=1521)
      (SEND_BUF_SIZE=11784))
  )
(CONNECT_DATA=
  (SERVICE_NAME=sales.us.example.com)))
net_service_name=
(DESCRIPTION=
  (SEND_BUF_SIZE=11784)
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp) (HOST=hr1-server) (PORT=1521)
      (SEND_BUF_SIZE=11784))
    (ADDRESS=(PROTOCOL=tcp) (HOST=hr2-server) (PORT=1521)
      (SEND_BUF_SIZE=11784))
  )
(CONNECT_DATA=
  (SERVICE_NAME=hr.us.example.com)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring this parameter

SOURCE_ROUTE**Purpose**

To enable routing through multiple protocol addresses.

When you set to on or yes, Oracle Net uses each address in order until the destination is reached.

To use Oracle Connection Manager, an initial connection from the client to Oracle Connection Manager is required, and a second connection from Oracle Connection Manager to the listener is required.

Put this parameter under either the DESCRIPTION_LIST parameter, the DESCRIPTION parameter, or the ADDRESS_LIST parameter.

Default

off

Values

- yes | on
- no | off

Example

```
net_service_name=
(DESCRIPTION=
  (SOURCE_ROUTE=on)
  (ADDRESS=(PROTOCOL=tcp) (HOST=cman-pc) (PORT=1630))
  (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-svr) (PORT=1521)))
(CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))
```

See Also: *Oracle Database Net Services Administrator's Guide* for complete configuration information

TYPE_OF_SERVICE

Purpose

To specify the type of service to use for an Oracle Rdb database. This parameter should only be used if the application supports both an Oracle Rdb and Oracle database service, and you want the application to load balance between the two.

Put this parameter under the `DESCRIPTION` parameter.

Example

```
net_service_name=
  (DESCRIPTION_LIST=
    (DESCRIPTION=
      (ADDRESS=...)
      (CONNECT_DATA=
        (SERVICE_NAME=generic)
        (RDB_DATABASE=[.mf]mf_personal.rdb)
        (GLOBAL_NAME=alpha5))
      (TYPE_OF_SERVICE=rdb_database))
    (DESCRIPTION=
      (ADDRESS=...)
      (CONNECT_DATA=
        (SERVICE_NAME=sales.us.example.com)
        (TYPE_OF_SERVICE=oracle11_database)))
```

Connection Data Section

The connection data section of the `tnsnames.ora` file specifies the name of the destination service. The following parameters are available:

- [CONNECT_DATA](#)
- [FAILOVER_MODE](#)
- [GLOBAL_NAME](#)
- [HS](#)
- [INSTANCE_NAME](#)
- [RDB_DATABASE](#)
- [SERVER](#)
- [SERVICE_NAME](#)

CONNECT_DATA

Purpose

To define the service to which to connect, such as `SERVICE_NAME`.

Put this parameter under the `DESCRIPTION` parameter.

Usage Notes

`CONNECT_DATA` permits the following additional parameters:

- [FAILOVER_MODE](#)
- [GLOBAL_NAME](#)

- HS
- INSTANCE_NAME
- RDB_DATABASE
- SERVER
- SERVICE_NAME

Example

```
net_service_name=  
(DESCRIPTION=  
  (ADDRESS= (PROTOCOL=tcp) (HOST=sales1-svr) (PORT=1521))  
  (ADDRESS= (PROTOCOL=tcp) (HOST=sales2-svr) (PORT=1521))  
  (CONNECT_DATA=  
    (SERVICE_NAME=sales.us.example.com)))
```

FAILOVER_MODE

Purpose

To instruct Oracle Net to fail over to a different listener if the first listener fails during run time. Depending upon the configuration, session or any `SELECT` statements which were in progress are automatically failed over.

This type of failover is called **Transparent Application Failover (TAF)** and should not be confused with the connect-time failover `FAILOVER` parameter.

Put this parameter under the `CONNECT_DATA` parameter.

Additional Parameters

`FAILOVER_MODE` supports the following parameters:

- **BACKUP**: Specify the failover node by its net service name. A separate net service name must be created for the failover node.
- **TYPE**: Specify the type of failover. Three types of Oracle Net failover functionality are available by default to **Oracle Call Interface (OCI)** applications:
 - **SESSION**: Fails over the session. For example, if a user's connection is lost, then a new session is automatically created for the user on the backup. This type of failover does not attempt to recover selects.
 - **SELECT**: Allows users with open cursors to continue fetching them after failure. However, this mode involves overhead on the client side in normal select operations.
 - **NONE**: This is the default, in which no failover functionality is used. This can also be explicitly specified to prevent failover from happening.
- **METHOD**: Specify how fast failover is to occur from the primary node to the backup node:
 - **BASIC**: Establishes connections at failover time. This option requires almost no work on the backup database server until failover time.
 - **PRECONNECT**: Pre-establishes connections. This provides faster failover but requires that the backup instance be able to support all connections from every supported instance.

- **RETRIES:** Specify the number of times to attempt to connect after a failover. If **DELAY** is specified, then **RETRIES** defaults to five retry attempts.
- **DELAY:** Specify the amount of time in seconds to wait between connect attempts. If **RETRIES** is specified, then **DELAY** defaults to one second.

Note: If a callback function is registered, then **RETRIES** and **DELAY** parameters are ignored.

See Also: *Oracle Database Net Services Administrator's Guide* for additional configuration information

GLOBAL_NAME

Purpose

To identify the Oracle Rdb database.

Put this parameter under the **CONNECT_DATA** parameter.

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS=...)
    (ADDRESS=...)
    (CONNECT_DATA=
      (SERVICE_NAME=generic)
      (RDB_DATABASE=[.mf]mf_personal.rdb)
      (GLOBAL_NAME=alpha5)))
```

HS

Purpose

To direct Oracle Net to connect to a non-Oracle system through [Heterogeneous Services](#).

Put this parameter under the **CONNECT_DATA** parameter.

Default

None

Values

ok

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS=...)
    (ADDRESS=...)
    (CONNECT_DATA=
      (SID=sales6)
      (HS=ok)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for complete configuration information

INSTANCE_NAME

Purpose

To identify the database instance to access. Set the value to the value specified by the `INSTANCE_NAME` parameter in the initialization parameter file.

Put this parameter under the `CONNECT_DATA` parameter.

Example

```
net_service_name=
(DESCRIPTION=
 (ADDRESS=...)
 (ADDRESS=...)
 (CONNECT_DATA=
 (SERVICE_NAME=sales.us.example.com)
 (INSTANCE_NAME=sales1)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about the use of `INSTANCE_NAME`

RDB_DATABASE

Purpose

To specify the file name of an Oracle Rdb database.

Put this parameter under the `CONNECT_DATA` parameter.

Example

```
net_service_name=
(DESCRIPTION=
 (ADDRESS=...)
 (ADDRESS=...)
 (CONNECT_DATA=
 (SERVICE_NAME=sales.us.example.com)
 (RDB_DATABASE= [.mf]mf_personal.rdb)))
```

SERVER

Purpose

To direct the listener to connect the client to a specific type of [service handler](#).

Put this parameter under the `CONNECT_DATA` parameter.

Values

- `dedicated` to specify whether client requests be served by [dedicated server](#)
- `shared` to specify whether client request be served by [shared server](#)
- `pooled` to get a connection from the connection pool if database resident connection pooling is enabled on the server

Notes: Shared server must be configured in the database initialization file in order for the client to connect to the database with a shared server process.

The [USE_DEDICATED_SERVER](#) parameter in the `sqlnet.ora` file overrides this parameter.

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS=...)
    (ADDRESS=...)
    (CONNECT_DATA=
      (SERVER_NAME=sales.us.example.com)
      (SERVER=dedicated)))
```

See Also:

- *Oracle Database Net Services Administrator's Guide* for additional information about database resident connection pooling
- *Oracle Call Interface Programmer's Guide* and *Oracle Database Administrator's Guide* for additional information about enabling and configuring database resident connection pooling

SERVICE_NAME

Purpose

To identify the Oracle Database database service to access. Set the value to a value specified by the `SERVICE_NAMES` parameter in the initialization parameter file.

Put this parameter under the `CONNECT_DATA` parameter.

Example

```
net_service_name=
  (DESCRIPTION=
    (ADDRESS=...)
    (ADDRESS=...)
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about the use of the `SERVICE_NAME` parameter

Security Section

The security section of the `tnsnames.ora` file specifies the following security-related parameters for use with Oracle Advanced Security features:

- [SECURITY](#)
- [SSL_SERVER_CERT_DN](#)

SECURITY

Purpose

To enable secure connections. Put this parameter under the `DESCRIPTION` parameter.

Usage Notes

`SECURITY` permits the [SSL_SERVER_CERT_DN](#) parameter.

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521))
  (CONNECT_DATA=
    (SERVICE_NAME=sales.us.example.com))
  (SECURITY=
    (SSL_SERVER_CERT_DN="cn=sales,cn=OracleContext,dc=us,dc=acme,dc=com")))
```

SSL_SERVER_CERT_DN

Purpose

To specify the **distinguished name (DN)** of the database server. The client uses this information to obtain the list of DNs it expects for each of the servers, enforcing the database server DN to match its service name.

Usage Notes

Use this parameter with the `sqlnet.ora` parameter [SSL_SERVER_DN_MATCH](#) to enable server DN matching.

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS=...)
  (ADDRESS=...)
  (CONNECT_DATA=
    (SERVICE_NAME=finance.us.example.com))
  (SECURITY=
    (SSL_SERVER_CERT_DN="cn=finance,cn=OracleContext,dc=us,dc=acme,dc=com")))
```

See Also: *Oracle Database Security Guide*

Timeout Parameters

The timeout section of the `tnsnames.ora` file provides the ability to specify timeout and retry configuration through the TNS connect string. The following parameters can now be set at the `DESCRIPTION` level of a connect string:

- [CONNECT_TIMEOUT](#)
- [RETRY_COUNT](#)
- [TRANSPORT_CONNECT_TIMEOUT](#)

CONNECT_TIMEOUT

Purpose

To specify the timeout duration in seconds for a client to establish an Oracle Net connection to an Oracle database. Put this parameter under the `DESCRIPTION` parameter.

Usage Notes

The timeout interval specified by `CONNECT_TIMEOUT` is a superset of the TCP connect timeout interval. It includes the time to be connected to the database instance providing the requested service, not just the duration of the TCP connection.

The default value of `CONNECT_TIMEOUT` is dependent on the `TCP_CONNECT_TIMEOUT` parameter. Oracle recommends setting the `CONNECT_TIMEOUT` value slightly greater than the `TCP_CONNECT_TIMEOUT` value.

The timeout interval is applicable for each `ADDRESS` in an `ADDRESS_LIST`, and each IP address to which a host name is mapped.

The `CONNECT_TIMEOUT` parameter is equivalent to the `sqlnet.ora` parameter `SQLNET.OUTBOUND_CONNECT_TIMEOUT` and overrides it.

Example

```
net_service_name=
  (DESCRIPTION=
    (CONNECT_TIMEOUT=10) (RETRY_COUNT=3)
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-svr) (PORT=1521))
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-svr) (PORT=1521)))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)))
```

RETRY_COUNT

Purpose

To specify the number of times an `ADDRESS` list is traversed before the connection attempt is terminated. The default value is 0.

Put this parameter under the `DESCRIPTION` parameter.

Usage Notes

When a `DESCRIPTION_LIST` is specified, each `DESCRIPTION` is traversed multiple times based on the specified number of retries. `RETRY_COUNT` is only supported at `DESCRIPTION` level in connect string.

When using `SCAN` listeners in an Oracle Real Application Clusters environment, with `FAILOVER = on`, setting the `RETRY_COUNT` parameter to 2 means the three `SCAN` IP addresses are traversed three times each, such that there are nine connect attempts (3 * 3).

Example

```
net_service_name=
  (DESCRIPTION_LIST=
    (DESCRIPTION=
      (CONNECT_TIMEOUT=10) (RETRY_COUNT=3)
      (ADDRESS_LIST=
```

```
(ADDRESS=(PROTOCOL=tcp)(HOST=sales1a-svr)(PORT=1521))
(ADDRESS=(PROTOCOL=tcp)(HOST=sales1b-svr)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=sales1.example.com))
(DESCRIPTION=
(CONNECT_TIMEOUT=60)(RETRY_COUNT=1)
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=tcp)(HOST=sales2a-svr)(PORT=1521))
(ADDRESS=(PROTOCOL=tcp)(HOST=sales2b-svr)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=sales2.us.example.com)))
```

TRANSPORT_CONNECT_TIMEOUT

Purpose

To specify the transportation timeout duration in seconds for a client to establish an Oracle Net connection to an Oracle Database.

This parameter is put under the `DESCRIPTION` parameter.

Usage Notes

The `TRANSPORT_CONNECT_TIMEOUT` parameter specifies the time, in seconds, for a client to establish a TCP connection to the database server. The default value is 60 seconds.

The timeout interval is applicable for each `ADDRESS` in an `ADDRESS_LIST` description, and each IP address that a host name is mapped. The `TRANSPORT_CONNECT_TIMEOUT` parameter is equivalent to the `sqlnet.ora` parameter `TCP.CONNECT_TIMEOUT`, and overrides it.

Example

```
net_service_name =
(DESCRIPTION=
(TRANSPORT_CONNECT_TIMEOUT=10)
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
(ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521))
(CONNECT_DATA=
(SERVICE_NAME=sales.us.example.com)))
```

Oracle Net Listener Parameters (listener.ora)

This chapter provides a complete listing of the `listener.ora` file configuration parameters.

This chapter contains the following topics:

- [Overview of Oracle Net Listener Configuration File](#)
- [Protocol Address Parameters](#)
- [Connection Rate Limiter Parameters](#)
- [Control Parameters](#)
- [ADR Diagnostic Parameters for Oracle Net Listener](#)
- [Non-ADR Diagnostic Parameters for Oracle Net Listener](#)
- [Class of Secure Transports Parameters](#)

Overview of Oracle Net Listener Configuration File

Oracle Net Listener configuration, stored in the `listener.ora` file, consists of the following elements:

- Name of the listener
- Protocol addresses that the listener is accepting connection requests on
- Database services
- Control parameters

Dynamic [service registration](#), eliminates the need for static configuration of supported services. However, static service configuration is required if you plan to use Oracle Enterprise Manager.

By default, the `listener.ora` file is located in the `ORACLE_HOME/network/admin` directory. The `listener.ora` file can also be stored the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value
- On Linux and UNIX operating systems, the global configuration directory. For example, on the Solaris Operating System, this directory is `/var/opt/oracle`.

See Also: Oracle operating system-specific documentation

It is possible to configure multiple listeners, each with unique name, in one `listener.ora` file. Multiple listener configurations are possible because each of the

top-level configuration parameters has a suffix of the listener name or is the listener name itself.

Note: It is often useful to configure multiple listeners in one `listener.ora` file. However, Oracle recommends running only one listener for each node in most customer environments.

[Example 7-1](#) shows a `listener.ora` file for a listener named `LISTENER`, which is the default name of the listener.

Example 7-1 *listener.ora File*

```
LISTENER=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sale-server)(PORT=1521))
      (ADDRESS=(PROTOCOL=ipc)(KEY=extproc)))
    SID_LIST_LISTENER=
      (SID_LIST=
        (SID_DESC=
          (GLOBAL_DBNAME=sales.us.example.com)
          (ORACLE_HOME=/oracle11g)
          (SID_NAME=sales))
        (SID_DESC=
          (SID_NAME=plsextproc)
          (ORACLE_HOME=/oracle11g)
          (PROGRAM=extproc)))
```

Protocol Address Parameters

The **protocol address** section of the `listener.ora` file defines the protocol addresses on which the listener is accepting connection requests. This section describes the most common parameters used in protocol addresses. The `ADDRESS_LIST` parameter is also supported.

See Also: [Chapter 4, "Protocol Address Configuration"](#) for additional information about the `ADDRESS_LIST` parameter

This section lists and describes the following parameters:

- [ADDRESS](#)
- [DESCRIPTION](#)
- [IP](#)
- [QUEUESIZE](#)
- [RECV_BUF_SIZE](#)
- [SEND_BUF_SIZE](#)

ADDRESS

Purpose

To specify a single listener protocol address.

Put this parameter under the `DESCRIPTION` parameter.

Example

```
listener_name=
  (DESCRIPTION=
    (ADDRESS= (PROTOCOL=tcp) (HOST=hr-server) (PORT=1521))
    (ADDRESS= (PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)))
```

See Also: [Chapter 4, "Protocol Address Configuration"](#) for descriptions of the correct parameters to use for each type of support protocol

DESCRIPTION**Purpose**

To contain listener protocol addresses.

Example

```
listener_name=
  (DESCRIPTION=
    (ADDRESS= (PROTOCOL=tcp) (HOST=hr-server) (PORT=1521))
    (ADDRESS= (PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)))
```

IP**Purpose**

To determine which IP address the listener listens on when a host name is specified. This parameter is only applicable when the `HOST` parameter specifies a host name.

Values

- `FIRST`
Listen on the first IP address returned by the DNS resolution of the host name. If the user wants the listener to listen on the first IP to which the specified host name resolves, then the address must be qualified with `(IP=FIRST)`.
- `V4_ONLY`
List only on IPv4 addresses.
- `V6_ONLY`
Listen only on IPv6 addresses.

Default

This feature is disabled by default.

Example

```
listener_name=
  (DESCRIPTION=
    (ADDRESS= (PROTOCOL=tcp) (HOST=rancode1-vip) (PORT=1522) (IP=V6_ONLY))
```

QUEUESIZE

Purpose

To specify the number of concurrent connection requests that the listener can accept on a TCP/IP or IPC listening endpoint (protocol address).

Put this parameter at the end of the protocol address with its value set to the expected number of concurrent connection requests.

Default

The default number of concurrent connection requests is operating system specific.

Usage Notes

The number of concurrent connection requests is dependent on the platform and listener usage scenarios. If the listener is heavily-loaded, then set the parameter to a higher number.

Example

```
listener_name=  
(DESCRIPTION=  
  (ADDRESS=(PROTOCOL=tcp) (HOST=hr-server) (PORT=1521) (QUEUESIZE=20)))
```

See Also: *Oracle Net Services Administrator's Guide* for additional information about configuring this parameter

RECV_BUF_SIZE

Purpose

To specify, in bytes, the buffer space for receive operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note: Additional protocols might support this parameter on certain operating systems. Refer to the operating system-specific documentation for additional information about additional protocols that support this parameter.

Put this parameter under the `DESCRIPTION` parameter or at the end of the protocol address with its value set to the expected number of bytes.

Default

The default value for this parameter is operating system-specific.

Example

```
listener_name=  
(DESCRIPTION=  
  (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)  
    (RECV_BUF_SIZE=11784))  
  (ADDRESS=(PROTOCOL=ipc) (KEY=extproc)  
    (RECV_BUF_SIZE=11784)))  
listener_name=  
(DESCRIPTION=  
  (RECV_BUF_SIZE=11784))  
  (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
```

```
(ADDRESS=(PROTOCOL=ipc)(KEY=extproc))
```

See Also: *Oracle Net Services Administrator's Guide* for additional information about configuring this parameter

SEND_BUF_SIZE

Purpose

To specify, in bytes, the buffer space for send operations of sessions. This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note: Additional protocols might support this parameter on certain operating systems. Refer to operating system-specific documentation for additional information about additional protocols that support this parameter.

Put this parameter under the DESCRIPTION parameter or at the end of the protocol address.

Default

The default value for this parameter is operating system-specific.

Example

```
listener_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)
        (SEND_BUF_SIZE=11280))
      (ADDRESS=(PROTOCOL=ipc)(KEY=extproc)
        (SEND_BUF_SIZE=11280)))
    listener_name=
      (DESCRIPTION=
        (ADDRESS_LIST=
          (SEND_BUF_SIZE=11280))
          (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)
            (ADDRESS=(PROTOCOL=ipc)(KEY=extproc))))
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring this parameter

Connection Rate Limiter Parameters

The connection rate limiter feature in the Oracle Net Listener enables a DBA to limit the number of new connections handled by the listener. When this feature is enabled, Oracle Net Listener imposes a user-specified maximum limit on the number of new connections handled by the listener every second.

Depending on the configuration, the rate can be applied to a collection of endpoints, or to a specific endpoint.

This feature is controlled through the following two `listener.ora` configuration parameters:

- [CONNECTION_RATE_listener name](#)
- [RATE_LIMIT](#)

CONNECTION_RATE_*listener name*

Purpose

To specify a global rate that is enforced across all listening endpoints that are rate-limited. When this parameter is specified, it overrides any endpoint-level numeric rate values that might be specified.

Syntax

```
CONNECTION_RATE_listener_name=number_of_connections_per_second
```

RATE_LIMIT

Purpose

To indicate that a particular listening endpoint is rate limited. The parameter is specified in the ADDRESS section of the listener endpoint configuration.

Syntax

```
LISTENER= (ADDRESS_LIST=  
  (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521) (RATE_LIMIT=yes))  
  )
```

- When the RATE_LIMIT parameter is set to *yes*, the end-point is included in the enforcement of a listener-wide connection rate. This is used with the CONNECTION_RATE_*listener_name* parameter.
- When the RATE_LIMIT parameter is set to a value greater than 0, the rate limit is enforced at that endpoint level.

Examples

The following examples use the CONNECTION_RATE_*listener name* and RATE_LIMIT parameters.

Example 1

```
CONNECTION_RATE_LISTENER=10
```

```
LISTENER=  
  (ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521) (RATE_LIMIT=yes))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1522) (RATE_LIMIT=yes))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1523))  
  )
```

In the preceding example, the total number of new connections through ports 1521 and 1522 is limited at 10 every second. Connections through port 1523 are not limited and do not count towards the overall rate of 10 connections every second.

Example 2

```
LISTENER= (ADDRESS_LIST=  
  (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521) (RATE_LIMIT=5))  
  (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1522) (RATE_LIMIT=10))  
  (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1523))  
  )
```

In the preceding example, the connection rates are enforced at the endpoint level. A maximum of 5 connections are processed through port 1521 every second. The limit

for connections through port 1522 is 10 every second. Connections through port 1523 are not limited.

Note: The global `CONNECTON_RATE_listener_name` parameter is not specified in the preceding configuration. If it is specified, then the limits on ports 1521 and 1522 are ignored, and the global value is used instead.

Control Parameters

This section describes the following parameters that control the behavior of the listener:

- [ADMIN_RESTRICTIONS_listener_name](#)
- [CRS_NOTIFICATION_listener_name](#)
- [DEFAULT_SERVICE_listener_name](#)
- [INBOUND_CONNECT_TIMEOUT_listener_name](#)
- [SAVE_CONFIG_ON_STOP_listener_name](#)
- [SSL_CLIENT_AUTHENTICATION](#)
- [SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name](#)
- [WALLET_LOCATION](#)

ADMIN_RESTRICTIONS_listener_name

Purpose

To restrict run-time administration of the listener.

Setting `ADMIN_RESTRICTIONS_listener_name=on` disables the run-time modification of parameters in `listener.ora`. That is, the listener refuses to accept `SET` commands that alter its parameters. To change any of the parameters in `listener.ora`, including `ADMIN_RESTRICTIONS_listener_name` itself, modify the `listener.ora` file manually and reload its parameters (using the [RELOAD](#) command) for the new changes to take effect without explicitly stopping and restarting the listener.

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about password security of the listener

Default

off

Example

```
ADMIN_RESTRICTIONS_listener=on
```

CRS_NOTIFICATION_listener_name

Purpose

To set notification. By default, the Oracle Net listener notifies Cluster Ready Services (CRS) when it is started or stopped. These notifications allow CRS to manage the

listener in an Oracle Real Application Clusters environment. This behavior can be prevented by setting the `CRS_NOTIFICATION_listener_name` parameter to `off`.

Default

on

Values

on | off

DEFAULT_SERVICE_listener_name

Purpose

To enable users to connect to the database without having to specify a service name from the client side.

In Oracle Database 11g, when a client tries to connect to the database the connection request passes through the listener. The listener may be servicing several different databases. If a service name is configured in this parameter, then users may not necessarily need to specify a service name in the connect syntax. If a user specifies a service name, then the listener connects the user to that specific database, otherwise the listener connects to the service name specified by the `DEFAULT_SERVICE_listener_name` parameter.

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about the Easy Connect naming method

Default

There is no default value for the `DEFAULT_SERVICE_listener_name` parameter. If this parameter is not configured and a user does not specify a fully-qualified service name in the connect syntax, then the connection attempt fails. This parameter only accepts one value.

Example

```
DEFAULT_SERVICE_LISTENER=sales.us.example.com
```

INBOUND_CONNECT_TIMEOUT_listener_name

Purpose

To specify the time, in seconds, for the client to complete its connect request to the listener after the network connection had been established.

If the listener does not receive the client request in the time specified, then it terminates the connection. In addition, the listener logs the IP address of the client and an `ORA-12525:TNS: listener has not received client's request in time allowed` error message to the `listener.log` file.

To protect both the listener and the database server, Oracle recommends setting this parameter in combination with the [SQLNET.INBOUND_CONNECT_TIMEOUT](#) parameter in the `sqlnet.ora` file. When specifying values for these parameters, consider the following recommendations:

- Set both parameters to an initial low value.
- Set the value of the `INBOUND_CONNECT_TIMEOUT_listener_name` parameter to a lower value than the `SQLNET.INBOUND_CONNECT_TIMEOUT` parameter.

For example, you can set the `INBOUND_CONNECT_TIMEOUT_listener_name` parameter to 2 seconds and the `INBOUND_CONNECT_TIMEOUT` parameter to 3 seconds. If clients are unable to complete connections within the specified time due to system or network delays that are normal for the particular environment, then increment the time as needed.

Default

60 seconds

Example

```
INBOUND_CONNECT_TIMEOUT_listener=2
```

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about configuring these parameters

SAVE_CONFIG_ON_STOP_listener_name

Purpose

To specify whether run-time configuration changes are saved into the `listener.ora` file.

When you set the parameter to `true`, any parameters that were modified while the listener was running using the Listener Control utility [SET](#) command are saved to the `listener.ora` file when the [STOP](#) command is issued. When you set the parameter to `false`, the Listener Control utility does not save the run-time configuration changes to the `listener.ora` file.

Default

false

Values

true | false

Example

```
SAVE_CONFIG_ON_STOP_listener=true
```

SSL_CLIENT_AUTHENTICATION

Purpose

To specify whether a client is authenticated using the [Secure Sockets Layer \(SSL\)](#).

Default

true

Values

true | false

Usage Notes

The database server authenticates the client. Therefore, this value should be set to `false`. If this parameter is set to `true`, then the listener attempts to authenticate the client, which can result in a failure.

Example

```
SSL_CLIENT_AUTHENTICATION=false
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name**Purpose**

To subscribe to Oracle Notification Service (ONS) notifications for downed events. By default, the listener subscribes to the ONS node down event on startup, if ONS is available. This subscription enables the listener to remove the affected service when it receives node down event notification from ONS. The listener uses asynchronous subscription for the event notification. Alter this behavior setting `SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name=off` in `listener.ora`.

Default

on

Values

on | off

WALLET_LOCATION**Purpose**

To specify the location of wallets. Wallets are certificates, keys, and trustpoints processed by SSL that allow for secure connections.

Syntax

Oracle wallets on file system:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=file)
(METHOD_DATA=
(DIRECTORY=directory)
[(PKCS11=TRUE/FALSE)]))
```

Microsoft certificate store:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=mcs))
```

Oracle wallets in the Microsoft Windows registry:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=reg)
(METHOD_DATA=
(KEY=registry_key))
```

Entrust wallets:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=entr)
```

```
(METHOD_DATA=
 (PROFILE=file.epf)
 (INIFILE=file.ini))
```

Additional Parameters

The following additional parameters are available for `WALLET_LOCATION`:

- `SOURCE`: Type of storage for wallets and storage location.
- `METHOD`: Type of storage.
- `METHOD_DATA`: Storage location.
- `DIRECTORY`: Location of Oracle wallets on file system.
- `KEY`: Wallet type and location in the Microsoft Windows registry.
- `PROFILE`: Entrust profile file (.epf).
- `INIFILE`: Entrust initialization file (.ini).

Default

None

Usage Notes

- The key/value pair for Microsoft certificate store (MCS) omits the `METHOD_DATA` parameter because MCS does not use wallets. Instead, Oracle PKI (public key infrastructure) applications obtain certificates, trustpoints and private keys directly from the user's profile.
- If an Oracle wallet is stored in the Microsoft Windows registry and the wallet's key (`KEY`) is `SALESAPP`, then the storage location of the encrypted wallet is `HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12`. The storage location of the decrypted wallet is `HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO`.

Examples

Oracle wallets on file system:

```
WALLET_LOCATION=
 (SOURCE=
 (METHOD=file)
 (METHOD_DATA=
 (DIRECTORY=/etc/oracle/wallets/databases)))
```

Microsoft certificate store:

```
WALLET_LOCATION=
 (SOURCE=
 (METHOD=mcs))
```

Oracle Wallets in the Microsoft Windows registry:

```
WALLET_LOCATION=
 (SOURCE=
 (METHOD=REG)
 (METHOD_DATA=
 (KEY=SALESAPP)))
```

Entrust Wallets:

```
WALLET_LOCATION=
(SOURCE=
(METHOD=entr)
(METHOD_DATA=
(PROFILE=/etc/oracle/wallets/test.epf)
(INIFILE=/etc/oracle/wallets/test.ini)))
```

See Also: *Oracle Database Advanced Security Administrator's Guide*

ADR Diagnostic Parameters for Oracle Net Listener

Beginning with Oracle Database 11g, Oracle Database includes an advanced fault diagnosability infrastructure for preventing, detecting, diagnosing, and resolving problems. The problems are critical errors such as those caused by database code bugs, metadata corruption, and customer data corruption.

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error, such as traces and dumps, are immediately captured and tagged with the incident number. The data is then stored in the **Automatic Diagnostic Repository (ADR)**, a file-based repository outside the database.

This section includes the parameters used when **ADR** is enabled (when `DIAG_ADR_ENABLED_listener_name` is set to `on`). "[Non-ADR Diagnostic Parameters for Oracle Net Listener](#)" on page 7-14 includes those used when ADR is disabled (when `DIAG_ADR_ENABLED_listener_name` is set to `off`). Non-ADR parameters listed in the `listener.ora` file are ignored when ADR is enabled.

- [ADR_BASE_listener_name](#)
- [DIAG_ADR_ENABLED_listener_name](#)
- [LOGGING_listener_name](#)
- [TRACE_LEVEL_listener_name](#)
- [TRACE_TIMESTAMP_listener_name](#)

ADR_BASE_listener_name

Purpose

To specify the base directory in to which tracing and logging incidents are stored when ADR is enabled.

Default

The default is `ORACLE_BASE`, or `ORACLE_HOME/log` if `ORACLE_BASE` is not defined.

Values

Any valid directory path to a directory with write permission.

Example

```
ADR_BASE=/oracle/network/trace
```

DIAG_ADR_ENABLED_*listener_name*

Purpose

To indicate whether ADR tracing is enabled.

Usage Notes

When the `DIAG_ADR_ENABLED_listener_name` parameter is set to `on`, then ADR file tracing is used. When the `DIAG_ADR_ENABLED_listener_name` parameter is set to `off`, then non-ADR file tracing is used.

Default

`on`

Values

`on` | `off`

Example

```
DIAG_ADR_ENABLED=on
```

LOGGING_*listener_name*

Purpose

To turn logging on or off. This parameter is also applicable when non-ADR tracing is used.

Default

`on`

Values

`on` | `off`

Example

```
LOGGING_listener_name=on
```

TRACE_LEVEL_*listener_name*

Purpose

To turn listener tracing on, at a specific level, or off. This parameter is also applicable when non-ADR tracing is used.

Default

`off` | `0`

Values

- `off` or `0` for no trace output
- `user` or `4` for user trace information
- `admin` or `10` for administration trace information
- `support` or `16` for Oracle Support Services trace information

Example

```
TRACE_LEVEL_listener=admin
```

TRACE_TIMESTAMP_listener_name**Purpose**

To add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi1` to every trace event in the trace file for the listener. This parameter is used with the [TRACE_LEVEL_listener_name](#) parameter. This parameter is also applicable when non-ADR tracing is used.

Default

on

Values

- on | true
- off | false

Example

```
TRACE_TIMESTAMP_listener=true
```

Non-ADR Diagnostic Parameters for Oracle Net Listener

This section lists the parameters used when ADR is disabled (when `DIAG_ADR_ENABLED_listener_name` is set to `off`). "[ADR Diagnostic Parameters for Oracle Net Listener](#)" on page 7-12 includes the parameters when ADR is enabled.

Notes: The default value of [DIAG_ADR_ENABLED_listener_name](#) is on. Therefore, the `DIAG_ADR_ENABLED_listener_name` parameter *must* explicitly be set to `off` to use non-ADR tracing.

- [LOG_DIRECTORY_listener_name](#)
- [LOG_FILE_listener_name](#)
- [TRACE_FILELEN_listener_name](#)
- [TRACE_FILENO_listener_name](#)

LOG_DIRECTORY_listener_name**Purpose**

To specify the destination directory of the listener log file. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/log

Example

```
LOG_DIRECTORY_listener=/oracle/network/admin/log
```

LOG_FILE_listener_name

Purpose

To specify the name of the log file for the listener. Use this parameter when ADR is not enabled.

Default

listener.log

Example

```
LOG_FILE_listener=list.log
```

TRACE_DIRECTORY_listener_name

Purpose

To specify the destination directory of the listener trace file. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/trace

Example

```
TRACE_DIRECTORY_listener=/oracle/network/admin/trace
```

TRACE_FILE_listener_name

Purpose

To specify the name of the trace file for the listener. Use this parameter when ADR is not enabled.

Default

listener.trc

Example

```
TRACE_FILE_listener=list.trc
```

TRACE_FILELEN_listener_name

Purpose

To specify the size of the listener trace files in kilobytes (KB). When the size is met, the trace information is written to the next file. The number of files is specified using the [TRACE_FILENO_listener_name](#) parameter. Use this parameter when ADR is not enabled.

Default

Unlimited

Example

```
TRACE_FILELEN_listener=100
```

TRACE_FILENO_listener_name

Purpose

To specify the number of trace files for listener tracing. When this parameter is set along with the [TRACE_FILELEN_listener_name](#) parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is re-used, and so on.

The trace file names are distinguished from one another by their sequence number. For example, if the default trace file of `listener.trc` is used, and this parameter is set to 3, then the trace files would be named `listener1.trc`, `listener2.trc` and `listener3.trc`.

In addition, trace events in the trace files are preceded by the sequence number of the file. Use this parameter when ADR is not enabled.

Default

1

Example

```
TRACE_FILENO_listener=3
```

Class of Secure Transports Parameters

The class of secure transports (COST) parameters specify a list of transports that are considered secure for administration and registration of a particular listener. The COST parameters identify which transports are considered secure for that installation and whether the administration of a listener requires secure transports. Configuring these parameters is optional.

The following are the COST parameters:

- [DYNAMIC_REGISTRATION_listener_name](#)
- [SECURE_CONTROL_listener_name](#)
- [SECURE_REGISTER_listener_name](#)
- [SECURE_PROTOCOL_listener_name](#)

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about COST parameters and listener security

DYNAMIC_REGISTRATION_listener_name

Purpose

To enable or disable dynamic registration. When set to `on`, the listener accepts dynamic registration. When set to `off`, the listener refuses dynamic registration. Static registrations are not affected by this parameter.

Syntax

```
DYNAMIC_REGISTRATION_listener_name={on|off}
```

Default

The default value is `on`. Unless this parameter is explicitly set to `off`, all registration connections are accepted.

SECURE_CONTROL_*listener_name*

Purpose

To specify the transports on which control commands are to be serviced.

Syntax

```
SECURE_CONTROL_listener_name =
[({}transport1[,transport2, ...,transportn])]
```

In the preceding syntax, *transport1*, *transport2*, and *transportn* are valid, installed transport protocol names.

Configuration

If the `SECURE_CONTROL_listener_name` parameter is configured with the list of transport names, then the control commands will be serviced only if the connection is one of the listed transports. Connections arriving by other transport protocols are refused. For example:

```
SECURE_CONTROL_listener1 = (TCPS, IPC)
```

In the preceding example, administration requests are accepted only on TCPS and IPC transports.

If no values are entered for this parameter, then the listener accepts any connection on any endpoint.

Example

```
LISTENER1=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
    (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522)) )
  SECURE_CONTROL_LISTENER1=tcps
```

SECURE_REGISTER_*listener_name*

Purpose

To specify the transports on which registration requests are to be accepted.

Syntax

```
SECURE_REGISTER_listener_name =
[({}transport1[,transport2, ...,transportn])]
```

In the preceding example, *transport1*, *transport2*, and *transportn* are valid, installed transport protocol names.

Configuration

If the `SECURE_REGISTER_listener_name` parameter is configured with the list of transport names, then only the connections arriving on the specified transports will be able to register the service with the listener. Connections arriving by other transport protocols are refused. For example:

```
SECURE_REGISTER_listener1 = (TCPS, IPC)
```

In the preceding example, registration requests are accepted only on TCPS and IPC transports.

If no values are entered for this parameter, then the listener accepts registration requests from any transport.

If this parameter and [SECURE_CONTROL_listener_name](#) are configured, then they override the [SECURE_PROTOCOL_listener_name](#) parameter.

Example

```
LISTENER1=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
    (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))    )
  SECURE_REGISTER_LISTENER1=tcps
```

SECURE_PROTOCOL_listener_name

Purpose

To specify the transports on which administration and registration requests are to be accepted.

Syntax

```
SECURE_PROTOCOL_listener_name =
[ (|transport1[,transport2, ...,transportn])
```

In the preceding example, *transport1*, *transport2*, and *transportn* are valid, installed transport protocol names.

Configuration

If this parameter is configured with the list of transport names, then the control commands and service registration can happen only if the connection belongs to the list of transports configured.

If this parameter is not present and neither [SECURE_CONTROL_listener_name](#) or [SECURE_REGISTER_listener_name](#) are configured, then all supported transports accept control and registration requests.

If the [SECURE_CONTROL_listener_name](#) and [SECURE_REGISTER_listener_name](#) parameters are configured, then they override the [SECURE_PROTOCOL_listener_name](#) parameter.

Example

```
LISTENER1=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
    (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))    )
  SECURE_PROTOCOL_LISTENER1=tcps
```

Using COST Parameters in Combination

COST parameters can also be used in combination to further control which transports accept service registration and control commands.

In [Example 7-2](#), control commands are accepted only on the IPC channel and the TCPS transport, and service registrations are accepted only on an IPC channel.

Example 7-2 Combining COST Parameters

```
LISTENER1=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
    (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))
  )
  SECURE_CONTROL_LISTENER1=(tcps,ipc)
  SECURE_REGISTER_LISTENER1=ipc
```

In [Example 7-3](#), control commands are accepted only on the TCPS transport and service registrations are accepted only on the IPC channel.

Example 7-3 Combining COST Parameters

```
LISTENER1=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
    (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))
  )
  SECURE_CONTROL_LISTENER1=tcps
  SECURE_PROTOCOL_LISTENER1=ipc
```

Oracle Connection Manager Parameters (cman.ora)

This chapter provides a complete listing of the `cman.ora` file configuration parameters.

This chapter contains the following topics:

- [Overview of Oracle Connection Manager Configuration File](#)
- [Oracle Connection Manager Parameters](#)
- [ADR Diagnostic Parameters for Oracle Connection Manager](#)
- [Non-ADR Diagnostic Parameters for Oracle Connection Manager](#)

Overview of Oracle Connection Manager Configuration File

Oracle Connection Manager configuration information, stored in the `cman.ora` file, consists of the following elements:

- Protocol address of the Oracle Connection Manager listener
- Access control parameters
- Performance parameters

By default, the `cman.ora` file is located in the `ORACLE_HOME/network/admin` directory. The `cman.ora` file can also be stored in the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value.
- On Linux and UNIX operating systems, the global configuration directory. For example, on the Solaris Operating System, this directory is `/var/opt/oracle`.

See Also: Oracle operating system-specific documentation

[Example 8-1](#) shows an sample of a `cman.ora` file.

Example 8-1 Sample cman.ora File

```
CMAN=
(CONFIGURATION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1521))
  (RULE_LIST=
    (RULE=(SRC=192.168.2.32/27)(DST=sales-server)(SRV=*)(ACT=accept))
    (ACTION_LIST=(AUT=on)(MCT=120)(MIT=30)))
    (RULE=(SRC=foo)(DST=hr-server)(SRV=cmon)(ACT=accept)))
(PARAMETER_LIST=
```

```
(MAX_GATEWAY_PROCESSES=8)
(MIN_GATEWAY_PROCESSES=3)
(DIAG_ADR_ENABLED=ON)
(ADR_BASE=/oracle/log))
```

The `cman.ora` configuration file consists of the following sections:

- **Listening address:** Preceded by `ADDRESS=`, this section contains information pertinent to the listener. The `ADDRESS` parameter is required.
- **Rule list:** Preceded by `RULE_LIST=`, this section contains rule information. The `RULE` parameter is listed in the rule list section of the file. The `RULE` parameter is required.
- **Parameter list:** Preceded by `PARAMETER_LIST=`, this section contains all other parameters including those listed in "[ADR Diagnostic Parameters for Oracle Connection Manager](#)" on page 8-10, and "[Non-ADR Diagnostic Parameters for Oracle Connection Manager](#)" on page 8-12.

The following parameters are allowed in the parameter list section of the `cman.ora` file. The default values are bold. To override the default setting for a parameter, enter the parameter and a nondefault value.

```
ASO_AUTHENTICATION_FILTER={off | on}
CONNECTION_STATISTICS={no | yes}
EVENT_GROUP={init_and_term | memory_ops | conn_hdlg | proc_mgmt
| reg_and_load | wake_up | timer | cmd_proc | relay}
IDLE_TIMEOUT=0 or greater
INBOUND_CONNECT_TIMEOUT=0 or greater
LOG_DIRECTORY=log_directory
LOG_LEVEL={off | user | admin | support}
MAX_CMCTL_SESSIONS= Any positive number
MAX_CONNECTIONS=[1 to 1024]
MAX_GATEWAY_PROCESSES= Any number greater than the minimum number of
gateway processes up to 64
MIN_GATEWAY_PROCESSES= Any positive number less than or equal to 64. Must be
less than or equal to the maximum number of gateway processes.
OUTBOUND_CONNECT_TIMEOUT=0 or greater
PASSWORD_instance_name= Value is the encrypted instance password, if one has
been set.
SESSION_TIMEOUT=0 or greater
TRACE_DIRECTORY=trace_directory
TRACE_FILELEN= Any positive number
TRACE_FILENO= Any positive number
TRACE_LEVEL={off | user | admin | support}
TRACE_TIMESTAMP={off | on}
```

Notes:

- INBOUND_CONNECT_TIMEOUT default value is 60.
- LOG_DIRECTORY default value is ORACLE_HOME/network/log.
- MAX_CMCTL_SESSIONS default value is 4.
- MAX_CONNECTIONS default value is 256.
- MAX_GATEWAY_PROCESSES default value is 16.
- MIN_GATEWAY_PROCESSES default value is 2.
- PASSWORD_instance_name default value is no value.
- TRACE_DIRECTORY default value is ORACLE_HOME/network/trace.
- TRACE_FILELEN default value is 0.
- TRACE_FILENO default value is 0.
- You cannot add the parameter PASSWORD_instance_name directly to the cman.ora file. The parameter is added when you issue the command SAVE_PASSWD.

Example 8-2 shows the parameter list section of a cman.ora file.

Example 8-2 Parameter List Section of a cman.ora File

```
(PARAMETER_LIST=
  (ASO_AUTHENTICATION_FILTER=ON)
  (CONNECTION_STATISTICS=NO)
  (EVENT_GROUP=INIT_AND_TERM, MEMORY_OPS, PROCESS_MGMT)
  (IDLE_TIMEOUT=30)
  (INBOUND_CONNECT_TIMEOUT=30)
  (LOG_DIRECTORY=/home/user/network/admin/log)
  (LOG_LEVEL=SUPPORT)
  (MAX_CMCTL_SESSIONS=6)
  (MAX_CONNECTIONS=512)
  (MAX_GATEWAY_PROCESSES=10)
  (MIN_GATEWAY_PROCESSES=4)
  (OUTBOUND_CONNECT_TIMEOUT=30)
  (SESSION_TIMEOUT=60)
  (TRACE_DIRECTORY=/home/user/network/admin/trace)
  (TRACE_FILELEN=100)
  (TRACE_FILENO=2)
  (TRACE_LEVEL=SUPPORT)
  (TRACE_TIMESTAMP=ON))
```

Oracle Connection Manager Parameters

This section lists and describes the following cman.ora file parameters:

- ADDRESS
- ASO_AUTHENTICATION_FILTER
- CONNECTION_STATISTICS
- EVENT_GROUP

- IDLE_TIMEOUT
- INBOUND_CONNECT_TIMEOUT
- LOG_DIRECTORY
- LOG_LEVEL
- MAX_CMCTL_SESSIONS
- MAX_CONNECTIONS
- MAX_GATEWAY_PROCESSES
- MIN_GATEWAY_PROCESSES
- OUTBOUND_CONNECT_TIMEOUT
- PASSWORD_instance_name
- RULE
- SESSION_TIMEOUT
- TRACE_FILE
- TRACE_FILELEN
- TRACE_FILENO
- TRACE_LEVEL
- TRACE_TIMESTAMP

ADDRESS

Purpose

To specify the protocol address of Oracle Connection Manager.

Syntax

```
(ADDRESS=(PROTOCOL=protocol) (HOST=host_name) (PORT=port_number))
```

Example

```
(ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
```

ASO_AUTHENTICATION_FILTER

Purpose

To specify whether Oracle Advanced Security authentication settings must be used by the client. The global setting can be overridden by a rule-level setting in ACTION_LIST.

Values

- `on` to instruct Oracle Connection Manager to reject connect requests that are not using Secure Network Services (SNS). SNS is part of the Oracle Advanced Security.
- `off` (default) to instruct Oracle Connection Manager not to check for SNS between the client and server

CONNECTION_STATISTICS

Purpose

To specify whether the `SHOW_CONNECTIONS` command displays connection statistics. The global setting can be overridden by a rule-level setting in `ACTION_LIST`.

Values

- `yes` to display statistics
- `no` (default) to not display statistics

EVENT_GROUP

Purpose

To specify which event groups are logged. Multiple events may be designated using a comma-delimited list.

Values

- `alert` for alert notifications
- `cmd_proc` for command processing
- `conn_hdlg` for connection handling
- `init_and_term` for initialization and termination
- `memory_ops` for memory operations
- `proc_mgmt` for process management
- `reg_and_load` for registration and load update
- `relay` for events associated with connection control blocks
- `timer` for gateway timeouts
- `wake_up` for events related to CMADMIN wake-up queue

Note: The event group `ALERT` cannot be turned off.

IDLE_TIMEOUT

Purpose

To specify the amount of time that an established connection can remain active without transmitting data. The global setting can be overridden by a rule-level setting in `ACTION_LIST`.

Values

- `0` (default) to disable the timeout.
- any number greater than `0` to enable the timeout. The number equals the timeout period in seconds

INBOUND_CONNECT_TIMEOUT

Purpose

To specify how long the Oracle Connection Manager listener waits for a valid connection from a client or another instance of Oracle Connection Manager.

Values

- 60 (default) to disable the timeout.
- any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

LOG_DIRECTORY

Purpose

To specify the directory for the Oracle Connection Manager log files.

LOG_LEVEL

Purpose

To specify the level for log messages.

Values

- `off` for no logging. This is the default.
- `user` for user-induced errors log information.
- `admin` for administration, such as installation-specific, log information.
- `support` for Oracle Support Services information.

MAX_CMCTL_SESSIONS

Purpose

To specify the maximum number of concurrent local or remote sessions of the Oracle Connection Manager control utility allowable for a given instance. One of these sessions must be a local session.

Values

Any number of sessions can be designated.

MAX_CONNECTIONS

Purpose

To specify the maximum number of connection slots that a gateway process can handle.

Values

This parameter accepts a range of 1 to 1024.

MAX_GATEWAY_PROCESSES

Purpose

To specify the maximum number of gateway processes that an instance of Oracle Connection Manager supports.

Values

The number designated must be greater than the minimum number of gateway processes. The maximum is 64.

MIN_GATEWAY_PROCESSES

Purpose

To specify the minimum number of gateway processes that an instance of Oracle Connection Manager must support.

Values

Any number of sessions can be designated up to 64.

OUTBOUND_CONNECT_TIMEOUT

Purpose

To specify the length of time that the Oracle Connection Manager instance waits for a valid connection to be established with the database server or with another Oracle Connection Manager instance.

Values

- 60 (default) to disable the timeout.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

PASSWORD_*instance_name*

Purpose

To specify the encrypted instance password, if one has been set.

RULE

Purpose

To specify an access control rule list to filter incoming connections. A rule list specifies which connections are accepted, rejected, or dropped.

Values

This parameter is listed in the rule list section of the `cman.ora` file preceded by `RULE_LIST=`.

Syntax

(`RULE_LIST=`

```
(RULE=  
  (SRC=host)  
  (DST=host)  
  (SRV=service_name)  
  (ACT={accept|reject|drop})  
  (ACTION_LIST=AUT={on|off})  
  ((CONN_STATS={yes|no}) (MCT=time) (MIT=time) (MOCT=time))  
(RULE= ...))
```

Additional Parameters

The `RULE` parameter filters a connection or group of connections using the following parameters:

`SRC`: Specify the source host name or IP address of the client.

`DST`: Specify the destination server host name or IP address of the database server.

`SRV`: Specify database **service name** of Oracle Database obtained from the `SERVICE_NAME` parameter in the initialization parameter file.

`ACT`: Specify `accept` to accept incoming requests, `reject` to reject incoming requests, or `drop` to reject incoming requests without sending an error message.

`ACTION_LIST`: Specify rule-level parameter settings for some parameters. These parameters are as follows:

- `AUT`: Oracle Advanced Security authentication on client side
- `CONN_STATS`: Log input and output statistics
- `MCT`: Maximum connect time
- `MIT`: Maximum idle timeout
- `MOCT`: Maximum outbound connect time

Rule-level parameters override their global counterparts.

Usage Notes

- If no rules are specified, then all connections are rejected.
- The source and destination can be a host name, IP address, or subnet mask.
- You must enter at least one rule for client connections and one rule for CMCTL connections. Omitting one or the other results in the rejection of all connections for the rule type omitted. The last rule in the example that follows is a CMCTL rule.
- Oracle Connection Manager does not support wildcards for partial IP addresses. If you use a wildcard, then use it in place of a full IP address. The IP address of the client may, for example, be `(SRC=*)`.
- Oracle Connection Manager supports only the `/nn` notation for subnet addresses. In the first rule in the example, `/27` represents a subnet mask that comprises 27 left-most bits.

Example

```
(RULE_LIST=  
  (RULE=  
    (SRC=client1-pc)  
    (DST=sales-server)  
    (SRV=sales.us.example.com)  
    (ACT=reject))  
  (RULE=
```

```

(SRC=192.168.2.45)
(DST=192.168.2.200)
(SRV=db1)
(ACT=accept))
(RULE=
(SRC=foo)
(DST=foobar)
(SRV=cmon)
(ACT=accept)))

```

SESSION_TIMEOUT

Purpose

To specify the maximum time allowed for a user session. The global setting can be overridden by a rule-level setting in `ACTION_LIST`. This parameter accepts the following values:

Values

- 0 (default) to disable the timeout.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

TRACE_FILE

Purpose

To specify the directory for Oracle Connection Manager trace files.

TRACE_FILELEN

Purpose

To specify the size of the trace file in KB. When the size is reached, the trace information is written to the next file. The number of files is specified with the `TRACE_FILENO` parameter.

TRACE_FILENO

Purpose

To specify the number of trace files. When this parameter is set along with the `TRACE_FILELEN` parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is reused, and so on.

TRACE_LEVEL

Purpose

To specify the level for trace messages.

Values

- `off` for no tracing. This is the default.

- `user` for user-induced errors trace information.
- `admin` for administration, such as installation-specific, trace information.
- `support` for Oracle Support Services information.

TRACE_TIMESTAMP

Purpose

To specify the use of a timestamp for the tracing logs. If the `TRACING` parameter is enabled, then a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi1` for every trace event in the trace file.

Values

- `off` for no timestamp is included in the file.
- `on` for timestamp to be included in the file.

ADR Diagnostic Parameters for Oracle Connection Manager

Beginning with Oracle Database 11g, Oracle Database includes an advanced fault diagnosability infrastructure for preventing, detecting, diagnosing, and resolving problems. The problems are critical errors such as those caused by database code bugs, metadata corruption, and customer data corruption.

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error, such as traces and dumps, are immediately captured and tagged with the incident number. The data is then stored in the [Automatic Diagnostic Repository \(ADR\)](#), a file-based repository outside the database.

This section describes the parameters used when [ADR](#) is enabled (when `DIAG_ADR_ENABLED` is set to `on`). "[Non-ADR Diagnostic Parameters for Oracle Connection Manager](#)" on page 8-12 describes the parameters used when ADR is disabled (when `DIAG_ADR_ENABLED` is set to `off`). Non-ADR parameters listed in the `cman.ora` file are ignored when ADR is enabled.

- [ADR_BASE](#)
- [DIAG_ADR_ENABLED](#)
- [LOG_LEVEL](#)
- [TRACE_LEVEL](#)
- [TRACE_TIMESTAMP](#)

ADR_BASE

Purpose

To specify the base directory into which tracing and logging incidents are stored when ADR is enabled.

Default

The default is `ORACLE_BASE`, or `ORACLE_HOME/log` if `ORACLE_BASE` is not defined.

Values

Any valid directory path to a directory with write permission.

Example

```
ADR_BASE=/oracle/network/trace
```

DIAG_ADR_ENABLED**Purpose**

To indicate whether ADR tracing is enabled.

Usage

When the `DIAG_ADR_ENABLED` parameter is set to `OFF`, non-ADR file tracing is used.

Values

on | off

Example

```
DIAG_ADR_ENABLED=on
```

LOG_LEVEL**Purpose**

To specify the level of logging performed by Oracle Connection Manager. This parameter is also applicable when non-ADR logging is used.

The following log files are used with Oracle Connection Manager:

- `instance-name_pid.log` for the listener
- `instance-name_cmadmin_pid.log` for CMADMIN
- `instance-name_cmgw_pid.log` for the gateway processes

The log files are located in the `ORACLE_HOME/network/log` directory.

Default

off or 0

Values

- `off` or `0` for no log output
- `user` or `4` for user log information
- `admin` or `10` for administration log information
- `support` or `16` for Oracle Support Services log information

Example

```
LOG_LEVEL=admin
```

TRACE_LEVEL**Purpose**

To specify the trace level for the Oracle Connection Manager instance. This parameter is also applicable when non-ADR tracing is used.

The following trace files are used with Oracle Connection Manager:

- `instance-name_pid.trc` for the listener
- `instance-name_cmadmin_pid.trc` for CMADMIN
- `instance-name_cmgw_pid.trc` for the gateway processes

The log files are located in the `ORACLE_HOME/network/log` directory.

Default

`off`

Values

- `off` for no trace output
- `user` for user trace information
- `admin` for administration trace information
- `support` for Oracle Support Services trace information

Example

```
TRACE_LEVEL=admin
```

TRACE_TIMESTAMP

Purpose

To add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi1` to every trace event in the trace file for the listener. This parameter is used with the [TRACE_LEVEL](#) parameter. This parameter is also applicable when non-ADR tracing is used.

Default

`on`

Values

- `on` or `true`
- `off` or `false`

Example

```
TRACE_TIMESTAMP=true
```

Non-ADR Diagnostic Parameters for Oracle Connection Manager

This section lists the parameters used when ADR is disabled (when `DIAG_ADR_ENABLED` is set to `off`):

Notes: The default value of [DIAG_ADR_ENABLED](#) is `on`. Therefore, the `DIAG_ADR_ENABLED` parameter *must* explicitly be set to `off` in order for non-ADR tracing to be used.

- [LOG_DIRECTORY](#)
- [TRACE_DIRECTORY](#)
- [TRACE_FILELEN](#)

- [TRACE_FILENO](#)

LOG_DIRECTORY

Purpose

To specify the location of Oracle Connection Manager log files. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/log

Values

Any valid directory path to a directory with write permission.

Example

```
LOG_DIRECTORY=/oracle/network/log
```

TRACE_DIRECTORY

Purpose

To specify the location of the Oracle Connection Manager trace files. Use this parameter when ADR is not enabled.

Default

ORACLE_HOME/network/trace

Values

Any valid directory path to a directory with write permission.

Example

```
TRACE_DIRECTORY=/oracle/network/admin/trace
```

TRACE_FILELEN

Purpose

To specify the size, in KB, of the trace file. When the size is met, the trace information is written to the next file. The number of files is specified with the [TRACE_FILENO](#) parameter. Any size can be designated. Use this parameter when ADR is not enabled.

Default

Unlimited

Example

```
TRACE_FILELEN=100
```

TRACE_FILENO

Purpose

To specify the number of trace files for Oracle Connection Manager tracing. When this parameter is set along with the [TRACE_FILELEN](#) parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is reused, and so on. Any number of files can be designated.

The trace file names are distinguished from one another by their sequence number. For example, if this parameter is set to 3, then the gateway trace files would be named *instance-name_cm gw1_pid.trc*, *instance_name_cm gw2_pid.trc* and *instance_name_cm gw3_pid.trc*.

In addition, trace events in the trace files are preceded by the sequence number of the file. Use this parameter when ADR is *not* enabled.

Default

1

Example

```
TRACE_FILENO=3
```

Directory Usage Parameters (ldap.ora)

This chapter provides a complete listing of the `ldap.ora` file configuration parameters.

This chapter contains the following topics:

- [Overview of Directory Server Usage File](#)
- [Directory Usage Parameters](#)

Overview of Directory Server Usage File

The `ldap.ora` file contains directory usage configuration parameters created by [Oracle Internet Directory](#) Configuration Assistant or [Oracle Net Configuration Assistant](#). Do not modify these parameters or their settings.

When created with Oracle Internet Directory Configuration Assistant, `ldap.ora` is located in the `ORACLE_HOME/ldap/admin` directory. When created with Oracle Net Configuration Assistant, the `ldap.ora` file is located in the `ORACLE_HOME/network/admin` directory. The `ldap.ora` file can also be stored in the directory specified by the `LDAP_ADMIN` or `TNS_ADMIN` environment variable.

Directory Usage Parameters

This section lists and describes the following `ldap.ora` file configuration parameters:

- [DIRECTORY_SERVERS](#)
- [DIRECTORY_SERVER_TYPE](#)
- [DEFAULT_ADMIN_CONTEXT](#)

DIRECTORY_SERVERS

Purpose

To list the host names and port number of the primary and alternate LDAP [directory servers](#).

Values

`host:port[:sslport]`

Example

```
DIRECTORY_SERVERS=(ldap-server:389, raffles:400:636)
```

DIRECTORY_SERVER_TYPE

Purpose

To specify the type of directory server that is being used.

Values

- `oid` for Oracle Internet Directory
- `ad` for Microsoft Active Directory

Example

```
DIRECTORY_SERVER_TYPE=oid
```

DEFAULT_ADMIN_CONTEXT

Purpose

To specify the default directory entry that contains an **Oracle Context** from which connect identifiers can be created, modified, or looked up.

Values

Valid **distinguished name (DN)**

Example

```
DEFAULT_ADMIN_CONTEXT="o=OracleSoftware,c=US"
```

Part III

Appendixes

This part contains the following appendixes:

- [Appendix A, "Features Not Supported in this Release"](#)
- [Appendix B, "Upgrade Considerations for Oracle Net Services"](#)
- [Appendix C, "LDAP Schema for Oracle Net Services"](#)

Features Not Supported in this Release

This appendix describes features no longer supported by Oracle Net Services.

This appendix contains the following topics:

- [Overview of Unsupported Features](#)
- [Unsupported Parameters](#)
- [Unsupported Control Utility Commands](#)

Overview of Unsupported Features

The following section describe the features and the configuration file that are no longer being supported in Oracle Database. This is based on an effort to streamline configuration and use of Oracle Database.

Oracle Names

Oracle Names is not supported as a [naming method](#) in Oracle Database 11g. You must migrate to [directory naming](#).

See Also: *Oracle Net Services Administrator's Guide* for additional information about migrating to directory naming

Unsupported Parameters

[Table A-1](#) describes the networking parameters no longer supported since Oracle9i.

Table A-1 *Unsupported Networking Parameters*

File	Parameter	Description	Last Supported Release
names.ora	All parameters	Oracle Names is no longer supported.	9.2
sqlnet.ora	SQLNET.CRYPTO_SEED	This parameter was used to seed a random number generator for Oracle Advanced Security. Starting with Oracle Database 10g, Oracle Advanced Security uses a random number generator that does not to require a user-supplied seed value.	9.2

Unsupported Control Utility Commands

[Table A-2](#) describes the control utility commands not supported since Oracle9i.

Table A-2 *Unsupported Network Control Utility Commands*

Control Utility	Commands	Description	Last Supported Release
Oracle Names Control Utility	All commands	Oracle Names is no longer supported.	9.2

Upgrade Considerations for Oracle Net Services

This appendix describes coexistence and upgrade issues for Oracle Net Services. This appendix covers the following topics:

- [Anonymous Access to Oracle Internet Directory](#)
- [Migrating a Non-default Listener to Oracle Database 11g Release 2 \(11.2\)](#)
- [Removing the Listener Password](#)
- [Client and Database Coexistence Issues](#)

Anonymous Access to Oracle Internet Directory

Typical users of directory naming (LDAP) require anonymous access to the Oracle Internet Directory for name lookup. If your Oracle Internet Directory software is upgraded to Oracle Internet Directory 11g, then it should be noted that the default setting for Oracle Internet Directory has changed. The default now is to disallow anonymous access to the directory. The directory administrator will need to configure the directory to enable anonymous binds after upgrading the directory to release 11g. In addition, the way anonymous binds are configured in Oracle Internet Directory has changed between Oracle Database 10g and Oracle Database 11g.

See Also: *Oracle Internet Directory Administrator's Guide* for additional information about anonymous binds

Migrating a Non-default Listener to Oracle Database 11g Release 2 (11.2)

When migrating Oracle Database 11g Release 1 (11.1.0.7) to Oracle Database 11g Release 2 (11.2) in an Oracle Real Application Clusters environment, non-default listeners are not migrated by Oracle Net Services Configuration Assistant. To migrate the non-default listeners, use the following procedure:

1. Use Oracle Database 11g Release 1 (11.1.0.7) Oracle Net Services Configuration Assistant to remove the non-default listeners.
2. Use Oracle Database 11g Release 2 (11.2) Oracle Net Services Configuration Assistant to create the non-default listeners.

Removing the Listener Password

In Oracle Database 11g Release 2 (11.2), the password feature is being deprecated. This does not cause a loss of security because authentication is enforced through local

operating system authentication. To migrate a listener that has a set password, do the following:

1. Remove all `PASSWORDS_listener_name` entries from the `listener.ora` file.
2. Reload the listener using the following command:

```
lsnrctl reload listener_name
```

If remote administration of a listener is required, then use one of the following methods to connect to and administer the listener.

- Connect to the host where listener is running using SSH or other secure method, and then perform local administration. Local administration is enforced by the operating system authentication.
- Use Oracle Enterprise Manager to administer the listener. Oracle Enterprise Manager uses HTTPS, which ensures security.

See Also: *Oracle Database Net Services Administrator's Guide* for more information about managing the listener

Client and Database Coexistence Issues

Clients and database servers require compatible releases of Oracle Net Services. For example, an Oracle9i client requires an installation of Oracle Net Services, and an Oracle9i database requires an installation of Oracle Net Services with the Oracle Net Listener.

Consider the following client-to-database connection issues before you decide if upgrading is appropriate for your environment:

- [Oracle9i Database Connections](#)
- [Oracle Names](#)

Oracle9i Database Connections

Connect descriptors, created for connections to an Oracle9i database, identify a database by its service name with the `SERVICE_NAME` parameter.

A connect descriptor to an Oracle9i or Oracle8 database uses the parameter `SERVICE_NAME`, as shown in the following example:

```
sales=
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521))
  (CONNECT_DATA=
    (SERVICE_NAME=sales.us.example.com)))
```

Connect descriptors that are currently configured with the `SID` parameter can remain. However, to take advantage of new features, such as client load balancing and connect-time failover, Oracle recommends replacing `SID` with `SERVICE_NAME`.

To modify a connect descriptor to use `SERVICE_NAME`, use the Oracle Net Manager's compatibility mode.

See Also: *Oracle Database Net Services Administrator's Guide* for additional information about database identification by `SERVICE_NAME` rather than `SID`

Oracle Names

Oracle Names is not supported as a centralized naming method. Because no new enhancements are being added to Oracle Names, consider using directory naming or upgrading an existing Oracle Names configuration to directory naming, as described in the *Oracle Database Net Services Administrator's Guide*.

LDAP Schema for Oracle Net Services

This appendix describes the **Oracle schema** object classes and **attributes** defined in the **directory server** for Oracle Net Services objects. It does not describe object classes and attributes reserved for future functionality or used by other Oracle products.

This appendix contains the following topics:

- [Structural Object Classes](#)
- [Attributes](#)

Structural Object Classes

The Oracle schema supports the structural object classes for Oracle Net **directory naming** lookups. [Table C-1](#) lists the structural object classes for Oracle Connection Manager.

Table C-1 Oracle Net Structural Object Classes

Object Class	Attributes	Description
orclDBServer	<ul style="list-style-type: none"> ▪ orclNetDescName ▪ orclVersion 	Defines the attributes for database service entries.
orclNetService	<ul style="list-style-type: none"> ▪ orclNetDescName ▪ orclVersion 	Defines the attributes for net service name entries.
orclNetServiceAlias	<ul style="list-style-type: none"> ▪ orclNetDescName ▪ orclVersion 	Defines the attributes for net service alias entries.
orclNetDescription	<ul style="list-style-type: none"> ▪ orclNetAddrList ▪ orclNetInstanceName ▪ orclNetConnParamList ▪ orclNetFailover ▪ orclNetLoadBalance ▪ orclNetSdu ▪ orclNetServiceName ▪ orclNetSourceRoute ▪ orclSid ▪ orclVersion 	Specifies a connect descriptor containing the protocol address of the listener and the connect information to the service
orclNetDescriptionList	<ul style="list-style-type: none"> ▪ orclNetDescList ▪ orclVersion 	Specifies a list of connect descriptors.

Table C-1 (Cont.) Oracle Net Structural Object Classes

Object Class	Attributes	Description
orclNetAddress	<ul style="list-style-type: none"> ▪ orclNetAddressString ▪ orclNetProtocol ▪ orclVersion 	Specifies a listener protocol address.
orclNetAddressList	<ul style="list-style-type: none"> ▪ orclNetAddrList ▪ orclNetFailover ▪ orclNetLoadBalance ▪ orclNetSourceRoute ▪ orclVersion 	Specifies a list of protocol addresses.
orclNetDescriptionAux1	<ul style="list-style-type: none"> ▪ orclNetSendBufSize ▪ orclNetReceiveBufSize ▪ orclNetFailoverModeString ▪ orclNetInstanceRole 	Specifies auxiliary object class to add attributes to an orclNetDescription entry.
orclNetAddressAux1	<ul style="list-style-type: none"> ▪ orclNetHostname 	Specifies an auxiliary object class to add attributes to an orclNetAddress entry.

Attributes

[Table C-2](#) lists the attributes used for the object classes. This list is subject to change.

Table C-2 LDAP Schema Attributes for Oracle Net Services

Attribute	Description
orclCommonContextMap	Allows the mapping of more than one default oracleContext in the directory server.
orclNetAddrList	Identifies one or more listener protocol addresses.
orclNetAddressString	Defines a listener protocol address.
orclNetConnParamList	Placeholder for connect data parameters.
orclNetDescList	Identifies one or more connect descriptors.
orclNetDescName	Identifies a connect descriptor or a list of connect descriptors.
orclNetFailover	Turns connect-time failover on for a protocol address list.
orclNetFailoverModeString	Instructs Oracle Net to fail over to a different listener if the first listener fails during run-time. Depending on the configuration, session or any SELECT statements that were in progress are automatically failed over.
orclNetHostname	Specifies the host name.
orclNetInstanceName	Specifies the instance name to access.
orclNetInstanceRole	Specifies a connection to the primary or secondary instance of an Oracle RAC configuration.
orclNetLoadBalance	Turns client load balancing on for a protocol address list.
orclNetProtocol	Identifies the protocol used in the orclAddressString attribute.
orclNetReceiveBufSize	Specifies the buffer space limit for receive operations of sessions.
orclNetSdu	Specifies the session data unit (SDU) size.
orclNetSendBufSize	Specifies the buffer space limit for send operations of sessions.

Table C-2 (Cont.) LDAP Schema Attributes for Oracle Net Services

Attribute	Description
orclNetServiceName	Specifies the database service name in the CONNECT_DATA portion.
orclNetSourceRoute	Instructs Oracle Net to use each address in order until the destination is reached.
orclSid	Specifies the Oracle System Identifier (SID) in the CONNECT_DATA portion of a connection descriptor.
orclVersion	Specifies the version of software used to create the entry.

Glossary

access control

A feature of Oracle Connection Manager that sets rules for denying or allowing certain clients to access designated servers.

access control list (ACL)

The group of access directives that you define. The directives grant levels of access to specific data for specific clients or groups of clients.

ACL

See [access control list \(ACL\)](#).

ADR

See [Automatic Diagnostic Repository \(ADR\)](#)

address

See [protocol address](#).

alias

An alternative name for a network object in an Oracle Names server. An alias stores the name of the object it is referencing. When a client requests a lookup of an alias, Oracle completes the lookup as if it is the referenced object.

application gateway

A host computer that runs the [Oracle Net Firewall Proxy](#). An application gateway looks and acts like a real server from the client's point of view, and a real client from the server's point of view. An application gateway sits between the Internet and company's internal network and provides middleman services (or proxy services) to users on either side.

ASCII character set

American Standard Code for Information Interchange character set, a convention for representing alphanumeric information using digital data. The collation sequence used by most computers with the exception of IBM and IBM-compatible computers.

attribute

A piece of information that describes some aspect of a directory entry. An entry comprises a set of attributes, each of which belongs to an [object class](#). Moreover, each attribute has both a type, which describes the kind of information in the attribute, and a value which contains the actual data.

authentication method

A security method that enables you to have confidence in the identity of users, clients, and servers in distributed environments. Network authentication methods can also provide the benefit of single sign-on for users. The following authentication methods are supported in Oracle Database, depending on whether [Oracle Advanced Security](#) is installed:

- RADIUS
- Kerberos
- [SSL](#)
- [Microsoft Windows NT native authentication](#)

Automatic Diagnostic Repository (ADR)

A systemwide tracing and logging central repository. The repository is a file-based hierarchical data store for depositing diagnostic information, including network tracing and logging information.

cache

Memory that stores recently-accessed data so that subsequent requests to access the same data can be processed quickly.

CIDR (Classless Inter-Domain Routing)**client**

A user, software application, or computer that requests the services, data, or processing from another application or computer. The client is the user process.

client load balancing

Load balancing, whereby if more than one listener services a single database, a client can randomly choose between the listeners for its connect requests. This randomization enables all listeners to share the burden of servicing incoming connect requests.

client profile

The properties of a client, which may include the preferred order of [naming methods](#), client and server [logging](#) and [tracing](#), the domain from which to request names, and other client options for [Oracle Advanced Security](#).

client/server architecture

Software architecture based on a separation of processing between two CPUs. One CPU acts as the client in the transaction, requesting and receiving services. The other acts as the server that provides service for the requests.

cman.ora file

A configuration file that specifies protocol addresses for incoming requests and administrative commands, as well as Oracle Connection Manager parameters and [access control](#) rules.

CMADMIN (Connection Manager Administration)

An [Oracle Connection Manager](#) process that monitors the health of the listener and Oracle Connection Manager gateway processes, shutting down and starting processes

as needed. CMADMIN registers information about gateway processes with the listener and processes commands executed with the Oracle Connection Manager Control utility.

CMGW (Connection Manager gateway)

An **Oracle Connection Manager** process that receives client connections screened and forwarded by the listener located at the Oracle Connection Manager instance. The gateway process forwards the requests to the database server. In addition, it can multiplex or process multiple client connections through a single protocol connection.

connect data

A portion of the **connect descriptor** that defines the destination database **service name** or **Oracle System Identifier (SID)**. In the following example, `SERVICE_NAME` defines a database service called `sales.us.example.com`:

```
(DESCRIPTION=
  (ADDRESS= (PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)
  (CONNECT_DATA=
    (SERVICE_NAME=sales.us.example.com)))
```

connect descriptor

A specially-formatted description of the destination for a network connection. A connect descriptor contains destination service and network route information.

The destination service is indicated by using its **service name**. The network route provides, at a minimum, the location of the listener through use of a network address.

connect identifier

A name, net service name, or service name that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a connect identifier in a connect string for the service to which they want to connect, for example:

```
CONNECT username@connect_identifier
```

connect string

Information the user passes to a service to connect, such as user name, password, and **connect identifier**:

```
CONNECT username@net_service_name
```

connect-time failover

A connect-time failover occurs when a client connect request fails over to a different address if the first protocol address fails. A statically configured global database name disables connect-time failover.

connection

An interaction between two processes on a network. Connections are originated by an initiator (client) that requests a connection with a destination (server).

connection load balancing

The method for balancing the number of active connections for the same service across the instances and dispatchers. Connection load balancing enables listeners to make routing decisions based on how many connections for each dispatcher and the load on the nodes.

connection pooling

A resource utilization and user scalability feature that enables you to maximize the number of sessions over a limited number of protocol connections to a [shared server](#).

connection request

A notification sent by an initiator and received by a listener that indicates that the initiator wants to start a connection.

data packet

See [packet](#).

database administrator (DBA)

A person responsible for operating and maintaining an Oracle Server or a database application.

An Oracle user name that has been given DBA privileges and can perform database administration functions. Usually the two meanings coincide. Many sites have multiple DBAs.

Database Configuration Assistant

A tool that enables you to create, delete, and modify a database.

database link

A pointer that defines a one-way communication path from an Oracle database server to another database server. The link is a defined entry in a data dictionary table. To access the link, the user must be connected to the local database that contains the data dictionary entry.

A client connected to local database A can use a link stored in database A to access information in remote database B. However, users connected to database B cannot use the same link to access data in database A. If local users on database B want to access data on database A, then a link must be defined and stored in the data dictionary of database B.

The following database links are supported:

- A [private database link](#) in a specific schema of a database. Only the owner of a private database link can use it.
- A [public database link](#) for a database. All users in the database can use it.

DBA

See [database administrator \(DBA\)](#)

dedicated server

A server process that is dedicated to one client connection. Compare to [shared server](#).

default domain

The [domain](#) within which most client requests take place. It could be the domain where the client resides, or it could be a domain from which the client requests network services often. Default domain is also the client configuration parameter that determines what domain should be appended to unqualified network name requests. A name request is unqualified if it does not have a period (.) character within it.

directory information tree (DIT)

A hierarchical tree-like structure in a [directory server](#) of the distinguished names (DNs) of the entries.

directory naming

A [naming method](#) that resolves a database service, [net service name](#), or [net service alias](#) to a [connect descriptor](#) stored in a central directory server. A [directory server](#) provides central administration of directory naming objects, reducing the work effort associated with adding or relocating services.

directory server

A directory server that is accessed with the [Lightweight Directory Access Protocol \(LDAP\)](#). Support of LDAP-compliant directory servers provides a centralized vehicle for managing and configuring a distributed Oracle network. The directory server can replace client-side and server-side localized `tnsnames.ora` files.

dispatcher

A process that enables many clients to connect to the same server without the need for a dedicated server process for each client. A dispatcher handles and directs multiple incoming network session requests to shared server processes. See also [shared server](#).

distinguished name (DN)

Name of entry in a [directory server](#). The DN specifies where the entry resides in the LDAP directory hierarchy, much the way a directory path specifies the exact location of a file.

distributed processing

Division of front-end and back-end processing to different computers. Oracle Net Services support distributed processing by transparently connecting applications to remote databases.

domain

Any tree or subtree within the [Domain Name System \(DNS\)](#) namespace. Domain most commonly refers to a group of computers whose host names share a common suffix, the domain name.

Domain Name System (DNS)

A system for naming computers and network services that is organized into a hierarchy of [domains](#). DNS is used in TCP/IP networks to locate computers through user-friendly names. DNS resolves a friendly name into an [IP address](#), which is understood by computers.

For Oracle Net Services, DNS translates the host name in a TCP/IP address into an IP address.

DNS

See [Domain Name System \(DNS\)](#).

enterprise role

An enterprise role is analogous to a regular database role, except that it spans authorization on multiple databases. An enterprise role is a category of roles that define privileges on a particular database. An enterprise role is created by the database administrator of a particular database. An enterprise role can be granted to or revoked

to one or more enterprise users. The information for granting and revoking these roles is stored in the directory server.

enterprise user

A user that has a unique identity across an enterprise. Enterprise users connect to individual databases through a schema. Enterprise users are assigned enterprise roles that determine their access privileges on databases.

entry

The building block of a directory server, it contains information about an object of interest to directory users.

external naming

A [naming method](#) that uses a third-party naming service, such as [NIS](#).

external procedure

Function or procedure written in a third-generation language (3GL) that can be called from PL/SQL code. Only C is supported for external procedures.

failover

See [connect-time failover](#).

firewall support

See [access control](#).

foreign domains

The set of domains not managed within a given administrative region. Domains are foreign only in relation to a region; they are not foreign in any absolute sense. A network administrator typically defines foreign domains relative to a particular region to optimize caching performance.

FTP

File Transfer Protocol. A client/server protocol which allows a user on one computer to transfer files to and from another computer over a TCP/IP network.

global database name

The full name of the database which uniquely identifies it from any other database. The global database name is of the form "*database_name.database_domain*," for example, *sales.us.example.com*.

The database name portion, *sales*, is a simple name to call a database. The database domain portion, *us.example.com*, specifies the database domain in which the database is located, making the global database name unique. When possible, Oracle recommends that your database domain mirror the network domain.

The global database name is the default service name of the database, as specified by the `SERVICE_NAMES` parameter in the initialization parameter file.

Heterogeneous Services

An integrated component that provides the generic technology for accessing non-Oracle systems from the Oracle database server. Heterogeneous Services enables you to:

- Use Oracle SQL to transparently access data stored in non-Oracle systems as if the data resides within an Oracle server.

- Use Oracle procedure calls to transparently access non-Oracle systems, services, or application programming interfaces (APIs), from your Oracle distributed environment.

hierarchical naming model

An infrastructure in which names are divided into multiple hierarchically-related domains. For Oracle Names, hierarchical naming model can be used with either central or delegated administration.

host naming

A **naming method** resolution that enables users in a TCP/IP environment to resolve names through their existing name resolution service. This name resolution service might be **Domain Name System (DNS)**, **Network Information Service (NIS)**, or simply a centrally-maintained set of `/etc/hosts` files. Host naming enables users to connect to an Oracle database server by simply providing the server computer's host name or host name alias. No client configuration is required to take advantage of this feature. This method is recommended for simple TCP/IP environments.

HTTP

Hypertext Transfer Protocol. A protocol that provides the language that enables Web browsers and application Web servers to communicate.

identity management realm

A collection of identities, all of which are governed by the same administrative policies. In an enterprise, all employees having access to the intranet may belong to one realm, while all external users who access the public applications of the enterprise may belong to another realm. An identity management realm is represented in the directory by a specific entry with a special object class associated with it.

instance

The combination of the **System Global Area (SGA)** and the Oracle background processes. When a database is started on a database server (regardless of the type of computer), Oracle allocates a memory area called the SGA, and starts one or more Oracle processes. The memory and processes of an instance efficiently manage the associated database data and serve the database users. You can connect to any instance to access information within a cluster database.

instance name

A name of an Oracle database instance. The instance name is identified by the `INSTANCE_NAME` parameter in the database initialization parameter file. `INSTANCE_NAME` corresponds to the **Oracle System Identifier (SID)** of the instance. Clients can connect to a specific instance by specifying the `INSTANCE_NAME` parameter in the connect descriptor.

The instance name is included in the **connect data** part of the **connect descriptor**.

Interprocess Communication (IPC)

A protocol used by client applications that resides on the same node as the listener to communicate with the database. IPC can provide a faster local connection than TCP/IP.

IP address

Used to identify a node on a network. Each computer on the network is assigned a unique IP address, which is made up of the network ID, and a unique host ID. This

address is typically represented in dotted-decimal notation, with the decimal value of each octet separated by a period, for example 192 . 168 . 2 . 22.

IPC

See [Interprocess Communication \(IPC\)](#).

Java Database Connectivity (JDBC) Driver

A driver that provides Java applications and applets access to an Oracle database.

JDBC OCI Driver

A Type II driver for use with client/server Java applications. This driver requires an Oracle client installation.

JDBC Thin Driver

A Type IV driver for Oracle JDBC applets and applications. Because it is written entirely in Java, this driver is platform-independent. It does not require any additional Oracle software on the client side. The Thin driver communicates with the server using [Two-Task Common \(TTC\)](#), a protocol developed by Oracle to access the database server.

keyword-value pair

The combination of a keyword and a value, used as the standard unit of information in connect descriptors and many configuration files. Keyword-value pairs may be nested; that is, a keyword may have another keyword-value pair as its value.

latency

The amount of time it takes for to send a request and receive an answer.

LDAP Data Interchange Format (LDIF)

The set of standards for formatting an input file for any of the LDAP command line utilities.

ldap.ora file

A file created by Oracle Internet Directory Configuration Assistant or Oracle Net Configuration Assistant that contains the following directory server access information:

- Type of directory server
- Location of the directory server
- Default Oracle Context that the client or server use to look up or configure connect identifiers for connections to database services

When created with Oracle Internet Directory Configuration Assistant, `ldap.ora` is located in the `ORACLE_HOME/ldap/admin` directory. When created with Oracle Net Configuration Assistant, `ldap.ora` is located in the `ORACLE_HOME/network/admin` directory.

Lightweight Directory Access Protocol (LDAP)

A standard, extensible directory access protocol. It is a industry-standard programmatic interface and a wire protocol which enables clients to access directory systems.

link qualifier

An extension to the database link name which specifies the connect name used to connect to the database. It provides alternate settings for the database user name and password credentials. For example, a link qualifier of `fieldrep` can be appended to a global database link of `sales.us.example.com`.

```
SQL> SELECT * FROM emp@sales.us.example.com@fieldrep
```

listener

A process that resides on the server whose responsibility is to listen for incoming client connection requests and manage the traffic to the server.

When a client requests a network session with a database server, a listener receives the actual request. If the client information matches the listener information, then the listener grants a connection to the database server.

Listener Control utility

A utility included with Oracle Net Services to control listener functions, such as starting, stopping, and getting the status of the listener.

listener.ora file

A configuration file for the listener that identifies the following for a [listener](#):

- Unique name
- Protocol addresses that it is accepting connection requests on
- Services it is listening for

The `listener.ora` file typically resides in the `ORACLE_HOME/network/admin` directory.

Oracle does not require identification of the database service because of [service registration](#). However, static service configuration is required if you plan to use Oracle Enterprise Manager.

load balancing

A feature by which client connections are distributed evenly among multiple listeners, dispatchers, instances, and nodes so that no single component is overloaded.

Oracle Net Services support [client load balancing](#) and [connection load balancing](#).

local naming

A [naming method](#) that locates network addresses by using information configured and stored on each individual client's [tnsnames.ora file](#). Local naming is most appropriate for simple distributed networks with a small number of services that change infrequently.

location transparency

A distributed database characteristic that enables applications to access data tables without knowing where they reside. All data tables appear to be in a single database, and the system determines the actual data location based on the table name. The user can reference data on multiple nodes in a single statement, and the system automatically and transparently routes (parts of) SQL statements to remote nodes for execution if needed. The data can move among nodes with no impact on the user or application.

logging

A feature in which errors, service activity, and statistics are written to a log file. The log file provides additional information for an administrator when the error message on the screen is inadequate to understand the failure. The log file, by way of the error stack, shows the state of the software at various layers.

See also [tracing](#).

loopback test

A connection from the server back to itself. Performing a successful loopback verifies that Oracle Net is functioning on the database server.

map

Files used by the [Network Information Service \(NIS\)](#) ypserv program to handle name requests.

Microsoft Active Directory

An LDAP-compliant directory server included with Microsoft Windows 2000 Server. It stores information about objects on the network, and makes this information available to users and network administrators. Active Directory also provides access to resources on the network using a single logon process.

Active Directory can be configured as a directory naming method to store service information that clients can access.

Microsoft Windows NT native authentication

An [authentication method](#) that enables a client single login access to a Microsoft Windows NT server and a database running on the server.

Named Pipes protocol

A high-level interface protocol providing interprocess communications between clients and servers using distributed applications. Named Pipes enables client/server conversation over a network using Named Pipes protocol.

naming context

A subtree that resides entirely on one directory server. It is a contiguous subtree, that is, it must begin at an entry that serves as the top of the subtree, and extend downward to either leaf entries or references to subordinate naming contexts. It can range in size from a single entry to the entire [directory information tree \(DIT\)](#).

An [Oracle Context](#) can be created under a naming context.

naming method

The resolution method used by a client application to resolve a [connect identifier](#) to a [connect descriptor](#) when attempting to connect to a database service. Oracle Net provides four naming methods:

- [Domain Name System \(DNS\)](#)
- [directory naming](#)
- Easy Connect naming
- [external naming](#)

net service alias

An alternative name for a [directory naming](#) object in a directory server. A directory server stores net service aliases for any defined [net service name](#) or database service. A net service alias entry does not have connect descriptor information. Instead, it only references the location of the object for which it is an alias. When a client requests a directory lookup of a net service alias, the directory determines that the entry is a net service alias and completes the lookup as if it was actually the entry it is referencing.

net service name

A simple name for a service that resolves to a [connect descriptor](#). Users initiate a connect request by passing a user name and password along with a net service name in a connect string for the service to which they want to connect:

```
CONNECT username/password@net_service_name
```

Depending on your needs, net service names can be stored in a variety of places, including:

- Local configuration file, `tnsnames.ora`, on each client
- Directory server
- External naming service, such as [NIS](#)

network

A group of two or more computers linked through hardware and software to allow the sharing of data and peripherals.

network administrator

The person who performs network management tasks such as installing, configuring, and testing network components. The administrator typically maintains the configuration files, connect descriptors and service names, aliases, and public and global database links.

network character set

As defined by Oracle, the set of characters acceptable for use as values in keyword-value pairs (that is, in connect descriptors and configuration files). The set includes alphanumeric uppercase, and lowercase, and some special characters.

Network Information Service (NIS)

Sun Microsystems' Yellow Pages (`yp`) client/server protocol for distributing system configuration data such as user and host names between computers on a network.

Network Interface (NI)

A network layer that provides a generic interface for Oracle clients, servers, or external processes to access Oracle Net functions. The NI layer handles the break and reset requests for a connection.

network listener

See [listener](#).

network object

Any service that can be directly addressed on a network; for example, a listener.

Network Program Interface (NPI)

An interface for server-to-server interactions that performs all of the functions that the [OCI](#) does for clients, allowing a coordinating server to construct SQL requests for additional servers.

network protocol

See [Oracle protocol support](#).

Network Session (NS)

A [session layer](#) that is used in typical Oracle Net connections to establish and maintain the connection between a client application and a database server.

NI

See [Network Interface \(NI\)](#).

NIS

See [Network Information Service \(NIS\)](#).

node

A computer or terminal that is part of a network

NPI

See [Network Program Interface \(NPI\)](#).

NS

See [Network Session \(NS\)](#).

NT

Network Transport. See [transport](#).

object class

In a directory server, a named group of attributes. When you want to assign attributes to an entry, you do so by assigning the object classes that hold those attributes to that entry.

All objects associated with the same object class share the attributes of that object class.

OCI

See [Oracle Call Interface \(OCI\)](#).

OPI

See [Oracle Program Interface \(OPI\)](#).

Open Systems Interconnection (OSI)

A network architecture model developed by ISO as a framework for international standards in heterogeneous computer network architecture.

The OSI architecture has seven layers, from lowest to highest:

1. Physical layer
2. Data link layer
3. Network layer
4. Transport layer

5. Session layer
6. Presentation layer
7. Application layer

Oracle Advanced Security

A comprehensive suite of security features to protect enterprise networks and securely extends corporate networks to the Internet. Oracle Advanced Security provides a single source of integration with network encryption and authentication solutions, single sign-on services, and security protocols. By integrating industry standards, it delivers unparalleled security to the network.

Oracle Call Interface (OCI)

An application programming interface (API) that enables you to create applications that use the native procedures or function calls of a third-generation language to access an Oracle database server and control all phases of SQL statement execution. OCI supports the data types, calling conventions, syntax, and semantics of a number of third-generation languages including C, C++, COBOL and FORTRAN.

Oracle Connection Manager

A router through which a client connection request may be sent either to its next hop or directly to the database server. Clients who route their connection requests through an Oracle Connection Manager can take advantage of the [session multiplexing](#), [access control](#), or [protocol conversion](#) features configured for that Oracle Connection Manager.

Oracle Connection Manager Control utility

A utility included with Oracle Net Services to control various functions, such as starting, stopping, and getting the status of the Oracle Connection Manager.

Oracle Context

A relative distinguished name ([RDN](#)) of `cn=OracleContext` in a [directory information tree \(DIT\)](#) that is located under a [naming context](#) or an unpublished directory entry. Oracle Context contains entries for use with Oracle features, such as Oracle Net [directory naming](#) and [Oracle Advanced Security enterprise user](#) security. There can be one or more Oracle Contexts in a directory server. [Oracle Internet Directory](#) automatically creates an Oracle Context at the root of the DIT structure. This root Oracle Context has a DN of `dn:cn=OracleContext`.

Oracle Enterprise Manager

A separate Oracle product that combines a graphical console, agents, common services, and tools to provide an integrated and comprehensive systems management platform for managing Oracle products.

Oracle Identity Management

An infrastructure enabling deployments to manage centrally and securely all enterprise identities and their access to various applications in the enterprise.

Oracle Internet Directory

A directory server implemented as an application on the Oracle database. It enables retrieval of information about dispersed users and network resources. It combines [Lightweight Directory Access Protocol \(LDAP\)](#) Version 3, the open Internet standard directory server access protocol, with the high performance, scalability, robustness, and availability of the Oracle database.

Oracle Net

Communication software that enables a network session from a client application to an Oracle database server. After a network session is established, Oracle Net acts as a data courier for the client application and the database server. It is responsible for establishing and maintaining the connection between the client application and database server, as well as exchanging messages between them. Oracle Net can perform these jobs because it is located on each computer in the network.

Oracle Net Configuration Assistant

A postinstallation tool that configures basic network components after installation, including:

- Listener names and protocol addresses
- Naming methods the client uses to resolve [connect identifiers](#)
- Net service names in a `tnsnames.ora` file
- Directory server usage

Oracle Net Firewall Proxy

Product offered by some firewall vendors that supplies [Oracle Connection Manager](#) functionality.

Oracle Net foundation layer

A networking communication layer that is responsible for establishing and maintaining the connection between the client application and server, as well as exchanging messages between them.

Oracle Net listener

See [listener](#).

Oracle Net Manager

A tool that combines configuration abilities with component control to provide an integrated environment for configuring and managing Oracle Net Services.

You can use Oracle Net Manager to configure the following network components:

- Naming
 - Define [connect identifiers](#) and map them to [connect descriptors](#) to identify the network location and identification of a service. Oracle Net Manager supports configuration of connect descriptors in a local `tnsnames.ora` file or directory server.
- Naming Methods
 - Configure the ways in which connect identifiers are resolved into connect descriptors.
- Listeners
 - Create and configure listeners to receive client connections.

Oracle Net Services

A suite of networking components that provide enterprise-wide connectivity solutions in distributed, heterogeneous computing environments. Oracle Net Services is comprised of [Oracle Net](#), [listener](#), [Oracle Connection Manager](#), [Oracle Net Configuration Assistant](#), and [Oracle Net Manager](#).

Oracle Program Interface (OPI)

A networking layer responsible for responding to each of the possible messages sent by [OCI](#). For example, an OCI request to fetch 25 rows would have an OPI response to return the 25 rows after they have been fetched.

Oracle protocol support

A software layer responsible for mapping [Transparent Network Substrate \(TNS\)](#) functionality to industry-standard protocols used in the client/server connection.

Oracle Rdb

A database for Digital's 64-bit platforms. Because Oracle Rdb has its own listener, the client interacts with Rdb in the same manner as it does with an Oracle database.

Oracle schema

A set of rules that determine what can be stored in a [directory server](#). Oracle has its own schema that is applied to many types of Oracle entries, including Oracle Net Services entries. The Oracle schema for Oracle Net Services entries includes the attributes the entries may contain.

Oracle System Identifier (SID)

A name that identifies a specific instance of a running an Oracle database earlier than release 8.1. For any database, there is at least one instance referencing the database.

For Oracle databases earlier than release 8.1, a SID is used to identify the database. The SID is included in the connect descriptor of a [tnsnames.ora file](#) and in the definition of the listener in the [listener.ora file](#).

Oracle XML DB

A high-performance XML storage and retrieval technology provided with Oracle database server. It is based on the W3C XML data model.

Oracle Real Application Clusters

An architecture that allows multiple instances to access a shared database of data files. Oracle Real Application Clusters is also a software component that provides the necessary cluster database scripts, initialization files, and data files needed for the Oracle Enterprise Edition and Oracle Real Application Clusters.

ORACLE_HOME

An alternate name for the top directory in the Oracle directory hierarchy on some directory-based operating systems.

OSI

See [Open Systems Interconnection \(OSI\)](#).

packet

A block of information sent over the network each time a connection or data transfer is requested. The information contained in packets depends on the type of packet, such as connect, accept, redirect, data, and so on. Packet information can be useful in troubleshooting.

PMON process

A process monitor (PMON) database process that performs process recovery when a user process fails. PMON is responsible for cleaning the cache and freeing resources that the process was using. PMON also checks on dispatcher and server processes and

restarts them if they have failed. As a part of [service registration](#), PMON registers instance information with the listener.

presentation layer

A networking communication layer that manages the representation of information that application layer entities either communicate or reference in their communication. [Two-Task Common \(TTC\)](#) is an example of presentation layer.

private database link

A database link created by one user for exclusive use.

See also [database link](#) and [public database link](#).

profile

A collection of parameters that specifies preferences for enabling and configuring Oracle Net Services features on the client or server. A profile is stored and implemented through the `sqlnet.ora` file.

protocol

A set of rules that defines how data is transported across the network.

protocol address

An address that identifies the network address of a network object.

When a connection is made, the client and the receiver of the request, such as the [listener](#) or [Oracle Connection Manager](#), are configured with identical protocol addresses. The client uses this address to send the connection request to a particular network object location, and the recipient listens for requests on this address. It is important to install the same protocols for the client and the connection recipient, as well as configure the same addresses.

protocol conversion

A feature of Oracle Connection Manager that enables a client and server with different networking protocols to communicate with each other. This feature replaces functionality previously provided by the Oracle Multi-Protocol Interchange with SQL*Net version 2.

protocol stack

Designates a particular [presentation layer](#) and [session layer](#) combination.

proxy server

A server that substitutes for a real server, forwarding client connection requests to the real server or to other proxy servers. Proxy servers provide access control, data and system security, monitoring, and caching.

public database link

A database link created by a DBA on a local database that is accessible to all users on that database.

See also [database link](#) and [private database link](#).

realm Oracle Context

An Oracle Context contained in each [identity management realm](#). It stores the following information:

- User naming policy of the identity management realm, that is, how users are named and located
- Mandatory authentication attributes
- Location of groups in the identity management realm
- Privilege assignments for the identity management realm, for example, who has privileges to add more users to the realm.
- Application specific data for that realm including authorizations

RDBMS

Relational Database Management System.

RDN

See [relative distinguished name \(RDN\)](#).

relative distinguished name (RDN)

A fully-qualified X.500 name. It is the local, most granular level entry name. In the example, `cn=sales, dc=us, dc=acme, dc=com`, the RDN is `cn=sales`.

root Oracle Context

In the [Oracle Identity Management](#) infrastructure, the root Oracle Context is an entry in Oracle Database Net Services Reference containing a pointer to the default [identity management realm](#) in the infrastructure. It also contains information about how to locate an identity management realm given the simple name of the realm.

RPC

Remote procedure call.

SDP

Sockets Direct Protocol.

Secure Sockets Layer (SSL)

An industry-standard protocol designed by Netscape Communications Corporation for securing network connections. SSL provides authentication, encryption, and data integrity using public key infrastructure (PKI).

server process

Database processes that handle a client request on behalf of a database.

service

A program that responds to requests from various clients or performs some operation. For example, the database is a service that stores and retrieves data for clients.

service handler

A process that acts a connection point from the listener to the database server. A service handler can be a [dispatcher](#) or [dedicated server](#).

service name

A logical representation of a database, which is the way a database is presented to clients. A database can be presented as multiple services and a service can be implemented as multiple database instances. The service name is a string that is the [global database name](#), that is, a name comprising the database name and domain

name, entered during installation or database creation. If you are not sure what the global database name is, then you can obtain it from the value of the `SERVICE_NAMES` parameter in the initialization parameter file.

The service name is included in the **connect data** part of the **connect descriptor**.

service registration

A feature by which the **PMON process** automatically registers information with a **listener**. Because this information is registered with the listener, the `listener.ora` file does not need to be configured with this static information.

Service registration provides the listener with information about:

- Service names for each running instance of the database
- Instance names of the database
- Service handlers (**dispatcher** or **dedicated server**) available for each instance
These enable the listener to direct a client request appropriately.
- Dispatcher, instance, and node load information
This load information enables the listener to determine which dispatcher can best handle a client connection request. If all dispatchers are blocked, then the listener can spawn a dedicated server for the connection.

session data unit (SDU)

A buffer that Oracle Net uses to place data before transmitting it across the network. Oracle Net sends the data in the buffer either when requested or when it is full.

session layer

A network layer that provides the services needed by the **protocol address** entities that enable them to organize and synchronize their dialog and manage their data exchange. This layer establishes, manages, and terminates network sessions between the client and server. An example of a session layer is **Network Session (NS)**.

session multiplexing

Combining multiple sessions for transmission over a single network connection to conserve the operating system's resources.

shared server

A database server that is configured to allow many user processes to share very few server processes, so the number of users that can be supported is increased. With shared server configuration, many user processes connect to a **dispatcher**. The dispatcher directs multiple incoming network session requests to a common queue. An idle shared server process from a shared pool of server processes picks up a request from the queue. This means that a small pool of server processes can serve a large number of clients. Contrast with **dedicated server**.

shared server process

A process type used with **shared server** configuration.

SID

See **Oracle System Identifier (SID)**.

SID_LIST_listener_name

A section of the `listener.ora` file that defines the **Oracle System Identifier (SID)** of the database served by the listener. This section is valid only for Oracle databases release 8.0, as information for Oracle8i or later instances is automatically registered with the listener. Static configuration is also required for other services, such as **external procedure** calls and **Heterogeneous Services**.

single sign-on

The ability for a user to log in to different servers using a single password. This permits the user to authenticate to all servers the user is authorized to access.

sqlnet.ora file

A configuration file for the client or server that specifies:

- Client domain to append to unqualified service names or net service names
- Order of naming methods the client should use when resolving a name
- Logging and tracing features to use
- Route of connections
- External naming parameters
- Oracle Advanced Security parameters

The `sqlnet.ora` file typically resides in the `ORACLE_HOME/network/admin` directory.

SSL

See **Secure Sockets Layer (SSL)**.

System Global Area (SGA)

A group of shared memory structures that contain data and control information for an Oracle **instance**.

TCP/IP

Transmission Control Protocol/Internet Protocol. The standard communication protocol used for client/server conversation over a network.

TCP/IP with SSL protocol

A protocol that enables an Oracle application on a client to communicate with remote Oracle databases through the **TCP/IP** and **Secure Sockets Layer (SSL)**.

tick

The amount of time it takes for a message to be sent and processed from the client to the server or from the server to the client

TNS

See **Transparent Network Substrate (TNS)**.

tnsnames.ora file

A configuration file that maps **net service names** to **connect descriptors**. This file is used for the **local naming** method. The `tnsnames.ora` file typically resides in the `ORACLE_HOME/network/admin` directory.

tracing

A facility that writes detailed information about an operation to an output file. The trace facility produces a detailed sequence of statements that describe the events of an operation as they are run. Administrators use the trace facility for diagnosing an abnormal condition. It is not normally turned on.

See also [logging](#).

Transparent Application Failover (TAF)

A run-time failover for high-availability environments, such as Oracle Real Application Clusters and Oracle Fail Safe, that refers to the failover and re-establishment of application-to-service connections. It enables client applications to automatically reconnect to the database if the connection fails, and, optionally, resume a `SELECT` statement that was in progress. This reconnect happens automatically from within the Oracle Call Interface (OCI) library.

Transparent Network Substrate (TNS)

A foundation technology, built into the [Oracle Net foundation layer](#) that works with any standard network transport protocol.

transport

A networking layer that maintains end-to-end reliability through data flow control and error recovery methods. The [Oracle Net foundation layer](#) uses [Oracle protocol support](#) for the transport layer.

TTC

See [Two-Task Common \(TTC\)](#).

Two-Task Common (TTC)

A [presentation layer](#) type that is used in a typical Oracle Net connection to provide character set and data type conversion between different character sets or formats on the client and server.

UPI

User Program Interface

virtual circuit

A piece of shared memory used by the [dispatcher](#) for client database connection requests and replies. The dispatcher places a virtual circuit on a common queue when a request arrives. An idle shared server picks up the virtual circuit from the common queue, services the request, and relinquishes the virtual circuit before attempting to retrieve another virtual circuit from the common queue.

WebDAV protocol

World Wide Web Distributed Authoring and Versioning. A protocol with a set of extensions to [HTTP](#) which allows users to manage files on remote Web servers.

Symbols

- # (quotation mark) symbol
 - reserved in configuration files, 3-2
- () (parenthesis) symbol
 - reserved in configuration files, 3-2
- = (equals sign) symbol
 - reserved in configuration files, 3-2

Numerics

- 1024 port, 4-3
- 1521 port, 4-3
- 1575 port, 4-3
- 1630 port, 4-3
- 1646 port, 5-22
- 1830 port, 4-3
- 2482 port, 4-3
- 2484 port, 4-3

A

- ACT networking parameter, 8-8
- ACTION_LIST networking parameter, 8-8
- ADDRESS networking parameter, 4-1, 6-5, 7-2, 8-4
- ADDRESS_LIST networking parameter, 4-2, 6-6
- ADMIN_RESTRICTIONS_*listener_name* control parameter, 7-7
- ADMINISTER command, 2-3
- ADR
 - described, 5-32, 7-12, 8-10
- ADR diagnostic parameters
 - listener.ora
 - DIAG_ADR_ENABLED_*listener_name*, 7-13
 - sqlnet.ora
 - ADR_BASE, 5-33
 - DIAG_ADR_ENABLED, 5-33
- ADR_BASE diagnostic parameter, 5-33, 8-10
- ADR_BASE_*listener_name* diagnostic parameter, 7-12
- ASO_AUTHENTICATION_FILTER networking parameter, 8-4
- attributes
 - orclCommonContextMap, C-2
 - orclDescList, C-2
 - orclDescName, C-2

- orclLoadBalance, C-2
- orclNetAddrList, C-2
- orclNetAddrString, C-2
- orclNetConnParamList, C-2
- orclNetFailover, C-2
- orclNetFailoverModeString, C-2
- orclNetHostname, C-2
- orclNetInstanceName, C-2
- orclNetInstanceRole, C-2
- orclNetProtocol, C-2
- orclNetReceiveBufSize, C-2
- orclNetSdu, C-2
- orclNetSendBufSize, C-2
- orclNetServiceName, C-3
- orclNetSourceRoute, C-3
- orclSid, C-3
- orclVersion, C-3
- authentication ability, 5-9
- automatic diagnostic repository
 - described, 5-32, 7-12, 8-10
 - See ADR

B

- BACKUP networking parameter, 6-12
- BEQUEATH_DETACH networking parameter, 5-3

C

- character sets
 - for net service name, 3-3
 - network, for keyword values, 3-2
- class of secure transports parameters
 - See COST parameters
- client load balancing
 - configuring, 6-7
 - with Oracle Connection Manager, 6-3
- CLOSE CONNECTIONS command, 2-4
- cman.ora file
 - diagnostic parameters
 - ADR_BASE, 8-10
 - DIAG_ADR_ENABLED, 8-11
 - LOG_DIRECTORY, 8-13
 - LOG_LEVEL, 8-11
 - TRACE_DIRECTORY, 8-13
 - TRACE_FILELEN, 8-13

- TRACE_FILENO, 8-14
- TRACE_LEVEL, 8-11
- example, 8-1
- listening address section, 8-2
- parameter list section, 8-2
- parameters
 - ACT, 8-8
 - ACTION_LIST, 8-8
 - ADDRESS, 8-4
 - ASO_AUTHENTICATION_FILTER, 8-4
 - CONNECTION_STATISTICS, 8-5
 - DST, 8-8
 - EVENT_GROUP, 8-5
 - IDLE_TIMEOUT, 8-5
 - INBOUND_CONNECT_TIMEOUT, 8-6
 - MAX_CMCTL_SESSIONS, 8-6
 - MAX_CONNECTIONS, 8-6
 - MAX_GATEWAY_PROCESSES, 8-7
 - MIN_GATEWAY_PROCESSES, 8-7
 - OUTBOUND_CONNECT_TIMEOUT, 8-7
 - PARAMETER_LIST, 8-3
 - PASSWORD_instance_name, 8-7
 - RULE, 8-7
 - SESSION_TIMEOUT, 8-9
 - SRC, 8-8
 - SRV, 8-8
- rule list section, 8-2
- cman.ora file parameters, 8-3
- comments in configuration files, 3-2
- connect descriptors, 6-1
- CONNECT_DATA networking parameter, 6-11
- CONNECT_TIMEOUT, 6-17
- CONNECT_TIMEOUT networking parameter, 6-17
- connection rate limiter, 7-5
- CONNECTION_RATE_listener_name configuration parameter, 7-6
- CONNECTION_STATISTICS networking parameter, 8-5
- connections
 - adjusting listener queue size to avoid errors, 7-4
- connect-time failover
 - configuring, 6-3, 6-7
 - with Oracle Connection Manager, 6-3
- control parameters
 - listener.ora
 - ADMIN_RESTRICTIONS_listener_name, 7-7
 - CRS_NOTIFICATION_listener_name, 7-8
 - DEFAULT_SERVICE_listener_name, 7-8
 - INBOUND_CONNECT_TIMEOUT_listener_name, 7-8
 - SAVE_CONFIG_ON_STOP_listener_name, 7-9
 - SSL_CLIENT_AUTHENTICATION, 7-9
 - SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name, 7-10
 - WALLET_LOCATION, 7-10
- control utilities
 - Listener Control utility, 1-19
 - Oracle Connection Manager Control utility, 2-2 to 2-25
- COST parameters, 7-16

- DYNAMIC_REGISTRATION_listener_name, 7-16
- SECURE_CONTROL_listener_name, 7-17
- SECURE_PROTOCOL_listener_name, 7-18
- SECURE_REGISTER_listener_name, 7-17
- CRS_NOTIFICATION_listener_name control parameter, 7-8

D

- database resident connection pooling, 6-14
- DEFAULT_ADMIN_CONTEXT networking parameter, 9-2
- DEFAULT_SDU_SIZE networking parameter, 5-4
- DEFAULT_SERVICE_listener_name control parameter, 7-8
- DELAY networking parameter, 6-13
- DESCRIPTION networking parameter, 6-5, 7-3
- DESCRIPTION_LIST networking parameter, 6-5
- DIAG_ADR_ENABLED diagnostic parameter, 5-33, 8-11
- DIAG_ADR_ENABLED_listener_name diagnostic parameter, 7-13
- diagnostic parameters
 - cman.ora
 - ADR_BASE, 8-10
 - DIAG_ADR_ENABLED, 8-11
 - LOG_DIRECTORY, 8-13
 - LOG_LEVEL, 8-11
 - TRACE_DIRECTORY, 8-13
 - TRACE_FILELEN, 8-13
 - TRACE_FILENO, 8-14
 - TRACE_LEVEL, 8-11
 - listener.ora
 - ADR_BASE_listener_name, 7-12
 - DIAG_ADR_ENABLED_listener_name, 7-13
 - LOG_DIRECTORY_listener_name, 7-14
 - LOG_FILE_listener_name, 7-15
 - LOGGING_listener_name, 7-13
 - TRACE_DIRECTORY_listener_name, 7-15
 - TRACE_FILE_listener_name, 7-15
 - TRACE_FILENO_listener_name, 7-16
 - TRACE_LEVEL_listener_name, 7-13
 - TRACE_TIMESTAMP_listener_name, 7-14
 - sqlnet.ora
 - ADR_BASE, 5-33
 - DIAG_ADR_ENABLED, 5-33
 - LOG_DIRECTORY_CLIENT, 5-35
 - LOG_DIRECTORY_SERVER, 5-36
 - LOG_FILE_CLIENT, 5-36
 - LOG_FILE_SERVER, 5-36
 - TRACE_DIRECTORY_CLIENT, 5-36
 - TRACE_DIRECTORY_SERVER, 5-37
 - TRACE_FILE_CLIENT, 5-37
 - TRACE_FILE_SERVER, 5-37
 - TRACE_FILELEN_CLIENT, 5-38
 - TRACE_FILELEN_SERVER, 5-38
 - TRACE_FILENO_CLIENT, 5-38
 - TRACE_FILENO_SERVER, 5-38
 - TRACE_LEVEL_CLIENT, 5-33

- TRACE_LEVEL_SERVER, 5-34
- TRACE_TIMESTAMP_CLIENT, 5-34
- TRACE_TIMESTAMP_SERVER, 5-34
- TRACE_UNIQUE_CLIENT, 5-39
- directory naming
 - configuring, 5-5
- DIRECTORY_SERVER_TYPE networking parameter, 9-2
- DIRECTORY_SERVERS, 9-1
- DISABLE_OOB networking parameter, 5-4
- DST networking parameter, 8-8
- DYNAMIC_REGISTRATION_*listener_name*
 - COST parameter, 7-16

E

- ENABLE networking parameter, 6-6
- error messages
 - ORA-12170, 5-16, 5-18, 5-26
 - ORA-12525, 1-9, 7-8
 - ORA-12535, 5-22, 5-23
 - ORA-12547, 5-16
 - ORA-12608, 5-23
 - ORA-12609, 5-22
- EVENT_GROUP networking parameter, 8-5
- EXIT command
 - Listener Control utility, 1-4
 - Oracle Connection Manager Control utility, 2-5
- external naming
 - Network Information Service (NIS), 5-5

F

- failover
 - connect-time, 6-3, 6-7
 - Transparent Application Failover
- FAILOVER networking parameter, 6-7
- FAILOVER_MODE networking parameter, 6-12

G

- GLOBAL_NAME networking parameter, 6-13

H

- HELP command
 - of Listener Control utility, 1-4
 - of Oracle Connection Manager Control utility, 2-6
- heterogeneous services, 6-13
- HOST networking parameter, 4-2, 4-3
- HS networking parameter, 6-13

I

- IDLE_TIMEOUT networking parameter, 8-5
- INBOUND_CONNECT_TIMEOUT networking parameter, 8-6
- INBOUND_CONNECT_TIMEOUT_*listener_name* control parameter, 7-8
- INBOUND_CONNECT_TIMEPUT_*listener_name*

- networking parameter, 7-8
- INSTANCE_NAME networking parameter, 6-14
- IPC protocol
 - addresses, 4-2
 - KEY parameter, 4-2
 - PROTOCOL parameter, 4-2
- IPC, parameters for addresses, 4-2

K

- keepalive feature, 6-6
- KEY networking parameter, 4-2
- keyword syntax rules for configuration files, 3-2
- keyword values and network character sets, 3-2

L

- LDAP schema
 - attributes, C-1 to C-3
 - object classes, C-1 to C-3
- ldap.ora file
 - DEFAULT_ADMIN_CONTEXT parameter, 9-2
 - DIRECTORY_SERVER_TYPE parameter, 9-2
- Listener Control utility
 - command reference, 1-19
 - commands
 - EXIT, 1-4
 - HELP, 1-4
 - QUIT, 1-5
 - RELOAD, 1-5
 - SAVE_CONFIG, 1-6
 - SERVICES, 1-7
 - SET, 1-2, 1-8
 - SET CONNECT_TIMEOUT, 1-8
 - SET CURRENT_LISTENER, 1-8
 - SET DISPLAYMODE, 1-9
 - SET INBOUND_CONNECT_TIMEOUT, 1-9
 - SET LOG_DIRECTORY, 1-10
 - SET LOG_FILE, 1-11
 - SET LOG_STATUS, 1-11
 - SET TRC_DIRECTORY, 1-12
 - SET TRC_FILE, 1-13
 - SET TRC_LEVEL, 1-13
 - SET USE_PLUGANDPLAY, 1-14
 - SHOW, 1-2, 1-14
 - SHOW CURRENT_LISTENER, 1-15
 - SHOW DISPLAYMODE, 1-15
 - SHOW INBOUND_CONNECT_TIMEOUT, 1-15
 - SHOW LOG_DIRECTORY, 1-15
 - SHOW LOG_FILE, 1-15
 - SHOW LOG_STATUS, 1-15
 - SHOW RAWMODE, 1-15
 - SHOW SAVE_CONFIG_ON_STOP, 1-15
 - SHOW TRC_DIRECTORY, 1-15
 - SHOW TRC_FILE, 1-15
 - SHOW TRC_LEVEL, 1-15
 - START, 1-15
 - STATUS, 1-17
 - STOP, 1-18

- TRACE, 1-18
- VERSION, 1-19
- distributed operation, 1-2
- function of and syntax format, 1-1
- remote administration, 1-2
- Listener Control utility access, 1-3
- listener.ora file
 - configuration parameter reference, 7-12
 - configuration parameters
 - CONNECTION_RATE_listener_name, 7-6
 - RATE_LIMIT, 7-6
 - control parameters
 - ADMIN_RESTRICTIONS_listener_name, 7-7
 - CRS_NOTIFICATION_listener_name, 7-8
 - DEFAULT_SERVICE_listener_name, 7-8
 - INBOUND_CONNECT_TIMEOUT_listener_name, 7-8
 - SAVE_CONFIG_ON_STOP_listener_name, 7-9
 - SSL_CLIENT_AUTHENTICATION, 7-9
 - SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name, 7-10
 - WALLET_LOCATION, 7-10
 - COST parameters, 7-16
 - diagnostic parameters
 - ADR_BASE_listener_name, 7-12
 - DIAG_ADR_ENABLED_listener_name, 7-13
 - LOG_DIRECTORY_listener_name, 7-14
 - LOG_FILE_listener_name, 7-15
 - LOGGING_listener_name, 7-13
 - TRACE_DIRECTORY_listener_name, 7-15
 - TRACE_FILE_listener_name, 7-15
 - TRACE_FILEN_listener_name, 7-15
 - TRACE_FILENO_listener_name, 7-16
 - TRACE_LEVEL_listener_name, 7-13
 - TRACE_TIMESTAMP_listener_name, 7-14
 - parameters
 - ADDRESS, 7-2
 - DESCRIPTION, 7-3
 - INBOUND_CONNECT_TIMEOUT_listener_name, 7-8
 - PRESPAWN_DESC, 7-7
 - PRESPAWN_MAX, 7-7
 - QUEUESIZE, 7-4
 - RECV_BUF_SIZE, 7-4
 - SEND_BUF_SIZE, 7-5
 - SID_LIST_listener_name, 7-7
- listeners
 - adjusting queue size for, 7-4
 - connect-request timeouts, 7-8
 - multiple, 7-1
- load balancing
 - client, 6-7
- LOAD_BALANCE networking parameter, 6-7
- local naming
 - configuring, 5-5
- LOG_DIRECTORY diagnostic parameter, 8-13
- LOG_DIRECTORY_CLIENT diagnostic parameter, 5-35
- LOG_DIRECTORY_listener_name diagnostic parameter, 7-14

- LOG_DIRECTORY_SERVER diagnostic parameter, 5-36
- LOG_FILE_CLIENT diagnostic parameter, 5-36
- LOG_FILE_listener_name diagnostic parameter, 7-15
- LOG_FILE_SERVER diagnostic parameter, 5-36
- LOG_LEVEL diagnostic parameter, 8-11
- LOGGING_listener_name diagnostic parameter, 7-13

M

- MAX_CMCTL_SESSIONS networking parameter, 8-6
- MAX_CONNECTIONS networking parameter, 8-6
- MAX_GATEWAY_PROCESSES networking parameter, 8-7
- METHOD networking parameter, 6-12
- MIN_GATEWAY_PROCESSES networking parameter, 8-7
- multiple listeners, 7-1

N

- Named Pipes protocol
 - PIPE parameter, 4-2
 - PROTOCOL parameter, 4-2
 - SERVER parameter, 4-2
- NAMESCTL.TRACE_UNIQUE networking parameter, 5-7
- NAMES.DEFAULT.DOMAIN networking parameter, 5-4
- NAMES.DIRECTORY_PATH networking parameter, 5-5
 - ezconnect, 5-5
 - hostname, 5-5
 - ldap, 5-5
 - nis, 5-5
 - tnsnames, 5-5
- NAMES.LADP_AUTHENTICATE_BIND networking parameter, 5-5
- NAMES.LDAP_CONN_TIMEOUT networking parameter, 5-6
- NAMES.LDAP_PERSISTENT_SESSION networking parameter, 5-6
- Net8
 - coexistence issues, B-2
 - Oracle9i databases, B-2
 - using SERVICE_NAME networking parameter, B-2
 - network character sets, keyword values, 3-2
 - network configuration files
 - listener.ora, 7-12
 - sqlnet.ora, 5-1 to 5-31
 - syntax rules, 3-1
 - tnsnames.ora, 6-16, 6-17
 - Network Information Service external naming configuring, 5-5
 - network performance, improving
 - client load balancing, 6-7
 - networking parameters
 - listener.ora configuration reference, 7-12

sqlnet.ora configuration reference, 5-1 to 5-31
tnsnames.ora configuration reference, 6-16, 6-17

O

object classes

- orclNetAddress, C-2
- orclNetAddressList, C-2
- orclNetDescription, C-1
- orclNetDescriptionList, C-1
- orclNetServiceAlias, C-1

obsolete parameters, A-1

ORA-12170 error message, 5-16, 5-18, 5-26

ORA-12525 error message, 1-9, 7-8

ORA-12535 error message, 5-22, 5-23

ORA-12547 error message, 5-16

ORA-12608 error message, 5-23

ORA-12609 error message, 5-22

Oracle Connection Manager

- client load balancing, 6-3
- connect-time failover, 6-3
- SOURCE_ROUTE networking parameter, 6-10

Oracle Connection Manager Control utility

- command reference, 2-2 to 2-25
- commands
 - ADMINISTER, 2-3
 - CLOSE CONNECTIONS, 2-4
 - EXIT, 2-5
 - HELP, 2-6
 - QUIT, 2-7
 - RELOAD, 2-7
 - RESUME GATEWAYS, 2-8
 - SAVE_PASSWORD, 2-8
 - SET, 2-9
 - SET ASO_AUTHENTICATION_FILTER, 2-9
 - SET CONNECTION_STATISTICS, 2-10
 - SET EVENT, 2-10, 2-11
 - SET IDLE_TIMEOUT, 2-11
 - SET INBOUND_CONNECT_TIMEOUT, 2-12
 - SET LOG_DIRECTORY, 2-12
 - SET LOG_LEVEL, 2-13, 2-16
 - SET OUTBOUND_CONNECT_TIMEOUT, 2-14
 - SET PASSWORD, 2-14
 - SET SESSION_TIMEOUT, 2-15
 - SET TRACE_DIRECTORY, 2-15
 - SET TRACE_LEVEL, 2-16
 - SHOW, 2-17
 - SHOW ALL, 2-17
 - SHOW CONNECTIONS, 2-18
 - SHOW DEFAULTS, 2-20
 - SHOW EVENTS, 2-20
 - SHOW GATEWAYS, 2-21
 - SHOW PARAMETERS, 2-21
 - SHOW RULES, 2-22
 - SHOW SERVICES, 2-23
 - SHOW STATUS, 2-24
 - SHOW VERSION, 2-25
 - SHUTDOWN, 2-25
 - STARTUP, 2-26, 2-27

- SUSPEND GATEWAY, 2-27

Oracle Connection Manager parameters

- ADDRESS, 8-4
- ASO_AUTHENTICATION_FILTER, 8-4
- CONNECTION_STATISTICS, 8-5
- EVENT_GROUP, 8-5
- IDLE_TIMEOUT, 8-5
- INBOUND_CONNECT_TIMEOUT, 8-6
- MAX_CMCTL_SESSIONS, 8-6
- MAX_CONNECTIONS, 8-6
- MAX_GATEWAY_PROCESSES, 8-7
- MIN_GATEWAY_PROCESSES, 8-7
- OUTBOUND_CONNECT_TIMEOUT, 8-7
- PASSWORD_instance_name, 8-7
- RULE, 8-7
- SESSION_TIMEOUT, 8-9

Oracle Names support, A-1

Oracle Net Services

- coexistence issues, B-2
 - using SERVICE_NAME networking parameter, B-2
 - using SID networking parameter, B-2
- SERVICE_NAME parameter, B-2

Oracle protocol support

- configuring addresses, 4-2
- IPC, 4-2
- Named Pipes, 4-2
- SDP, 4-2
- TCP/IP, 4-2
- TCP/IP with SSL, 4-3

Oracle Real Application Clusters

- connect-time failover, 6-7
- FAILOVER networking parameter, 6-7
- FAILOVER_MODE networking parameter, 6-12
- INSTANCE_NAME networking parameter, 6-14
- LOAD_BALANCE networking parameter, 6-7

Oracle schema

- attributes, C-1 to C-3
- object classes, C-1 to C-3

orclCommonContextMap attribute, C-2

orclDescList attribute, C-2

orclDescName attribute, C-2

orclLoadBalance attribute, C-2

orclNetAddress object class, C-2

orclNetAddressList object class, C-2

orclNetAddrList attribute, C-2

orclNetAddrString attribute, C-2

orclNetConnParamList attribute, C-2

orclNetDescription object class, C-1

orclNetDescriptionList object class, C-1

orclNetFailover attribute, C-2

orclNetFailoverModeString attribute, C-2

orclNetHostname attribute, C-2

orclNetInstanceName attribute, C-2

orclNetInstanceRole attribute, C-2

orclNetReceiveBufSize attribute, C-2

orclNetSdu attribute, C-2

orclNetSendBufSize attribute, C-2

orclNetServiceAlias object class, C-1

orclNetServiceName attribute, C-3

- orclNetSourceRoute attribute, C-3
- orclProtocol attribute, C-2
- orclSid attribute, C-3
- orclVersion attribute, C-3
- outbound connect timeout interval, 5-18
- OUTBOUND_CONNECT_TIMEOUT networking parameter, 8-7

P

- PARAMETER_LIST networking parameter, 8-3
- PASSWORD_*instance_name* networking parameter, 8-7
- PIPE networking parameter, 4-2
- port 1024, 4-3
- port 1521, 4-3
- port 1575, 4-3
- port 1630, 4-3
- port 1646, 5-22
- port 1830, 4-3
- port 2483, 4-3
- port 2484, 4-3
- PORT networking parameter, 4-2, 4-3
- port numbers, allowed, 4-3
- ports
 - privileged, 4-3
- PRESPAWN_DESC networking parameter, 7-7
- PRESPAWN_MAX networking parameter, 7-7
- privileged ports, 4-3
- PROTOCOL networking parameter, 4-2, 4-3
- protocols, 4-2
 - authentication ability, 5-9
 - configuring addresses, 4-2
 - IPC, 4-2
 - Named Pipes, 4-2
 - parameters, 4-2
 - SDP, 4-2
 - TCP/IP, 4-2
 - TCP/IP with SSL, 4-3

Q

- QUEUESIZE networking parameter, 7-4
- QUEUESIZE parameter
 - for adjusting listener queue size, 7-4
- QUIT command
 - of Listener Control utility, 1-5
 - of Oracle Connection Manager Control utility, 2-7

R

- randomizing requests among listeners, 6-7
- RATE_LIMIT configuration parameter, 7-6
- RDB_DATABASE networking parameter, 6-14
- RECV_BUF_SIZE, 5-7
- RECV_BUF_SIZE networking parameter, 6-8, 7-4
- reference
 - for Listener Control utility commands, 1-19
 - for listener.ora, 7-12
 - for Oracle Connection Manager Control utility

- commands, 2-2 to 2-25
- for sqlnet.ora, 5-1 to 5-31
- for tnsnames.ora, 6-16, 6-17
- RELOAD command, 2-7
 - of Listener Control utility, 1-5
- RESUME GATEWAYS command, 2-8
- RETRIES networking parameter, 6-13
- RETRY_COUNT, 6-17
- RETRY_COUNT networking parameter, 6-17
- RULE networking parameter, 8-7
- rules, syntax for network configuration files, 3-1

S

- SAVE_CONFIG command
 - of Listener Control utility, 1-6
- SAVE_CONFIG_ON_STOP_*listener_name* control parameter, 7-9
- SAVE_PASSWD command, 2-8
- SDP protocol
 - parameters for addresses, 4-2
- SDP.PF_INET_SDP networking parameter, 5-7
- SDU, 5-4
- SDU networking parameter, 6-9
- SEC_USER_AUDIT_ACTION_BANNER networking parameter, 5-7
- SEC_USER_UNAUTHORIZED_ACCESS_BANNER networking parameter, 5-8
- SECURE_CONTROL_*listener_name*
 - COST parameter, 7-17
- SECURE_PROTOCOL_*listener_name*
 - COST parameter, 7-18
- SECURE_REGISTER_*listener_name*
 - COST parameter, 7-17
- security
 - database server
 - client network timeouts, 5-22
 - connect-request timeouts, 5-8, 5-9
 - listeners
 - connect-request timeouts, 7-8
 - restricting runtime administration, 7-7
- SECURITY networking parameter, 6-16
- SEND_BUF_SIZE networking parameter, 5-8, 6-9, 7-5
- SERVER networking parameter, 4-2, 6-14
- server type
 - dedicated, 6-14
 - pooled, 6-14
 - shared, 6-14
- service name
 - character set keyword values, 3-3
- SERVICE_NAME networking parameter, 6-15
- SERVICE_NAME parameter, B-2
- SERVICES command, 1-7
- SESSION_TIMEOUT networking parameter, 8-9
- SET ASO_AUTHENTICATION_FILTER command, 2-9
- SET command
 - of Listener Control utility, 1-8
 - of Oracle Connection Manager Control

- utility, 2-9
- SET CONNECT_TIMEOUT command, 1-8
- SET CONNECTION_STATISTICS command, 2-10
- SET CURRENT_LISTENER command, 1-8
- SET DISPLAYMODE command
 - of Listener Control utility, 1-9
- SET EVENT command, 2-10, 2-11
- SET IDLE_TIMEOUT command, 2-11
- SET INBOUND_CONNECT_TIMEOUT command
 - of Listener Control utility, 1-9
- SET INBOUND_CONNECT_TIMEOUT command, of
 - Oracle Connection Manager Control utility, 2-12
- SET LOG_DIRECTORY command
 - of Listener Control utility, 1-10
 - of Oracle Connection Manager Control Utility, 2-12
- SET LOG_FILE command, 1-11
- SET LOG_LEVEL command, 2-13, 2-16
- SET LOG_STATUS command, 1-11
- SET OUTBOUND_CONNECT_TIMEOUT command, 2-14
- SET PASSWORD command
 - of Oracle Connection Manager Control utility, 2-14
- SET SAVE_CONFIG_ON_STOP command, 1-12
 - of Listener Control utility, 1-12
- SET SESSION_TIMEOUT command, 2-15
- SET TRACE_DIRECTORY command, 2-15
- SET TRACE_LEVEL command, 2-16
- SET TRC_DIRECTORY command, 1-12
- SET TRC_FILE command, 1-13
- SET TRC_LEVEL command, 1-13
- SET USE_PLUGANDPLAY command, 1-14
- SHOW ALL command, 2-17
- SHOW command
 - of Listener Control utility, 1-14
 - of Oracle Connection Manager Control utility, 2-17
- SHOW CONNECTIONS command, 2-18
- SHOW CURRENT_LISTENER command, 1-15
- SHOW DEFAULTS command, 2-20
- SHOW DISPLAYMODE command
 - of Listener Control utility, 1-15
- SHOW EVENTS command, 2-20
- SHOW GATEWAYS command, 2-21
- SHOW INBOUND_CONNECT_TIMEOUT command, 1-15
- SHOW LOG_DIRECTORY command, 1-15
- SHOW LOG_FILE command, 1-15
- SHOW LOG_STATUS command, 1-15
- SHOW PARAMETERS command, 2-21
- SHOW RAWMODE command, 1-15
- SHOW RULES command, 2-22
- SHOW SAVE_CONFIG_ON_STOP command, 1-15
- SHOW SERVICES command, 2-23
- SHOW STATUS command, 2-24
- SHOW TRC_DIRECTORY command, 1-15
- SHOW TRC_FILE command, 1-15
- SHOW TRC_LEVEL command, 1-15
- SHOW VERSION command, 2-25
- SHUTDOWN command, 2-25
- SID_LIST_listener_name networking parameter, 7-7
- SOURCE_ROUTE networking parameter, 6-10
- SQL*Net
 - coexistence issues, B-2
 - Oracle9i databases, B-2
 - using SERVICE_NAME networking parameter, B-2
- SQLNET.ALLOWED_LOGON_VERSION networking parameter, 5-9
- SQLNET.ALTERNATE_PORT networking parameter, 5-19
- SQLNET.AUTHENTICATION_KERBEROS5_SERVICE networking parameter, 5-7
- SQLNET.AUTHENTICATION_SERVICES networking parameter, 5-11
- SQLNET.CLIENT_REGISTRATION networking parameter, 5-12
- SQLNET.CRYPTO_CHECKSUM_CLIENT networking parameter, 5-12
- SQLNET.CRYPTO_CHECKSUM_SERVER networking parameter, 5-12
- SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT networking parameter, 5-13
- SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER networking parameter, 5-13
- SQLNET.CRYPTO_SEED networking parameter, A-1
- SQLNET.ENCRYPTION_CLIENT networking parameter, 5-13
- SQLNET.ENCRYPTION_SERVER networking parameter, 5-14
- SQLNET.ENCRYPTION_TYPES_CLIENT networking parameter, 5-14
- SQLNET.ENCRYPTION_TYPES_SERVER networking parameter, 5-15
- SQLNET.EXPIRE_TIME networking parameter, 5-15
- SQLNET.KERBEROS5_CC_NAME networking parameter, 5-17
- SQLNET.KERBEROS5_CLOCKSKEW networking parameter, 5-17
- SQLNET.KERBEROS5_CONF networking parameter, 5-17
- SQLNET.KERBEROS5_KEYTAB networking parameter, 5-18
- SQLNET.KERBEROS5_REALMS networking parameter, 5-18
- sqlnet.ora file
 - configuration parameter reference, 5-1 to 5-31
 - diagnostic parameters
 - ADR_BASE, 5-33
 - DIAG_ADR_ENABLED, 5-33
 - LOG_DIRECTORY_CLIENT, 5-35
 - LOG_DIRECTORY_SERVER, 5-36
 - LOG_FILE_CLIENT, 5-36
 - LOG_FILE_SERVER, 5-36
 - TRACE_DIRECTORY_CLIENT, 5-36
 - TRACE_DIRECTORY_SERVER, 5-37
 - TRACE_FILE_CLIENT, 5-37

TRACE_FILE_SERVER, 5-37
 TRACE_FILELEN_CLIENT, 5-38
 TRACE_FILELEN_SERVER, 5-38
 TRACE_FILENO_CLIENT, 5-38
 TRACE_FILENO_SERVER, 5-38
 TRACE_LEVEL_CLIENT, 5-33
 TRACE_LEVEL_SERVER, 5-34
 TRACE_TIMESTAMP_CLIENT, 5-34
 TRACE_TIMESTAMP_SERVER, 5-34
 TRACE_UNIQUE_CLIENT, 5-39

parameters

BEQUEATH_DETACH, 5-3
 DEFAULT_SDU_SIZE, 5-4
 DISABLE_OOB, 5-4
 NAMES_DIRECTORY_PATH, 5-5
 NAMESCTL.TRACE_UNIQUE, 5-7
 NAMES.DEFAULT_DOMAIN, 5-4
 NAMES.LADP_AUTHENTICATE_BIND, 5-5
 NAMES.LDAP_CONN_TIMEOUT, 5-6
 NAMES.LDAP_PERSISTENT_SESSION, 5-6
 RECV_BUF_SIZE, 5-7
 SDP.PF_INET_SDP, 5-7
 SEC_USER_AUDIT_ACTION_BANNER, 5-7
 SEC_USER_UNAUTHORIZED_ACCESS_BANNER, 5-8
 SEND_BUF_SIZE, 5-8
 SQLNET.ALLOWED_LOGON_VERSION, 5-9
 SQLNET.ALTERNATE_PORT, 5-19
 SQLNET.AUTHENTICATION_KERBEROS5_SERVICE, 5-7
 SQLNET.AUTHENTICATION_SERVICES, 5-11
 SQLNET.CLIENT_REGISTRATION, 5-12
 SQLNET.CRYPTO_CHECKSUM_CLIENT, 5-12
 SQLNET.CRYPTO_CHECKSUM_SERVER, 5-12
 SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT, 5-13
 SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER, 5-13
 SQLNET.ENCRYPTION_SERVER, 5-14
 SQLNET.ENCRYPTION_TYPES_CLIENT, 5-14
 SQLNET.ENCRYPTION_TYPES_SERVER, 5-15
 SQLNET.ENCYPRTION_CLIENT, 5-13
 SQLNET.EXPIRE_TIME, 5-15
 SQLNET.KERBEROS5_CC_NAME, 5-17
 SQLNET.KERBEROS5_CLOCKSKEW, 5-17
 SQLNET.KERBEROS5_CONF, 5-17
 SQLNET.KERBEROS5_KEYTAB, 5-18
 SQLNET.KERBEROS5_REALMS, 5-18
 SQLNET.OUTBOUND_CONNECT_TIMEOUT, 5-18
 SQLNET.RADIUS_ALTERNATE, 5-19
 SQLNET.RADIUS_ALTERNATE_RETRIES, 5-19
 SQLNET.RADIUS_AUTHENTICATION, 5-20
 SQLNET.RADIUS_AUTHENTICATION_INTERFACE, 5-20
 SQLNET.RADIUS_AUTHENTICATION_PORT, 5-20
 SQLNET.RADIUS_AUTHENTICATION_RETRIES, 5-20
 SQLNET.RADIUS_AUTHENTICATION_TIMEOUT, 5-21
 SQLNET.RADIUS_CHALLENGE_RESPONSE, 5-21
 SQLNET.RADIUS_SECRET, 5-21
 SQLNET.RADIUS_SEND_ACCOUNTING, 5-22
 SQLNET.RECV_TIMEOUT, 5-22
 SQLNET.SEND_TIMEOUT, 5-22
 SSL_CERT_FILE, 5-23
 SSL_CERT_PATH, 5-24
 SSL_CERT_REVOCATION, 5-23
 SSL_CIPHER_SUITES, 5-24
 SSL_SERVER_DN_MATCH, 5-25
 SSL_VERSION, 5-26
 SSL.CLIENT_AUTHENTICATION, 5-25
 TCP.CONNECT_TIMEOUT, 5-26
 TCP.EXCLUDED_NODES, 5-26
 TCP.INVITED_NODES, 5-27
 TCP.NODELAY, 5-27
 TCP.VALIDNODE_CHECKING, 5-27
 TNSPING.TRACE_DIRECTORY, 5-28
 TNSPING.TRACE_LEVEL, 5-28
 USE_DEDICATED_SERVER, 5-29
 WALLET_LOCATION, 5-29

unsupported parameters

SQLNET.CRYPTO_SEED, A-1
 SQLNET.OUTBOUND_CONNECT_TIMEOUT networking parameter, 5-18
 SQLNET.RADIUS_ALTERNATE networking parameter, 5-19
 SQLNET.RADIUS_ALTERNATE_RETRIES networking parameter, 5-19
 SQLNET.RADIUS_AUTHENTICATION networking parameter, 5-20
 SQLNET.RADIUS_AUTHENTICATION_INTERFACE networking parameter, 5-20
 SQLNET.RADIUS_AUTHENTICATION_PORT networking parameter, 5-20
 SQLNET.RADIUS_AUTHENTICATION_RETRIES networking parameter, 5-20
 SQLNET.RADIUS_AUTHENTICATION_TIMEOUT networking parameter, 5-21
 SQLNET.RADIUS_CHALLENGE_RESPONSE networking parameter, 5-21
 SQLNET.RADIUS_SECRET networking parameter, 5-21
 SQLNET.RADIUS_SEND_ACCOUNTING networking parameter, 5-22
 SQLNET.RECV_TIMEOUT networking parameter, 5-22
 SQLNET.SEND_TIMEOUT networking parameter, 5-22
 SRC networking parameter, 8-8
 SRV networking parameter, 8-8

- SSL_CERT_FILE networking parameter, 5-23
- SSL_CERT_PATH networking parameter, 5-24
- SSL_CERT_REVOCATION networking parameter, 5-23
- SSL_CIPHER_SUITES networking parameter, 5-24
- SSL_CLIENT_AUTHENTICATION control parameter, 7-9
- SSL_SERVER_CERT_DN networking parameter, 6-16, 6-17
- SSL_SERVER_DN_MATCH networking parameter, 5-25
- SSL_VERSION networking parameter, 5-26
- SSL.CLIENT_AUTHENTICATION networking parameter, 5-25
- START command
 - of Listener Control utility, 1-15
- STARTUP command, 2-26, 2-27
- STATUS command
 - of Listener Control utility, 1-17
- STOP command
 - of Listener Control utility, 1-18
- SUBSCRIBE_FOR_NODE_DOWN_EVENT_*listener_name* control parameter, 7-10
- SUSPEND GATEWAY command, 2-27
- syntax
 - rules for network configuration files, 3-1

T

- TAF
 - see Transparent Application Failover (TAF)
- TCP.CONNECT_TIMEOUT networking parameter, 5-26
- TCP.EXCLUDED_NODES networking parameter, 5-26
- TCP.INVITED_NODES networking parameter, 5-27
- TCP/IP protocol
 - HOST parameter, 4-2
 - parameters for addresses, 4-2
 - PORT parameter, 4-2, 4-3
 - PROTOCOL parameter, 4-2
- TCP/IP with SSL protocol
 - HOST parameter, 4-3
 - parameters for addresses, 4-3
 - PORT parameter, 4-3
 - PROTOCOL parameter, 4-3
- TCP.NODELAY networking parameter, 5-27
- TCP.VALIDNODE_CHECKING networking parameter, 5-27
- terminated connection detection
 - limitations, 5-16
 - SQLNET.EXPIRE_TIME parameter, 5-15
- TNS_ADMIN environment variable, 6-1
- tnsnames.ora file
 - configuration parameter reference, 6-16, 6-17
 - parameters
 - ADDRESS, 6-5
 - ADDRESS_LIST, 6-6
 - BACKUP, 6-12
 - CONNECT_DATA, 6-11

- CONNECT_TIMEOUT, 6-17
- DELAY, 6-13
- DESCRIPTION, 6-5
- DESCRIPTION_LIST, 6-5
- ENABLE, 6-6
- FAILOVER, 6-7
- FAILOVER_MODE, 6-12
- GLOBAL_NAME, 6-13
- HS, 6-13
- INSTANCE_NAME, 6-14
- LOAD_BALANCE, 6-7
- METHOD, 6-12
- RDB_DATABASE, 6-14
- RECV_BUF_SIZE, 6-8
- RETRIES, 6-13
- RETRY_COUNT, 6-17
- SDU, 6-9
- SECURITY, 6-16
- SEND_BUF_SIZE, 6-9
- SERVER, 6-14
- SERVICE_NAME, 6-15, B-2
- SOURCE_ROUTE, 6-10
- SSL_SERVER_CERT_DN, 6-16, 6-17
- TYPE, 6-12
- TYPE_OF_SERVICE, 6-11
- TNSPING.TRACE_DIRECTORY networking parameter, 5-28
- TNSPING.TRACE_LEVEL networking parameter, 5-28
- TRACE command, 1-18
- trace files
 - cycling
 - client, 5-38
 - database server, 5-38
- TRACE_DIRECTORY diagnostic parameter, 8-13
- TRACE_DIRECTORY_CLIENT diagnostic parameter, 5-36
- TRACE_DIRECTORY_*listener_name* diagnostic parameter, 7-15
- TRACE_DIRECTORY_SERVER diagnostic parameter, 5-37
- TRACE_FILE_CLIENT diagnostic parameter, 5-37
- TRACE_FILE_*listener_name* diagnostic parameter, 7-15
- TRACE_FILE_SERVER diagnostic parameter, 5-37
- TRACE_FILELEN diagnostic parameter, 8-13
- TRACE_FILELEN_CLIENT diagnostic parameter, 5-38
- TRACE_FILELEN_SERVER diagnostic parameter, 5-38
- TRACE_FILEN_*listener_name* diagnostic parameter, 7-15
- TRACE_FILENO diagnostic parameter, 8-14
- TRACE_FILENO_CLIENT diagnostic parameter, 5-38
- TRACE_FILENO_*listener_name* diagnostic parameter, 7-16
- TRACE_FILENO_SERVER diagnostic parameter, 5-38
- TRACE_LEVEL diagnostic parameter, 8-11

TRACE_LEVEL_CLIENT diagnostic parameter, 5-33
TRACE_LEVEL_listener_name diagnostic
parameter, 7-13
TRACE_LEVEL_SERVER diagnostic
parameter, 5-34
TRACE_TIMESTAMP_CLIENT diagnostic
parameter, 5-34
TRACE_TIMESTAMP_listener_name diagnostic
parameter, 7-14
TRACE_TIMESTAMP_SERVER diagnostic
parameter, 5-34
TRACE_UNIQUE_CLIENT diagnostic
parameter, 5-39
tracing
 cycling files
 client, 5-38
 database server, 5-38
Transparent Application Failover (TAF)
 parameters
TYPE networking parameter, 6-12
TYPE_OF_SERVICE networking parameter, 6-11

U

USE_DEDICATED_SERVER networking
parameter, 5-29

V

VERSION command
 of Listener Control utility, 1-19

W

WALLET_LOCATION control parameter, 7-10
WALLET_LOCATION networking parameter, 5-29
WALLET_OVERRIDE, 5-31