

Oracle9i Net Services

Reference Guide

Release 2 (9.2)

March 2002

Part No. A96581-01

Oracle9i Net Services Reference Guide, Release 2 (9.2)

Part No. A96581-01

Copyright © 2001, 2002 Oracle Corporation. All rights reserved.

Primary Author: Deborah Steiner

Graphic Designer: Valarie Moore

Contributors: Henry Abrecht, Robert Achacoso, Gaurav Agarwal, Mark Bauer, Ruth Baylis, Omar Bellal, David Chen, Howard Cohen, Toby Close, Pramodini Gattu, Stanley Guan, Harvey Eneman, Stella Kister, Shuvayu Kanjila, Cynthia Kibbe, Vincent Lee, Catherine Luu, Scot McKinley, Ed Miner, Peter Povinec, Richard Smith, Juliet Ta, Norman Woo, and Jose Wong

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent and other intellectual and industrial property laws. Reverse engineering, disassembly or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle Corporation disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and Oracle Names, Oracle Rdb, Oracle Store, Oracle9i, Oracle8i, Oracle8, PL/SQL, Secure Network Services, SQL*Net, and SQL*Plus are trademarks or registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	xxi
Preface.....	xxiii
Audience	xxiv
Organization.....	xxiv
Related Documentation	xxv
Conventions.....	xxvi
Documentation Accessibility	xxxii
Part I Control Utilities	
1 Listener Control Utility	
Listener Control Utility Overview	1-2
SET and SHOW Commands of the Listener Control Utility	1-3
Distributed Operations.....	1-4
Listener Security	1-4
Listener Control Utility Commands.....	1-5
CHANGE_PASSWORD	1-5
EXIT	1-7
HELP	1-8
QUIT	1-9
RELOAD	1-10
SAVE_CONFIG.....	1-11
SERVICES	1-12
SET	1-14

SET CURRENT_LISTENER.....	1-16
SET DISPLAYMODE.....	1-17
SET LOG_DIRECTORY	1-18
SET LOG_FILE	1-19
SET LOG_STATUS	1-20
SET PASSWORD.....	1-21
SET SAVE_CONFIG_ON_STOP	1-22
SET STARTUP_WAITTIME.....	1-23
SET TRC_DIRECTORY	1-24
SET TRC_FILE.....	1-25
SET TRC_LEVEL.....	1-26
SHOW	1-27
SPAWN	1-29
START.....	1-30
STATUS.....	1-32
STOP	1-34
TRACE.....	1-35
VERSION	1-36

2 Oracle Connection Manager Control Utility

Oracle Connection Manager Control Utility Overview	2-2
SET Commands of the Oracle Connection Manager Control Utility	2-3
Distributed Operations.....	2-4
Oracle Connection Manager Control Utility Commands	2-5
ACCEPT_CONNECTIONS.....	2-5
CLOSE_RELAY	2-6
EXIT	2-7
HELP.....	2-8
QUIT	2-9
SET	2-10
SET AUTHENTICATION_LEVEL.....	2-11
SET DISPLAYMODE.....	2-12
SET LOG_LEVEL.....	2-13
SET RELAY_STATISTICS.....	2-14
SHOW	2-15

SHOW ADDRESS.....	2-16
SHOW ALL.....	2-17
SHOW DISPLAYMODE.....	2-19
SHOW PROFILE.....	2-20
SHOW RELAY	2-21
SHOW RULES.....	2-23
SHUTDOWN.....	2-24
START	2-25
STATS.....	2-27
STATUS.....	2-29
STOP	2-31
STOPNOW.....	2-32
VERSION	2-33

3 Oracle Names Control Utility

Oracle Names Control Utility Overview.....	3-2
SET and SHOW Commands of the Oracle Names Control Utility.....	3-3
Distributed Operations.....	3-3
Oracle Names Server Security.....	3-5
Confirmation Mode in the Oracle Names Control Utility.....	3-6
Oracle Names Control Utility Commands.....	3-6
DELEGATE_DOMAIN.....	3-6
DOMAIN_HINT.....	3-7
DUMP_ALIAS.....	3-9
DUMP_LDAP	3-13
DUMP_TNSNAMES.....	3-17
EXIT	3-18
FLUSH.....	3-19
FLUSH_NAME	3-20
HELP	3-21
LIST_DELEGATED	3-23
LIST_DOMAINS.....	3-24
LIST_OBJECTS.....	3-25
LOAD_TNSNAMES.....	3-26
LOG_STATS	3-28

PASSWORD.....	3-29
PING	3-30
QUERY	3-31
QUIT	3-33
REGISTER	3-34
REGISTER_NS.....	3-37
RELOAD	3-39
REORDER_NS.....	3-40
REPEAT	3-42
RESET_STATS	3-44
RESTART	3-45
SAVE_CONFIG.....	3-46
SET	3-47
SET CACHE_CHECKPOINT_INTERVAL.....	3-48
SET DEFAULT_DOMAIN.....	3-49
SET FORWARDING_AVAILABLE	3-50
SET LOG_FILE_NAME.....	3-52
SET LOG_STATS_INTERVAL.....	3-53
SET NAMESCTL_TRACE_LEVEL.....	3-54
SET PASSWORD.....	3-56
SET REQUESTS_ENABLED.....	3-57
SET RESET_STATS_INTERVAL	3-58
SET SAVE_CONFIG_INTERVAL	3-59
SET SAVE_CONFIG_ON_STOP	3-60
SET SERVER	3-61
SET TRACE_FILE_NAME.....	3-62
SET TRACE_LEVEL	3-63
SHOW	3-64
SHOW CACHE_CHECKPOINT_INTERVAL	3-66
SHOW DEFAULT_DOMAIN	3-67
SHOW FORWARDING_AVAILABLE.....	3-68
SHOW LOG_FILE_NAME	3-69
SHOW LOG_STATS_INTERVAL	3-70
SHOW NAMESCTL_TRACE_LEVEL	3-71
SHOW REQUESTS_ENABLED	3-72

SHOW RESETS_STATS_INTERVAL	3-73
SHOW SAVE_CONFIG_INTERVAL	3-74
SHOW SAVE_CONFIG_ON_STOP	3-75
SHOW SERVER	3-76
SHOW STATUS	3-77
SHOW SYSTEM_QUERIES.....	3-78
SHOW TRACE_FILE_NAME.....	3-79
SHOW TRACE_LEVEL	3-80
SHOW VERSION.....	3-81
SHUTDOWN.....	3-82
START	3-83
START_CLIENT_CACHE.....	3-85
STARTUP.....	3-86
STATUS.....	3-87
STOP	3-88
TIMED_QUERY	3-89
UNREGISTER	3-91
UNREGISTER_NS.....	3-93
VERSION	3-95

Part II Configuration Parameters

4 Syntax Rules for Configuration Files

Configuration File Syntax Overview	4-2
Further Syntax Rules for Configuration Files	4-3
Network Character Set.....	4-4
Character Set.....	4-4

5 Protocol Address Configuration

ADDRESSES and ADDRESS_LISTs	5-2
ADDRESS.....	5-2
ADDRESS_LIST	5-2
Protocol Parameters	5-3
Recommended Port Numbers	5-5
Port Number Limitations	5-6

6 Profile Parameters (sqlnet.ora)

Overview of Profile Configuration File	6-2
Profile Parameters	6-3
BEQUEATH_DETACH.....	6-3
DISABLE_OOB.....	6-4
LOG_DIRECTORY_CLIENT	6-4
LOG_DIRECTORY_SERVER	6-5
LOG_FILE_CLIENT	6-5
LOG_FILE_SERVER.....	6-5
NAMES.CONNECT_TIMEOUT	6-6
NAMES.DCE.PREFIX	6-6
NAMES.DEFAULT_DOMAIN	6-7
NAMES.DIRECTORY_PATH.....	6-7
NAMES.INITIAL_RETRY_TIMEOUT.....	6-9
NAMES.MAX_OPEN_CONNECTIONS	6-9
NAMES.MESSAGE_POOL_START_SIZE.....	6-10
NAMES.NIS.META_MAP.....	6-10
NAMES.PREFERRED_SERVERS	6-11
NAMES.REQUEST_RETRIES	6-11
NAMESCTL.ECHO	6-12
NAMESCTL.INTERNAL_ENCRYPT_PASSWORD	6-13
NAMESCTL.INTERNAL_USE	6-13
NAMESCTL.NO_INITIAL_SERVER.....	6-14
NAMESCTL.NOCONFIRM.....	6-14
NAMESCTL.SERVER_PASSWORD.....	6-15
NAMESCTL.TRACE_LEVEL.....	6-15
NAMESCTL.TRACE_FILE.....	6-16

NAMESCTL.TRACE_DIRECTORY.....	6-16
NAMESCTL.TRACE_UNIQUE.....	6-17
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE.....	6-17
SQLNET.AUTHENTICATION_GSSAPI_SERVICE.....	6-18
SQLNET.AUTHENTICATION_SERVICES.....	6-18
SQLNET.CLIENT_REGISTRATION.....	6-19
SQLNET.CRYPTO_CHECKSUM_CLIENT.....	6-19
SQLNET.CRYPTO_CHECKSUM_SERVER.....	6-20
SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT.....	6-21
SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER.....	6-21
SQLNET.CRYPTO_SEED.....	6-22
SQLNET.ENCRYPTION_CLIENT.....	6-22
SQLNET.ENCRYPTION_SERVER.....	6-23
SQLNET.ENCRYPTION_TYPES_CLIENT.....	6-24
SQLNET.ENCRYPTION_TYPES_SERVER.....	6-25
SQLNET.EXPIRE_TIME.....	6-26
SQLNET.KERBEROS5_CC_NAME.....	6-27
SQLNET.KERBEROS5_CLOCKSKEW.....	6-27
SQLNET.KERBEROS5_CONF.....	6-28
SQLNET.KERBEROS5_KEYTAB.....	6-28
SQLNET.KERBEROS5_REALMS.....	6-29
SQLNET.RADIUS_ALTERNATE.....	6-30
SQLNET.RADIUS_ALTERNATE_PORT.....	6-30
SQLNET.RADIUS_ALTERNATE_RETRIES.....	6-31
SQLNET.RADIUS_AUTHENTICATION.....	6-31
SQLNET.RADIUS_AUTHENTICATION_INTERFACE.....	6-32
SQLNET.RADIUS_AUTHENTICATION_PORT.....	6-32
SQLNET.RADIUS_AUTHENTICATION_RETRIES.....	6-33
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT.....	6-33
SQLNET.RADIUS_CHALLENGE_RESPONSE.....	6-34
SQLNET.RADIUS_SECRET.....	6-34
SQLNET.RADIUS_SEND_ACCOUNTING.....	6-35
SSL_CIPHER_SUITES.....	6-35
SSL_CLIENT_AUTHENTICATION.....	6-36
SSL_SERVER_DN_MATCH.....	6-36
SSL_VERSION.....	6-37

TCP.EXCLUDED_NODES	6-38
TCP.INVITED_NODES.....	6-38
TCP.VALIDNODE_CHECKING.....	6-39
TCP.NODELAY	6-39
TNSPING.TRACE_DIRECTORY	6-40
TNSPING.TRACE_LEVEL	6-40
TRACE_DIRECTORY_CLIENT.....	6-41
TRACE_DIRECTORY_SERVER	6-41
TRACE_FILE_CLIENT	6-42
TRACE_FILE_SERVER.....	6-42
TRACE_FILELEN_CLIENT	6-43
TRACE_FILELEN_SERVER.....	6-43
TRACE_FILENO_CLIENT	6-44
TRACE_FILENO_SERVER.....	6-44
TRACE_LEVEL_CLIENT	6-45
TRACE_LEVEL_SERVER.....	6-46
TRACE_TIMESTAMP_CLIENT	6-46
TRACE_TIMESTAMP_SERVER.....	6-47
TRACE_UNIQUE_CLIENT	6-47
USE_CMAN.....	6-48
USE_DEDICATED_SERVER.....	6-49
WALLET_LOCATION.....	6-50

7 Local Naming Parameters (tnsnames.ora)

Overview of Local Naming Parameters.....	7-2
General Syntax of tnsnames.ora.....	7-2
Multiple Descriptions in tnsnames.ora	7-3
Multiple Address Lists in tnsnames.ora.....	7-4
Connect-Time Failover and Client Load Balancing with Oracle Connection Managers.....	7-5
Local Naming Parameters.....	7-7
Connect Descriptor Descriptions.....	7-7
DESCRIPTION	7-7
DESCRIPTION_LIST.....	7-8
Protocol Address Section.....	7-8
ADDRESS	7-8
ADDRESS_LIST	7-9

Optional Parameters for Lists	7-10
FAILOVER.....	7-10
LOAD_BALANCE	7-11
SDU.....	7-12
SOURCE_ROUTE.....	7-13
TYPE_OF_SERVICE.....	7-14
Connect Data Section	7-15
CONNECT_DATA.....	7-15
FAILOVER_MODE	7-16
GLOBAL_NAME.....	7-17
HS	7-18
INSTANCE_NAME	7-18
RDB_DATABASE.....	7-19
SERVER.....	7-20
SERVICE_NAME.....	7-21
SID	7-22
Security Section.....	7-23
SECURITY	7-23
SSL_SERVER_CERT_DN	7-24

8 Listener Parameters (listener.ora)

Overview of Listener Configuration File	8-2
Listener Parameters	8-3
Protocol Address Section.....	8-4
DESCRIPTION.....	8-4
ADDRESS	8-4
QUEUESIZE.....	8-5
Static Service Registration (SID_LIST) Section.....	8-6
SID_LIST	8-6
SID_DESC.....	8-7
ENVS.....	8-7
GLOBAL_DBNAME.....	8-9
ORACLE_HOME.....	8-9
PROGRAM.....	8-10
SID_NAME.....	8-10

Control Parameters.....	8-11
ADMIN_RESTRICTIONS_listener_name.....	8-11
LOG_DIRECTORY_listener_name.....	8-12
LOG_FILE_listener_name.....	8-12
LOGGING_listener_name.....	8-13
PASSWORDS_listener_name.....	8-13
SAVE_CONFIG_ON_STOP_listener_name.....	8-14
SSL_CLIENT_AUTHENTICATION.....	8-14
STARTUP_WAIT_TIME_listener_name.....	8-15
TRACE_DIRECTORY_listener_name.....	8-15
TRACE_FILE_listener_name.....	8-16
TRACE_FILELEN_listener_name.....	8-16
TRACE_FILENO_listener_name.....	8-17
TRACE_LEVEL_listener_name.....	8-17
TRACE_TIMESTAMP_listener_name.....	8-18
WALLET_LOCATION.....	8-18

9 Oracle Connection Manager Parameters (cman.ora)

Overview of Oracle Connection Manager Configuration File.....	9-2
Oracle Connection Manager Parameters.....	9-3
Listening Endpoint List.....	9-3
CMAN.....	9-3
CMAN_ADMIN.....	9-4
Route List.....	9-4
CMAN_RULES.....	9-4
Parameter List.....	9-6
CMAN_PROFILE.....	9-6
ANSWER_TIMEOUT.....	9-8
AUTHENTICATION_LEVEL.....	9-8
LOG_LEVEL.....	9-8
MAX_FREELIST_BUFFERS.....	9-8
MAXIMUM_CONNECT_DATA.....	9-9
MAXIMUM_RELAYS.....	9-9
RELAY_STATISTICS.....	9-9
REMOTE_ADMIN.....	9-10

SHOW_TNS_INFO	9-10
TRACING	9-10
TRACE_DIRECTORY	9-11
TRACE_FILELEN.....	9-11
TRACE_FILENO	9-11
TRACE_TIMESTAMP	9-11
USE_ASYNC_CALL	9-12

10 Oracle Names Parameters (names.ora)

Overview of Oracle Names Server Configuration File.....	10-2
Oracle Names Parameters	10-3
NAMES.ADDRESSES	10-3
NAMES.ADMIN_REGION.....	10-4
NAMES.AUTHORITY_REQUIRED	10-7
NAMES.AUTO_REFRESH_EXPIRE.....	10-7
NAMES.AUTO_REFRESH_RETRY.....	10-8
NAMES.CACHE_CHECKPOINT_FILE	10-8
NAMES.CACHE_CHECKPOINT_INTERVAL	10-9
NAMES.CONFIG_CHECKPOINT_FILE.....	10-9
NAMES.CONNECT_TIMEOUT	10-10
NAMES.DEFAULT_FORWARDERS	10-11
NAMES.DEFAULT_FORWARDERS_ONLY.....	10-12
NAMES.DOMAIN_CHECKPOINT_FILE.....	10-12
NAMES.DOMAINS	10-13
NAMES.DOMAIN_HINTS	10-15
NAMES.FORWARDING_AVAILABLE	10-17
NAMES.FORWARDING_DESIRED	10-17
NAMES.KEEP_DB_OPEN	10-18
NAMES.LOG_DIRECTORY	10-18
NAMES.LOG_FILE	10-19
NAMES.LOG_STATS_INTERVAL.....	10-19
NAMES.LOG_UNIQUE	10-20
NAMES.MAX_OPEN_CONNECTIONS	10-20
NAMES.MAX_REFORWARDS.....	10-21
NAMES.MESSAGE_POOL_START_SIZE.....	10-21

NAMES.NO_MODIFY_REQUESTS	10-22
NAMES.NO_REGION_DATABASE	10-23
NAMES.PASSWORD	10-23
NAMES.RESET_STATS_INTERVAL	10-24
NAMES.SAVE_CONFIG_ON_STOP	10-24
NAMES.SERVER_NAME	10-25
NAMES.TOPOLOGY_CHECKPOINT_FILE	10-26
NAMES.TRACE_DIRECTORY	10-26
NAMES.TRACE_FILE	10-27
NAMES.TRACE_FUNC	10-27
NAMES.TRACE_LEVEL	10-28
NAMES.TRACE_UNIQUE	10-28

11 Directory Usage Parameters (ldap.ora)

Overview of Directory Server Usage File	11-2
Directory Usage Parameters	11-2
DIRECTORY_SERVERS	11-2
DIRECTORY_SERVER_TYPE	11-3
DEFAULT_ADMIN_CONTEXT	11-3

Part III Appendixes

A LDAP Schema for Oracle Net Services

Structural Object Classes	A-2
orclDBServer	A-2
orclNetService	A-2
orclNetServiceAlias	A-3
orclNetDescription	A-3
orclNetDescriptionList	A-4
orclNetAddress	A-4
orclNetAddressList	A-4
Attributes	A-5

B Commands and Parameters Not Supported in This Release

Overview of Unsupported Features	B-2
Unsupported Parameters	B-4
Unsupported Control Utility Commands	B-6

Glossary

Index

List of Figures

7-1	Basic Format of tnsnames.ora File	7-2
7-2	Net Service Name with Multiple Connect Descriptors in tnsnames.ora	7-3
7-3	Multiple Address Lists in tnsnames.ora	7-4
7-4	Multiple Oracle Connection Manager Addresses in tnsnames.ora	7-5
7-5	Client Load Balancing in tnsnames.ora.....	7-6
8-1	Example listener.ora File	8-3
9-1	Example cman.ora File.....	9-2
10-1	Example names.ora File.....	10-2

List of Tables

5-1	Protocol-Specific Parameters	5-3
5-2	Recommended Port Numbers	5-5
A-1	LDAP Schema Attributes for Oracle Net Services.....	A-5
B-1	Unsupported Networking Parameters.....	B-4
B-2	Unsupported Network Control Utility Commands.....	B-6

Send Us Your Comments

Oracle9i Net Services Reference Guide, Release 2 (9.2)

Part No. A96581-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: infodev_us@oracle.com
- FAX: (650) 506-7227 Attn: Server Technologies Documentation Manager
- Postal service:

Oracle Corporation
Server Technologies Documentation
500 Oracle Parkway, Mailstop 4op11
Redwood Shores, CA 94065
USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

The *Oracle9i Net Services Reference Guide* provides the information you need to understand and use Oracle Net Services and its related applications and components.

This preface contains these topics:

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

Audience

Oracle9i Net Services Reference Guide is intended for network administrators who are responsible for configuring and administering network components.

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

Organization

This document contains:

Part I, "Control Utilities"

Chapter 1, "Listener Control Utility"

This chapter describes the **Listener Control utility** commands.

Chapter 2, "Oracle Connection Manager Control Utility"

This chapter describes the **Oracle Connection Manager Control utility** commands.

Chapter 3, "Oracle Names Control Utility"

This chapter describes the **Oracle Names Control utility** commands.

Part II, "Configuration Parameters"

Chapter 4, "Syntax Rules for Configuration Files"

This chapter describes the syntax rules for networking configuration files.

Chapter 5, "Protocol Address Configuration"

This chapter describes how to configure **protocol addresses**.

Chapter 6, "Profile Parameters (sqlnet.ora)"

This chapter describes the `sqlnet.ora` file parameters.

Chapter 7, "Local Naming Parameters (tnsnames.ora)"

This chapter describes the `tnsnames.ora` file parameters.

Chapter 8, "Listener Parameters (listener.ora)"

This chapter describes the `listener.ora` file parameters.

Chapter 9, "Oracle Connection Manager Parameters (cman.ora)"

This chapter describes the `cman.ora` file parameters.

Chapter 10, "Oracle Names Parameters (names.ora)"

This chapter describes the `names.ora` file parameters.

Chapter 11, "Directory Usage Parameters (ldap.ora)"

This chapter describes the `ldap.ora` file parameters.

Part III, "Appendixes"

Appendix A, "LDAP Schema for Oracle Net Services"

This appendix describes the **Oracle schema** for Oracle Net Services.

Appendix B, "Commands and Parameters Not Supported in This Release"

This appendix describes the control utility commands and parameters no longer supported by Oracle Net Services.

Glossary

Related Documentation

For more information, see these Oracle resources:

- *Oracle Net Services Administrator's Guide*
- Oracle9i documentation set

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 *Oracle9i Sample Schemas* for information on how these schemas were created and how you can use them yourself.

In North America, printed documentation is available for sale in the Oracle Store at

<http://oraclestore.oracle.com/>

Customers in Europe, the Middle East, and Africa (EMEA) can purchase documentation from

<http://www.oraclebookshop.com/>

Other customers can contact their Oracle representative to purchase printed documentation.

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

<http://otn.oracle.com/admin/account/membership.html>

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

<http://otn.oracle.com/docs/index.htm>

To access the database documentation search engine directly, please visit

<http://tahiti.oracle.com>

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- [Conventions in Text](#)
- [Conventions in Code Examples](#)
- [Conventions for Windows Operating Systems](#)

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an index-organized table .
<i>Italics</i>	Italic typeface indicates book titles or emphasis.	<i>Oracle9i Database Concepts</i> Ensure that the recovery catalog and target database do <i>not</i> reside on the same disk.
UPPERCASE monospace (fixed-width) font	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a NUMBER column. You can back up the database by using the BACKUP command. Query the TABLE_NAME column in the USER_TABLES data dictionary view. Use the DBMS_STATS.GENERATE_STATS procedure.
lowercase monospace (fixed-width) font	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter sqlplus to open SQL*Plus. The password is specified in the orapwd file. Back up the datafiles and control files in the /disk1/oracle/dbs directory. The department_id, department_name, and location_id columns are in the hr.departments table. Set the QUERY_REWRITE_ENABLED initialization parameter to true. Connect as oe user. The JRepUtil class implements these methods.
lowercase italic monospace (fixed-width) font	Lowercase italic monospace font represents placeholders or variables.	You can specify the <i>parallel_clause</i> . Run <i>Uold_release</i> .SQL where <i>old_release</i> refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[]	Brackets enclose one or more optional items. Do not enter the brackets.	DECIMAL (<i>digits</i> [, <i>precision</i>])
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces.	{ENABLE DISABLE}
	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	{ENABLE DISABLE} [COMPRESS NOCOMPRESS]
...	Horizontal ellipsis points indicate either: <ul style="list-style-type: none"> That we have omitted parts of the code that are not directly related to the example That you can repeat a portion of the code 	CREATE TABLE ... AS <i>subquery</i> ; SELECT <i>col1</i> , <i>col2</i> , ... , <i>coln</i> FROM <i>employees</i> ;
. . . .	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	SQL> SELECT NAME FROM V\$DATAFILE; NAME ----- /fsl/dbs/tbs_01.dbf /fsl/dbs/tbs_02.dbf
Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;

Convention	Meaning	Example
<i>Italics</i>	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/ <i>system_password</i> DB_NAME = <i>database_name</i>
UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;
lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files. Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;

Conventions for Windows Operating Systems

The following table describes conventions for Windows operating systems and provides examples of their use.

Convention	Meaning	Example
Choose Start >	How to start a program.	To start the Database Configuration Assistant, choose Start > Programs > Oracle - <i>HOME_NAME</i> > Configuration and Migration Tools > Database Configuration Assistant.
File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotes. If the file name begins with \\, then Windows assumes it uses the Universal Naming Convention.	c:\winnt\"system32 is the same as C:\WINNT\SYSTEM32

Convention	Meaning	Example
C:\>	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the <i>command prompt</i> in this manual.	C:\oracle\oradata>
Special characters	The backslash (\) special character is sometimes required as an escape character for the double quotation mark (") special character at the Windows command prompt. Parentheses and the single quotation mark (') do not require an escape character. Refer to your Windows operating system documentation for more information on escape and special characters.	C:\>exp scott/tiger TABLES=emp QUERY=\"WHERE job='SALESMAN' and sal<1600\" C:\>imp SYSTEM/password FROMUSER=scott TABLES=(emp, dept)
HOME_NAME	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	C:\> net start OracleHOME_NAME\TNSListener

Convention	Meaning	Example
<i>ORACLE_HOME</i> and <i>ORACLE_BASE</i>	<p>In releases prior to Oracle8i release 8.1.3, when you installed Oracle components, all subdirectories were located under a top level <i>ORACLE_HOME</i> directory that by default used one of the following names:</p> <ul style="list-style-type: none"> ■ C:\orant for Windows NT ■ C:\orawin98 for Windows 98 <p>This release complies with Optimal Flexible Architecture (OFA) guidelines. All subdirectories are not under a top level <i>ORACLE_HOME</i> directory. There is a top level directory called <i>ORACLE_BASE</i> that by default is C:\oracle. If you install the latest Oracle release on a computer with no other Oracle software installed, then the default setting for the first Oracle home directory is C:\oracle\orann, where <i>nn</i> is the latest release number. The Oracle home directory is located directly under <i>ORACLE_BASE</i>.</p> <p>All directory path examples in this guide follow OFA conventions.</p> <p>Refer to <i>Oracle9i Database Getting Starting for Windows</i> for additional information about OFA compliances and for information about installing Oracle products in non-OFA compliant directories.</p>	Go to the <i>ORACLE_BASE\ORACLE_HOME\rdms\admin</i> directory.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Part I

Control Utilities

Oracle Net Services provides control utilities to administer listeners, Oracle Names servers, and Oracle Connection Managers. 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"](#)
- [Chapter 3, "Oracle Names Control Utility"](#)

Listener Control Utility

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

This chapter contains these topics:

- [Listener Control Utility Overview](#)
- [SET and SHOW Commands of the Listener Control Utility](#)
- [Distributed Operations](#)
- [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]
```

where *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 a majority of 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 needs to 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 on UNIX operating systems and the `ORACLE_HOME\network\admin` directory on Windows operating systems
- Naming method, for example, a `tnsnames.ora` file

If the listener name is `LISTENER` and it cannot be resolved, a protocol address of TCP/IP, port 1521 is assumed.

The Listener Control utility supports several types of commands:

- Operational commands, such as `START`, `STOP`, and so forth.
- Modifier commands, such as `SET TRC_LEVEL`
- Informational commands, such as `STATUS` and `SHOW LOG_FILE`
- Operational commands, such as `EXIT`, `QUIT`, and `HELP`

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, 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:

1. Ensure that the Listener Control utility (`lsnrctl`) executable is installed.
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 1-2.

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.

Listener Security

It is important to provide security through a password for the listener. With a password, privileged operations, such as saving configuration changes or stopping the listener, used from the Listener Control utility will require a password.

Use the Listener Control utility's `CHANGE_PASSWORD` command or Oracle Net Manager to set or modify an encrypted password in the `PASSWORDS_listener_name` parameter in the `listener.ora` file. If the `PASSWORDS_listener_name` parameter is set to an unencrypted password, you must manually remove it from the `listener.ora` file prior to modifying it. If the unencrypted password is not removed, you will be unable to successfully set an encrypted password.

If the `PASSWORDS_listener_name` parameter is set in the `listener.ora` file or the `CHANGE_PASSWORD` command has been used to create a new, encrypted password, then the Listener Control utility will require a `SET PASSWORD` command prior to any protected command, such as `STOP`.

Note: If you are administering the listener remotely over an insecure network and require maximum security, configure the listener with a secure protocol address that uses the [TCP/IP with SSL protocol](#). If the listener has multiple protocol addresses, ensure that the TCP/IP with SSL protocol address is listed first in the `listener.ora` file.

Listener Control Utility Commands

This section lists and describes the Listener Control utility commands.

CHANGE_PASSWORD

Purpose

Use the `CHANGE_PASSWORD` command to establish an encrypted password or change an encrypted password set with the `PASSWORDS_listener_name` parameter in the `listener.ora` file.

Prerequisites

None

Password Required If One Has Been Set:

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl CHANGE_PASSWORD [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> CHANGE_PASSWORD [listener_name]
```

Arguments

[*listener_name*]: Specify the listener name, if the default name of `LISTENER` is not used.

Usage Notes

The Listener Control utility prompts you for the old password and then for the new one. It asks you to re-enter the new one, and then changes it. Neither the old nor the new password displays during this procedure. `CHANGE_PASSWORD` is usually followed by the `SAVE_CONFIG` command to save the new password in the `listener.ora` file. If a `SAVE_CONFIG` command is not issued, then the new password will be in effect only until the listener is shut down.

See Also: *Oracle9i Net Services Administrator's Guide* for further information about password security of the listener

Example

The following shows a new password of `lsnrc9i` being set:

```
LSNRCTL> CHANGE_PASSWORD
Old password:
New password: lsnrc9i
Reenter new password: lsnrc9i
Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris))
Password changed for LISTENER
The command completed successfully
LSNRCTL> SAVE_CONFIG
The command completed successfully
```

The following shows the password being changed from `lsnrc80` to `lsnrc90`:

```
LSNRCTL> CHANGE_PASSWORD
Old password: lsnrc8i
New password: lsnrc9i
Reenter new password: lsnrc9i
Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris))
Password changed for LISTENER
The command completed successfully
LSNRCTL> SAVE_CONFIG
The command completed successfully
```

EXIT

Purpose

Use the `EXIT` command to exit from the Listener Control utility.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

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

Use the command `HELP` to provide a list of all the Listener Control utility commands or provide syntax help for a particular Listener Control utility command.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl HELP [command]
```

From the Listener Control utility:

```
LSNRCTL> 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 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:
change_password
exit
quit
reload
services
set*
show*
spawn
start
status
stop
trace
version
```

QUIT

Purpose

Use the `QUIT` command to exit the Listener Control utility and return to the operating system prompt.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

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

Use the `RELOAD` command 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 will be unregistered and subsequently registered again.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the [SET PASSWORD](#) command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl RELOAD [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> RELOAD [listener_name]
```

Arguments

[*listener_name*]: Specify 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

Use the `SAVE_CONFIG` command to compare 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. Prior to modification of the `listener.ora` file, a backup of the file, called `listener.bak`, is created.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SAVE_CONFIG [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> SAVE_CONFIG [listener_name]
```

Arguments

[*listener_name*]: Specify the listener name, if the default name of `LISTENER` is not used.

Usage Notes

This command enables you to save all runtime configuration changes to the `listener.ora` file, which can be especially useful for saving changed encrypted passwords.

Example

```
LSNRCTL> SAVE_CONFIG listener
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
Saved LISTENER configuration parameters.
Listener Parameter File  /oracle9i/network/admin/listener.ora
Old Parameter File      /oracle9i/network/admin/listener.bak
The command completed successfully
```

SERVICES

Purpose

Use the `SERVICES` command 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

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SERVICES [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> SERVICES [listener_name]
```

Arguments

[*listener_name*]: Specify the listener name, if the default name of `LISTENER` is not used.

Usage Notes

See Also: *Oracle9i Net Services Administrator's Guide* for a complete description of SERVICES output

The `SET DISPLAYMODE` command changes the format and the detail level of the 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.acme.com` and `sales2.us.acme.com`, with a total of three service handlers.
- Service `sales1.us.acme.com` is handled by one dispatcher only.
- Service `sales2.us.acme.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.acme.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.acme.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

Use the `SET` command to alter a listener's parameter values. Parameter values 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

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SET [parameter]
```

From the Listener Control utility:

```
LSNRCTL> SET [parameter]
```

Arguments

[*parameter*]: Specify 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, 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  
log_file  
log_directory  
log_status  
password  
raw_mode  
save_config_on_stop  
startup_waittime  
trc_file  
trc_directory  
trc_level
```

SET CURRENT_LISTENER

Purpose

Use the `SET CURRENT_LISTENER` command to set the name of the listener to administer. Subsequent commands that would normally require *listener_name* can be issued without it.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Listener Control utility

```
LSNRCTL> SET CURRENT_LISTENER [listener_name]
```

Arguments

[*listener_name*]: Specify 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

Use the `SET DISPLAYMODE` command to change the format and level of detail for the `SERVICES` and `STATUS` commands.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Listener Control utility:

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

Arguments

[`compat`]: Specify to display output that is compatible with older versions of the listener.

[`normal`]: Specify to display output in a formatted and descriptive output. Oracle Corporation recommends this mode.

[`verbose`]: Specify to display all data received from the listener in a formatted and descriptive output.

[`raw`]: Specify to display all data received from the listener without any formatting. This output should be used only if recommended by Oracle Support Services.

Example

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

SET LOG_DIRECTORY

Purpose

Use the command `SET LOG_DIRECTORY` 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 on UNIX operating systems and the `ORACLE_HOME\network\log` directory on Windows NT.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SET LOG_DIRECTORY {directory}
```

From the Listener Control utility:

```
LSNRCTL> SET LOG_DIRECTORY {directory}
```

Arguments

`{directory}`: Specify the directory path of the listener log file.

Example

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

SET LOG_FILE

Purpose

Use the command `SET LOG_FILE` to set the name for the listener log file. By default, the log file name is `listener.log`.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

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}`: Specify file name of the listener log.

Example

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

SET LOG_STATUS

Purpose

Use the command `SET LOG_STATUS` to turn listener logging on or off

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

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

From the Listener Control utility:

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

Arguments

[on]: Specify to turn logging on.

[off]: Specify to turn logging off.

Example

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

SET PASSWORD

Purpose

Use the command `SET PASSWORD` prior to privileged Listener Control utility commands, such as [SAVE_CONFIG](#) and [STOP](#).

The password entered should match the one established for the `PASSWORDS_listener_name` parameter in the `listener.ora` file or set by the [CHANGE_PASSWORD](#) command.

Syntax

From the Listener Control utility:

```
LSNRCTL> SET PASSWORD  
Password: password
```

Arguments

None

Usage Notes

You can enter this command when you start up the Listener Control utility or at any time during the session as needed.

See Also: ["Listener Security"](#) on page 1-4

Example

```
LSNRCTL> SET PASSWORD  
Password: lnrc9i  
The command completed successfully
```

SET SAVE_CONFIG_ON_STOP

Purpose

Use the command `SET SAVE_CONFIG_ON_STOP` to specify whether or not changes made to the listener's parameter values 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. Prior to modification of the `listener.ora` file, a back up of the file, called `listener.bak`, is created.

To have all parameters saved right away, use the `SAVE_CONFIG` command.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this 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]: Specify to save configuration to `listener.ora`.

[off]: Specify 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 STARTUP_WAITTIME

Note: This command is deprecated in Oracle9i and will be desupported in a future release. If you require this command to run the listener, please notify Oracle Support Services.

Purpose

Use the command `SET STARTUP_WAITTIME` to specify the amount of time for the listener to wait before responding to a `START` command.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SET STARTUP_WAITTIME {time}
```

From the Listener Control utility:

```
LSNRCTL> SET STARTUP_WAITTIME {time}
```

Arguments

`{time}`: Specify the time, in seconds.

Example

```
LSNRCTL> SET STARTUP_WAITTIME 10
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
listener parameter "startup_waittime" set to 10
The command completed successfully
```

SET TRC_DIRECTORY

Purpose

Use the command `SET TRC_DIRECTORY` to set the destination directory where the listener trace files are written. By default, the trace files are written to the `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows NT.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SET TRC_DIRECTORY {directory}
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_DIRECTORY {directory}
```

Arguments

`{directory}`: Specify the directory path of the listener trace files.

Example

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

SET TRC_FILE

Purpose

Use the command `SET TRC_FILE` to set the name of the listener trace file. By default, the trace file name is `listener.trc`.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

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}`: Specify the file name of the listener trace.

Example

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

SET TRC_LEVEL

Purpose

Use the command `SET TRC_LEVEL` to set a specific level of tracing for the listener.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SET TRC_LEVEL {level}
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_LEVEL {level}
```

Arguments

`{level}`: Specify 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
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
listener parameter "trc_level" set to admin
The command completed successfully
```

SHOW

Purpose

Use the command `SHOW` to view the current listener's parameter values.

All of the `SET` parameters, except `SET PASSWORD`, have equivalent `SHOW` parameters.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to using this command.

Syntax

From the operating system:

```
lsnrctl SHOW [parameter]
```

From the Listener Control utility:

```
LSNRCTL> SHOW [parameter]
```

Arguments

[*parameter*]: Specify a `SHOW` parameter to view its configuration settings. 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.

Example

```
LSNRCTL> SHOW
```

The following properties are available with SHOW:

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

current_listener

displaymode

log_file

log_directory

log_status

rawmode

save_config_on_stop

startup_waittime

trc_file

trc_directory

trc_level

Note: SHOW STARTUP_WAITTIME is deprecated in Oracle9i and will be desupported in a future release. If you require this command to run the listener, please notify Oracle Support Services.

SPAWN

Purpose

Use the command `SPAWN` 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

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

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*]: Specify the listener name, if the default name of `LISTENER` is not used.

{*alias*}: Specify the alias name of the program specified by the `PROGRAM` parameter in the `listener.ora` file.

[(arguments='arg1,arg2,...')]: Specify the arguments sent to by program that is to be spawned.

Example

```
LSNRCTL> SPAWN nstest_alias (ARGUMENTS='')
```

START

Purpose

Use the command `START` to start the named listener.

Prerequisites

Listener must not already be running.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl START [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> START [listener_name]
```

Arguments

[*listener_name*]: Specify 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/dsteiner/sales/bin/tnslsnr: please wait...

TNSLSNR for Solaris: Version 9.0.1.0.0
System parameter file is /oracle9i/network/admin/listener.ora
Log messages written to /oracle9i/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 Solaris: Version 9.0.1.0.0
Start Date           23-JAN-2002 18:02:25
Uptime               0 days 0 hr. 0 min. 0 sec
Trace Level          off
Security             OFF
SNMP                 OFF
Listener Parameter File /oracle9i/network/admin/listener.ora
Listener Log File    /oracle9i/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

Note: You can also obtain the status of the listener through the Oracle Enterprise Manager Console. See the *Oracle Enterprise Manager Administrator's Guide* for further information.

Purpose

Use the command `STATUS` 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.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl STATUS [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> STATUS [listener_name]
```

Arguments

[*listener_name*]: Specify the listener name, if the default name of `LISTENER` is not used.

Usage Notes

See Also: *Oracle9i Net Services Administrator's Guide* for a complete description of `STATUS` output

The `SET DISPLAYMODE` command changes the format and level of the detail of the 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 Solaris: Version 9.2.0.1.0 -
Development
Start Date           18-JAN-2002 12:02:00
Uptime               0 days 0 hr. 5 min. 29 sec
Trace Level          support
Security             OFF
SNMP                 OFF
Listener Parameter File /oracle9i/network/admin/listener.ora
Listener Log File    /oracle9i/network/log/listener.log
Listener Trace File   /oracle9i/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.acme.com" has 1 instance(s).
  Instance "sales", status READY, has 1 handler(s) for this service...
Service "sales2.us.acme.com" has 1 instance(s).
  Instance "sales", status READY, has 2 handler(s) for this service...
The command completed successfully
```

STOP

Purpose

Use the command `STOP` to stop the named listener.

Prerequisites

The listener must be running.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl STOP [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> STOP [listener_name]
```

Arguments

[*listener_name*]: Specify the listener name, if the default name of `LISTENER` is not used.

Example

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

TRACE

Purpose

Use the command `TRACE` to turn on tracing for the listener.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl trace {level}[listener_name]
```

From the Listener Control utility:

```
LSNRCTL> trace {level} [listener_name]
```

Arguments

`{level}`: Specify 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
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
Opened trace file: /oracle9i/network/trace/listener.trc
The command completed successfully
```

VERSION

Purpose

Use the command `VERSION` to display the current version of Listener Control utility.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl VERSION [listener_name]
```

From the Listener Control utility:

```
LSNRCTL> VERSION [listener_name]
```

Arguments

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

Example

```
LSNRCTL> VERSION listener1
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
TNSLSNR for Solaris: Version 9.2.0.1.0
  TNS for Solaris: Version 9.2.0.1.0
    Oracle Bequeath NT Protocol Adapter for Solaris: Version 9.2.0.1.0
      Unix Domain Socket IPC NT Protocol Adaptor for Solaris: Version
9.2.0.1.0
        TCP/IP NT Protocol Adapter for Solaris: Version 9.2.0.1.0
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 these topics:

- [Oracle Connection Manager Control Utility Overview](#)
- [SET Commands of the Oracle Connection Manager Control Utility](#)
- [Distributed Operations](#)
- [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.

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

```
cmctl command [process_type]
```

where *process_type* is the name of the Oracle Connection Manager process. The choices are as follows:

- `cm` for both the **CMGW process** (gateway process) and the **CMADMIN process** (administrative process)
- `cm` for the CMGW process
- `adm` for the CMADMIN process

For example, the following command starts both the CMGW and CMADMIN processes:

```
CMCTL START cm
```

Note: In general, Oracle Corporation recommends using `cm` for commands. Once an Oracle Connection Manager is configured, the CMADMIN administrative process is not needed. If is necessary to reserve resources, you can then use the `cm`. The CMGW gateway process performs all Oracle Connection Manager basic functions and can run without the CMADMIN process. There is no need for using the `adm` type.

See Also: *Oracle9i Net Services Administrator's Guide* for an overview of the Oracle Connection Manager processes

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

```
cmctl
CMCTL> command [process_type]
```

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:

```
cmctl @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.

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

- Operational commands, such as `START`
- Modifier commands, such as `SET LOG_LEVEL`
- Informational commands, such as `STATUS` and `SHOW ADDRESS`
- Command utility operational commands, such as `EXIT`, `QUIT`, and `HELP`

Note: You must statically set passwords for Oracle Connection Manager in the `cman.ora` file. The Oracle Connection Manager Control utility has no commands for setting or changing passwords.

SET Commands of the Oracle Connection Manager Control Utility

You can use the `SET` command to alter parameter values for an Oracle Connection Manager. Parameter values remain in effect until the Oracle Connection Manager is shut down. You cannot permanently save these temporary changes to the `cman.ora` file.

You can use the `SHOW` command to display the current value of a configuration setting.

Distributed Operations

The Oracle Connection Manager Control utility can perform operations on a local or a remote Oracle Connection Manager.

To set up a computer to remotely administer an Oracle Connection Manager:

1. Configure the local computer that runs the Oracle Connection Manager:

- a. Configure a `cman.ora` file that includes `REMOTE_ADMIN=yes` in the `CMAN_PROFILE` section.

```
CMAN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1630))
CMAN_ADMIN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1830))
CMAN_PROFILE=
  (PARAMETER_LIST=
    (REMOTE_ADMIN=yes))
```

- b. Start the CMGW process. From the operating system, enter either of the following commands:

```
cmctl START cm
cmctl START cman
```

2. Configure the remote computer:

- a. Ensure that Oracle Connection Manager is installed.
- b. Configure a `cman.ora` file with the same values for the `CMAN` and `CMAN_ADMIN` parameters as you did for local Oracle Connection Manager computer. For example:

```
CMAN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1630))
CMAN_ADMIN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1830))
```

All commands except `START` can be issued when an Oracle Connection Manager is administered remotely. The Oracle Connection Manager Control utility can only start the Oracle Connection Manager on the same computer from where the utility is running.

Oracle Connection Manager Control Utility Commands

This section lists and describes the Oracle Connection Manager Control utility commands.

ACCEPT_CONNECTIONS

Purpose

Use the `ACCEPT_CONNECTIONS` command to enable or disable Oracle Connection Manager to accept new connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl ACCEPT_CONNECTIONS [on | off]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> ACCEPT_CONNECTIONS [on | off]
```

Arguments

[on] (default): Specify to enable Oracle Connection Manager to accept new connections.

[off]: Specify to reject new connections to Oracle Connection Manager.

Usage Notes

If set to `off`, existing connections are not affected.

Example

```
CMCTL> ACCEPT_CONNECTIONS off
Profile of the CMAN
-----
Current state           offline
```

CLOSE_RELAY

Purpose

Use the `CLOSE_RELAY` command to enable a connection identified by relay number to be shut down.

Note: Use the `SHOW RELAY` command to obtain the relay numbers for connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl CLOSE_RELAY [relay_num | ALL]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> CLOSE_RELAY [relay_num | ALL]
```

Arguments

[*relay_num*]: Specify the relay number of the connection to terminate immediately.

[ALL]: Specify to immediately terminate all connections

Usage Notes

Use this command with caution. From the time that `SHOW RELAY` command is executed to the time this command is executed, the same relay may have been reused by another connection.

Example

```
CMCTL> CLOSE_RELAY 0  
Relay is not active
```

```
CMCTL> CLOSE_RELAY 0  
The command completed successfully
```

EXIT

Purpose

Use the `EXIT` command to exit from the Oracle Connection Manager Control utility.

Prerequisites

None

Syntax

From the operating system:

```
cmctl EXIT
```

From the Oracle Connection Manager Control utility:

```
CMCTL> EXIT
```

Usage Notes

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

Example

```
CMCTL> EXIT
```

HELP

Purpose

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

Prerequisites

None

Syntax

From the operating system:

```
cmctl HELP [command]
```

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:  
accept_connections  
close_relay  
exit  
quit  
set*  
show*  
shutdown  
start
```

```
stats
status
stop
stopnow
version
```

QUIT

Purpose

Use the `QUIT` command 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
```

SET

Purpose

Use the `SET` command to alter the Oracle Connection Manager's parameter values. Parameter values changes remain in effect until the Oracle Connection Manager is shut down. To make the changes permanent, manually update the parameter settings in the `cmn.ora` file.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SET [parameter]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET [parameter]
```

Arguments

[*parameter*]: Specify a `SET` parameter to modify its configuration setting. Parameters are shown in the example output.

When you enter `SET` without an argument, the Oracle Connection Manager Control utility displays a list of all the parameters.

Example

```
CMCTL> SET
```

The following operations are available

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

```
authentication_level
```

```
displaymode
```

```
log_level
```

```
relay_statistics
```

SET AUTHENTICATION_LEVEL

Purpose

Use the `SET AUTHENTICATION_LEVEL` command to set the level of security for the Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET AUTHENTICATION_LEVEL [0 | 1]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET AUTHENTICATION_LEVEL [0 | 1]
```

Arguments

[0] (default): Specify so that no authentication is required for client connections.

[1]: Specify to reject connections that are not employing Secure Network Service (SNS) to perform client authentication. SNS is part of the Oracle Advanced Security.

Example

```
CMCTL> SET AUTHENTICATION_LEVEL 0
Profile of the CMAN
-----
AUTHENTICATION_LEVEL    = 0
```

SET DISPLAYMODE

Purpose

Use the `SET DISPLAYMODE` command to change the format and level of detail for the `START`, `STATS`, `STATUS`, `STOP`, and `VERSION` commands.

Prerequisites:

None

Syntax

From the operating system:

```
cmctl SET DISPLAYMODE [compat | verbose]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET DISPLAYMODE [compat | verbose]
```

Arguments

[`compat`] (default): Specify to display output that is compatible with older versions of Oracle Connection Manager.

[`verbose`] (recommended): Specify to display a formatted and descriptive output.

Example:

```
CMCTL> SET DISPLAYMODE compat  
Current display mode is COMPATible
```

SET LOG_LEVEL

Purpose

Use the `SET LOG_LEVEL` command to set the log level for the Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET LOG_LEVEL [level]
```

From the Oracle Connection Manager Control utility:

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

Arguments

[*level*]: Specify one of the following log levels:

- 0 (default) for no logging
- 1 for basic reporting
- 2 for `RULE_LIST` matching lookup reporting
- 3 for relay blocking reporting
- 4 for relay I/O counts reporting

Example

```
CMCTL> SET LOG_LEVEL 0
```

Profile of the CMAN

```
-----  
LOG_LEVEL                = 0
```

SET RELAY_STATISTICS

Purpose

Use the `SET RELAY_STATISTICS` command to turn statistic collection pertaining to the I/O of connections of the Oracle Connection Manager on or off.

When a relay closes and this command is set to on, the following statistics are recorded in the appropriate `LOG_RECORD` of the `cman_pid.log` file on UNIX operating systems and `cmanpid.log` file on Windows NT:

- Number of inbound (IN) bytes
- Number of outbound (OUT) bytes
- Number of IN packets
- Number of OUT packets

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET RELAY_STATISTICS [on | off]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET RELAY_STATISTICS [on | off]
```

Arguments

[on]: Specify to turn statistic collection on.

[off] (default): Specify to turn statistic collection off.

Example

```
CMCTL> SET RELAY_STATISTICS off
```

```
Profile of the CMAN
```

```
-----  
RELAY_STATISTICS      = no
```

SHOW

Purpose

Use the `SHOW` command to view the Oracle Connection Manager's parameter values.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHOW [parameter]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW [parameter]
```

Arguments

[*parameter*]: Specify a `SHOW` parameter to view its configuration setting. Parameters are shown in the example output.

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

Example

```
CMCTL> SHOW
```

```
The following operations are available
```

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

```
address
```

```
ALL
```

```
displaymode
```

```
profile
```

```
relay
```

```
rules
```

SHOW ADDRESS

Purpose

Use the `SHOW ADDRESS` command to lists the **protocol address** of the Oracle Connection Manager.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHOW ADDRESS
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW ADDRESS
```

Arguments

None

Usage Notes

The address is set with the **CMAN** parameter in the `cmn.ora` file.

Example

```
CMCTL> SHOW ADDRESS
```

```
Address List
```

```
-----
```

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

SHOW ALL

Purpose

Use the `SHOW ALL` command to display the output for `SHOW ADDRESS`, `SHOW PROFILE`, and `SHOW RULES` commands.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHOW ALL
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW ALL
```

Arguments

None

Example

```
CMCTL> SHOW ALL
Address List
-----
(ASSOCIATION=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1630))

Profile of the CMAN
-----
MAXIMUM_RELAYS           = 2048
RELAY_STATISTICS         = yes
AUTHENTICATION_LEVEL     = 0
LOG_LEVEL                = 4
ANSWER_TIMEOUT           = 0
MAXIMUM_CONNECT_DATA    = 1024
USE_ASYNC_CALL           = yes
TRACING                  = yes
TRACE_DIRECTORY         = default
MAX_FREELIST_BUFFERS    = 0
REMOTE_ADMIN             = no

Rule List
-----
(rule=(src=144.25.185.60)(dst=sales-server)(srv=sales.us.acme.com)(act=accept))
(rule=(src=client-pc)(dst=sales-server)(srv=sales.us.acme.com)(act=accept))
```

SHOW DISPLAYMODE

Purpose

Use the `SHOW DISPLAYMODE` to show the current display mode used for the `START`, `STATS`, `STATUS`, `STOP`, and `VERSION` commands.

Syntax

From the operating system:

```
cmctl SHOW DISPLAYMODE
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW DISPLAYMODE
```

Example

```
CMCTL> SHOW DISPLAYMODE  
Current display mode is VERbose
```

SHOW PROFILE

Purpose

Use the `SHOW PROFILE` command to display the current parameter settings for the Oracle Connection Manager. Information is obtained from the [CMAN_PROFILE](#) parameters in the `cman.ora` file and any changes made with the [SET](#) command.

Syntax

From the operating system:

```
cmctl SHOW PROFILE
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW PROFILE
```

Example

```
CMCTL> SHOW PROFILE
```

```
Profile of the CMAN
```

```
-----  
MAXIMUM_RELAYS           = 2048  
RELAY_STATISTICS         = yes  
AUTHENTICATION_LEVEL     = 0  
LOG_LEVEL                 = 4  
SHOW_INS_INFO            = yes  
ANSWER_TIMEOUT           = 0  
MAXIMUM_CONNECT_DATA     = 1024  
USE_ASYNC_CALL           = yes  
TRACING                   = yes  
TRACE_DIRECTORY          = default  
MAX_FREELIST_BUFFERS     = 0  
REMOTE_ADMIN             = no
```

SHOW RELAY

Purpose

Use the `SHOW RELAY` command to display the current status of a selected relay (connection) or all active relays for the Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW RELAY {relay_num | ACTIVE}
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW RELAY {relay_num | ACTIVE}
```

Arguments

[*relay_num*]: Specify to show the status information for a specific relay.

[ACTIVE]: Specify to show the list of active relays.

Usage Notes

For active relays, only relay numbers are shown. For a given relay number, the following information is displayed:

- Relay number
- Source address (clientside endpoint)
- Destination address (serverside endpoint)
- Number of inbound (IN) bytes
- Number of IN packets
- Number of IN probes
- Number of outbound (OUT) bytes
- Number of OUT packets
- Number of OUT probes

Examples

```
CMCTL> SHOW RELAY active
```

```
Active Relays
```

```
-----  
0000 0001
```

```
CMCTL> SHOW RELAY 0
```

```
Relay Information
```

```
-----  
Relay number          0  
Start-up time         22-JAN-2002 19:47:17  
Src  
(ADDRESS=(PROTOCOL=tcp)(HOST=144.25.185.60)(PORT=35279))  
Dest  
(ADDRESS=(PROTOCOL=tcp)(HOST=144.25.185.60)(PORT=1521))  
Number of IN bytes    438  
Number of IN packets  7  
Number of IN DCD probes 0  
Number of OUT bytes   364  
Number of OUT packets 7  
Number of OUT DCD probes 0
```

SHOW RULES

Purpose

Use the `SHOW RULES` to display the current access rules used by Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SHOW RULES
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW RULES
```

Arguments:

None

Usage Notes:

Rules are set with the [CMAN_RULES](#) parameter in the `cmn.ora` file.

Example

```
CMCTL> SHOW RULES
```

```
Rule List
```

```
-----  
(rule=(src=144.25.185.60)(dst=sales-server)(srv=sales.us.acme.com)(act=accept))  
(rule=(src=client-pc)(dst=sales-server)(srv=sales.us.acme.com)(act=accept))
```

SHUTDOWN

Purpose

Use the `SHUTDOWN` command to shut down the Oracle Connection Manager processes.

Prerequisites

None

Syntax

From the operating system:

```
cmctl SHUTDOWN [normal | abort] [cman]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHUTDOWN [normal | abort] [cman]
```

Arguments

`[normal]` (default): Specify to accept new connections and terminate after all existing connections close.

`[abort]`: Specify to shut down Oracle Connection Manager immediately, closing down all open connections.

`[cman]`: Specify to start both CMGW and CMADMIN processes. `normal` and `abort` use this argument. Therefore, it is not necessary to explicitly specify it.

Example

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

```
CMCTL> SHUTDOWN ABORT  
The command completed successfully
```

START

Purpose

Use the `START` command to start Oracle Connection Manager.

Prerequisites

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

Syntax

From the operating system:

```
cmctl START [cman | cm]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> START [cman | cm]
```

Arguments

[`cman`] (default): Specify to start both CMGW and CMADMIN processes.

[`cm`]: Specify to start the CMGW process.

Usage Notes

The `SET DISPLAYMODE` command changes the format and the level of detail of the output.

Examples

compat mode:

```
CMCTL> START cman
ADMIN Status:
 (STATUS=(VERSION=9.2.0.1.0)(STARTED=22-JAN-2002 19:34:15)(STATE=RUNNING))
CMAN Status:
 (STATUS=(VERSION=9.2.0.1.0)(STARTED=22-JAN-2002 19:34:17)(STATE=running))
```

verbose mode:

```
CMCTL> START cman
Starting /vobs/oracle/bin/cmadmin: please wait...
STATUS of the cman_admin
```

```
-----
ADMIN Version          9.2.0.1.0
Start-up time         22-JAN-2002 19:40:00
Current state         RUNNING
```

```
Starting /vobs/oracle/bin/cmfw: please wait...
STATUS of the cman
```

```
-----
CMAN Version          9.2.0.1.0
Start-up time         22-JAN-2002 19:40:02
Current state         running
```

STATS

Purpose

Use the `STATS` command to display statistical information for Oracle Connection Manager.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl STATS [cman | cm]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> STATS [cman | cm]
```

Arguments

[`cman`] (default): Specify to display statistics both the CMGW and CMADMIN processes.

[`cm`]: Specify to display status information for the CMGW process.

Usage Notes

The following statistics are displayed:

- `TOTAL_RELAYS (compat) | Total number of connections handled (verbose)`

The total number of connections that Oracle Connection Manager has established since it started

- `ACTIVE_DELAYS (compat) | Number of currently active relays (verbose)`

The number of currently active connections

- `MOST_RELAYS (compat)` | Peak active relays (verbose)
The maximum number of concurrent connections that Oracle Connection Manager has ever held since it started
- `OUT_OF_RELAY (compat)` | Total refusals due to `max_relays` exceeded (verbose)
The total number of connect request refusals due to out-of-relays since Oracle Connection Manager started
- `TOTAL_REFUSED (compat)` | Total number of connections refused (verbose)
The total number of connect request refusals since Oracle Connection Manager started

Usage Notes

The `SET DISPLAYMODE` command changes the format and the level of detail of the output.

Examples

compat mode:

```
CMCTL> STATS
```

```
CMAN Status:
```

```
(STATISTICS=(TOTAL_RELAYS=0)(ACTIVE_RELAYS=0)(MOST_RELAYS=0)(OUT_OF_RELAY=0)(TOTAL_REFUSED=0))
```

verbose mode:

```
CMCTL> STATS
```

```
STATISTICS of CMAN
```

```
-----  
Total number of connections handled           0  
Number of currently active relays            0  
Peak active relays                           0  
Total refusals due to max_relays exceeded    0  
Total number of connections refused          0
```

STATUS

Purpose

Use the `STATUS` command to display basic status information, including version, start time, and current statistics.

Prerequisites

None

Syntax

From the operating system:

```
cmctl STATUS [cman | cm]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> STATUS [cman | cm]
```

Arguments

[cman] (default): Specify to display status information for both the CMGW and CMADMIN processes.

[cm]: Specify to display status information for the CMGW process.

Usage Notes

The `SET DISPLAYMODE` command changes the format and the level of detail of the output.

Examples

compat mode:

```
CMCTL> STATUS
```

```
CMAN Status:
```

```
(STATUS=(VERSION=9.2.0.1.0)(STARTED=22-JAN-2002 19:34:17)(STATE=running))
```

```
ADMIN Status:
```

```
(STATUS=(VERSION=9.2.0.1.0)(STARTED=22-JAN-2002 19:34:15)(STATE=RUNNING))
```

verbose mode:

```
CMCTL> STATUS
```

```
STATUS of the cman
```

```
-----
```

CMAN Version	9.2.0.1.0
Start-up time	22-JAN-2002 19:40:02
Current state	running

```
STATUS of the cman_admin
```

```
-----
```

ADMIN Version	9.2.0.1.0
Start-up time	22-JAN-2002 19:40:00
Current state	RUNNING

STOP

Purpose

Use the `STOP` command to shut down Oracle Connection Manager immediately.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl STOP [cman | cm]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> STOP [cman | cm]
```

Arguments

[cman] (default): Specify to stop both CMGW and CMADMIN processes.

[cm]: Specify to stop both CMGW and CMADMIN processes.

Usage Notes

Oracle recommends the [SHUTDOWN](#) command to this command.

If you issue a stop command while connections remain active, you are prompted to confirm the stop.

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

Examples

compat mode:

```
CMCTL> STOP  
The command completed successfully.
```

verbose mode:

```
CMCTL> STOP  
The command completed successfully.
```

STOPNOW

Purpose

Use the `STOPNOW` command to shut down Oracle Connection Manager immediately, closing down all open connections without warning.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl STOPNOW [cman | cm]
```

From the Oracle Connection Manager Control utility:

```
CMCTL> STOPNOW [cman | cm]
```

Arguments

[`cman`] (default): Specify to stop both CMGW and CMADMIN processes.

[`cm`]: Specify to stop both CMGW and CMADMIN processes.

Usage Notes

Oracle recommends the [SHUTDOWN](#) command to this command.

Example

```
CMCTL> STOPNOW
```

VERSION

Purpose

Use the `VERSION` command to display the current version and name of the Oracle Connection Manager Control utility.

Prerequisites

None

Syntax

From the operating system:

```
cmctl VERSION
```

From the Oracle Connection Manager Control utility:

```
CMCTL> VERSION
```

Usage Notes

The `SET DISPLAYMODE` command changes the format of the output.

Examples

compat mode:

```
CMCTL> VERSION  
CMCTL Version          9.2.0.1.0
```

verbose mode:

```
CMCTL> VERSION  
CMCTL Version          9.2.0.1.0  
CMAN Version           9.2.0.1.0
```

Oracle Names Control Utility

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

This chapter contains these topics:

- Oracle Names Control Utility Overview
- SET and SHOW Commands of the Oracle Names Control Utility
- Distributed Operations
- Oracle Names Server Security
- Confirmation Mode in the Oracle Names Control Utility
- Oracle Names Control Utility Commands

Oracle Names Control Utility Overview

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

The basic syntax of Oracle Names Control utility administrative commands is as follows:

```
namesctl command [onames_server]
```

where *onames_server* is the name of the Oracle Names server to be administered.

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

```
namesctl  
NAMESCTL> command [onames_server]
```

When you start the Oracle Names Control utility, it starts a session with an Oracle Names server. The session is started with the first Oracle Names server listed in the discovery file (`.sdns.ora` on UNIX operating systems and `sdns.ora` on Windows operating systems) or the `sqlnet.ora` file (by the `NAMES.PREFERRED_SERVERS` parameter). The `SHOW SERVER` commands display the Oracle Names server that the Oracle Names Control utility is currently managing. If you want to start a session with another Oracle Names server, then use the `SET SERVER` command.

Note: The `START` command starts a session with the Oracle Names server that is started, even if the Oracle Names Control utility had a session with another Oracle Names server.

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:

```
namesctl @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.

The Oracle Names Control utility supports several types of administrative commands:

- Operational commands, such as `START`, `STOP`, and `RESTART`
- Modifier commands, such as `SET TRACE_LEVEL`
- Informational commands, such as `STATUS`, `SHOW LOG_FILE_NAME`, and `PING`
- Command utility operational commands, such as `EXIT`, `QUIT`, and `HELP`

The Oracle Names Control utility also provides data operations command, such as `QUERY` and `REGISTER`.

SET and SHOW Commands of the Oracle Names Control Utility

You can use the `SET` command to change some parameter values for an Oracle Names server or the Oracle Names Control utility environment. Parameter values remain in effect until the Oracle Names server is shut down. If you want these settings to persist, use the `SAVE_CONFIG`, `SET SAVE_CONFIG_ON_STOP`, or `SET SAVE_CONFIG_INTERVAL` commands to save changes to the `names.ora`.

You can use the `SHOW` command to display the current value of a configuration setting.

Distributed Operations

The Oracle Names Control utility can perform operations on a local or a remote Oracle Names server. This is useful when a single administrator is managing all of the Oracle Names servers in a region, or wants to check the availability of a specific Oracle Names server.

To set up a computer to remotely administer an Oracle Names server, ensure that the Oracle Names Control utility (`namesctl`) executable is installed.

All commands except [START](#) can be issued when an Oracle Names server is administered remotely. The Oracle Names Control utility can only start the Oracle Names server on the same computer from where the utility is running.

When issuing commands, specify the name of the Oracle Names server as an argument. For example:

```
NAMESCTL> SHOW SYSTEM_QUERIES dolphin.us.acme.com
```

If the Oracle Names Control utility cannot locate the Oracle Names server, then a name lookup error appears:

```
NNL-00406: name "dolphin.us.acme.com" does not exist
```

If you know the [protocol address](#) of the particular Oracle Names server, then enter it in place of the name of the Oracle Names server. For example:

```
NAMESCTL> SHOW SYSTEM_QUERIES  
(ADDRESS=(PROTOCOL=tcp)(HOST=dolphin.us.acme.com)(PORT=1575))
```

When the name omitted, the Oracle Names server that the Oracle Names Control utility has a session with is assumed. If a session cannot be established with any Oracle Names server, then the following error appears:

```
NNL-00005: no server has been set. Use the "SET SERVER" command first
```

Use the [SET SERVER](#) command to establish a session with an Oracle Name server.

Once remote access is established, all commands except [START](#) can be issued.

See Also: ["Oracle Names Control Utility Overview"](#) on page 3-2 for further information about session establishment

Oracle Names Server Security

If the `NAMES.PASSWORD` parameter is set in the `names.ora`, then the Oracle Names Control utility requires a `SET PASSWORD` command for any sensitive operations, such as stopping an Oracle Names server.

If the `NAMESCTL.SERVER_PASSWORD` parameter is set in the `sqlnet.ora` file on the node running the Oracle Names Control utility, you are prompted to use the `SET PASSWORD` command each time a secure operation is performed.

If you are concerned with the security implications of explicitly putting an Oracle Names server password in the client `sqlnet.ora` file, you can omit the `NAMESCTL.SERVER_PASSWORD` parameter and always use the command:

```
NAMESCTL> SET PASSWORD
```

When passed over the network, the password is encrypted, regardless of how it was set in the `names.ora` file. However, if the `NAMESCTL.INTERNAL_ENCRYPT_PASSWORD` parameter is set to `false` in the `sqlnet.ora` file, then the password is not encrypted.

Confirmation Mode in the Oracle Names Control Utility

Some of the Oracle Names Control utility commands require confirmation before they are executed. When you issue the command, you are prompted:

```
confirm:[yes or no]
```

Enter `yes` to execute the command; enter `no` to cancel the command.

You can turn confirmation mode off by setting the parameter `NAMESCTL.NOCONFIRM` to `true` in the `sqlnet.ora` file.

Oracle Names Control Utility Commands

This section lists and describes the Oracle Names Control utility commands.

DELEGATE_DOMAIN

Purpose

Use the command `DELEGATE_DOMAIN` to define a domain as the start of a subregion for the current administrative region.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl DELEGATE_DOMAIN {domain}{onames_server}{(ADDRESS=...)}
```

From the Oracle Names Control utility:

```
NAMESCTL> DELEGATE_DOMAIN {domain}{onames_server}{(ADDRESS=...)}
```

Arguments

{*domain*}: Specify the domain name.

{*onames_server*}: Specify the Oracle Names server name.

{(ADDRESS= . . .)}: Specify the Oracle Names server protocol address.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about defining protocol addresses

Usage Notes

This command provides a dynamic way to subdivide the namespace.

Unless a domain is delegated from a region, the Oracle Names servers in that region assume authority over all subdomains. In order to delegate a domain, you must first create a new region.

Once a domain is delegated, the Oracle Names servers in the current administrative region forward subsequent operations to the subregion where the domain is administered by Oracle Names servers.

Examples

```
NAMECTL> DELEGATE_DOMAIN webwidgets.acme.com ns1.webwidgets.acme.com  
(ADDRESS=(PROTOCOL=tcp)(HOST=fred.webwidgets.acme.com)(PORT=1575))
```

DOMAIN_HINT

Purpose

Use the command `DOMAIN_HINT` to specify domain hints for requests for data from remote regions. A domain hint contains the name of a remote domain and at least one address of an Oracle Names server in that domain. A hint enables the Oracle Names server to forward the request to a specific address, reducing network traffic.

Without a domain hint, an Oracle Names server forwards a request to whatever remote Oracle Names servers it knows, which then forwards the request to the root Oracle Names server in its region. The root Oracle Names server forwards the request to the Oracle Names server which has information on the domain that the request refers to.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl DOMAIN_HINT {domain}{onames_server}{(ADDRESS=...)}
```

From the Oracle Names Control utility:

```
NAMESCTL> DOMAIN_HINT {domain}{onames_server}{(ADDRESS=...)}
```

Arguments

{*domain*}: Specify the domain name.

{*onames_server*}: Specify the Oracle Names server name.

{(ADDRESS=...) }: Specify the Oracle Names server protocol address.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about defining protocol addresses

Usage Notes

Any region that is not the root region will need at least the root region defined using this command in order to find objects in any other region. You can provide additional hints as optimizations to provide local Oracle Names servers with direct access to certain other regions.

Examples

```
NAMESCTL> DOMAIN_HINT ACME.COM ns0.acme.com (ADDRESS=(PROTOCOL=tcp)  
(HOST=top.acme.com)(PORT=1575))
```

DUMP_ALIAS

Purpose

Use the command `DUMP_ALIAS` to query all the **alias** data in a domain or domain subtree and export the data to a LDAP-compliant directory service or into an **LDAP Data Interchange Format (LDIF)** file, which can later be loaded into a directory.

Notes:

- Net service aliases are not supported using Microsoft Active Directory.
 - This command does not verify that the object the alias is referencing exists in the directory. If an alias is exported and the object it is referencing is not exported, then the `DUMP_ALIAS` command exports the alias without verifying that the referenced database service or net service name exists.
 - The command should use the same destination as applied with the `DUMP_LDAP` command to the object the alias is referencing. If an alias and the object it is referencing are exported with different destinations, then the net service alias will not contain the correct name for the object it is referencing. This can occur in a case of tree rearrangement. See *Oracle9i Net Services Administrator's Guide* for further information.
-
-

Prerequisites

The directory must already have the **Oracle schema** and one or more **Oracle Contexts**.

Password Required If One Has Been Set

A password for the directory may be required.

Syntax

Exporting Data to an LDIF File

```
NAMESCTL> DUMP_ALIAS [source] [destination] [options] {-f [filename]}
```

Exporting Data To a Directory

```
NAMESCTL> DUMP_ALIAS [source] [destination] [options] {-h host} {-p port} {-D user_dn} {-w password}
```

Arguments

[*source*]: Specify the source Oracle Names domain and, optionally, the `-R` argument:

{*domain*}: Specify the domain name for aliases to be exported. The default domain is the root. The operation is forwarded to an Oracle Names server that is authoritative for that domain if the Oracle Names server which the client contacts is not authoritative.

[`-R`]: Specify to recursively descend the Oracle Names tree structure. Without `-R`, this command looks only at objects in the specified domain.

[*destination*]: Specify the **distinguished name (DN)** in the **directory information tree (DIT)** where to export net service aliases. A DN can be specified in one of following ways:

- A DN where all exported aliases will be located

For example, `(dn:dc=acme,dc=com)` specifies to export all aliases to `cn=OracleContext` under `dc=acme,dc=com`. By default, `cn=OracleContext` is automatically inserted.

- A template to model the DIT structure where objects will be located

You create a template by specifying attributes for mapping subdomains to the corresponding attribute/value pairs in a DN. A destination of `(dn:o,ou,dc)`, for example, maps to a DIT structure that contains an organization (`o`), organization unit (`ou`), and a domain component (`dc`).

When you use `-R` in the source argument, the domain in the source maps to the segment of the destination argument that contains the fully-specified attribute/value pairs, and subdomains of the source domain map to nodes specified in the destination argument by attributes without values. For example, if Oracle Names contains a root domain of `acme.com` and delegated domains of `sales.acme.com` and `dev.acme.com`, you can use a template of `(dn:ou,o=acme,c=us)` to create a DIT with a common structure of

`o=acme,c=us` and unique subtrees of `ou=sales` and `ou=dev`. Using the `-R` argument exports data in one of two ways:

- If the destination is specified with a DN, then the `DUMP_ALIAS` command recursively descends the Oracle Names tree structure and exports all aliases in that structure to the destination node in the DIT. All the aliases in all the subdomains map to a single DIT node.

Continuing with an example of an Oracle Names root domain of `acme.com` and delegated domains of `sales.acme.com` and `dev.acme.com`, issuing `DUMP_ALIAS acme.com (dn:dc=acme,dc=com) -R` exports all net services in `acme.com`, `sales.acme.com`, and `dev.acme.com` to `dc=OracleContext,dc=acme,dc=com`.

- If the destination is specified with a template, then the `DUMP_ALIAS` command recursively descends the Oracle Names tree structure and exports the aliases in the source domain and subdomains according to the template.

Issuing `DUMP_ALIAS acme.com (dn:ou,dc=acme,dc=com) -R` exports aliases in `acme.com` to `dc=OracleContext,dc=acme,dc=com`, aliases in `sales.acme.com` to `dc=OracleContext,ou=sales,dc=acme,dc=com`, and aliases in `dev.acme.com` to `dc=OracleContext,ou=dev,dc=acme,dc=com`.

[*options*]: Arguments that specify how the export of the aliases is to occur:

- c: Specify that the export should continue on error.
- n: Specify to not perform an actual export. This argument enables you to perform a test run. The results display to the screen.
- m: Specify that existing entries in the DIT are to be modified.
- x: Specify to not include `cn=OracleContext` in each object's DN.

[`-f filename`]: Specify that the exported data dumped into an LDIF file, which can later be loaded into a directory. The default file name is `onames.ldif`. To use `onames.ldif` as the default name, specify `-f` as the last argument.

These arguments specify the location of the directory server:

`[-h host]`: Specify the host name of the directory server.

`[-p port]`: Specify the port number the directory is configured to listen on. The default TCP/IP port number is 389.

These arguments specify the authentication credentials of the directory server:

`[-D user_dn]`: Specify a directory administrator that has been given add and modify privileges. For example, `cn=Mary` is the DN for an administrator named Mary.

`[-w password]`: Specify the password for the directory administrator.

Usage Notes

See Also: *Oracle9i Net Services Administrator's Guide*

Examples

Exporting Net Service Aliases to a LDIF File

```
NAMECTL> DUMP_ALIAS acme.com (dn:dc=sales,dc=com) -R -f test.ldif
```

Exporting Net Service Aliases Directly Into a Directory

```
NAMECTL> DUMP_ALIAS acme.com (dn:dc=sales,dc=com) -R -h ldap-server -p  
389 -D cn=orcladmin -w welcome
```

DUMP_LDAP

Purpose

Use the command `DUMP_LDAP` to query all the addresses of database objects in a domain or region and export the data to a LDAP-compliant directory service or into an LDIF file, which can later be loaded into a directory.

Note: Because database objects are exported as net service names, the username and password credentials for a **global database link** or **link qualifier** is not exported with the address information. In the same way you used global database links in Oracle Names, you can use a net service name from a directory to access an object in a database. However, the net service name will behave as a connected user database link.

See Also:

- *Oracle9i Net Services Administrator's Guide* for further information about using global database links
- *Oracle9i Database Administrator's Guide* for information about the supported types of database link users
- *Oracle Advanced Security Administrator's Guide* for information about configuring database links in a directory

Prerequisites

The directory must already have an Oracle schema and one or more Oracle Contexts.

Password Required If One Has Been Set

A password for the directory may be required.

Syntax

Exporting Data to an LDIF File

```
NAMESCTL> DUMP_LDAP [source] [destination] [options] {-f [filename]}
```

Exporting Data To a Directory

```
NAMESCTL> DUMP_LDAP [source] [destination] [options] {-h host} {-p port} {-D user_dn} {-w password}
```

Arguments

[*source*]: Specify the source Oracle Names domain and, optionally, the `-R` argument:

{*domain*}: Specify the domain name for objects to be exported. The default domain is the root. The operation is forwarded to an Oracle Names server that is authoritative for that domain if the Oracle Names server which the client contacts is not authoritative.

[`-R`]: Specify to recursively descend the Oracle Names tree structure. Without `-R`, this command looks only at objects in the specified domain.

[*destination*]: Specify the DN in the DIT where to export objects. A DN can be specified in one of following ways:

- A DN where all exported aliases will be located

For example, `(dn:dc=acme,dc=com)` specifies to export objects to `cn=OracleContext` under `dc=acme,dc=com`. By default, `cn=OracleContext` is automatically inserted.

- A template to model the DIT structure where objects will be located

You create a template by specifying attributes for mapping subdomains to the corresponding attribute/value pairs in a DN. A destination of `(dn:o,ou,dc)`, for example, maps to a DIT structure that contains an organization (`o`), organization unit (`ou`), and a domain component (`dc`).

When you use `-R` in the source argument, the domain in the source maps to the segment of the destination argument that contains the fully-specified attribute/value pairs, and subdomains of the source domain map to nodes specified in the destination argument by attributes without values. For example, if Oracle Names contains a root domain of `acme.com` and delegated domains of `sales.acme.com` and `dev.acme.com`, you can use a template of `(dn:ou,o=acme,c=us)` to create a DIT with a common structure of `o=acme,c=us` and unique subtrees of `ou=sales` and `ou=dev`. Using the `-R` argument exports data in one of two ways:

- If the destination is specified with a DN, then the `DUMP_ALIAS` command recursively descends the Oracle Names tree structure and exports all net service names and database services in that structure to the destination node in the DIT. All the objects in all the subdomains map to a single DIT node.

Continuing with the example of an Oracle Names root domain of `acme.com` and delegated domains of `sales.acme.com` and `dev.acme.com`, issuing

```
DUMP_LDAP acme.com (dn:dc=acme,dc=com) -R
exports all net service names and database services in acme.com,
sales.acme.com, and dev.acme.com to
dc=OracleContext,dc=acme,dc=com.
```

- If the destination is specified with a template, then the `DUMP_ALIAS` command recursively descends the Oracle Names tree structure and exports the net service names and database services in the source domain and subdomains according to the template.

```
Issuing DUMP_LDAP acme.com (dn:ou,dc=acme,dc=com) -R
exports objects in acme.com to dc=OracleContext,dc=acme,dc=com,
objects in sales.acme.com to
dc=OracleContext,ou=sales,dc=acme,dc=com, and objects in
dev.acme.com to dc=OracleContext,ou=dev,dc=acme,dc=com.
```

[*options*]: Arguments that specify how the export of data is to occur:

`-c`: Specify that the export should continue on error.

`-n`: Specify to not perform an actual export. This argument enables you to perform a test run. The results display to the screen.

`-m`: Specify that existing entries in the DIT are to be modified.

`-x`: Specify to not include `cn=OracleContext` in each object's DN.

[`-f filename`]: Specify that the exported data be dumped into an LDIF file, which can later be loaded into a directory. The default file name is `onames.ldif`. To use `onames.ldif` as the default name, specify `-f` as the last argument.

These arguments specify the location of the directory server:

[`-h host`]: Specify the host name of the directory server.

[`-p port`]: Specify the port number the directory is configured to listen on. The default TCP/IP port number is 389.

These arguments specify the authentication credentials of the directory server:

`[-D user_dn]`: Specify a directory administrator that has been given add and modify privileges. For example, `cn=mary` is the DN for an administrator named Mary.

`[-w password]`: Specify the password for the directory administrator.

Usage Notes

See Also: *Oracle9i Net Services Administrator's Guide*

Examples

Exporting Data to an LDIF File

```
NAMESTCL> DUMP_LDAP sj.us.sczl.com (dn:ou=sj,dc=us,dc=sczi,dc=com) -R -f
test.ldif
```

Exporting Data Directly Into a Directory

```
NAMESTCL> DUMP_LDAP sj.us.sczl.com (dn:ou=sj,dc=us,dc=sczi,dc=com) -R -h
ldap-server -p 389 -D cn=orcladmin -w welcome
```

DUMP_TNSNAMES

Purpose

Use the command `DUMP_TNSNAMES` to write the address information defined in the local region to a `tnsnames.ora` file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl DUMP_TNSNAMES
```

From the Oracle Names Control utility:

```
NAMESCTL> DUMP_TNSNAMES
```

Arguments

None

Usage Notes

`DUMP_TNSNAMES` writes the addresses defined in the local region into `tnsnames.ora`, that is, everything with an address-type record, A.SMD.

The command creates or partially overwrites the `tnsnames.ora` file. Any entries in the existing `tnsnames.ora` that are not defined in the Oracle Names servers will remain. Any definitions in `tnsnames.ora` that are also defined in the Oracle Names servers will be overwritten. Entries that are defined in the Oracle Names servers but not in the `tnsnames.ora` are added.

Example

```
NAMESCTL> DUMP_TNSNAMES
```

EXIT

Purpose

Use the command `EXIT` to exit from the Oracle Names Control utility.

Prerequisites

The Oracle Names Control utility must be loaded.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Oracle Names Control utility:

```
NAMESTL> EXIT
```

Arguments

None

Usage Notes

`EXIT` has no affect on any Oracle Names servers; it affects only the Oracle Names Control utility.

This command is identical to the `QUIT` command.

Example

```
namctl> EXIT  
NL-00851: NAMESTL finished
```

FLUSH

Purpose

Use the command `FLUSH` to instruct the Oracle Names server to clear all remote region information from its local cache checkpoint file, which has a default of `ckpcch.ora`.

Prerequisites

This command is relevant with an environment with multiple regions where there is authoritative data.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl FLUSH [onames_server] [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> FLUSH [onames_server] [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then only the current Oracle Names server cache is flushed of the foreign names.

Usage Notes

`FLUSH` erases *all* remote data that has been cached. Typically, you should flush the foreign data cache for the following reasons:

- A large volume of data changes in the network and the normal time-to-live (TTL) aging mechanism will take too long.
- Unidentifiable errors in name resolution of cached foreign data are occurring. Flushing all foreign data from the cache forces it to be looked up again when it is requested the next time.

Names are flushed from the current Oracle Names server. The current Oracle Names server is either the default preferred Oracle Names server or the one set by using the `SET SERVER` command.

Examples

```
NAMECTL> FLUSH
Confirm [yes or no]: yes
```

FLUSH_NAME

Purpose

Use the command `FLUSH_NAME` to instruct the Oracle Names server to clear information for a specific region from its local cache checkpoint file, which has a default of `ckpccch.ora`.

Prerequisites

This parameter is useful for an environment with multiple regions. (In central administration, there is no authoritative data.)

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl flush_name {domain}
```

From the Oracle Names Control utility:

```
NAMECTL> flush_name {domain}
```

Arguments

[*domain*]: Specify the domain name.

Usage Notes

`FLUSH_NAME` erases only data cached from outside the region of the Oracle Names server, that is, non-authoritative data. Data is typically flushed when a name is behaving unusually, suggesting the source copy may have changed.

Names are flushed from the current Oracle Names server. The current Oracle Names server is either the default preferred Oracle Names server or the one set by using the `SET SERVER` command.

Example

```
NAMESCTL> FLUSH_NAME mountain.acme.com
```

HELP

Purpose

Use the command `HELP` to provide a list of all the Oracle Names Control utility commands or provide syntax help for a particular Oracle Names Control utility command.

Prerequisites

None

Password Required If One Has Been Set:

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl HELP [command]
```

From the Oracle Names Control utility:

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

Example

```
NAMECTL> HELP
The following operations are available:
An asterisk (*) denotes a modifier or extended command:
exit
flush
flush_name
log_stats
ping
query
quitreload
repeat*
reset_stats
restart
save_config
set*
show
shutdown
start
startup
status
stop
version
```

LIST_DELEGATED

Purpose

Use the command `LIST_DELEGATED` to list all the delegated domains for the root region or a specified domain.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl LIST_DELEGATED [domain]
```

From the Oracle Names Control utility

```
NAMESCTL> LIST_DELEGATED [domain]
```

Arguments

[*domain*]: Specify the domain name.

Usage Notes

Before exporting data from an Oracle Names server to a directory server, use this command to find out the current domain structure.

See Also: *Oracle9i Net Services Administrator's Guide*

Example

```
NAMESCTL> LIST_DELEGATED
europe.acme.com
asia.acme.com
africa.acme.com
```

LIST_DOMAINS

Purpose

Use the command `LIST_DOMAINS` to list all the domains in the root region or subdomains for a specified domain.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system

```
namesctl LIST_DOMAIN [domain]
```

From the Oracle Names Control utility:

```
NAMESTL> LIST_DOMAIN [domain]
```

Arguments

[*domain*]: Specify the domain name.

Usage Notes

Before exporting data from an Oracle Names server to a directory server, use this command to determine the current domain structure.

See Also: *Oracle9i Net Services Administrator's Guide*

Example

```
NAMESTL> LIST_DOMAINS
com
sczi.com
us.sczi.com
sj.us.sczi.com
```

LIST_OBJECTS

Purpose

Use the command `LIST_OBJECTS` to list all the network objects for the root region or a specified domain.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl LIST_OBJECT [-R] [domain]
```

From the Oracle Names Control utility:

```
NAMESCTL> LIST_OBJECT [-R] [domain]
```

Arguments

`[-R]`: Specify that all authoritative subdomains of the given domain be listed.

`[domain]`: Specify the domain name.

Usage Notes

Before exporting data from an Oracle Names server to a directory server, use this command to determine the objects stored in a domain.

See Also: *Oracle9i Net Services Administrator's Guide*

Example

```
NAMESCTL> LIST_OBJECTS
partstdb.widgets.acme.com
toolsdb.widgets.acme.com
partstdb.components.widgets.acme.com
sparepartstdb.gadgets.widgets.acme.com
```

LOAD_TNSNAMES

Purpose

Use the command `LOAD_TNSNAMES` to load all connect descriptors defined in one or more `tnsnames.ora` files into an Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl LOAD_TNSNAMES {directory_path/tnsnames.ora} [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> LOAD_TNSNAMES {directory_path/tnsnames.ora}[...]
```

Arguments

`{directory_path/tnsnames.ora}`: Specify one or more `tnsnames.ora` files.

Usage Notes

This command would typically be used once when a site begins using Oracle Names after having used `tnsnames.ora` files. Run this command once for each region. The names defined during this operation will be defined permanently and will be propagated from one Oracle Names server to all the rest in the region.

Example

```
NAMECTL> LOAD_TNSNAMES /oracle9i/network/admin/tnsnames.ora
Name: koala.lab.npd.us.acme.com
Response status: normal, successful completion
Name: devdd.rdbms.us.acme.com
Response status: normal, successful completion
Name: envyd.lab.npd.us.acme.com
Response status: normal, successful completion
Name: stealth.npd.us.acme.com
Response status: normal, successful completion
Name: null.us.acme.com
Response status: normal, successful completion
Name: slime.lab.npd.us.acme.com
Response status: normal, successful completion
Name: felix.hp.us.acme.com
Response status: normal, successful completion
Name: dtnet1.dec.acme.com
Response status: normal, successful completion
Name: devds.rdbms.us.acme.com
Response status: normal, successful completion
```

LOG_STATS

Purpose

Use the command `LOG_STATS` to log the current set of statistics to the configured log file for that Oracle Names server. The log file has a default of `names.log`.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl LOG_STATS [onames_server] [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> LOG_STATS [onames_server] [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. When no names are supplied, then only the statistics for the current Oracle Names server are reset.

Usage Notes

Statistics may be logged if the `STATUS` command or other behavior indicates some data that you would like to capture in the log. This command does not affect the current log statistics interval.

Example

```
NAMESCTL> LOG_STATS  
Statistics counters logged.
```

PASSWORD

Purpose

Use the command `PASSWORD` to set an encrypted password for privileged Oracle Names Control utility commands, such as `STOP`, `RESTART`, and `RELOAD`.

Prerequisites

The Oracle Names Control utility must be loaded.

Password Required If One Has Been Set

Not applicable

Syntax

From the Oracle Names Control utility:

```
NAMESCTL> PASSWORD [password]
```

Arguments

Text string matching the value encrypted in the `NAMES.PASSWORD` parameter in the `names.ora` file.

Usage Notes

This command does not change a password already established with the `NAMES.PASSWORD` parameter in the `names.ora` file. It simply sets an Oracle Names Control utility variable. Then, the value stored is sent from the Oracle Names Control utility with any command request to the Oracle Names server, and the value is compared to that on the Oracle Names server. If they match, then operations requiring passwords are allowed.

Only privileged operations are affected, that is, operations that alter the functioning of the Oracle Names server. Operations such as `SHOW` or `STATUS` are not considered privileged, and do not require a password.

The password can either be passed as an argument of the `PASSWORD` command, or, if no argument is given, you are prompted for the password. Note that the input is not displayed on the screen as it is entered.

When passed over the network, the password is always encrypted, regardless of how it is set.

Examples

```
NAMESCTL> PASSWORD open_sesame
```

```
NAMESCTL> PASSWORD
```

```
Enter name server password: names9i
```

PING

Purpose

Use the command `PING` to contact an Oracle Names server and display the request/response time.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl PING [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> PING [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then only the current Oracle Names server is pinged.

Usage Notes

Ping ensures that an Oracle Names server is functioning and shows typical response times from the location of the Oracle Names Control utility user to an Oracle Names server.

Example

```
NAMECTL> PING nserver.com
Round trip time is 0.04 seconds
```

QUERY

Purpose

Use the command `QUERY` to retrieve the contents of a network object stored in the Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl QUERY object_name [record_type] [modifiers]
```

From the Oracle Names Control utility:

```
NAMECTL> QUERY object_name [record_type] [modifiers]
```

Arguments

[*record_type*]: Specify one of the following record types:

- `A.SMD` for a network address
- `CNAME.SMD` for an alias
- `DL.RDBMS.OMD` for a global database link
- `DLCR.RDBMS.OMD` for a global database link qualifier
- `NS.SMD` for an Oracle Names server
- `V1ADD.NPO.OMD` for a SQL*Net Version 1 connect descriptor

[*modifier*]: Specify one of the following modifier types:

- **authority**: Forces the query to be resolved at the source of the data (in the administrative region where the data is considered local) even if the data is in the local cache. Use this modifier if you suspect that the data has changed at the source.
- **noforward**: Queries for the data, but does not forward the request. When the data is not local and no forward is specified, the query is not be resolved.
- **trace**: Enables a trace of the path to the answer. This is useful whenever you want to find out which Oracle Names servers the request went to.

Usage Notes

QUERY can be used to verify that a defined piece of data can be found, and that the contents are correct.

If this command is used with just a name as a parameter, then Oracle Names server responds with the number of pieces of data with that name, and the time required to complete the operation.

If this command is used with the name and type supplied as arguments, the specific name is looked up and returned to the user.

The QUERY command can take multiple arguments. For example:

```
QUERY sales.com a.smd authority trace
```

This command operates on the current Oracle Names server, either the default, or as specified using the [SET SERVER](#) command.

Example

```
NAMECTL> QUERY bones.dem.medicine a.smd
Total response time:0.04 seconds
Response status:normal, successful completion
Authoritative answer:yes
Number of answers:1
Canonical name:bones.dem.medicine
TTL: 1 day
Alias translations:
    from:bones.dem.medicine
    to: bones.dem.medicine
Answers:
    data type is "a.smd"
    Syntax is ADDR:...(DESCRIPTION=(ADDRESS=
(PROTOCOL=TCP)(Host=bones-pc)
(Port=1521))(CONNECT_DATA=(SERVICE_NAME=bones.dem.medicine)))
```

QUIT

Purpose

Use the command `QUIT` to quit the Oracle Names Control utility.

Prerequisites

The Oracle Names Control utility must be loaded.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Oracle Names Control utility:

```
NAMECTL> QUIT
```

Arguments

None

Usage Notes

QUIT has no effect on any Oracle Names servers; it affects only the Oracle Names Control utility.

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

Example

```
NAMESCTL> QUIT
NL-00851: NAMESCTL finished
```

REGISTER

Purpose

Use the command REGISTER to register a network object to an Oracle Names server.

Password Required If One Has Been Set

No. If a password is set, the [SET PASSWORD](#) command does not need to be issued prior to this command.

Syntax

From the operating system

```
namesctl REGISTER {object_name} [-t service_type]
[-d[(DESCRIPTION=)(ADDRESS=...)((CONNECT_DATA=(SERVICE_NAME|SID=service_
name|SID)))]]) [-h host] [-l listener_name]
```

From the Oracle Names Control utility:

```
NAMESCTL> REGISTER {object_name} [-t service_type]
[-d[(DESCRIPTION=)(ADDRESS=...)((CONNECT_DATA=(SERVICE_NAME|SID=service_
name|SID)))]]) [-h host] [-l listener_name]
```

Arguments

{object_name}: Specify the object name.

[-t service_type]: Specify the service type of the object:

- ORACLE_DATABASE for an Oracle database
- ORACLE_LISTENER for a listener
- ORACLE_NAMESERVER for an Oracle Names server

Note: To register an Oracle Names server, use the [REGISTER_NS](#) command rather than the REGISTER command.

[-d]: Specify the protocol address of the listener or Oracle Names server object, or complete connect descriptor of the database object, net service name, alias, or global database link.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about protocol addresses and parameters

[-h host]: Specify the host name that the object resides on.

[-l listener_name]: Specify the name of the listener object.

The service type, address description, host, and listener name options are not necessary to make the registration process appear to work. However, they are necessary to make the registration useful. In other words, an object name registered without an address cannot be used.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about protocol addresses and parameters

Note: The protocol address cannot contain any spaces.

Usage Notes

This command provides a mechanism for registering a service, its type, its hostname, and its address. Both the type of service and the data can be any valid string, but the typical registration has either `ORACLE_DATABASE` or `ORACLE_LISTENER` as type of service, and the address as the data.

The object registration is propagated to all other Oracle Names servers in the region.

If the `sqlnet.ora` file is configured with the parameter `NAMES.DEFAULT_DOMAIN` to set the same domain name as the database domain, then the database name does not need to be qualified with the domain.

Example

The following example shows the registration of database service `sales.us.acme.com`.

```
NAMESCTL> REGISTER sales -t oracle_database -d (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1575)) (CONNECT_DATA=(SERVICE_NAME=sales.us.acme.com)))
```

REGISTER_NS

Purpose

Use the REGISTER_NS command to define an Oracle Names server and its authoritative domain.

Prerequisites

None

Password required if one has been set

No. If a password is set, the SET PASSWORD command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl REGISTER_NS {onames_server}{(ADDRESS=...)}{domain}
```

From Oracle Names Control utility:

```
NAMECTL> REGISTER_NS {onames_server}{(ADDRESS=...)}{domain}
```

Arguments

{onames_server}: Specify the Oracle Names server name.

{(ADDRESS=...)}: Specify the Oracle Names server protocol address.

{domain}: Specify the domain name.

Usage Notes

This command provides a mechanism for registering an Oracle Names server as an authoritative server for a given domain. The command adds a network session record type, NS . SMD, for the Oracle Names server to the domain, and provides the Oracle Names server with an address record, A . SMD.

This command will fail if either the domain exists and has non-NS records or the Oracle Names server exists and has a type of service record that is other than 'ORACLE_NAMESERVER'.

Ordinarily, the Oracle Names servers maintain their own data by registering themselves when they start. This command is provided as a manual way to manage domain and Oracle Names server data if for some reason the Oracle Names server cannot. This may occur if the region database tables are set up as read-only for security reasons.

If the Oracle Names servers are not registering themselves, then use this command to define the region topology data. Each Oracle Names server in the region should be defined using this command for each top-level domain in the region. Usually, the top level consists of a single parent domain, for example, `acme.com`. However, a region may also have multiple sibling parent domains, for example, a region covering North America would have `US`, `CA`, and `MX` as its top-level parent domains.

Note the regions which were defined using the Oracle Network Manager in SQL*Net version 2 have `NS.SMD` records defined for every domain in the administrative region, but in Oracle Net only the top-level parent domains need to have `ns.smd` records defined for each server in the region.

Use the Oracle Names Control utility `DELEGATE DOMAIN` command to define Oracle Names servers which are delegation points for subregions.

Use the `NAMES.DOMAIN_HINTS` parameter in the `names.ora` file to provide data about any other Oracle Names servers in foreign regions.

Example

```
NAMECTL> REGISTER_NS namesrv1
(ADDRESS=(PROTOCOL=tcp)(HOST=namesvr1)(PORT=1575))
Total response time: 7 minutes 59.14 seconds
Response status: normal, successful completion
```

RELOAD

Purpose

Use the command `RELOAD` to force the Oracle Names server to check immediately for data changes in its administrative region. If there are any changes, then the Oracle Names server reloads all database service names, net service names, global database links, and aliases.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, then the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl RELOAD [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> RELOAD [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then only the current Oracle Names server is reloaded.

Usage Notes

All Oracle Names servers load their data directly from the database specified by the `NAMES.ADMIN_REGION` parameter in the `names.ora` file.

In an environment with multiple regions, `RELOAD` affects only the data for the current administrative region. All foreign data in the cache is unchanged.

Example

```
NAMESCTL> RELOAD  
Server reloaded.
```

REORDER_NS

Purpose

Use the command `REORDER_NS` to create the file that lists local Oracle Names servers and their protocol addresses.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl REORDER_NS [(ADDRESS=...)]
```

From the Oracle Names Control utility:

```
NAMESCTL> REORDER_NS [(ADDRESS=...)]
```

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about protocol addresses and parameters

Arguments

[(ADDRESS=...)]: Specify an optional Oracle Names server address to be used as the initial Oracle Names server to contact.

Usage Notes

This command generates the file which defines Oracle Names server names and addresses to enable clients to contact Oracle Names servers for name lookup.

The `REORDER_NS` command performs the following tasks:

1. Searches for the first Oracle Names server in the following order:
 - a. A **preferred Oracle Names server** configured in the `sqlnet.ora` file with the `NAMES.PREFERRED_SERVERS` parameter
 - b. A **well-known Oracle Names server**
 - c. A local Oracle Names server configured with TCP/IP on port 1575
2. Sends a query for all the Oracle Names servers in the local region
3. Sends a ping to each of these Oracle Names servers
4. Sorts the list of Oracle Names servers by increasing order of response time
5. Writes an Oracle Names server list with the sorted list of names and addresses

Example

```
NAMESCTL> REORDER_NS (ADDRESS=(PROTOCOL=tcp)(HOST=namesrv-server)(PORT=1575))
```

REPEAT

Purpose

Use the command `REPEAT` to perform [QUERY](#), [REGISTER](#), [TIMED_QUERY](#), or [UNREGISTER](#) multiple times to compute average return rates.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl REPEAT {number} QUERY|REGISTER|TIMED_QUERY|UNREGISTER [record_type]
```

From the Oracle Names Control utility:

```
NAMESCTL> REPEAT {number} QUERY [record_type]
```

Arguments

{number}: Specify an integer

[record_type]: Specify one of the following record types:

- `A.SMD` for a network address
- `CNAME.SMD` for an alias
- `DL.RDBMS.OMD` for a global database link
- `DLCR.RDBMS.OMD` for a global database link qualifier
- `NS.SMD` for an Oracle Names server
- `V1ADD.NPO.OMD` for a SQL*Net Version 1 connect descriptor

Usage Notes

This command is useful for understanding the average response time over a number of requests.

Do not specify too large a number here; while the number of iterations are occurring, the Oracle Names Control utility cannot perform any other operation.

Example

```
NAMECTL> REPEAT 10 QUERY manatee a.smd
Number of requests: 10
Average response time: 0.01 seconds
Minimum response time: 0.01 seconds
Maximum response time:0.04 seconds
Total response time:0.14 seconds
Response status:normal, successful completion
Authoritative answer:yes
Number of answers: 1
TTL: 1 day
Answers:
    data type is "a.smd"
    Syntax is ADDR:(DESCRIPTION=(ADDRESS=
(PROTOCOL=TCP)(Host=salmon)
(Port=1522))(CONNECT_DATA=(SID=otter)))
```

RESET_STATS

Purpose

Use the command `RESET_STATS` to reset the Oracle Names server statistics to the original values of the Oracle Names server at startup.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl RESET_STATS [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> RESET_STATS [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then only the current Oracle Names server's statistics are reset.

Example

```
NAMESCTL> RESET_STATS  
Confirm [yes or no]: yes  
Server statistics reset.
```

RESTART

Purpose

Use the command `RESTART` to initiate a reset of an Oracle Names server to its original state at startup.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system prompt:

```
namesctl RESTART [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> RESTART [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then only the current Oracle Names server is restarted.

Usage Notes

`RESTART` is the same as `STARTUP`, except that you use it when the Oracle Names server is already running.

Data is reloaded, statistics are reset, and all foreign data is flushed. Valid foreign cache data, that is, data with a TTL greater than 0, is retrieved from the checkpoint files.

Example

```
NAMESCTL> RESTART
Confirm [yes or no]: yes
Server restarted.
```

SAVE_CONFIG

Purpose

Use the `SAVE_CONFIG` command to compare the current configuration state of the Oracle Names server, including trace level, trace file, trace directory, and logging to the `names.ora` file. Any changes are stored in `names.ora`, preserving formatting, comments, and case as much as possible.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
lsnrctl SAVE_CONFIG
```

From the Oracle Names Control utility:

```
LSNRCTL> SAVE_CONFIG
```

Arguments

None

Example

```
NAMESCTL> SAVE_CONFIG  
Server saving the config file now
```

SET

Purpose

Use the `SET` command to alter an Oracle Names server's parameter values. These changes remain in effect until the Oracle Names is stopped. To make the changes permanent, use the `SAVE_CONFIG`, `SET SAVE_CONFIG_ON_STOP`, or `SET SAVE_CONFIG_INTERVAL` commands to save changes to the `names.ora` file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
NAMESCTL SET [parameter]
```

From the Oracle Names Control utility:

```
namesctl> SET [parameter]
```

Arguments

[*parameter*]: Specify a `SET` parameter to modify its configuration setting. Parameters are shown in the example output.

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

Example

```
NAMECTL> SET
The following operations are available after set
An asterisk (*) denotes a modifier or extended command:
cache_checkpoint_interval
default_domain
forwarding_available
log_file_name
log_stats_interval
NAMECTL_trace_level
password
requests_enabled
reset_stats_interval
save_config_interval
save_config_on_stop
server
trace_file_name
trace_level
```

SET CACHE_CHECKPOINT_INTERVAL

Purpose

Use the command `SET CACHE_CHECKPOINT_INTERVAL` to specify the time, in seconds, of how often to save all collected information about remote regions to the local cache file. By default, the cache checkpoint file name is `ckpccch.ora`.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET CACHE_CHECKPOINT_INTERVAL [time]
```

From the Oracle Names Control utility:

```
NAMECTL> SET CACHE_CHECKPOINT_INTERVAL [time]
```

Arguments

{time}: Specify the number of seconds.

For example, to increase the interval to 36 hours, set the following:

```
NAMESCTL> SET CACHE_CHECKPOINT_INTERVAL 129600
```

Usage Notes

Minimum Value: 10

Maximum Value: 259200 (3 days)

Default Value: 0 (disabled)

Example

```
NAMESCTL> SET CACHE_CHECKPOINT_INTERVAL 12
```

SET DEFAULT_DOMAIN

Purpose

Use the command `SET DEFAULT_DOMAIN` to set the domain from which the Oracle Names Control utility most often looks up names resolution requests. The domain set is used for the duration of the session, ignoring the `NAMES.DEFAULT_DOMAIN` parameter configured in the `sqlnet.ora` file.

Prerequisites

The Oracle Names Control utility must be loaded.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Oracle Names Control utility:

```
NAMESCTL> SET DEFAULT_DOMAIN [domain_name]
```

Arguments

{domain}: Specify the domain name. The root domain is the default domain.

Usage Notes

When a default domain is set, it is automatically appended to any unqualified net service name or service name. For example, if the default domain is set to `us.acme.com`, the global name `sales.us.acme.com` can be queried using:

```
NAMESCTL> QUERY sales
```

Example

```
NAMESCTL> SET DEFAULT_DOMAIN us.acme.com  
Default domain is now "US.ACME.COM"
```

SET FORWARDING_AVAILABLE

Purpose

Use the command `SET FORWARDING_AVAILABLE` to set forwarding to remote Oracle Names servers for client name requests.

Prerequisites

Oracle Names server must be running.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET FORWARDING_AVAILABLE [yes | no] [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET FORWARDING_AVAILABLE [yes | no] [onames_server] [...]
```

Arguments

[*yes* | *no*]: Specify *yes* or *on* to set forwarding on. Specify *no* or *off* (default) to set forwarding off.

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current Oracle Names server.

Usage Notes

This command is intended for Oracle Names servers that have no local clients and are exclusively handling requests from remote Oracle Names servers. This usually would apply only to Oracle Names servers in the root region when the root is configured without clients or services. If such an Oracle Names server is a performance bottleneck in cross-region request processing, then disabling forwarding in that Oracle Names server can reduce its workload in half. Rather than forward the request and return the answer, the Oracle Names server simply tells the requestor the address of the Oracle Names server that can answer the request. Note that there is no overall reduction in work; the work is simply displaced from the non-forwarding Oracle Names server to the requesting Oracle Names server.

Caution: If `SET FORWARDING_AVAILABLE` is set to `off` or `no`, any clients that rely directly on that Oracle Names server will be unable to resolve remote names. Clients are not capable of redirecting their requests as Oracle Names servers are. Their requests will fail at that point, even if other Oracle Names servers are listed in the `NAMES.PREFERRED_SERVERS` parameter in the `sqlnet.ora` file.

Example

```
NAMECTL> SET FORWARDING_AVAILABLE off
Request processing is now disabled.
```

SET LOG_FILE_NAME

Purpose

Use the command `SET LOG_FILE_NAME` to set the name for the Oracle Names server log file. By default, the log file name is `names.log`.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET LOG_FILE_NAME [file_name]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET LOG_FILE_NAME [file_name]
```

Arguments

`{file_name}`: Specify file name of the Oracle Names trace. The default file name is `names.log`.

Usage Notes

The `LOG_FILE_NAME` changes the destination of all logging messages.

Example

```
NAMESCTL> SET LOG_FILE_NAME namesvr1
```

SET LOG_STATS_INTERVAL

Purpose

Use the command `SET LOG_STATS_INTERVAL` to change the frequency at which the statistics are logged to the log file. By default, the log file name is `names.log`.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET LOG_STATS_INTERVAL [time]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET LOG_STATS_INTERVAL [time]
```

Arguments

`{time}`: Specify the time in seconds or in [*n* day[s]] [*hh:mm:ss*]. For example, to increase the `LOG_STATS_INTERVAL` to 36 hours, both of the following can be set:

```
set log_stats_interval 129600
set log_stats_interval 1 day 12:00:00
```

You can specify any valid combination, such as the number of days combined with number of hours, minutes, and seconds, or just the number in hours.

Restrictions

Minimum Value: 10

Maximum Value: None

Special Value: 0 (never reset)

Default value: 0 (no logging)

Usage Notes

The `LOG_STATS_INTERVAL` value is initially set based on the value configured in `NAMES.LOG_STATS_INTERVAL` parameter in the `sqlnet.ora` file when the Oracle Names server is loaded. By default, the value is 0 (no logging). This command is intended to override that value during server operation.

Example

```
NAMESCTL> SET LOG_STATS_INTERVAL 7200
Statistic counter logging interval is now 2 hours
```

SET NAMESCTL_TRACE_LEVEL

Purpose

Use the command `SET NAMESCTL_TRACE_LEVEL` to set the level at which the Oracle Names Control utility can be traced.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET NAMESCTL_TRACE_LEVEL {level}
```

From the Oracle Names Control utility:

```
NAMESCTL> SET NAMESCTL_TRACE_LEVEL {level}
```

Arguments

`{level}`: Specify 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

Tracing assists in diagnosing unexpected or unidentifiable failures in processing the Oracle Names Control utility. Tracing writes a series of events from normal Oracle Names Control utility processing to an operating system file for review by the administrator.

When no level is specified, the setting is reset to the level specified by the `NAMESCTL.TRACE_LEVEL` parameter in the `sqlnet.ora` file.

Example

```
NAMESCTL> SET NAMESCTL_TRACE_LEVEL admin  
Controller's local trace level changed from 0 to 4
```

SET PASSWORD

Purpose

Use the command `SET PASSWORD` to set the password for privileged Oracle Names Control utility commands, such as `STOP`, `RESTART` and `RELOAD`.

The password entered should match the one set for the `NAMES.PASSWORD` parameter in the `names.ora` file.

Prerequisites

The Oracle Names Control utility must be loaded.

Password Required If One Has Been Set

Not applicable

Syntax

From the Oracle Names Control utility:

```
NAMECTL> SET PASSWORD [password]
```

Arguments

`{password}`: Specify the password.

Usage Notes

You can enter this command when you start up the Oracle Names Control utility or at any time during the session as needed.

When passed over the network, the password is always encrypted, regardless of how it is set.

See Also: ["Oracle Names Server Security"](#) on page 3-5

Example

```
NAMECTL> SET PASSWORD open_sesame
```

```
NAMECTL> SET PASSWORD  
enter name server password: onames9i
```

SET REQUESTS_ENABLED

Purpose

Use the command `SET REQUESTS_ENABLED` to determine whether the current Oracle Names server will respond to requests.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET REQUESTS_ENABLED [on | off]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET REQUESTS_ENABLED [on | off]
```

Arguments

[on] (default): Specify to have the Oracle Names server respond to requests.

[off]: Specify to send refusals to all clients that approach with names resolution requests. This is primarily useful for diagnostics when an Oracle Names server is functioning unexpectedly.

Example

```
NAMESCTL> SET REQUESTS_ENABLED OFF
Confirm [yes or no]: yes
General request processing is now disabled
```

SET RESET_STATS_INTERVAL

Purpose

Use the command `SET RESET_STATS_INTERVAL` to change the time between the statistics being reset to 0 or initial values in the current Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET RESET_STATS_INTERVAL [time]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET RESET_STATS_INTERVAL [time]
```

Arguments

`{time}`: Specify the time in seconds or in [*n* day[*s*]] [*hh:mm:ss*]. For example, to increase the `RESET_STATS_INTERVAL` to 72 hours, the following can be set:

```
SET RESET_STATS_INTERVAL 259200
```

```
SET RESET_STATS_INTERVAL 3 days
```

Minimum Value: 10 seconds

Maximum Value: no maximum

Default value: 0 (never reset)

Usage Notes

The `RESET_STATS_INTERVAL` value is initially set based on the `NAMES.RESET_STATS_INTERVAL` parameter in the `names.ora` file. This command is intended to override that value during Oracle Names server operation.

Example

```
NAMECTL> SET RESET_STATS_INTERVAL 1 day
Statistic counter reset interval is now 24 hours
```

SET SAVE_CONFIG_INTERVAL

Purpose

Use the command `SET SAVE_CONFIG_INTERVAL` to save any changes made by `SET` commands to the `names.ora` at an interval.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET SAVE_CONFIG_INTERVAL [time]
```

From the Oracle Names Control utility:

```
NAMECTL> SET SAVE_CONFIG_INTERVAL [time]
```

Arguments

`{time}`: Specify the time in seconds.

Example

```
NAMECTL> SET SAVE_CONFIG_INTERVAL 10
Scheduled configuration save to occur in 3 minutes
```

SET SAVE_CONFIG_ON_STOP

Purpose

Use the command `SET SAVE_CONFIG_ON_STOP` to specify whether or not changes made to an Oracle Names server's parameter values by the `SET` commands are to be saved to the `names.ora` file at the time the Oracle Names server is stopped with the `SHUTDOWN` or `STOP` commands.

When changes are saved, the Oracle Names Control utility tries to preserve formatting, comments, and letter case.

To have all parameters saved right away, use the `SAVE_CONFIG` command.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET SAVE_CONFIG_ON_STOP [on | off]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET SAVE_CONFIG_ON_STOP [on | off]
```

Arguments

[on]: Specify to save configuration to `names.ora`.

[off] (default): Specify to not save configuration to `names.ora`.

Example

```
NAMESCTL> SET SAVE_CONFIG_ON_STOP on
```

SET SERVER

Purpose

Use the command `SET SERVER` to set Oracle Names server for the Oracle Names Control utility to administer.

Prerequisites

The Oracle Names Control utility must be loaded

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the Oracle Names Control utility:

```
NAMESCTL> SET SERVER [onames_server | (ADDRESS=(protocol_address_information))]
```

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about protocol addresses and parameters

Arguments

[*onames_server* | (ADDRESS=(*protocol_address_information*))]:
Specify a valid Oracle Names server or Oracle Names server address.

If you specify no argument, this command defaults to the values set by the `NAMES.PREFERRED_SERVERS` parameter in the `sqlnet.ora` file.

Usage Notes

`SET SERVER` enables switching between multiple Oracle Names servers while running the Oracle Names Control utility. The qualifier can be a name where the name is defined in the memory of the current Oracle Names server, or it can be the TNS address of any Oracle Names server.

The Oracle Names server name specified is resolved through normal name lookup. Another Oracle Names server can be set only if the current Oracle Names server knows or can retrieve its address. If no current Oracle Names server is set, you must enter an address to complete this command.

Example

```
NAMESCTL> SET SERVER server1.us.acme
```

SET TRACE_FILE_NAME

Purpose

Use the command `SET TRACE_FILE_NAME` to set the name of the Oracle Names server trace file. By default, the trace file name is `names.trc`.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET TRACE_FILE_NAME [file_name]
```

From the Oracle Names Control utility:

```
NAMESCTL> SET TRACE_FILE_NAME [file_name]
```

Arguments

`{file_name}`: Specify the file name of the Oracle Names trace.

Example

```
NAMESCTL> SET TRACE_FILE_NAME namesvr1
```

SET TRACE_LEVEL

Purpose

Use the command `SET TRACE_LEVEL` to set a specific level of tracing for the Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SET TRACE_LEVEL {level}
```

From the Oracle Names Control utility:

```
NAMESCTL> SET TRACE_LEVEL {level}
```

Arguments

`{level}`: Specify 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

Tracing assists in diagnosing unexpected or unidentifiable failures in processing the current Oracle Names server. It writes a series of events from normal Oracle Names server processing to an operating system file for review by the administrator.

After the `TRACE_LEVEL` is set, tracing begins immediately. All operations are traced until it is reset to trace level `off`.

SHOW

Trace files can grow very large. Remember to turn trace level off after diagnosing the problem.

Example

```
NAMECTL> SET TRACE_LEVEL admin
Trace level is now 6.
```

SHOW

Purpose

Use the command `SHOW` to view the Oracle Names server's parameter values.

All of the `SET` commands listed except `SET PASSWORD` have equivalent `SHOW` commands.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW
namesctl SHOW [command]
```

From the Oracle Names Control utility:

```
NAMECTL> SHOW
NAMECTL> SHOW [command]
```

Arguments

[*parameter*]: Specify a `SHOW` parameter to view its configuration setting. Parameters are shown in the example output.

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

Example

```
NAMECTL> SHOW
```

The following operations are available after show

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

```
cache_checkpoint_interval
```

```
default_domain
```

```
forwarding_available
```

```
log_file_name
```

```
log_stats_interval
```

```
NAMECTL_trace_level
```

```
requests_enabled
```

```
reset_stats_interval
```

```
save_config_interval
```

```
save_config_on_stop
```

```
server
```

```
status
```

```
system_queries
```

```
trace_file_name
```

```
trace_level
```

```
version
```

SHOW CACHE_CHECKPOINT_INTERVAL

Purpose

Use the command `SHOW CACHE_CHECKPOINT_INTERVAL` to show the frequency at which the Oracle Names server's cache is written to the cache checkpoint file. By default, the cache checkpoint file name is `ckpccch.ora`.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW CACHE_CHECKPOINT_INTERVAL
```

From the Oracle Names Control utility

```
NAMESCTL> SHOW CACHE_CHECKPOINT_INTERVAL
```

Arguments

None

Usage Notes

The interval is initially set with the value in `NAMES.CACHE_CHECKPOINT_INTERVAL` parameter in the `names.ora` file. By default, the value is 0, which disables cache checkpointing. Data written to the cache checkpoint file includes net service names, protocol addresses, and Oracle Names server addresses that were learned by the Oracle Names server as a result of forwarding a query to a foreign region on behalf of the client.

Example

```
NAMESCTL> SHOW CACHE_CHECKPOINT_INTERVAL  
Cache checkpoint interval is currently 8 minutes 20 seconds
```

SHOW DEFAULT_DOMAIN

Purpose

Use the command `SHOW DEFAULT_DOMAIN` to show the domain set by the [SET DEFAULT_DOMAIN](#) command or the `NAMES.DEFAULT_DOMAIN` parameter in the `sqlnet.ora` file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW DEFAULT_DOMAIN
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW DEFAULT_DOMAIN
```

Arguments

None

Usage Notes

When a default domain is set, it is automatically appended to any unqualified net service name or service name. For example, if the default domain is set to `us.acme.com`, the global name `sales.us.acme.com` can be queried using the following syntax:

```
NAMESCTL> QUERY sales
```

Example

```
NAMESCTL> SHOW DEFAULT_DOMAIN  
Current default domain is "com"
```

SHOW FORWARDING_AVAILABLE

Purpose

Use the command `SHOW FORWARDING_AVAILABLE` to show whether the Oracle Names server is forwarding client requests to remote Oracle Names servers.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW FORWARDING_AVAILABLE [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW FORWARDING_AVAILABLE [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current Oracle Names server.

Usage Notes

By default, all Oracle Names servers forward requests to remote Oracle Names servers. If forwarding is disabled, then requests to remote Oracle Names server are redirected to an Oracle Names server in the region that is authoritative to the requested name.

Disabling forwarding can reduce the load on a particular Oracle Names server, but makes it impossible to direct clients to remote Oracle Names servers.

Use the `SET FORWARDING_AVAILABLE` command to turn forwarding on or off.

Example

```
NAMESCTL> SHOW FORWARDING_AVAILABLE
Request forwarding is currently enabled
```

SHOW LOG_FILE_NAME

Purpose

Use the command `SHOW LOG_FILE_NAME` to show the name of the file where the Oracle Names server writes logging information.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW LOG_FILE_NAME
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW LOG_FILE_NAME
```

Arguments

None

Usage Notes

The log file name is initially set with the value of the `NAMES.LOG_FILE` parameter in the `names.ora` file. By default, the log file name is `names.log`.

Example

```
NAMESCTL> SHOW LOG_FILE_NAME
Log file name is currently
/oracle9i/network/log/names.log
```

SHOW LOG_STATS_INTERVAL

Purpose

Use the command `SHOW LOG_STATS_INTERVAL` to display the frequency at which statistics are logged to the log file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW [onames_server] [...] LOG_STATS_INTERVAL
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW [onames_server] [...] LOG_STATS_INTERVAL
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current server.

Usage Notes

The interval is initially set with the value of the `NAMES.LOG_STATS_INTERVAL` parameter in the `names.ora` file. By default, the value is 0, or no logging.

Example

```
NAMESCTL> SHOW LOG_STATS_INTERVAL  
Statistic counter logging is currently disabled
```

SHOW NAMESCTL_TRACE_LEVEL

Purpose

Use the command `SHOW NAMESCTL_TRACE_LEVEL` to display the level at which the Oracle Names Control utility is being traced.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW NAMESCTL_TRACE_LEVEL
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW NAMESCTL_TRACE_LEVEL
```

Arguments

None

Usage Notes

Tracing assists in diagnosing unexpected or unidentifiable failures in processing the Oracle Names Control utility. Tracing writes a series of events from normal Oracle Names Control utility processing to an operating system file for review by the administrator.

Tracing output is at four levels

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

SHOW REQUESTS_ENABLED

Example

```
NAMESCTL> SHOW NAMESCTL_TRACE_LEVEL  
Controller's trace level is currently 0
```

SHOW REQUESTS_ENABLED

Purpose

Use the command `SHOW REQUESTS_ENABLED` to show whether the Oracle Names server is responding to requests.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW REQUESTS_ENABLED [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW REQUESTS_ENABLED [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current server.

Usage Notes

If requests are disabled, all requests to the Oracle Names server will be refused.

Example

```
NAMESCTL> SHOW REQUESTS_ENABLED  
General request processing is currently enabled
```

SHOW RESETS_STATS_INTERVAL

Purpose

Use the command `SHOW RESETS_STATS_INTERVAL` to display the interval set for dumping statistics to the log file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW RESET_STATS_INTERVAL
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW RESET_STATS_INTERVAL
```

Arguments

None

Usage Notes

The interval is initially set with the value of the `NAMES.RESET_STATS_INTERVAL` parameter in the `names.ora` file. By default the value is set to 0, or no reset.

For example, if statistics are reset every day, then the statistics will represent totals for the day rather than the entire time the Oracle Names server has been running.

Example

```
NAMESCTL> SHOW RESET_STATS_INTERVAL  
Statistic counter reset interval is currently 5 minutes
```

SHOW SAVE_CONFIG_INTERVAL

Purpose

Use the command `SHOW SAVE_CONFIG_INTERVAL` to display the interval of when the `SET` command is scheduled to save to the `names.ora` file.

Prerequisites

The command only displays an interval if the `SET SAVE_CONFIG_INTERVAL` command was set.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW SAVE_CONFIG_INTERVAL
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW SAVE_CONFIG_INTERVAL
```

Arguments

None

Example

```
NAMESCTL> SHOW SAVE_CONFIG_INTERVAL  
Configuration will be saved in 1 minute 35 seconds
```

SHOW SAVE_CONFIG_ON_STOP

Purpose

Use the command `SHOW SAVE_CONFIG_ON_STOP` to show whether `SET` commands changes are scheduled to be saved to the `names.ora` file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW SAVE_CONFIG_ON_STOP
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW SAVE_CONFIG_ON_STOP
```

Arguments

None

Example

```
NAMESCTL> SHOW SAVE_CONFIG_ON_STOP  
Save_config_on_stop is currently ON
```

SHOW SERVER

Purpose

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the [SET PASSWORD](#) command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW SERVER
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW SERVER
```

Arguments

None

Usage Notes

SHOW SERVER displays the current Oracle Names server that commands will operate on.

Example

```
NAMESCTL> SHOW SERVER
currently managing name server "NameServer.us.acme.com"
Version banner is "Oracle Names for SunOS: Version 9.2.0.1.0"
```

SHOW STATUS

Purpose

Use the command `SHOW STATUS` to display general status information about the Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW STATUS [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESTCL> SHOW STATUS [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current server.

Usage Notes

This command is identical to the `STATUS` command.

Example

```
NAMESTCL> SHOW STATUS  
Version Banner is "Oracle Names for SunOS: Version 9.2.0.1.0" Server has been  
running for:1 day 2 hours 3 minutes 35.16 seconds....
```

SHOW SYSTEM_QUERIES

Purpose

Use the command `SHOW SYSTEM_QUERIES` to display the next occurrence of all system queries.

Prerequisites

This is relevant only for distributed configurations. There are no system queries with only one administrative region.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW SYSTEM_QUERIES
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW SYSTEM_QUERIES
```

Arguments

None

Usage Notes

System queries are performed at intervals to keep information among Oracle Names servers current.

There is no specific action that can change the activities listed as system queries. Being able to show them gives the administrator an understanding of when a system change will occur, and can assist in a decision to `RESTART`.

Example

```
NAMESCTL> SHOW SYSTEM_QUERIES
System query index number:1
Query ID:49824
Query next issued in:2 hours 55 min 3.84 seconds
Query state:2
Name:""
Desired data type:ns.smd
```

SHOW TRACE_FILE_NAME

Purpose

Use the command `SHOW TRACE_FILE_NAME` to display the directory path and file name of the current Oracle Names server's trace file.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW TRACE_FILE_NAME
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW TRACE_FILE_NAME
```

Arguments

None

Usage Notes

The trace file name is initially set with the value of the `NAMES.TRACE_FILE` parameter in the `names.ora` file. The default value is `names.trc`.

Example

```
NAMESCTL> SHOW TRACE_FILE_NAME
Trace file name is currently
/oracle9i/network/names.trc
```

SHOW TRACE_LEVEL

Purpose

Use the command `SHOW TRACE_LEVEL` to display the level of tracing for the current Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW TRACE_LEVEL
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW TRACE_LEVEL
```

Arguments

None

Usage Notes

Tracing assists in diagnosing unexpected or unidentifiable failures in processing the current Oracle Names server. Tracing writes a series of events from normal Oracle Names server processing to an operating system file for review by the administrator.

Tracing is available at the following levels:

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

Example

```
NAMESCTL> SHOW TRACE_LEVEL
Trace level is currently 0
```

SHOW VERSION

Purpose

Use the command `SHOW VERSION` to display the current version and name of the Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHOW VERSION [onames_server] [...] VERSION
```

From the Oracle Names Control utility:

```
NAMESCTL> SHOW VERSION [onames_server] [...] VERSION
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current server.

Usage Notes

This command identifies the Oracle Names server by name and version, which can be useful when clearing up minor difficulties. This command is enabled every time you connect Oracle Names Control utility to an Oracle Names server.

Example

```
NAMESCTL> SHOW VERSION
Currently managing Oracle Names server "NameServer.com"
Version banner is "Oracle Names for SunOS: Version 9.2.0.1.0"
```

SHUTDOWN

Purpose

Use the command `SHUTDOWN` to stop one or more Oracle Names servers.

Prerequisites

The Oracle Names server(s) must be started.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl SHUTDOWN [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> SHUTDOWN [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. When no arguments are supplied, only the current Oracle Names server is shut down.

Usage Notes

`SHUTDOWN` stops the current Oracle Names server and unloads the program from memory. An Oracle Names server should be shut down only for operational reasons like upgrades or computer maintenance. The preferred way to stop and start an Oracle Names server is using the `RESTART` command because you can perform it from anywhere in the network. If `SHUTDOWN` and `START` are processed individually, then they must occur on the Oracle Names server computer.

This command is identical to the `STOP` command.

Example

```
NAMESCTL> SHUTDOWN
Confirm [yes or no] yes
Server shut down.
```

START

Purpose

Use the command `START` to load the Oracle Names server and start loading system and local administrative region data.

Prerequisites

Oracle Names server must be stopped.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl START
```

From the Oracle Names Control utility:

```
NAMESCTL> START
```

Arguments

None

Usage Notes

`START` is the command to initially load an Oracle Names server into memory. At startup, the Oracle Names server reads its configuration files to set up its operating parameters, and then loads all data for the administrative region.

Security on Oracle Names server startup is supplied through the operating system that Oracle Names is installed on. Because an Oracle Names server must be started from a local session, network security is not an issue.

This command is identical to the `STARTUP` command.

Example

```
NAMECTL> START
Starting "/private/dsteiner/sales/bin/names"...server successfully started

Currently managing name server "namesrv1.us.acme.com"
Version banner is "Oracle Names for Solaris: Version 9.2.0.1.0"

Server name:                               namesrv1.us.acme.com
Server has been running for:                0.16 seconds
Request processing enabled:                 yes
Request forwarding enabled:                 yes
Requests received:                          0
Requests forwarded:                        0
Foreign data items cached:                  0
Region data next checked for reload in:    not set
Region data reload check failures:         0
Cache next checkpointed in:                 not set
Cache checkpoint interval:                  not set
Cache checkpoint file name:
/private/dsteiner/sales/network/names/ckpcch.ora
Statistic counters next reset in:          not set
Statistic counter reset interval:          not set
Statistic counters next logged in:         not set
Statistic counter logging interval:        not set
Trace level:                               0
Trace file name:
/private/dsteiner/sales/network/trace/names_10841.trc
Log file name:
/private/dsteiner/sales/network/log/names.log
System parameter file name:
/private/dsteiner/sales/network/admin/names.ora
Command-line parameter file name:         ""
Administrative region name:                 ""
```

Administrative region description:	" "
ApplTable Index:	0
Contact	" "
Operational Status	0
Save Config on Stop	yes

START_CLIENT_CACHE

Purpose

Use the command `START_CLIENT_CACHE` to start the client cache process. The client cache process finds a list of the local Oracle Names servers in a static list of preferred Oracle Names servers in the `sqlnet.ora` file or in the discovery list file (`.sdns.ora` or `sdns.ora`) the first time it runs. Afterwards, it relies on its cache checkpoint file. Once the client cache has contact with the local region server(s), it begins to cache data on all local servers, various foreign servers, and all the data it has been asked to find.

See Also: *Oracle9i Net Services Administrator's Guide* for more information about the usage of this command

Prerequisites

The client cache process must be stopped.

An Oracle Names server list must exist before you run the client cache process.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl START_CLIENT_CACHE
```

From the Oracle Names Control utility:

```
NAMESCTL> START_CLIENT_CACHE
```

Arguments

None

Usage Notes

Once started, the client cache daemon process stores all information received from an Oracle Names server, making lookups faster.

Example

```
NAMESCTL> START_CLIENT_CACHE
Starting ".../onrsd"...server successfully started
```

STARTUP

Purpose

Use the command `STARTUP` to load the Oracle Names server and start loading system and local administrative region data.

Prerequisites

Oracle Names server must be stopped.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl STARTUP
```

From the Oracle Names Control utility:

```
NAMESCTL> STARTUP
```

Arguments

None

Usage Notes

This command is identical to the `START` command.

Example

See example for `START`.

STATUS

Purpose

The command `STATUS` display statistics for one or more Oracle Names servers.

Prerequisites

Oracle Names server must be started.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl STATUS [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESTCL> STATUS [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. When no arguments are supplied, status is given only for the current Oracle Names server.

Usage Notes

`STATUS` shows the activity of the Oracle Names server over time and its state at a point in time.

Example

```
NAMESTCL> STATUS  
Version banner is "Oracle Names for SunOS:  
9.2.0.1.0"  
Server name: NSERVER.com  
Server has been running for: 1 day 20 hours .....
```

STOP

Purpose

Use the command `STOP` to stop one or more Oracle Names servers.

Prerequisites

Oracle Names server must be started.

Password Required If One Has Been Set

Yes. If a password is set, the `SET PASSWORD` command must be issued prior to this command.

Syntax

From the operating system:

```
namesctl STOP [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMESCTL> STOP [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. When no arguments are supplied, only the current Oracle Names server is stopped.

Usage Notes

`STOP` stops the current Oracle Names server and unloads the program from memory. An Oracle Names server should be shut down only for operational reasons like upgrades or computer maintenance. The preferred way to stop and start an Oracle Names server is using the `RESTART` command because you can issue it from anywhere in the network. If `STOP` and `START` are processed individually, they must occur on the Oracle Names server computer.

This command is identical to the `SHUTDOWN` command.

Example

```
NAMESCTL> STOP
```

```
Confirm [yes or no]: yes
Server shut down
```

TIMED_QUERY

Purpose

Use the command `TIMED_QUERY` to show all data in the Oracle Names server cache.

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl TIMED_QUERY [timestamp]
```

From the Oracle Names Control utility:

```
NAMESTCL> TIMED_QUERY [timestamp]
```

Arguments

Time in seconds

Usage Notes

The first time you issue this command, do not use the [*timestamp*] argument. The output returned includes all the objects registered. At the end of returned output is a `Last timestamp` field that provides a time stamp.

After there has been an update of data, issue the command again. This time, pass the time stamp provided from the first output. The command displays the data updated since the last the command was issued.

Example

In the following output, `TIMED_QUERY` is issued for the first time:

```
NAMESTCL> TIMED_QUERY
Total response time: 0.26 seconds
```

```
Response status:      normal, successful completion
Number of answers:    7
Name:                 [root]
    data type is "ns.smd"
    Syntax is DOMAIN: namesrv1
Name:                 us.acme.com
    data type is "ns.smd"
    Syntax is DOMAIN: us.acme.com
Name:                 hr.us.acme.com
    data type is "a.smd"
    Syntax is ADDR:
    ...(address=(protocol=tcp)(host=namesrv1-server)(port=1575))
Name:                 namesrv2
    data type is "a.smd"
    Syntax is ADDR:
    ...(address=(protocol=tcp)(Port=1575)(host=namesrv3-server))
    data type is "tos.npd.ond"
    Syntax is CTEXT: "ORACLE_NAMESERVER"
    data type is "host.nm.ond"
    Syntax is TEXT: "namesrv3-server"
Last timestamp:      1072
```

Note that the Last timestamp is 1072.

In the following output, TIMED_QUERY is issued before new data is registered.

```
NAMESCTL> TIMED_QUERY 1072
Total response time:  0.19 seconds
Response status:     NNC-00408: name "1072" exists but desired data does not
```

In the following output, a new object is registered:

```
NAMESCTL> REGISTER sales -t oracle_database -d (DESCRIPTION=(ADDRESS=
(PROTOCOL=TCP)(HOST=sales-server)(PORT=1575))(CONNECT_DATA=(SERVICE_
NAME=sales.us.acme.com)))
Total response time:  0.29 seconds
Response status:     normal, successful completion
```

When `TIMED_QUERY` is issued again, the newly registered data displays in the output:

```

NAMECTL> TIMED_QUERY 1072
Total response time: 0.13 seconds
Response status:    normal, successful completion
Number of answers:  1
Name:               sales
                   data type is "a.smd"
                   Syntax is ADDR:

... (description=(address=(protocol=tcp)(host=sales-server)(port=1575))(connect_
data=(service_name=sales.us.acme.com)
Last timestamp:     6107

```

UNREGISTER

Purpose

Use the command `UNREGISTER` to remove a network object from an Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system

```

namesctl UNREGISTER {object_name} [-t service_type]
[-d[(DESCRIPTION=)(ADDRESS=...)](CONNECT_DATA=(SERVICE_NAME|SID=service_
name|SID)))] [-h host] [-l listener_name]

```

From the Oracle Names Control utility:

```

NAMECTL> UNREGISTER {object_name} [-t service_type]
[-d[(DESCRIPTION=)(ADDRESS=...)](CONNECT_DATA=(SERVICE_NAME|SID=service_
name|SID)))] [-h host] [-l listener_name]

```

Arguments

`{object_name}`: Specify the object name.

`[-t service_type]`: Specify the service type of the object:

ORACLE_DATABASE for an Oracle database

ORACLE_LISTENER for a listener

ORACLE_NAMESERVER for an Oracle Names server

Note: To remove an Oracle Names server, use the [UNREGISTER_NS](#) command rather than the `UNREGISTER` command.

`[-d]`: Specify the protocol address of the listener or Oracle Names server object, or complete connect descriptor of the database object, net service name, alias, or global database link

See Also: [Chapter 5, "Protocol Address Configuration"](#) for further information about protocol addresses and parameters

`[-h host]`: Specify the host name the object resides on.

`[-l listener_name]`: Specify the name of the listener object.

The service type, address description, host, and listener name options are not necessary to make the registration process appear to work. However, they are necessary to make the registration useful. In other words, an object name registered without an address cannot be used.

Note: The protocol address cannot contain any spaces.

Usage Notes

This command provides a manual mechanism for unregistering a service. The definition for that object is removed from the Oracle Names servers in the region. If the object was registered with an address, listener name, or a host name, then the address, listener name, or host name must be provided on the command line in order to unregister the object.

Example

```
NAMESCTL> UNREGISTER sales -t oracle_database -d
(description=(address=(protocol=tcp)(host=sales-server)(port=1575))(connect_
data=(service_name=db3)))
```

UNREGISTER_NS

Purpose

Use the UNREGISTER_NS command to undefine an Oracle Names server and its authoritative domain.

Prerequisites

None

Password required if one has been set

No. If a password is set, the SET PASSWORD command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl UNREGISTER_NS {onames_server}{domain}
```

From Oracle Names Control utility:

```
NAMESCTL> UNREGISTER_NS {onames_server}{domain}
```

Arguments

{onames_server}: Specify the Oracle Names server name. If the `sqlnet.ora` file parameter `NAMES.DEFAULT_DOMAIN` is set to the same domain name as the Oracle Names server's domain, then the Oracle Name server does not need to be qualified with its domain.

{domain}: Specify the domain name.

Usage Notes

This command provides a mechanism for unregistering an Oracle Names server as an authoritative server for a given domain. This command removes the NS . SMD record for the Oracle Names from the domain, and deletes the Oracle Names server and its A . SMD address record.

This command will fail if either the domain exists and has non-NS records or the server exists and has a type of service record that is other than 'ORACLE_NAMESERVER'.

Ordinarily, Oracle Names servers maintain their own data by registering themselves when they start. This command is provided as a manual way to manage domain and Oracle Names server data if for some reason the Oracle Names server cannot. This can occur if the region database tables are set up as read-only for security reasons.

If the Oracle Names servers are not registering themselves, then this command should be used to define the region topology data. Each Oracle Names server in the region should be defined using this command for each top-level domain in the region. Usually, the top level consists of a single parent domain, for example, acme . com. However, a region may also have multiple sibling parent domains, for example, a region covering North America would have US, CA and MX as its top-level parent domains.

Note the regions which were defined using the Oracle Network Manager in SQL*Net version 2 have ns . smd records defined for every domain in the administrative region, but in Oracle Net only the top-level parent domains need to have NS . SMD records defined for each server in the region.

Example

```
NAMECTL> UNREGISTER_NS namesrv1.us.acme.com us.acme.com
Total response time: 7 minutes 59.14 seconds
Response status: normal, successful completion
```

VERSION

Purpose

Use the command `VERSION` to display the current version and name of the Oracle Names server.

Prerequisites

None

Password Required If One Has Been Set

No. If a password is set, the `SET PASSWORD` command does not need to be issued prior to this command.

Syntax

From the operating system:

```
namesctl VERSION [onames_server] [...]
```

From the Oracle Names Control utility:

```
NAMECTL> VERSION [onames_server] [...]
```

Arguments

[*onames_server*]: Specify zero or more Oracle Names servers separated by a space. If no names are supplied, then the setting is displayed for the current server.

Usage Notes

The output identifies the Oracle Names server by name and version, which can be useful when clearing up minor difficulties.

Example

```
NAMECTL> VERSION
Currently managing Oracle Names server "NameServer.com"
Version banner is "Oracle Names for SunOS: Version 9.2.0.1.0"
```


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 4, "Syntax Rules for Configuration Files"](#)
- [Chapter 5, "Protocol Address Configuration"](#)
- [Chapter 6, "Profile Parameters \(sqlnet.ora\)"](#)
- [Chapter 7, "Local Naming Parameters \(tnsnames.ora\)"](#)
- [Chapter 8, "Listener Parameters \(listener.ora\)"](#)
- [Chapter 9, "Oracle Connection Manager Parameters \(cman.ora\)"](#)
- [Chapter 10, "Oracle Names Parameters \(names.ora\)"](#)
- [Chapter 11, "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 these topics:

- [Configuration File Syntax Overview](#)
- [Further Syntax Rules for Configuration Files](#)
- [Network Character Set](#)
- [Character Set](#)

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.

Even if you do not choose to indent your files in this way, you must indent a wrapped line by at least one space, or it will be misread as a new parameter. The following layout is acceptable:

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

The following layout is *not* acceptable:

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

Further 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, it is interpreted as a continuation of the previous line.
- All characters must belong to the network character set

See Also: ["Network Character Set"](#) on page 4-4

- Keywords are not case sensitive. Values may be case sensitive, depending on the operating system and protocol.
- Spaces around the "=" sign are optional in keyword-value pairs.
- There is a hierarchy of keywords in 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 quotes (") or single quotes (').
- The maximum length of a connect descriptor is 4 KB
- Comments can be included using the pound sign # at the beginning of a line. Anything following the sign to the end of the line is considered a comment.
- If the keyword-value pair consists of a single word or a concatenation of words on either side of the equal sign, no parentheses are needed.

Network Character Set

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 is quoted. Either single or double quotes can be used to enclose a value containing reserved symbols. To include a quote within a value that is surrounded by quotes, use different quote types. 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

The listener name, net service name, and Oracle Names server are limited to the following character set:

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

The first character must be an alphabetical 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 (originally limited to eight characters), 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**, **Oracle Connection Manager**, or **Oracle Names server**) 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 these topics:

- [ADDRESSes and ADDRESS_LISTs](#)
- [Protocol Parameters](#)
- [Recommended Port Numbers](#)
- [Port Number Limitations](#)

ADDRESSES and ADDRESS_LISTS

Protocol addresses are comprised of [ADDRESS](#) and [ADDRESS_LIST](#) elements.

ADDRESS

Purpose

The `ADDRESS` parameter defines a protocol address.

Embed this parameter under an `ADDRESS_LIST` or `DESCRIPTION` parameter. A `DESCRIPTION` is used in a `tnsnames.ora` or a `listener.ora` file.

See Also: "[Protocol Parameters](#)" on page 5-3 for each protocol's required parameters

Example

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

ADDRESS_LIST

Purpose

The `ADDRESS_LIST` parameter defines 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)))
(ADDRESS_LIST=
 (ADDRESS=
 (PROTOCOL=tcp)
 (HOST=finance-server)
 (PORT=1521)))
```

Protocol Parameters

The listener, Oracle Names server and Oracle Connection Manager are identified by protocol addresses. [Table 5–1](#) describes the parameters used by the supported Oracle protocols:.

Table 5–1 Protocol-Specific Parameters

Protocol	Parameter	Description
IPC	PROTOCOL	Specify <code>ipc</code> as the value.
	KEY	Specify a unique name for the service. Oracle Corporation 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 computer.
	PIPE	Specify the pipe name you used to connect to the database server (the same <code>PIPE</code> keyword you specified on server with Named Pipes). This name can be any arbitrary name. Example: <code>(PROTOCOL=nmp)(SERVER=sales)(PIPE=dbpipe0)</code>
TCP/IP	PROTOCOL	Specify <code>tcp</code> as the value.
	HOST	Specify the host name or IP address of the computer.
	PORT	Specify the listening port number. Example: <code>(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)</code> <code>(PROTOCOL=tcp)(HOST=44.25.186.204)(PORT=1521)</code> See Also: "Recommended Port Numbers" on page 5-5

Protocol	Parameter	Description
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: (<code>PROTOCOL=tcps</code>) (<code>HOST=sales-server</code>) (<code>PORT=2484</code>) (<code>PROTOCOL=tcps</code>) (<code>HOST=44.25.186.204</code>) (<code>PORT=2484</code>)
		See Also: " Recommended Port Numbers " on page 5-5

Recommended Port Numbers

Table 5–2 lists the recommends the port numbers.

Table 5–2 Recommended Port Numbers

Port	Description
1521	Default listening port for client connections to the listener. In future releases, this port number may change to the officially registered port number of 2483 for TCP/IP and 2484 for TCP/IP with SSL.
1575	Default and officially registered listening port for client connections to an Oracle Names server using TCP/IP or TCP/IP with SSL.
1630	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 Corporation allows port numbers from 1 to 65535. Port numbers less than 1024 are reserved for use by privileged processes on many operating systems.

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, follow these general steps. Your operating system may require different procedures.

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

See Also: *Oracle9i Net Services Administrator's Guide*

2. Log in as super user (`root`) and 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 super user.

Ensure that the permissions of the individual directories found in the path names to these files, starting with the root directory, are also modified in the same way.

3. Start the listener as `root`.

At the operating system prompt, enter `tnslsnr` with optional command line arguments. The usage is as follows:

```
tnslsnr [listener_name] [-user user] [-group group]
```

where:

<code><i>listener_name</i></code>	Specify the name of the listener. If omitted, the default name LISTENER will be used.
<code>-user <i>user</i></code>	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.
<code>-group <i>group</i></code>	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.

The listener will temporarily switch to the provided user and group immediately after startup. All subsequent operations will be done with the specified user and group privileges, except the system calls necessary to listen

on configured endpoints. The listener will revert to super user (`root`) for a short period of time 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 will switch to the specified user and group irreversibly. Therefore, the listener will give up the `root` privilege that it initially had. In the current release, `-user` and `-group` command line arguments only accept user and group identifiers specified in numeric form.

For example, to execute a root listener 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. Note that 37555 could be the identifier for user `oracle` and 16 could be the identifier for the `dba` group.

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

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

Important Notes:

- Oracle Corporation recommends that the user under which the listener process runs be `oracle`, as described in the example in Step 3, or whichever user the listener process normally runs as on the operating system.
- Do not leave the listener process running as root because doing so is a security vulnerability.

Profile Parameters (sqlnet.ora)

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

This chapter contains these topics:

- [Overview of Profile Configuration File](#)
- [Profile Parameters](#)

Overview of Profile Configuration File

The `sqlnet.ora` file enables you to:

- 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, `sqlnet.ora` is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and the `ORACLE_HOME\network\admin` directory on Windows operating systems. `sqlnet.ora` can also be stored in the directory specified by the `TNS_ADMIN` environment variable.

Profile Parameters

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

BEQUEATH_DETACH

Purpose

Use the parameter `BEQUEATH_DETACH` to turn signal handling on or off for UNIX systems.

Default

`no`

Values

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

Example

```
BEQUEATH_DETACH=yes
```

DISABLE_OOB

Purpose

If turned `off`, the parameter `DISABLE_OOB` enables Oracle Net to send and receive "break" messages using urgent data provided by the underlying protocol.

If turned `on`, disables the ability to send and receive "break" messages using urgent data provided by the underlying protocol. Once enabled, this feature applies to all protocols used by this client.

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.

Default

`off`

Example

`DISABLE_OOB=on`

LOG_DIRECTORY_CLIENT

Purpose

Use the parameter `LOG_DIRECTORY_CLIENT` to specify the destination directory for the client log file.

Default

Current directory from which the executable is started

Example

`LOG_DIRECTORY_CLIENT=/oracle/network/log`

LOG_DIRECTORY_SERVER

Purpose

Use the parameter `LOG_DIRECTORY_SERVER` to specify the destination directory for the database server log file.

Default

Current directory from which the executable is started

Example

```
LOG_DIRECTORY_SERVER=/oracle/network/log
```

LOG_FILE_CLIENT

Purpose

Use the parameter `LOG_FILE_CLIENT` to specify the name of the log file for the client.

Default

```
sqlnet.log
```

Example

```
LOG_FILE_CLIENT=client
```

LOG_FILE_SERVER

Purpose

Use the parameter `LOG_FILE_SERVER` to specify the name of the log file for the database server.

Default

```
sqlnet.log
```

Example

```
LOG_FILE_SERVER=svr.log
```

NAMES.CONNECT_TIMEOUT

Purpose

Use the parameter `NAMES.CONNECT_TIMEOUT` to specify the amount of time, in seconds, for the client to wait for a connection to an **Oracle Names server** to complete.

Default

3

Minimum Value

1

Maximum Value

600

Example

```
NAMES.CONNECT_TIMEOUT=8
```

NAMES.DCE.PREFIX

Purpose

Use the parameter `NAMES.DCE.PREFIX` to specify the Distributed Computing Environment (DCE) cell name (prefix) to use for name lookups.

Default

```
././subsys/oracle/names
```

Example

```
NAMES.DCE.PREFIX=././subsys/oracle/names
```

NAMES.DEFAULT_DOMAIN

Purpose

Use the parameter `NAMES.DEFAULT_DOMAIN` 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.acme.com`, the connect string `CONNECT scott/tiger@sales` gets searched as `sales.us.acme.com`. If the connect string includes the domain extension, such as `CONNECT scott/tiger@sales.acme.com`, the domain is not appended.

Default

None

Example

```
NAMES.DEFAULT_DOMAIN=acme.com
```

NAMES.DIRECTORY_PATH

Purpose

Use the parameter `NAMES.DIRECTORY_PATH` to specify the order of the naming methods used for client name resolution lookups.

Default

```
NAMES.DIRECTORY_PATH=(tnsnames, onames, hostname)
```

Values

Naming Method Value	Description
tnsnames local naming naming method	Set to resolve a net service name through the <code>tnsnames.ora</code> file on the client. See Also: <i>Oracle9i Net Services Administrator's Guide</i>
ldap directory naming naming method	Set to resolve a database service name, net service name, or net service alias through a directory server . See Also: <i>Oracle9i Net Services Administrator's Guide</i>
onames Oracle Names method	Set to resolve database objects through a Oracle Names server. See Also: <i>Oracle9i Net Services Administrator's Guide</i>
hostname host naming method	Set to resolve a host name alias through an existing names resolution service or a centrally-maintained set of <code>/etc/hosts</code> files. See Also: <i>Oracle9i Net Services Administrator's Guide</i>
cds Cell Directory Services (CDS) external naming method	Set to resolve an Oracle database name in a Distributed Computing Environment (DCE) environment. See Also: <i>Oracle Advanced Security Administrator's Guide</i>
nis Network Information Service (NIS) external naming method	Set to resolve service information through an existing NIS. See Also: <i>Oracle9i Net Services Administrator's Guide</i>

Example

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

NAMES.INITIAL_RETRY_TIMEOUT

Purpose

Use the parameter `NAMES . INITIAL_RETRY_TIMEOUT` to determine how long a client waits for a response from an Oracle Names server before reiterating the request to the next Oracle Names server in the preferred servers list.

Default:

15

Minimum Value

1

Maximum Value

600

Example

```
NAMES . INITIAL_RETRY_TIMEOUT=20
```

NAMES.MAX_OPEN_CONNECTIONS

Purpose

Use the parameter `NAMES . MAX_OPEN_CONNECTIONS` to determine how many connections an Oracle Names client can have open at one time.

Default

10

Minimum Value

3

Maximum Value

64

Example

```
NAMES.MAX_OPEN_CONNECTIONS=3
```

NAMES.MESSAGE_POOL_START_SIZE

Purpose

Use the parameter `NAMES.MESSAGE_POOL_START_SIZE` to determine the initial number of messages allocated in the client's message pool for message requests.

Default

10

Minimum Value

3

Maximum Value

256

Example

```
NAMES.MESSAGE_POOL_START_SIZE=10
```

NAMES.NIS.META_MAP

Purpose

Use the parameter `NAMES.NIS.META_MAP` to specify the **map** file to be used to map **Network Information Service (NIS)** attributes to an NIS mapname

Default

`sqlnet.maps`

Example

```
NAMES.NIS.META_MAP=sqlnet.maps
```

NAMES.PREFERRED_SERVERS

Purpose

Use the parameter `NAMES.PREFERRED_SERVERS` to indicate the name, protocol addresses, and order of Oracle Names servers that are used for a client's name requests.

Default

None

Example

```
NAMES.PREFERRED_SERVERS=  
  (ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=icp)(KEY=n23))  
    (ADDRESS=(PROTOCOL=tcp)(HOST=nineva)(PORT=1575))  
    (ADDRESS=(PROTOCOL=tcp)(HOST=cicada)(PORT=1575)))
```

NAMES.REQUEST_RETRIES

Purpose

Use the parameter `NAMES.REQUEST_RETRIES` to specify the number of times the client should try each Oracle Names server in the list of preferred Oracle Names servers before allowing the operation to fail.

Default

1

Minimum Value

1

Maximum Value

5

Example

```
NAMES.REQUEST_RETRIES=5
```

NAMESCTL.ECHO

Purpose

If the parameter `NAMESCTL.ECHO` is set to `true`, then the Oracle Names Control utility echoes commands with prompts in the output. You can use this information to better interpret the output. You can use this information to better interpret the output, especially when the Oracle Names Control utility is run with a command script. The commands from the script appear before their output.

Default

`false`

Values

`true | false`

Example

```
NAMESCTL.ECHO=true
```

When the `QUERY . ns.smd` command is entered in the Oracle Names Control utility, the output that follows displays. Notice that `QUERY . ns.smd` command is echoed on the second line.

```
NAMESCTL> QUERY . ns.smd
NAMESCTL> QUERY . ns.smd
Total response time: 0.11 seconds
Response status:    normal, successful completion
Authoritative answer: yes
Number of answers: 3
TTL:                1 day
Answers:
  data type is "ns.smd"
    Syntax is DOMAIN: sales
  data type is "ns.smd"
    Syntax is DOMAIN: dev
  data type is "ns.smd"
    Syntax is DOMAIN: rootserv1.acme.com
```

NAMESTL.INTERNAL_ENCRYPT_PASSWORD

Purpose

If the parameter `NAMESTL.INTERNAL_ENCRYPT_PASSWORD` is set to `true`, then the Oracle Names Control utility encrypts the password when it is sent to the Oracle Names server.

If set to `false`, the Oracle Names Control utility does not encrypt the password. A `false` setting enables unencrypted passwords to be set in the `names.ora` file with the `NAMESTL.PASSWORD` parameter.

Default

`true`

Values

`true` | `false`

Example

```
NAMESTL.INTERNAL_ENCRYPT_PASSWORD=true
```

NAMESTL.INTERNAL_USE

Purpose

If the parameter `NAMESTL.INTERNAL_USE` is set to `true`, then the Oracle Names Control utility enables a set of internal undocumented commands. All internal commands are preceded by an underscore to distinguish them as internal.

Default

`false`

Values

`true` | `false`

Example

```
NAMESTL.INTERNAL_USE=true
```

NAMESCTL.NO_INITIAL_SERVER

Purpose

If the parameter `NAMESCTL.NO_INITIAL_SERVER` is set to `true`, then the Oracle Names Control utility suppresses any error messages when the client is unable to connect to a default Oracle Names server.

Default

`false`

Values

`true` | `false`

Example

```
NAMESCTL.NO_INITIAL_SERVER=true
```

NAMESCTL.NOCONFIRM

Purpose

Use the parameter `NAMESCTL.NOCONFIRM` to indicate whether sensitive commands, such as `STOP`, `RELOAD`, and `RESTART`, should be prompted with a confirmation when running the [Oracle Names Control utility](#).

Default

`off`

Values

`on` | `off`

Example

```
NAMESCTL.NOCONFIRM=on
```

NAMESTL.SERVER_PASSWORD

Purpose

Use the parameter `NAMESTL.SERVER_PASSWORD` to indicate the value that matches the configured password set in the `names.ora` file with the `NAMESTL.PASSWORD` parameter. This eliminates the need to enter the password with the `SET PASSWORD` command each time you use the Oracle Names Control utility to use secure commands, such as `STOP`, `RESTART`, and `RELOAD`.

Example

```
NAMESTL.SERVER_PASSWORD=secret
```

NAMESTL.TRACE_LEVEL

Purpose

Use the parameter `NAMESTL.TRACE_LEVEL` to turn Oracle Names Control utility tracing on, at a specific level, or 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

```
NAMESTL.TRACE_LEVEL=admin
```

NAMESCTL.TRACE_FILE

Purpose

Use the parameter `NAMESCTL.TRACE_FILE` to specify the file in which the Oracle Names Control utility trace output is placed.

Default

`namesctl_pid.trc`

Example

```
NAMESCTL.TRACE_FILE=nmsctl
```

NAMESCTL.TRACE_DIRECTORY

Purpose

Use the parameter `NAMESCTL.TRACE_DIRECTORY` to specify the directory where trace output from the Oracle Names Control utility is placed.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows NT

Example

```
NAMESCTL.TRACE_DIRECTORY=/oracle/trace
```

NAMESTL.TRACE_UNIQUE

Purpose

Use the parameter `NAMESTL.TRACE_UNIQUE` to indicate whether or not a process identifier is appended to the name of each trace file generated. If this parameter is enabled, several trace files can co-exist.

Default

on

Values

on | off

Example

```
NAMESTL.TRACE_UNIQUE=on
```

SQLNET.AUTHENTICATION_KERBEROS5_SERVICE

Purpose

Use the parameter `SQLNET.AUTHENTICATION_KERBEROS5_SERVICE` to define the name of the service used to obtain a Kerberos service ticket.

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

Default

None

Example

```
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE=oracle
```

SQLNET.AUTHENTICATION_GSSAPI_SERVICE

Purpose

Use the parameter `SQLNET.AUTHENTICATION_GSSAPI_SERVICE` to define the CyberSAFE service principal.

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

SQLNET.AUTHENTICATION_SERVICES

Purpose

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

Default

None

Values

Authentication Methods Available with Oracle Net Services:

- `none` for no authentication methods. A valid username and password can be used to access the database.
- `all` for all authentication methods
- `nts` for **Windows NT native authentication**

Authentication Methods Available with Oracle Advanced Security:

- `kerberos5` for Kerberos authentication
- `cybersafe` for Cybersafe authentication
- `radius` for RADIUS authentication
- `dcegssapi` for DCE GSSAPI authentication

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

Example

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

SQLNET.CLIENT_REGISTRATION

Purpose

Use the parameter `SQLNET.CLIENT_REGISTRATION` to set a unique identifier for this 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

Use the parameter `SQLNET.CRYPTO_CHECKSUM_CLIENT` to specify the checksum behavior for the client.

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

Default

rejected

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

Use the parameter `SQLNET.CRYPTO_CHECKSUM_SERVER` to specify the checksum behavior for the database server.

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

Default

`rejected`

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
```

SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT

Purpose

Use the parameter `SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT` to specify a list of crypto-checksum algorithms for the client to use.

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

Default

md5

Values

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

Example

```
SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT=(MD5)
```

SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER

Purpose

Use the parameter `SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER` to specify a list of crypto-checksum algorithms for the database server to use.

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

Default

md5

Values

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

Example

```
SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER=(md5)
```

SQLNET.CRYPTO_SEED

Purpose

Use the parameter `SQLNET.CRYPTO_SEED` to specify the characters used when generating cryptographic keys. The more random the characters are, the stronger the keys are. The string should be 10-70 random characters. This is required for when encryption or checksumming are turned on. Encryption is turned on if the [SQLNET.ENCRYPTION_CLIENT](#) parameter is specified for the client and the [SQLNET.ENCRYPTION_SERVER](#) parameter is specified for the database server; checksumming is turned on if the [SQLNET.CRYPTO_CHECKSUM_CLIENT](#) parameter is specified for the client and the [SQLNET.CRYPTO_CHECKSUM_SERVER](#) parameter is specified for the database server.

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

Default

```
qwertyuiopasdfghjkl;zxcvbnm,.s1
```

Example

```
SQLNET.CRYPTO_SEED="qwertyuiopasdfghjkl;zxcvbnm,.s1"
```

SQLNET.ENCRYPTION_CLIENT

Purpose

Use the parameter `SQLNET.ENCRYPTION_CLIENT` to turn encryption on for the client.

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

Default

```
rejected
```

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
```

SQLNET.ENCRYPTION_SERVER

Purpose

Use the parameter `SQLNET.ENCRYPTION_SERVER` to turn encryption on for the database server.

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

Default

```
rejected
```

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
```

SQLNET.ENCRYPTION_TYPES_CLIENT

Purpose

Use the parameter `SQLNET.ENCRYPTION_TYPES_CLIENT` to specify a list of encryption algorithms for the client to use.

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

Default

All algorithms are used if none are specified.

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)
```

SQLNET.ENCRYPTION_TYPES_SERVER

Purpose

Use the parameter `SQLNET.ENCRYPTION_TYPES_SERVER` to specify a list of encryption algorithms for the database server to use.

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

Default

All algorithms are used if none are specified.

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, ...)
```

SQLNET.EXPIRE_TIME

Purpose

Use parameter `SQLNET.EXPIRE_TIME` to specify a the time interval, in seconds, to send a probe to verify that client/server connections are active. 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, 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 downgrading network performance.

Default

0

Minimum Value

0

Recommended Value

10

Example

```
SQLNET.EXPIRE_TIME=10
```

SQLNET.KERBEROS5_CC_NAME

Purpose

Use the parameter `SQLNET.KERBEROS5_CC_NAME` to specify the complete path name to the Kerberos credentials cache file.

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

Default

`/usr/tmp/krbcache` on UNIX operating systems and `c:\tmp\krbcache` on Windows operating systems

Example

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

SQLNET.KERBEROS5_CLOCKSKEW

Purpose

Use the parameter `SQLNET.KERBEROS5_CLOCKSKEW` to specify how many seconds can pass before a Kerberos credential is considered out of date.

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

Default

300

Example

```
SQLNET.KERBEROS5_CLOCKSKEW=1200
```

SQLNET.KERBEROS5_CONF

Purpose

Use the parameter `SQLNET.KERBEROS5_CONF` 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.

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

Default

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

Example

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

SQLNET.KERBEROS5_KEYTAB

Purpose

Use the parameter `SQLNET.KERBEROS5_KEYTAB` 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.

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

Default

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

Example

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

SQLNET.KERBEROS5_REALMS

Purpose

Use the parameter `SQLNET.KERBEROS5_REALMS` 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.

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

Default

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

Example

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

SQLNET.RADIUS_ALTERNATE

Purpose

Use the parameter `SQLNET.RADIUS_ALTERNATE` 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.

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

Default

None

Example

```
SQLNET.RADIUS_ALTERNATE=radius2
```

SQLNET.RADIUS_ALTERNATE_PORT

Purpose

Use the parameter `SQLNET.RADIUS_ALTERNATE_PORT` to specify the listening port of the alternate RADIUS server.

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

Default

1645

Example

```
SQLNET.RADIUS_ALTERNATE_PORT=1667
```

SQLNET.RADIUS_ALTERNATE_RETRIES

Purpose

Use the parameter `SQLNET.RADIUS_ALTERNATE_RETRIES` to specify the number of times the database server should resend messages to the alternate RADIUS server.

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

Default

3

Example

```
SQLNET.RADIUS_ALTERNATE_RETRIES=4
```

SQLNET.RADIUS_AUTHENTICATION

Purpose

Use the parameter `SQLNET.RADIUS_AUTHENTICATION` to specify the location of the primary RADIUS server, either by its host name or IP address.

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

Default

Local host

Example

```
SQLNET.RADIUS_AUTHENTICATION=officeacct
```

SQLNET.RADIUS_AUTHENTICATION_INTERFACE

Purpose

Use the parameter `SQLNET.RADIUS_AUTHENTICATION_INTERFACE` to specify the class containing the user interface used to interact with the user.

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

Default

`DefaultRadiusInterface`

Example

`SQLNET.RADIUS_AUTHENTICATION_INTERFACE=DefaultRadiusInterface`

SQLNET.RADIUS_AUTHENTICATION_PORT

Purpose

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

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

Default

1645

Example

`SQLNET.RADIUS_AUTHENTICATION_PORT= 1667`

SQLNET.RADIUS_AUTHENTICATION_RETRIES

Purpose

Use the parameter `SQLNET.RADIUS_AUTHENTICATION_RETRIES` to specify the number of times the database server should resend messages to the primary RADIUS server.

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

Default

3

Example

```
SQLNET.RADIUS_AUTHENTICATION_RETRIES=4
```

SQLNET.RADIUS_AUTHENTICATION_TIMEOUT

Purpose

Use the parameter `SQLNET.RADIUS_AUTHENTICATION_TIMEOUT` to specify the time, in seconds, that the database server should wait for a response from the primary RADIUS server.

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

Default

5

Example

```
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT=10
```

SQLNET.RADIUS_CHALLENGE_RESPONSE

Purpose

Use the parameter `SQLNET.RADIUS_CHALLENGE_RESPONSE` to turn challenge response on or off.

Default

off

Values

on | off

Example

```
SQLNET.RADIUS_CHALLENGE_RESPONSE=on
```

SQLNET.RADIUS_SECRET

Purpose:

Use the parameter `SQLNET.RADIUS_SECRET` to specify the location of the RADIUS secret key.

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

Default

The `$ORACLE_HOME/network/security/radius.key` file on UNIX operating systems and the `ORACLE_HOME\network\security\radius.key` file on Windows NT

Example

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

SQLNET.RADIUS_SEND_ACCOUNTING

Purpose

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

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

Default

off

Values

on | off

Example

```
SQLNET.RADIUS_SEND_ACCOUNTING=on
```

SSL_CIPHER_SUITES

Purpose

Use the parameter `SSL_CIPHER_SUITES` to control what combination of encryption and data integrity is used by the [Secure Sockets Layer \(SSL\)](#).

Default

None

Values

See Also: *Oracle Advanced Security Administrator's Guide* for further information about cipher suite values

Example

```
SSL_CIPHER_SUITE=(ssl_rsa_with_rc4_138_md5)
```

SSL_CLIENT_AUTHENTICATION

Purpose

Use the parameter `SSL_CLIENT_AUTHENTICATION` to specify whether or not a client—in addition to the database server—is authenticated using SSL.

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

Default

true

Values

true | false

Example

```
SSL_CLIENT_AUTHENTICATION=true
```

SSL_SERVER_DN_MATCH

Purpose

Use the parameter `SSL_SERVER_DN_MATCH` to enforce that the database server's **distinguished name (DN)** 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.

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

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 to not enforce a match. If 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.

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

Example

```
SSL_SERVER_DN_MATCH=yes
```

SSL_VERSION

Purpose

Use the parameter `SSL_VERSION` to force the version of the SSL connection. Clients and database servers must use a compatible version.

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

Default

```
undetermined
```

Values

```
undetermined | 2.0 | 3.0
```

Example

```
SSL_VERSION=2.0
```

TCP.EXCLUDED_NODES

Purpose

Use the parameter `TCP.EXCLUDED_NODES` to specify which clients are denied access to the database.

Syntax

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

Example

```
TCP.EXCLUDED_NODES=(finance.us.acme.com, mktg.us.acme.com, 144.25.5.25)
```

TCP.INVITED_NODES

Purpose

Use the parameter `TCP.INVITED_NODES` 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, ...)
```

Example

```
TCP.INVITED_NODES=(sales.us.acme.com, hr.us.acme.com, 144.185.5.73)
```

TCP.VALIDNODE_CHECKING

Purpose

Use the parameter `TCP.VALIDNODE_CHECKING` to check for the `TCP.INVITED_NODES` and `TCP.EXCLUDED_NODES` to determine which clients to allow or deny access.

Default

no

Values

yes | no

Example

```
TCP.VALIDNODE_CHECKING=yes
```

TCP.NODELAY

Purpose

Use the parameter `TCP.NODELAY` to preempt delays in buffer flushing within the TCP/IP protocol stack.

Default

no

Values

yes | no

Example

```
TCP.NODELAY=yes
```

TNSPING.TRACE_DIRECTORY

Purpose

Use the parameter `TNSPING.TRACE_DIRECTORY` to specify the destination directory for the TNSPING utility trace file.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows operating systems

Example

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

TNSPING.TRACE_LEVEL

Purpose

Use the parameter `TNSPING.TRACE_LEVEL` to turn TNSPING utility tracing on, at a specific level, or 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
```

TRACE_DIRECTORY_CLIENT

Purpose

Use the parameter `TRACE_DIRECTORY_CLIENT` to specify the destination directory for the client trace file.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows operating systems

Example

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

TRACE_DIRECTORY_SERVER

Purpose

Use the parameter `TRACE_DIRECTORY_SERVER` to specify the destination directory for the database server trace file.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows NT

Example

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

TRACE_FILE_CLIENT

Purpose

Use the parameter `TRACE_FILE_CLIENT` to specify the name of the client trace file.

Default

`sqlnet.trc`

Example

```
TRACE_FILE_CLIENT=clientsqlnet.trc
```

TRACE_FILE_SERVER

Purpose

Use the parameter `TRACE_FILE_SERVER` to specify the name of the database server trace file

Default

`svr_pid.trc`

Example

```
TRACE_FILE_SERVER=svrsqlnet.trc
```

TRACE_FILELEN_CLIENT

Purpose

Use the parameter `TRACE_FILELEN_CLIENT` 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.

Example

```
TRACE_FILELEN_CLIENT=100
```

TRACE_FILELEN_SERVER

Purpose

Use the parameter `TRACE_FILELEN_SERVER` 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.

Example

```
TRACE_FILELEN_SERVER=100
```

TRACE_FILENO_CLIENT

Purpose

Use the parameter `TRACE_FILENO_CLIENT` to specify the number of trace files for client tracing. When this parameter is set along 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, 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.

Default

None

Example

```
TRACE_FILENO_CLIENT=3
```

TRACE_FILENO_SERVER

Purpose

Use the parameter `TRACE_FILENO_SERVER` to specify the number of trace files for database server tracing. When this parameter is set along 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, 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.

Default

None

Example`TRACE_FILENO_SERVER=3`**TRACE_LEVEL_CLIENT****Purpose**

Use the parameter `TRACE_LEVEL_CLIENT` to turn client tracing on, at a specific level, or 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`TRACE_LEVEL_CLIENT=user`

TRACE_LEVEL_SERVER

Purpose

Use the parameter `TRACE_LEVEL_SERVER` to turn server tracing on, at a specific level, or 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

```
TRACE_LEVEL_SERVER=admin
```

TRACE_TIMESTAMP_CLIENT

Purpose

Use the parameter `TRACE_TIMESTAMP_CLIENT` to add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mil` to every trace event in the client trace file, which has a default name of `sqlnet.trc`.

Default

`on`

Values

`on or true` | `off or false`

Example

```
TRACE_TIMESTAMP_SERVER=true
```

TRACE_TIMESTAMP_SERVER

Purpose

Use the parameter `TRACE_TIMESTAMP_SERVER` to add a time stamp in form of *dd-mon-yyyy hh:mi:ss:mil* to every trace event in the database server trace file, which has a default name of `svr_pid.trc`.

Default

`off`

Values

`on or true | off or false`

Example

```
TRACE_TIMESTAMP_SERVER=true
```

TRACE_UNIQUE_CLIENT

Purpose

Use the parameter `TRACE_UNIQUE_CLIENT` to determine whether or not a unique trace file is created for each client trace file. If the value is set to `on`, a process identifier is appended to the name of each trace file, enabling several files named `sqlnetpid.trc` to coexist. If the value is set to `off`, when a new trace file is created for a client, it overwrites the existing file.

Default

`on`

Values

`on or off`

Example

```
TRACE_UNIQUE_CLIENT=on
```

USE_CMAN

Purpose

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

The following example shows two address lists. While the first address list routes the client to an Oracle Connection Manager, the second address list routes the client directly to a listener.

```
sales=
  (DESCRIPTION=
    (LOAD_BALANCE=on)
    (FAILOVER=on)
    (ADDRESS_LIST=
      (SOURCE_ROUTE=yes)
      (ADDRESS=(PROTOCOL=tcp)(HOST=host1)(PORT=1630))
      (ADDRESS=(PROTOCOL=tcp)(HOST=host2)(PORT=1521)))
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=host3)(PORT=1521)))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.acme.com))
```

Without `USE_CMAN=true`, 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, connections are routed through any available listener address.

Note: If you are using Oracle Connection Manager with Oracle Names, this option must be set on clients and Oracle Names servers.

Default

`false`

Values

true | false

Example

```
USE_CMAN=true
```

USE_DEDICATED_SERVER

Purpose

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

This parameter adds `(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.

See Also: *Oracle9i Net Services Administrator's Guide* for complete configuration information

Default

off

Values

- `on` to append `(SERVER=dedicated)` to a connect descriptor's connect data.
- `off` to hand off requests to existing server processes

Example

```
USE_DEDICATED_SERVER=on
```

WALLET_LOCATION

Purpose

Use the parameter `WALLET_LOCATION` to specify the location of wallets. Wallets are certificates, keys, and trustpoints processed by SSL.

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

Syntax

Oracle wallets on file system:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=file)  
    (METHOD_DATA=  
      (DIRECTORY=directory)))
```

Microsoft certificate store:

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

Oracle wallets in the Windows NT 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)))

```

Subparameters

WALLET_LOCATION supports the following subparameters:

SOURCE: Specify the type of storage for wallets and storage location.

METHOD: Specify the type of storage.

METHOD_DATA: Specify the storage location.

DIRECTORY: Specify the location of Oracle wallets on file system.

KEY: Specify the wallet type and location in the Windows NT registry.

PROFILE: Specify the Entrust profile file (.epf).

INIFILE: Specify the Entrust initialization file (.ini).

Default

None

Usage Notes

- The key/value pair for Microsoft's 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 Windows NT registry and the wallet's key (KEY) is SALESAPP, 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 Windows NT 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)))
```

Local Naming Parameters (tnsnames.ora)

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

This chapter contains these 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](#)
- [Local Naming 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, `tnsnames.ora` is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and in the `ORACLE_HOME\network\admin` directory on Windows operating systems. `tnsnames.ora` can also be stored the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value
- On UNIX operating systems, the global configuration directory. For example, on Sun SPARC Solaris, this directory is `/var/opt/oracle`.

See Also: Oracle operating system-specific documentation

General Syntax of `tnsnames.ora`

The basic syntax for a `tnsnames.ora` file is shown in [Figure 7-1](#). `DESCRIPTION` contains the connect descriptor, `ADDRESS` contains the protocol address, and `CONNECT_DATA` contains the database service identification information.

Figure 7-1 Basic Format of `tnsnames.ora` File

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

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.

[Figure 7-2](#) shows two connect descriptors with multiple addresses. `DESCRIPTION_LIST` defines a list of connect descriptors.

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

Figure 7-2 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))))
```

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 [Figure 7-3](#), 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 either the first or second protocol address at random, then tries protocol addresses three and four sequentially.

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

Figure 7-3 *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: Protocol address lists do not have to be embedded in an `ADDRESS_LIST` if there is only one list, as was the case prior to release 8.1.

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.

Figure 7-4 illustrates failover of multiple Oracle Connection Manager protocol addresses.

Figure 7-4 Multiple Oracle Connection Manager Addresses in `tnsnames.ora`

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

In Figure 7-4:

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 then 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 then connects to the database service using the following protocol address:

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

Figure 7-5 illustrates client load balancing among two Oracle Connection Managers and two protocol addresses:

Figure 7-5 Client Load Balancing in tnsnames.ora

```
sample2=
  (DESCRIPTION=
    (LOAD_BALANCE=on)
    (FAILOVER=on)
    (ADDRESS_LIST=
      (SOURCE_ROUTE=yes)
      (ADDRESS=(PROTOCOL=tcp)(HOST=host1)(PORT=1630))
      (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.acme.com))
```

In **Figure 7-5**:

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.

Local Naming Parameters

This section lists and describes the `tnsnames.ora` file parameters that comprise connect descriptors. Configuration parameters fall into the following categories:

- [Connect Descriptor Descriptions](#)
- [Protocol Address Section](#)
- [Optional Parameters for Lists](#)
- [Connect Data Section](#)
- [Security Section](#)

Connect Descriptor Descriptions

Each connect descriptor is contained within a `DESCRIPTION` parameter. Multiple connect descriptors are characterized by the `DESCRIPTION_LIST` parameter. These parameters are described next.

DESCRIPTION

Purpose

Use the `DESCRIPTION` parameter as a container for a connect descriptor.

Embed this parameter under the `DESCRIPTION_LIST` parameter.

Example

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

DESCRIPTION_LIST

Purpose

Use the `DESCRIPTION_LIST` parameter defines a list of connect descriptors for a particular net service name.

Example

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

Protocol Address Section

The protocol address section of the `tnsnames.ora` file specifies the protocol addresses of the listener.

ADDRESS

Purpose

Use the parameter `ADDRESS` to define a single listener protocol address.

Embed this parameter under either the `ADDRESS_LIST` parameter or the `DESCRIPTION` parameter.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for descriptions of the correct parameters to use for each protocol

Example

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

ADDRESS_LIST

Purpose

Use the parameter `ADDRESS_LIST` to define a list of protocol addresses. If there is only address list, `ADDRESS_LIST` is not necessary.

Embed 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)))  
  (ADDRESS_LIST=  
    (FAILOVER=on)  
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))  
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521))  
    (CONNECT_DATA=(SERVICE_NAME=sales.us.acme.com)))
```

Optional Parameters for Lists

For multiple addresses, the following parameters are available for usage:

- `FAILOVER`
- `LOAD_BALANCE`
- `SDU`
- `SOURCE_ROUTE`
- `TYPE_OF_SERVICE`

FAILOVER

Purpose

Use the parameter `FAILOVER` 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.

Embed this parameter under either 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_LISTS`, `DESCRIPTIONS`, and `ADDRESS_LISTS`

Values

`on` | `off` | `yes` | `no` | `true` | `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.acme.com))
```

LOAD_BALANCE

Purpose

Use the parameter `FAILOVER` to enable or disable client load balancing for multiple protocol addresses.

When you set the parameter to `on`, `yes`, or `true`, Oracle Net progresses through 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 protocol addresses sequentially until one succeeds.

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

Default

`on` for `DESCRIPTION_LISTS`

Values

`on` | `off` | `yes` | `no` | `true` | `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.acme.com))
```

SDU

Purpose

Use the parameter `SDU` to instruct Oracle Net to optimize the transfer rate of data packets being sent across the network with the **session data unit (SDU)** size you specify.

Embed this parameter under the `DESCRIPTION` parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for complete configuration information

Default

2 KB

Usage

The SDU size can range from 512 bytes to 32 KB. The default SDU for the client and the database is 2 KB.

Optimal SDU size depends on the maximum segment size (MSS) and message fragmentation. For **Two-Task Common (TTC)** connections, configuring an SDU size larger than the 2 KB default requires configuring the SDU on both the client and server computers. When the configured values do not match, the lower of the two values will be used.

To minimize packet header overhead and message fragmentation, set the SDU size as a multiple of the MSS. When **Oracle Advanced Security** encryption is not used, increase the SDU size by one (1). For example, the TCP/IP version 4 MSS on Ethernet is 1460 bytes. Use a multiple of 1460 for the SDU size if encryption is used. If encryption is not used, increase the SDU size to 1461.

Example

```
net_service_name=
(DESCRIPTION=
  (SDU=2920)
  (ADDRESS=...)
  (ADDRESS=...)
  (CONNECT_DATA=
    (SERVER_NAME=sales.us.acme.com))
```

SOURCE_ROUTE

Purpose

Use the parameter `SOURCE_ROUTE` 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.

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

See Also: *Oracle9i Net Services Administrator's Guide* for complete configuration information

Default

`off`

Values

`yes` | `no` | `on` | `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.acme.com))
```

TYPE_OF_SERVICE

Purpose

Use the parameter `TYPE_OF_SERVICE` parameter 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.

Embed 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.acme.com)  
      (TYPE_OF_SERVICE=oracle9_database)) ) )
```

Connect Data Section

The connection data section of the `tnsnames.ora` file specifies the name of the destination service.

CONNECT_DATA

Purpose

Use the parameter `CONNECT_DATA` to define the service to which to connect.

Embed this parameter under the `DESCRIPTION` parameter.

Usage Notes

`CONNECT_DATA` permits the following subparameters:

- `FAILOVER_MODE`
- `GLOBAL_NAME`
- `HS`
- `INSTANCE_NAME`
- `RDB_DATABASE`
- `SERVER`
- `SERVICE_NAME`
- `SID`

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.acme.com)))
```

FAILOVER_MODE

Purpose

Use the parameter `FAILOVER_MODE` to instruct Oracle Net to fail over to a different listener if the first listener fails during runtime. 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.

Embed this parameter under the `CONNECT_DATA` parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for complete configuration information

Subparameters

`FAILOVER_MODE` supports the following subparameters:

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; that is, if a user's connection is lost, 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 on 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, **RETRIES** defaults to five retry attempts.

DELAY: Specify the amount of time in seconds to wait between connect attempts. If **RETRIES** is specified, **DELAY** defaults to one second.

Example

See Also: *Oracle9i Net Services Administrator's Guide* for implementation examples

GLOBAL_NAME

Purpose

Use the parameter **GLOBAL_NAME** to identify the Oracle Rdb database.

Embed 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

Use the parameter HS to instruct Oracle Net to connect to a non-Oracle system through **Heterogeneous Services**.

Embed this parameter under the CONNECT_DATA parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for complete configuration information

Default

None

Values

ok

Example

```
net_service_name=  
(DESCRIPTION=  
  (ADDRESS=...)   
  (ADDRESS=...)   
  (CONNECT_DATA=  
    (SID=sales6)  
    (HS=ok) ) )
```

INSTANCE_NAME

Purpose

Use the parameter INSTANCE_NAME to identify the database instance to access. Set the value to the value specified by the INSTANCE_NAME parameter in the **initialization parameter file**.

Embed this parameter under the CONNECT_DATA parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for information about the use of INSTANCE_NAME

Example

```
net_service_name=  
(DESCRIPTION=  
  (ADDRESS=...)  
  (ADDRESS=...)  
  (CONNECT_DATA=  
    (SERVICE_NAME=sales.us.acme.com)  
    (INSTANCE_NAME=sales1)))
```

RDB_DATABASE

Purpose

Use the parameter RDB_DATABASE parameter to specify the file name of an Oracle Rdb database.

Embed this parameter under the CONNECT_DATA parameter.

Example

```
net_service_name=  
(DESCRIPTION=  
  (ADDRESS=...)  
  (ADDRESS=...)  
  (CONNECT_DATA=  
    (SERVICE_NAME=sales.us.acme.com)  
    (RDB_DATABASE= [.mf]mf_personal.rdb)))
```

SERVER

Purpose

Use the parameter `SERVER` to instruct the listener to connect the client to a specific type of **service handler**.

Embed this parameter under the `CONNECT_DATA` parameter.

Values

- `dedicated` to specify that client requests be served by **dedicated server**
- `shared` to specify that client request be served by **shared server**

Note: 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. See the *Oracle9i Net Services Administrator's Guide* for configuration information.

Note: 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.acme.com)  
    (SERVER=dedicated)))
```

SERVICE_NAME

Purpose

Use the parameter `SERVICE_NAME` to identify the Oracle9i or Oracle8i database service to access. Set the value to a value specified by the `SERVICE_NAMES` parameter in the initialization parameter file.

Embed this parameter under the `CONNECT_DATA` parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for information about the use of the `SERVICE_NAME` parameter

Example

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

SID

Purpose

Use the parameter `SID` to identify the Oracle8 database instance by its **Oracle System Identifier (SID)**. If the database is Oracle9i or Oracle8i, use the `SERVICE_NAME` parameter rather than the `SID` parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for information about the use of `SID`

Embed this parameter under the `CONNECT_DATA` parameter.

Example

```
net_service_name=  
(DESCRIPTION=  
  (ADDRESS=...)  
  (ADDRESS=...)  
  (CONNECT_DATA=  
    (SID=sales)))
```

Security Section

The security section of the `tnsnames.ora` file specifies security-related parameters for use with Oracle Advanced Security features.

SECURITY

Purpose

Use the parameter `SECURITY` to enable secure connections.

Embed this parameter under the `DESCRIPTION` parameter.

Usage Notes

`SECURITY` permits the [SSL_SERVER_CERT_DN](#) subparameter.

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.acme.com))
  (SECURITY=
    (SSL_SERVER_CERT_DN="cn=sales,cn=OracleContext,dc=us,dc=acme,dc=com")))
```

SSL_SERVER_CERT_DN

Purpose

Use the parameter `SSL_SERVER_CERT_DN` 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 server's DN to match its service name.

Usage Notes

Use this parameter in conjunction with the `sqlnet.ora` parameter `SSL_SERVER_DN_MATCH` to enable server DN matching.

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

Example

```
net_service_name=
(DESCRIPTION=
  (ADDRESS=...)
  (ADDRESS=...)
  (CONNECT_DATA=
    (SERVICE_NAME=finance.us.acme.com))
  (SECURITY=
    (SSL_SERVER_CERT_DN="cn=finance,cn=OracleContext,dc=us,dc=acme,dc=com")))
```

Listener Parameters (listener.ora)

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

This chapter contains these topics:

- [Overview of Listener Configuration File](#)
- [Listener Parameters](#)

Overview of Listener Configuration File

Listener configuration, stored in the `listener.ora` file, consists of the following elements:

- Listener's name
- Protocol addresses that it is accepting connection requests on
- Services it is listening for

Dynamic **service registration**, a feature of Oracle9i and Oracle8i, eliminates the need for static configuration of supported services. However, static service configuration is required if you plan to use Oracle Enterprise Manager.

- Control parameters

By default, the `listener.ora` file is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and the `ORACLE_HOME\network\admin` directory on Windows NT. `listener.ora` can also be stored the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value
- On UNIX operating systems, the global configuration directory. For example, on Sun SPARC Solaris, 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 configuration is 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 Corporation recommends running only one listener for each node in most customer environments.

Figure 8–1 shows a `listener.ora` file for a listener named `LISTENER`, which is the default name of the listener.

Figure 8–1 Example 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.acme.com)
      (ORACLE_HOME=/oracle9i)
      (SID_NAME=sales))
    (SID_DESC=
      (SID_NAME=plsextproc)
      (ORACLE_HOME=/oracle9i)
      (PROGRAM=extproc)))
```

Listener Parameters

This section lists and describes the `listener.ora` file parameters. Listener configuration parameters fall into the following categories:

- [Protocol Address Section](#)
- [Static Service Registration \(SID_LIST\) Section](#)
- [Control Parameters](#)

Protocol Address Section

The **protocol address** section of the `listener.ora` file defines the protocol addresses that the listener is accepting connection requests on. Discussed next are the most common parameters used in protocol addresses. Note that `ADDRESS_LIST` is also supported.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for information about the `ADDRESS_LIST` parameter

DESCRIPTION

Purpose

Use the parameter `DESCRIPTION` as a container for listener protocol addresses.

Example

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

ADDRESS

Purpose

Use the parameter `ADDRESS` to specify a single listener protocol address.

Embed this parameter under the `DESCRIPTION` parameter.

See Also: [Chapter 5, "Protocol Address Configuration"](#) for descriptions of the correct parameters to use for each type of support protocol

Example

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

QUEUESIZE

Purpose

Use the parameter `QUEUESIZE` to specify the number of concurrent connection requests that the listener can accept on a TCP/IP listening endpoint (protocol address).

Embed 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. Following are the defaults for Sun SPARC Solaris and Windows NT:

- Sun SPARC Solaris: 5
- Windows NT 4.0 Workstation: 5
- Windows NT 4.0 Server: 50

Usage Notes

Specify this parameter if you expect the listener to handle more than the default number of concurrent connection requests.

Example

```
listener_name=  
(DESCRIPTION=  
  (ADDRESS= (PROTOCOL=tcp) (HOST=hr-server) (PORT=1521) (QUEUESIZE=20)))
```

Static Service Registration (SID_LIST) Section

You can use the `SID_LIST` section of the `listener.ora` to statically configure service information for the listener.

The `SID_LIST` is required for backward compatibility with Oracle8 databases, as well as **external procedure** calls and **Heterogeneous Services**, and some management tools, including Oracle Enterprise Manager.

Oracle9i or Oracle8i database information is dynamically registered with the listener during instance startup. Therefore, the `SID_LIST` is not required, unless Oracle Enterprise Manager is used to monitor an Oracle9i or Oracle8i database.

```
SID_LIST_listener_name=  
  (SID_LIST=  
    (SID_DESC=  
      (GLOBAL_DBNAME=global_database_name)  
      (SID_NAME=sid)  
      (ORACLE_HOME=oracle_home))  
    (SID_DESC=...))
```

SID_LIST

Purpose

Use the parameter `SID_LIST` to identify a list of SID descriptions.

Example

```
SID_LIST_listener_name=  
  (SID_LIST=  
    (SID_DESC=...)  
    (SID_DESC=...))
```

SID_DESC

Purpose

Use the parameter `SID_DESC` to specify service information for a specific database instance or a non-database service.

Embed this parameter under the `SID_LIST` parameter.

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
  (SID_DESC=...)  
  (SID_DESC=...))
```

`SID_DESC` permits the following parameters:

- `ENVS`
- `GLOBAL_DBNAME`
- `ORACLE_HOME`
- `PROGRAM`
- `SID_NAME`

ENVS

Purpose

Use the parameter `ENVS` to specify environment variables for the listener to set prior to executing (as a child process) a dedicated server program or an executable specified with the `PROGRAM` parameter.

Embed this parameter under the `SID_DESC` parameter.

Note: This parameter is not supported on Windows NT. Any process spawned by the listener will simply inherit the listener's environment.

Usage Notes

Enclose an environment variable and its value within double quotes ("):

```
(ENVS="variable=value")
```

A list of environment variables and their values can be specified. Enclose the list within double quotes, from end to end, and separate environment variable definitions with commas and no space.

```
(ENVS="variable=value,variable=value")
```

Note: Single quotes (') are supported for backward compatibility.

The use of the following characters within the environment variable or its value definition (ENVS="variable=value") are not supported:

- Comma (,)
- Single quotes (')
- Double quotes (")
- Equal sign (=)

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
  (SID_DESC=  
    (SID_NAME=plsextproc)  
    (ORACLE_HOME=/oracle9i)  
    (PROGRAM=extproc)  
    (ENVS="LD_LIBRARY_PATH=/oracle/9i/lib:/oracle/9i/ctx/lib"))  
  (SID_DESC=  
    (SID_NAME=test)  
    (PROGRAM=/tmp/myexec)  
    (ENVS="LD_LIBRARY_  
PATH=/private/xpm/lib:/private/mylibs,MYPATH=/usr/ucb:/usr/local/packages,APL_  
ENV_FILE=/apl/conf/env.txt")))
```

GLOBAL_DBNAME

Purpose

Use the parameter GLOBAL_DBNAME to identify the **global database name** of the database, a name comprised of the database name and database domain. You can obtain the GLOBAL_DBNAME value from the SERVICE_NAMES parameter in the initialization parameter file.

Embed this parameter under the SID_DESC parameter, and ensure it matches the value of the SERVICE_NAMES parameter in the initialization parameter file.

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
(SID_DESC=  
  (GLOBAL_DBNAME=sales.us.acme.com)  
  (SID_NAME=sales)  
  (ORACLE_HOME=/usr/oracle)))
```

ORACLE_HOME

Purpose

Use the parameter ORACLE_HOME to identify the Oracle home location of the service.

Embed this parameter under the SID_DESC parameter.

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
(SID_DESC=  
  (SID_NAME=extproc)  
  (ORACLE_HOME=/usr/oracle)  
  (PROGRAM=extproc)))
```

PROGRAM

Purpose

Use the parameter `PROGRAM` to identify the service's executable program name.

Embed this parameter under the `SID_DESC` parameter.

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
(SID_DESC=  
(SID_NAME=extproc)  
(ORACLE_HOME=/usr/oracle)  
(PROGRAM=extproc)))
```

SID_NAME

Purpose

Use the parameter `SID_NAME` to identify the **Oracle System Identifier (SID)** of the instance. You can obtain the SID value from the `INSTANCE_NAME` parameter in the initialization parameter file.

Embed this parameter under the `SID_DESC` parameter.

Example

```
SID_LIST_listener_name=  
(SID_LIST=  
(SID_DESC=  
(GLOBAL_DBNAME=sales.us.acme.com)  
(SID_NAME=sales)  
(ORACLE_HOME=/usr/oracle)))
```

Control Parameters

The sections describes the parameters that control the behavior of the listener.

ADMIN_RESTRICTIONS_*listener_name*

Purpose

Use the parameter `ADMIN_RESTRICTIONS_listener_name` to restrict runtime administration of the listener. The parameter is useful if the listener is not password-protected.

Setting `ADMIN_RESTRICTIONS_listener_name=on` disables the runtime modification of parameters in `listener.ora`. That is, the listener will refuse 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 (with the [RELOAD](#) command) for the new changes to take effect without explicitly stopping and restarting the listener.

Oracle Corporation recommends establishing a password to secure the listener. To establish an encrypted password, use either the Listener Control utility [CHANGE_PASSWORD](#) command or Oracle Net Manager.

See Also: *Oracle9i Net Services Administrator's Guide* for further information about password security of the listener

Default

off

Example

```
ADMIN_RESTRICTIONS_listener_name=on
```

LOG_DIRECTORY_*listener_name*

Purpose

Use the parameter `LOG_DIRECTORY_listener_name` to specify the destination directory of the listener log file.

Default

The `$ORACLE_HOME/network/log` directory on UNIX operating systems and the `ORACLE_HOME\network\log` directory on Windows NT

Example

```
LOG_DIRECTORY_listener=/oracle/network/admin/log
```

LOG_FILE_*listener_name*

Purpose

Use the parameter `LOG_FILE_listener_name` to specify the name of the log file for the listener.

Default

```
listener.log
```

Example

```
LOG_FILE_listener=list.log
```

LOGGING_*listener_name*

Purpose

Use the parameter LOGGING_*listener_name* to turn logging on or off.

Default

on

Values

on | off

Example

```
LOGGING_listener=on
```

PASSWORDS_*listener_name*

Purpose

Use the parameter PASSWORDS_*listener_name* to store an encrypted password for a listener, so that certain privileges operations, such as [SAVE_CONFIG](#) and [STOP](#), used from the Listener Control utility are secure. An encrypted password can be set using either the Listener Control utility [CHANGE_PASSWORD](#) command or Oracle Net Manager.

See Also: *Oracle9i Net Services Administrator's Guide* for further information about password security of the listener

Example

```
PASSWORDS_LISTENER=(2D6C48144CF753AC)
```

SAVE_CONFIG_ON_STOP_listener_name

Purpose

Use the parameter `SAVE_CONFIG_ON_STOP_listener_name` to specify whether or not runtime 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 runtime configuration changes to the `listener.ora` file.

Default

`false`

Values

`true` | `false`

Example

```
SAVE_CONFIG_ON_STOP_listener=true
```

SSL_CLIENT_AUTHENTICATION

Purpose

Use the parameter `SSL_CLIENT_AUTHENTICATION` to specify whether or not 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`, the listener attempts to authenticate the client, which can result in a failure.

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

Example

```
SSL_CLIENT_AUTHENTICATION=true
```

STARTUP_WAIT_TIME_listener_name

Note: This parameter is deprecated in Oracle9i and will be desupported in a future release. If you require this parameter to run the listener, please notify Oracle Support Services.

Purpose

Use the parameter `STARTUP_WAIT_TIME_listener_name` to set the number of seconds that the listener waits before responding to a Listener Control utility [START](#) command.

Default

0

Example

```
STARTUP_WAIT_TIME_listener=5
```

TRACE_DIRECTORY_listener_name

Purpose

Use the parameter `TRACE_DIRECTORY_listener_name` to specify the destination directory of the listener trace files.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows NT

Example

```
TRACE_DIRECTORY_listener=/oracle/network/admin/trace
```

TRACE_FILE_listener_name

Purpose

Use the parameter `TRACE_FILE_listener_name` to specify the name of the trace file for the listener.

Default

```
listener.trc
```

Example

```
TRACE_FILE_listener=list.trc
```

TRACE_FILELEN_listener_name

Purpose

Use the parameter `TRACE_FILELEN_listener_name` 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 with the [TRACE_FILENO_listener_name](#) parameter.

Default

Unlimited

Example

```
TRACE_FILELEN_listener=100
```

TRACE_FILENO_*listener_name*

Purpose

Use the parameter `TRACE_FILENO_listener_name` 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, 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.

Default

1

Example

```
TRACE_FILENO_listener=3
```

TRACE_LEVEL_*listener_name*

Purpose

Use the parameter `TRACE_LEVEL_listener_name` to turn listener tracing on, at a specific level, or 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

```
TRACE_LEVEL_listener=admin
```

TRACE_TIMESTAMP_listener_name

Purpose

Use the parameter `TRACE_TIMESTAMP_listener_name` to add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mi` to every trace event in the listener trace file.

Default

on

Values

on or true | off or false

Example

```
TRACE_TIMESTAMP_listener=true
```

WALLET_LOCATION

Purpose

Use the parameter `WALLET_LOCATION` to specify the location of wallets. Wallets are certificates, keys, and trustpoints processed by SSL that allow for secure connections.

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

Syntax

Oracle wallets on file system:

```
WALLET_LOCATION=  
  (SOURCE=  
    (METHOD=file)  
    (METHOD_DATA=  
      (DIRECTORY=directory)))
```

Microsoft certificate store:

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

Oracle wallets in the Windows NT 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)))
```

Subparameters

WALLET_LOCATION supports the following subparameters:

SOURCE: Specify the type of storage for wallets and storage location.

METHOD: Specify the type of storage.

METHOD_DATA: Specify the storage location.

DIRECTORY: Specify the location of Oracle wallets on file system.

KEY: Specify the wallet type and location in the Windows NT registry.

PROFILE: Specify the Entrust profile file (.epf).

INIFILE: Specify the Entrust initialization file (.ini).

Default

None

Usage Notes

- The key/value pair for Microsoft's 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 Windows NT registry and the wallet's key (KEY) is SALESAPP, 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 Windows NT 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)))
```

Oracle Connection Manager Parameters (cman.ora)

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

This chapter contains these topics:

- [Overview of Oracle Connection Manager Configuration File](#)
- [Oracle Connection Manager Parameters](#)

Overview of Oracle Connection Manager Configuration File

Oracle Connection Manager configuration, stored in the `cman.ora` file, consists of the following elements:

- Listening **protocol addresses** of the **CMGW process** (gateway process) and the **CMADMIN process** (administrative process)
- Access control parameters
- Control parameters

By default, the `cman.ora` file is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and in the `ORACLE_HOME\network\admin` directory on Windows NT. `cman.ora` can also be stored the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value
- On UNIX operating systems, the global configuration directory. For example, on Sun SPARC Solaris, this directory is `/var/opt/oracle`.

See Also: Oracle operating system-specific documentation

[Figure 9–1](#) shows an example `cman.ora` file.

Figure 9–1 Example `cman.ora` File

```
CMAN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1630))
  (ADDRESS=(PROTOCOL=tcps)(HOST=144.25.22.217)(PORT=2484))
CMAN_ADMIN=
  (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=1830))
CMAN_RULES=
  (RULE=(SRC=206.62.226.32/27)(DST=sales-server)(SRV=*)(ACT=accept))
CMAN_PROFILE=
  (PARAMETER_LIST=
    (LOG_LEVEL=2)
    (TRACING=on))
```

Oracle Connection Manager Parameters

This section lists and describes the `cmn.ora` file parameters. Configuration parameters fall into the following categories:

- [Listening Endpoint List](#)
- [Route List](#)
- [Parameter List](#)

Listening Endpoint List

CMAN

Purpose

Use the parameter `CMAN` to specify the protocol addresses of the CMGW process.

Default

```
CMAN=(ADDRESS=(PROTOCOL=tcp)(HOST=local_host)(PORT=1630))
```

Syntax

```
CMAN=  
  ([ADDRESS_LIST=  
    (ADDRESS= ...)  
    [(ADDRESS= ...)])
```

CMAN_ADMIN

Purpose

Use the parameter `CMAN_ADMIN` to specify the protocol addresses of the Oracle Connection Manager `CMADMIN` process.

Default

```
CMAN_ADMIN=(ADDRESS=(PROTOCOL=tcp)(HOST=anyhost)(PORT=1830))
```

Syntax

```
CMAN_ADMIN=  
([ADDRESS_LIST= ]  
  (ADDRESS= ...)  
  [(ADDRESS= ...)])
```

Route List

CMAN_RULES

Purpose

Use the parameter `CMAN_RULES` to specify an access control rule list to filter incoming connections. A rule list specifies which connections are accepted, rejected, or dropped.

Syntax

```
CMAN_RULES=  
(RULE_LIST=  
  (RULE=  
    (SRC=host)  
    (DST=host)  
    (SRV=service_name|sid)  
    (ACT=accept|reject))  
  [(RULE= ...)])
```

Subparameters

The `RULE` parameter filters a connection or group of connections using the following subparameters:

`SRC`: Specify the source host name or IP address in dot notation of the client.

`DST`: Specify the destination server host name or IP address in dot notation of the database server.

`SRV`: Specify database **service name** of the Oracle9i or Oracle8i database (obtained from the `SERVICE_NAME` parameter in the initialization parameter file) or **Oracle System Identifier (SID)** value of the pre-release 8.1 database (obtained from the `ORACLE_SID` environment variable or registry value).

`ACT`: Specify to `accept` to accept incoming request or `reject` to reject incoming requests.

Usage Notes

- If no rules are specified, all connections are accepted.
- For rejected connections, Oracle Connection Manager displays an error message. For dropped connections, it displays no message.
- The wildcard for host name is an asterisk (*). In the case of an IP address (d . d . d . d), you can wildcard the individual d's with this character.
- If client's connect descriptors contains both `SID` and `SERVICE_NAME` parameters, then both names requested need to be permitted by the rules respectively in order for the client's access to be allowed.

Example

```
CMAN_RULES=
(RULE_LIST=
(RULE=
(SRC=client1-pc)
(DST=sales-server)
(SRV=sales.us.acme.com)
(ACT=reject))
(RULE=
(SRC=144.25.23.45)
(DST=144.25.187.200)
(SRV=db1)
(ACT=accept)))
```

Parameter List

CMAN_PROFILE

Purpose

Use the parameter `CMAN_PROFILE` to specify the set attributes for an Oracle Connection Manager. To override a parameter's default setting, enter the parameter and its nondefault value in the `PARAMETER_LIST`.

Default Parameters

`ANSWER_TIMEOUT=0`

`AUTHENTICATION_LEVEL=0`

`LOG_LEVEL=0`

`MAX_FREELIST_BUFFERS=0`

`MAXIMUM_CONNECT_DATA=1024`

`MAXIMUM_RELAYS=128`

`RELAY_STATISTICS=NO`

`REMOTE_ADMIN=NO`

`SHOW_TNS_INFO=NO`

`TRACE_DIRECTORY=`**The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME\network\trace` directory on Windows NT**

`TRACE_FILELEN=unlimited`

`TRACE_FILENO=1`

`TRACE_TIMESTAMP=ON`

`TRACING=NO`

`USE_ASYNC_CALL=YES`

Allowed Values of Parameters

```

ANSWER_TIMEOUT=[ 0 to n ]
AUTHENTICATION_LEVEL=[ 0 | 1 ]
LOG_LEVEL=[ 0-4 ]
MAXIMUM_CONNECT_DATA=[ 257 to 4096 ]
MAX_FREELIST_BUFFERS=[ 0 to 10240 ]
MAXIMUM_RELAYS=[ 1 to 2048 ]
RELAY_STATISTICS=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]
REMOTE_ADMIN=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]
SHOW_TNS_INFO=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]
TRACE_DIRECTORY=directory
TRACE_FILELEN=size of file in kilobytes
TRACE_FILENO=[ 1 to 8 ]
TRACE_TIMESTAMP=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]
TRACING=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]
USE_ASYNC_CALL=[ YES | TRUE | ON | 1 | NO | FALSE | OFF | 0 ]

```

Example

```

CMAN_PROFILE=
(PARAMETER_LIST=
(MAXIMUM_RELAYS=512)
(LOG_LEVEL=1)
(RELAY_STATISTICS=YES)
(REMOTE_ADMIN=YES)
(SHOW_TNS_INFO=YES)
(USE_ASYNC_CALL=YES)
(AUTHENTICATION_LEVEL=0)
(TRACING=YES)
(TRACE_TIMESTAMP=YES)
(TRACE_FILELEN=100)
(TRACE_FILENO=2))

```

ANSWER_TIMEOUT

Use the parameter `ANSWER_TIMEOUT` to determine the time, in seconds, that Oracle Connection Manager uses to time out the protocol handshake associated with an incoming connection request

The parameter accepts a range of 0 to *n*.

AUTHENTICATION_LEVEL

Use the parameter `AUTHENTICATION_LEVEL` to specify the level of security. This parameter accepts the following values:

- 1 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.
- 0 (default) to instruct Oracle Connection Manager not to check for SNS between the client and server

LOG_LEVEL

The `LOG_LEVEL` parameter specifies the level of logging performed by Oracle Connection Manager. This parameter accepts four log levels, ranging from 0 to 4:

- 0 (default) for no logging
- 1 for basic reporting
- 2 for `RULE_LIST` matching lookup reporting
- 3 for relay blocking reporting
- 4 for relay I/O counts reporting

The CMGW process creates a log file called `cman_pid.log`, and the CMADMIN process creates a log file called `cmadm_pid.log`. The log files are located in the `$ORACLE_HOME/network/log` directory on UNIX operating systems and the `ORACLE_HOME\network\log` directory on Windows NT.

MAX_FREELIST_BUFFERS

The `MAX_FREELIST_BUFFERS` parameter specifies the maximum number of buffers that **Transparent Network Substrate (TNS)** keeps in its freelist for later reuse instead of returning them to the operating system after a relay gets closed.

The parameter accepts a range of 0 to 10240.

MAXIMUM_CONNECT_DATA

Use the parameter `MAXIMUM_CONNECT_DATA` to limit the connect data string length of the incoming connection requests.

The parameter accepts a range of 257 to 4096.

MAXIMUM_RELAYS

Use the parameter `MAXIMUM_RELAYS` to specify the maximum number of concurrent connections supported.

The parameter accepts a range of 1 to 2048.

RELAY_STATISTICS

Use the parameter `RELAY_STATISTICS` to specify whether or not I/O statistics are recorded. This parameter accepts the following values:

- `yes` to instruct Oracle Connection Manager to maintain statistics pertaining to relay I/O activities, such as:
 - Number of inbound (IN) bytes
 - Number of outbound (OUT) bytes
 - Number of IN packets
 - Number of OUT packets

This information is stored in the `cman_pid.log` file.

- `no` to record no I/O statistics

REMOTE_ADMIN

Use the parameter `REMOTE_ADMIN` to specify whether or not remote access to an Oracle Connection Manager is allowed. This parameter accepts the following values:

- `yes` to allow access from a remote Oracle Connection Manager Control utility session to Oracle Connection Manager
- `no` to allow only access to the local Oracle Connection Manager. This value prevents a user running a remote Oracle Connection Manager Control utility session access to Oracle Connection Manager.

See Also: ["Distributed Operations"](#) on page 2-4 for configuration details

SHOW_TNS_INFO

Use the parameter `SHOW_TNS_INFO` to specify whether or not TNS information is to be recorded. This parameter accepts the following values:

- `yes` to instruct Oracle Connection Manager to include TNS information in the `cman_pid.log` file
- `no` to instruct Oracle Connection Manager to not include TNS events in the log file

TRACING

Use the parameter `TRACING` to specify whether or not tracing is enabled for the Oracle Connection Manager. This parameter accepts the following values:

- `yes` to enable tracing for Oracle Connection Manager. The CMGW process creates a trace file called `cman_pid.trc`, and the CMADMIN process creates a trace file called `cmadm_pid.trc`.
- `no` to disable tracing

TRACE_DIRECTORY

Use the parameter `TRACE_DIRECTORY` to specify the destination directory for the trace files. You must set this parameter to a nondefault location if you plan to use the `TRACE_FILELEN` or `TRACE_TIMESTAMP` parameters.

TRACE_FILELEN

Use the parameter `TRACE_FILELEN` to specify the size of the trace file 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` parameter.

Note: This parameter requires a nondefault trace directory be specified with the `TRACE_DIRECTORY` parameter.

TRACE_FILENO

Use the parameter `TRACE_FILENO` 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.

The trace file names are distinguished from one another by their sequence number. For example, if this parameter is set to 3, the CMGW gateway trace files would be named `cman1_pid.trc`, `cman2_pid.trc` and `cman3_pid.trc` and the CMADMIN administrative trace files would be named `cmadm1_pid.trc`, `cmadm2_pid.trc` and `cmadm3_pid.trc`.

In addition, trace events in the trace files are preceded by the sequence number of the file.

TRACE_TIMESTAMP

Use the parameter `TRACE_TIMESTAMP` to add a time stamp in the form of `dd-mon-yyyy hh:mi:ss:mil` to every trace event in the trace file.

Note: This parameter requires a nondefault trace directory be specified with the `TRACE_DIRECTORY` parameter.

USE_ASYNC_CALL

Use the parameter `USE_ASYNC_CALL` to specify whether or not Oracle Connection Manager is to use asynchronous functions while in the answering or calling phase of establishing an Oracle Net connection. This parameter accepts the following values:

- `yes` to instruct Oracle Connection to use all asynchronous functions
- `no` to instruct Oracle Connection Manager not to use asynchronous functions

Oracle Names Parameters (names.ora)

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

This chapter contains these topics:

- [Overview of Oracle Names Server Configuration File](#)
- [Oracle Names Parameters](#)

Overview of Oracle Names Server Configuration File

Oracle Names server configuration, stored in the `names.ora` file, consists of the following elements:

- Oracle Names server's name
- Protocol addresses that it is accepting connection requests on
- Location of **administrative region** information
- Domain information
- Control parameters

By default, `names.ora` is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and the `ORACLE_HOME\network\admin` directory on Windows NT. `names.ora` can also be stored the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value
- On UNIX operating systems, the global configuration directory. For example, on Sun SPARC Solaris, this directory is `/var/opt/oracle`.

See Also: Oracle operating system-specific documentation

Figure 10–1 shows a `names.ora` file for an Oracle Names server named `namesvr2.us.acme.com`.

Figure 10–1 Example names.ora File

```
NAMES.SERVER_NAME=namesvr2.us.acme.com
NAMES.ADDRESSES=
  (ADDRESS=(PROTOCOL=tcp)(HOST=namesvr2-server)(PORT=1575))
NAMES.ADMIN_REGION=
  (REGION=
    (DESCRIPTION=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521))
      (CONNECT_DATA=
        (SERVICE_NAME=sales.us.acme.com))
      (USERID=system)
      (PASSWORD=manager)
      (NAME=local_region)
      (REFRESH=86400)
      (RETRY=60)
      (EXPIRE=600))
```

```

NAMES.DOMAIN_HINTS=
  (HINT_DESC=
    (HINT_LIST=
      (HINT=
        (NAME=namesvr1)
        (ADDRESS=(PROTOCOL=tcp)(HOST=namesvr1-server)(PORT=1575))
      )
    )
  )
NAMES.DOMAINS=
  (DOMAIN=
    (NAME=us.acme.com)
    (MIN_TTL=86400))

```

Oracle Names Parameters

This section lists and describes the `names.ora` file parameters.

NAMES.ADDRESSES

Purpose

Use the parameter `NAMES.ADDRESSES` to list the **protocol addresses** on which the Oracle Names server listens. Any valid `ADDRESS` or `ADDRESS_LIST` is allowed.

See Also:

- ["ADDRESSes and ADDRESS_LISTs"](#) on page 5-2 for a description of address syntax
- ["Protocol Parameters"](#) on page 5-3 for descriptions of the correct parameters to use for each protocol

Syntax

```

NAMES.ADDRESSES=
  [ (ADDRESS_LIST=
    (ADDRESS=...)
    (ADDRESS=...)[ ] ]

```

Default

```

NAMES.ADDRESSES=
  (ADDRESS=(PROTOCOL=tcp)(HOST=namesvr-server)(PORT=1575))

```

Example

```
NAMES.ADDRESSES=  
  (ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=tcp)(HOST=namesrv2-server)(PORT=1575))  
    (ADDRESS=(PROTOCOL=tcp)(HOST=namesrv2-server)(PORT=1375)))
```

NAMES.ADMIN_REGION

Purpose

Use the parameter `NAMES.ADMIN_REGION` to specify the data source for an administrative region.

Syntax

```
NAMES.ADMIN_REGION=  
  (REGION=  
    (DESCRIPTION=  
      (ADDRESS=...)  
      (CONNECT_DATA=(SERVICE_NAME|SID=service_name|sid)))  
    (USERID=user)  
    (PASSWORD=password)  
    (REFRESH=seconds)  
    (RETRY=seconds)  
    (EXPIRE=seconds)  
    (VERSION=version)  
    (TYPE=ldap|ldif)  
    (HOST=directory_host)  
    (PORT=port)  
    (TIMEOUT=time)  
    (SUBTREE_LIST=  
      (SUBTREE=(BASE=base_DN)[(SCOPE=sub|one)]))
```

Subparameters

REGION supports the following subparameters:

DESCRIPTION: Specify the connect descriptor that the Oracle Names server uses to connect to the **region database**.

USERID: If loading data from the region database, specify a database user. If loading restricted data from a LDAP-compliant **directory server**, specify a directory user with read privileges in the form of a **distinguished name (DN)**.

PASSWORD: Specify the password for the database user or directory user.

REFRESH: Specify the interval in seconds when the local Oracle Names server's memory cache (not the cache checkpoint file) will be refreshed from the region database.

The interval value for this parameter is irrelevant for **Oracle Names LDAP Proxy servers**, because the directory has no mechanism for enabling the Oracle Names LDAP Proxy server to determine if there is new data. Even if this parameter is specified, an Oracle Names LDAP Proxy server always has a reload interval of 248 days, 13 hours, and 20 minutes

RETRY: Specify the interval in seconds the Oracle Names server will wait after a failure to connect to the region database before trying again.

EXPIRE: Specify the time in seconds to wait before terminating the retry attempt.

VERSION: Specify an arbitrary value. Even though the value is irrelevant, this parameter is still necessary.

The following parameters are reserved for Oracle Names LDAP Proxy servers:

TYPE: Specify how the Oracle Names LDAP Proxy server is to load data from a directory server. This parameter supports the following values:

ldap: Specifies that the Oracle Names LDAP Proxy server is to load data directly from a directory server

ldif: Specifies that the Oracle Names LDAP Proxy server is to use data from an **LDAP Data Interchange Format (LDIF)** file

The following subparameters are for (TYPE=ldap):

HOST: Specify the directory server host name.

PORT: Specify the listening TCP/IP port for the directory server. If you do not specify this option, the default port (389) is used.

TIMEOUT: Specify the time limit in seconds in which the Oracle Names LDAP Proxy server can spend performing a search of directory objects. This

time limit cannot be greater than the time limit set for searches in the directory server. By default, the time limit is set to 10 seconds, which is sufficient for most searches.

SUBTREE_LIST | SUBTREE: Use the **SUBTREE** parameter to specify the DN. Use the **SUBTREE_LIST** parameter if there are multiple DNs. The **SUBTREE** parameter supports the following subparameters:

BASE: Specify a subtree location in the **directory information tree (DIT)** from which the Oracle Names Proxy server should load data. The location is specified as a DN of a LDAP directory entry. The Oracle Names Proxy loads directory naming data from one or more Oracle Contexts located in the subtree following the entry specified.

Note: Do not prefix the DN with "dn:". For example, (BASE=dn:dc=com) is not acceptable.

SCOPE: Specifies where the Oracle Names Proxy server looks for directory data to load. **one** (default) specifies that the Oracle Names Proxy server look only in one Oracle Context directly following the entry specified in the **BASE** parameter. **sub** specifies the Oracle Names Proxy server look in all the Oracle Contexts in the subtree following the entry specified in the **BASE** parameter.

See Also: *Oracle9i Net Services Administrator's Guide* for more information about configuring the **NAMES.ADMIN_REGION** parameter for Oracle Names LDAP Proxy servers

Example

```
NAMES.ADMIN_REGION=
  (REGION=
    (DESCRIPTION=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1575))
      (CONNECT_DATA=(SERVICE_NAME=sales.us.acme.com)))
    (USERID=system)
    (PASSWORD=manager)
    (REFRESH=172800)
    (RETRY=2700)
    (EXPIRE=8700)
    (VERSION=34619392))
```

NAMES.AUTHORITY_REQUIRED

Purpose

Use the parameter `NAMES.AUTHORITY_REQUIRED` to specify whether or not system queries require authoritative answers.

Default

`false`

Example

```
NAMES.AUTHORITY_REQUIRED=true
```

NAMES.AUTO_REFRESH_EXPIRE

Purpose

Use the parameter `NAMES.AUTO_REFRESH_EXPIRE` to specify the amount of time, in seconds, for the Oracle Names server to cache other region's database server addresses which have been obtained through the `NAMES.DOMAIN.HINTS` parameter. At the end of this interval, the Oracle Names server issues a query to the other region database servers to refresh the address.

Default

`600`

Acceptable Values

`60 to 1209600`

Example

```
NAMES.AUTO_REFRESH_EXPIRE=1200000
```

NAMES.AUTO_REFRESH_RETRY

Purpose

Use the parameter `NAMES.AUTO_REFRESH_RETRY` to specify the interval, in seconds, that the Oracle Names server retries the other Oracle Names servers on its domain hint list.

Default

180

Minimum Value

60

Maximum Value

3600

Example

```
NAMES.AUTO_REFRESH_RETRY=180
```

NAMES.CACHE_CHECKPOINT_FILE

Purpose

Use the parameter `NAMES.CACHE_CHECKPOINT_FILE` to specify the name and directory path of the file to which the Oracle Names server writes its checkpoint file.

Default

The `$ORACLE_HOME/network/names/ckpcch.ora` file on UNIX operating systems and the `ORACLE_HOME\network\names\ckpcch.ora` file on Windows NT

Example

```
NAMES.CACHE_CHECKPOINT_FILE=c:\oracle\network\names\cacheck.ora
```

NAMES.CACHE_CHECKPOINT_INTERVAL

Purpose

Use the parameter `NAMES.CACHE_CHECKPOINT_INTERVAL` to specify the interval, in seconds, in which the Oracle Names server writes a checkpoint of its stored data to a checkpoint file. Each Oracle Names server can periodically write its cached data to a file to protect against startup failures.

Default

0 (disabled)

Minimum Value

10

Maximum Value

259200 (3 days)

Example

```
NAMES.CACHE_CHECKPOINT_INTERVAL=24
```

NAMES.CONFIG_CHECKPOINT_FILE

Purpose

Use the parameter `NAMES.CONFIG_CHECKPOINT_FILE` to specify the name and directory path of the file used to checkpoint Oracle Names server configuration settings.

Default

The `$ORACLE_HOME/network/names/ckpcfg.ora` file on UNIX operating systems and the `ORACLE_HOME\network\names\ckpcfg.ora` file on Windows NT

Example

```
NAMES.CONFIG_CHECKPOINT_FILE=c:\oracle\network\names\configck.ora
```

NAMES.CONNECT_TIMEOUT

Purpose

Use the parameter `NAMES.CONNECT_TIMEOUT` to specify the amounts of time, in seconds, for the Oracle Names server to wait for the connection from a client to complete.

Default

3

Minimum Value

1

Maximum Value

600

Example

```
NAMES.CONNECT_TIMEOUT=8
```

NAMES.DEFAULT_FORWARDERS

Purpose

Use the parameter `NAMES.DEFAULT_FORWARDERS` to provide an address list of other Oracle Names servers used to forward queries.

Syntax

```
NAMES.DEFAULT_FORWARDERS=  
(FORWARDER_LIST=  
  (NAME=onames_server)  
  (ADDRESS=...))
```

Subparameters

`FORWARDER_LIST` supports the following subparameters:

NAME: Specify the Oracle Names server name to forward queries.

ADDRESS: Specify the protocol address of the Oracle Names server.

See Also: ["Protocol Parameters"](#) on page 5-3 for descriptions of the correct parameters to use for each protocol

Example

```
NAMES.DEFAULT_FORWARDERS=  
(FORWARDER_LIST=  
  (FORWARDER=  
    (NAME=rootsevl.com)  
    (ADDRESS=(PROTOCOL=tcp)(HOST=root-server)(PORT=4200))))
```

NAMES.DEFAULT_FORWARDERS_ONLY

Purpose

Use the parameter `NAMES.DEFAULT_FORWARDERS_ONLY` to specify how the Oracle Names server forwards queries.

When you set the parameter to `true`, the Oracle Names server forwards queries only to those Oracle Names servers listed as default forwarders with the `NAMES.DEFAULT_FORWARDERS` parameter. When you set the parameter to `false`, the Oracle Names server calls the Oracle Names servers listed as default forwarders before other Oracle Names servers found in the cache.

Default

`false`

Example

```
NAMES.DEFAULT_FORWARDERS_ONLY=c:\oracle\network\names\domainck.ora
```

NAMES.DOMAIN_CHECKPOINT_FILE

Purpose

Use the parameter `NAMES.DOMAIN_CHECKPOINT_FILE` to specify the name and path of the file used for checkpoint authoritative domain data.

Default

The `$ORACLE_HOME/network/names/ckpdom.ora` file on UNIX operating systems and the `ORACLE_HOME\network\names\ckpdom.ora` file on Windows NT

Example

```
NAMES.DOMAIN_CHECKPOINT_FILE=c:\oracle\network\names\domainck.ora
```

NAMES.DOMAINS

Purpose

Use the parameter `NAMES.DOMAINS` to provide a list of domains in the Oracle Names server's local region, as well as the default time to live (TTL) in seconds for data in those domains.

Syntax

```
NAMES.DOMAINS=  
(DOMAIN_LIST=  
  (DOMAIN=  
    (NAME=domain)  
    (MIN_TTL=ttl))
```

Subparameters

`NAMES.DOMAINS` supports the following subparameters:

`DOMAIN`: Specify the domain name.

Note: Specify the root domain with a dot (.) or a null value.

`MIN_TTL`: Specify the TTL for the domain's data.

Example

In the following example, `NAMES.DOMAINS` contains a listing for the root (value of null), `com`, `acme.com`, and `hq.oracle.com` domains. All the domain precedent to `hq.acme.com` must be defined in order to define `hq.acme.com`.

```
NAMES.DOMAINS=
(DOMAIN_LIST=
(DOMAIN=
(NAME=)
(MIN_TTL=86400))
(DOMAIN=
(NAME=com)
(MIN_TTL=86400))
(DOMAIN=
(NAME=acme.com)
(MIN_TTL=86400))
(DOMAIN=
(NAME=hq.acme.com)
(MIN_TTL=86400)))
```

NAMES.DOMAIN_HINTS

Purpose

Use the parameter `NAMES.DOMAINS_HINTS` in one of two ways:

- Identify the protocol address of an Oracle Names server in the root administrative region for delegated administrative regions
- Identify the name of the domain and at least one protocol address of an Oracle Names server in that domain for optimizing forwarded requests to remote administrative regions

Subparameters

`NAMES.DOMAINS_HINTS` supports the following subparameters:

HINT_LIST: Specify a list of hints. Each hint contains the name and address of Oracle Names server in a remote administrative region. `HINT_LIST` uses the following syntax:

```
(HINT_LIST=  
  (HINT=(NAME=onames_server) (ADDRESS=...))
```

DOMAIN_LIST: Specify the list of remote domains. `DOMAIN_LIST` uses the following syntax:

```
(DOMAIN_LIST=  
  (DOMAIN=domain))
```

Usage Notes

See Also: *Oracle9i Net Services Administrator's Guide* for usage notes

Examples

In the following example, `NAMES.DOMAIN_HINTS` contains a domain hint for Oracle Names server `rootsvr.com` that is located in the root domain of the remote administrative region. The `DOMAIN` parameter is left null, meaning that the hint is for the root domain.

```
NAMES.DOMAIN_HINTS=
  (HINT_DESC=
    (HINT_LIST=
      (HINT=
        (NAME=rootsvr.com)
        (ADDRESS=(PROTOCOL=tcp)(HOST=root-server)(PORT=1575)))
      (DOMAIN_LIST=
        (DOMAIN=)))
```

The following example shows a hint to query two domains, the root domain and the `us.acme.com` domain, for Oracle Names servers `rootsvr.com` and `namesvr.us.acme.com`.

```
NAMES.DOMAIN_HINTS=
  (HINT_DESC=
    (HINT_LIST=
      (HINT=
        (NAME=rootsvr.com)
        (ADDRESS=(PROTOCOL=tcp)(HOST=root-server)(PORT=1575)))
      (HINT=
        (NAME=namesrv.us.acme.com)
        (ADDRESS=(PROTOCOL=tcp)(HOST=hr-server)(PORT=1575)))
      (DOMAIN_LIST=
        (DOMAIN=)
        (DOMAIN=us.acme.com)))
```

NAMES.FORWARDING_AVAILABLE

Purpose

Use the parameter `NAMES.FORWARDING_AVAILABLE` to specify how the Oracle Names server forwards client requests.

When you set the parameter to `on`, the Oracle Names server forwards client requests to remote Oracle Names server. When you set the parameter to `off`, clients outside the local domain without access to the network are unable to resolve names.

Default

`on`

Values

`on` | `off`

Example

```
NAMES.FORWARDING_AVAILABLE=off
```

NAMES.FORWARDING_DESIRED

Purpose

Use the parameter `NAMES.FORWARDING_DESIRED` to specify how the Oracle Names server handles requests for remote Oracle Names servers.

When you set the parameter to `true`, the Oracle Names server provides protocol address of remote Oracle Names server to clients. This way, clients are redirected to the appropriate Oracle Names server. When you set the parameter to `false`, the Oracle Names server connects to the remote Oracle Names server on behalf of clients.

Default

`true`

Values

`true` | `false`

Example

```
NAMES.FORWARDING_DESIRED=true
```

NAMES.KEEP_DB_OPEN

Purpose

Use the parameter `NAMES.KEEP_DB_OPEN` to specify whether to attempt to keep the connection to the region database open between operations.

When you set the parameter to `true`, the connection is left open after each load, reload, or reload-check. When you set the parameter to `false`, the connection is closed after each load, reload, or reload-check.

Default

```
true
```

Values

```
true | false
```

Example

```
NAMES.KEEP_DB_OPEN=false
```

NAMES.LOG_DIRECTORY

Purpose

Use the parameter `NAMES.LOG_DIRECTORY` to specify the destination directory where the log file for Oracle Names server operational events are written

Default

The `$ORACLE_HOME/network/log` directory on UNIX operating systems and the `ORACLE_HOME\network\log` directory on Windows NT

Example

```
NAMES.LOG_DIRECTORY=c:\oracle\network\names
```

NAMES.LOG_FILE

Purpose

Use the parameter `NAMES.LOG_FILE` to indicate the name of the output file to which Oracle Names server operational events are written. The file name extension is always `.log`. Do not enter an extension for this parameter.

Default

`names`

Example

```
NAMES.LOG_FILE=onames
```

NAMES.LOG_STATS_INTERVAL

Purpose

Use the parameter `NAMES.LOG_STATS_INTERVAL` to specify the number of seconds between full statistical dumps in the log file.

Default

0 (disabled)

Minimum Value

10 seconds

Maximum Value

none

Example

```
NAMES.LOG_STATS_INTERVAL=12
```

NAMES.LOG_UNIQUE

Purpose

Use the parameter `NAMES.LOG_UNIQUE` to indicate whether or not log files are given unique names.

When you set the parameter to `on`, a process identifier is appended to the name of each log file generated, allowing multiple log files to coexist. For example, log files named `namespid.log` are created if the default log file name, `names.log`, is used.

Default

`false`

Values

`true` | `false`

Example

```
NAMES.LOG_UNIQUE=true
```

NAMES.MAX_OPEN_CONNECTIONS

Purpose

Use the parameter `NAMES.MAX_OPEN_CONNECTIONS` to specify the number of connections that the Oracle Names server can have open at any given time. The value is generated as the value 10 or the sum of one connection for listening, 5 for clients, plus one for each remote domain defined in the local administrative region, whichever is greater.

The calculated value is acceptable for most installations.

Default

Calculated based on entered data.

Minimum Value

2

Maximum Value

64

Example

```
NAMES.MAX_OPEN_CONNECTIONS=52
```

NAMES.MAX_REFORWARDS**Purpose**

Use the parameter `NAMES.MAX_REFORWARDS` to specify the maximum number of times the Oracle Names server attempts to forward an operation.

Default

2

Minimum Value

1

Maximum Value

15

Example

```
NAMES.MAX_REFORWARDS=2
```

NAMES.MESSAGE_POOL_START_SIZE**Purpose**

Use the parameter `NAMES.MESSAGE_POOL_START_SIZE` to determine the initial number of messages allocated in the Oracle Names server's message pool that are used for incoming or outgoing forwarded messages.

Default

10

Minimum Value

3

Maximum Value

256

Example

```
NAMES.MESSAGE_POOL_START_SIZE=10
```

NAMES.NO_MODIFY_REQUESTS

Purpose

Use the parameter `NAMES.NO_MODIFY_REQUESTS` to allow or disallow operations that modify data in the Oracle Names server's region.

When the parameter `NAMES.NO_MODIFY_REQUESTS` is set to `true`, the Oracle Names server refuses any operations that modify data in its region. When the parameter `NAMES.NO_MODIFY_REQUESTS` is set to `false`, the Oracle Names server accepts operations that modify data in its region.

Default

`false`

Values

`true` | `false`

Example

```
NAMES.NO_MODIFY_REQUESTS=true
```

NAMES.NO_REGION_DATABASE

Purpose

Use the parameter `NAMES.NO_REGION_DATABASE` to enable or disable the Oracle Names server from loading data from a region database.

When you set the parameter to `true`, the Oracle Names server does not load data from a region database. Instead, it loads data from its checkpoint files. When you set the parameter to `false`, the Oracle Names server loads data from a region database.

Default

`false`

Values

`true` | `false`

Example

```
NAMES.NO_REGION_DATABASE=true
```

NAMES.PASSWORD

Purpose

Use the parameter `NAMES.PASSWORD` to set an encrypted password for an Oracle Names server, so that certain privileged operations, such as [STOP](#), [RESTART](#) and [RELOAD](#), used from the Oracle Names Control utility are secure.

If this parameter is set with Oracle Net Manager, then the password is encrypted. A clear-text password can be made manually. If the password is clear-text, ensure that the `NAMESCTL.INTERNAL_ENCRYPT_PASSWORD` parameter in the `sqlnet.ora` file is set to `false`.

Default

None

Example

```
NAMES.PASSWORD=625926683431aa55
```

NAMES.RESET_STATS_INTERVAL

Purpose

Use the parameter `NAMES.RESET_STATS_INTERVAL` to specify the number of seconds during which the statistics collected by the Oracle Names servers should accumulate. At the frequency specified, they are reset to 0. The default value of 0 means never reset statistics.

Default

0 (never reset)

Minimum Value

10

Maximum Value

None

Example

```
NAMES.RESET_STATS_INTERVAL=15
```

NAMES.SAVE_CONFIG_ON_STOP

Purpose

Use the parameter `NAMES.SAVE_CONFIG_ON_STOP` to specify whether or not runtime configuration changes are saved into the `names.ora` file.

When you set the parameter to `true`, the Oracle Names Control utility saves runtime configuration changes to `names.ora` file. Any parameters which were modified through Oracle Names Control utility `SET` operations replace prior `names.ora` settings. When you set the parameter to `false`, the Oracle Names Control utility does not save runtime configuration changes to the `names.ora` file.

Default`false`**Example**`NAMES.SAVE_CONFIG_ON_STOP=false`**NAMES.SERVER_NAME**

Use the `NAMES.SERVER_NAME` parameter to uniquely identify an Oracle Names server by a name. All configuration references to a particular Oracle Names server use this name.

Default`ONAMES_ onames_server`**Example**`NAMES.SERVER_NAME=namesrv1.us.acme.com`

NAMES.TOPOLOGY_CHECKPOINT_FILE

Purpose

Use the parameter `NAMES.TOPOLOGY_CHECKPOINT_FILE` to specify the name and path of the file used for checkpoint topology data. This file defines the domains in the administrative region and the Oracle Names servers authoritative for each domain. Specifically, topology data consists of:

- Domains, defined as a list of Oracle Names servers names with `NS.SMD` record types that are authoritative for the domain
- Oracle Names servers, specified with a `A.SMD` record type and listening protocol addresses

Default

The `$ORACLE_HOME/network/names/ckptop.ora` file on UNIX operating systems and the `ORACLE_HOME\network\names\ckptop.ora` on Windows NT

Example

```
NAMES.TOPOLOGY_CHECKPOINT_FILE=c:\oracle\network\names\topck.ora
```

NAMES.TRACE_DIRECTORY

Purpose

Use the parameter `NAMES.TRACE_DIRECTORY` to specify the destination directory of the output trace files from Oracle Names trace sessions.

Default

The `$ORACLE_HOME/network/trace` directory on UNIX operating systems and the `ORACLE_HOME/network/trace` directory on Windows NT

Example

```
NAMES.TRACE_DIRECTORY=/oracle/network/admin/trace
```

NAMES.TRACE_FILE

Purpose

Use the parameter `NAMES.TRACE_FILE` to specify the name of the output trace files for Oracle Names trace sessions. The file name extension is always `.trc`.

Default

`names`

Example

```
NAMES.TRACE_FILE=onames
```

NAMES.TRACE_FUNC

Purpose

Use the parameter `NAMES.TRACE_FUNC` to enable the internal mechanism to control tracing by a function name.

Default

`false`

Example

```
NAMES.TRACE_FUNC=false
```

NAMES.TRACE_LEVEL

Purpose

Use the parameter `NAMES.TRACE_LEVEL` to turn Oracle Names tracing on, at a specific level, or 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

```
NAMES.TRACE_LEVEL=admin
```

NAMES.TRACE_UNIQUE

Purpose

Use the parameter `NAMES.TRACE_UNIQUE` to indicate whether or not trace files are given unique names. When the value is set to `on`, a process identifier is appended to the name of the trace file generated, allowing multiple trace files to coexist.

For example, trace files named `namespid.trc` are created if the default trace file name, `names.trc`, is used.

Default

`on`

Values

on | off

Example

```
NAMES.TRACE_UNIQUE=on
```

Directory Usage Parameters (ldap.ora)

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

This chapter contains these 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 Net Configuration Assistant. Do not modify these parameters or their settings.

By default, `ldap.ora` is located in the `$ORACLE_HOME/network/admin` directory on UNIX operating systems and the `ORACLE_HOME\network\admin` directory on Windows operating systems. `ldap.ora` can also be stored in the directory specified by the `TNS_ADMIN` environment variable.

Directory Usage Parameters

This section lists and describes the `ldap.ora` file configuration parameters.

DIRECTORY_SERVERS

Purpose

Use the parameter `DIRECTORY_SERVERS` 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

Use the parameter `DIRECTORY_SERVER_TYPE` 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

Use the parameter `DEFAULT_ADMIN_CONTEXT` 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, "LDAP Schema for Oracle Net Services"](#)
- [Appendix B, "Commands and Parameters Not Supported in This Release"](#)

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 these topics:

- [Structural Object Classes](#)
- [Attributes](#)

Structural Object Classes

The Oracle schema supports the following structural object classes for Oracle Net **directory naming** lookups:

- `orclDBServer`
- `orclNetService`
- `orclNetServiceAlias`
- `orclNetDescription`
- `orclNetDescriptionList`
- `orclNetAddress`
- `orclNetAddressList`

orclDBServer

Description

Defines the attributes for database service entries

Attributes

`orclNetDescName`

`orclVersion`

orclNetService

Description

Defines the attributes for **net service name** entries

Attributes

`orclNetDescName`

`orclVersion`

orclNetServiceAlias

Description

Defines the attributes for **net service alias** entries

Attributes

`orclNetDescName`

`orclVersion`

orclNetDescription

Description

Specifies a **connect descriptor** containing the **protocol address** of the **listener** and the connect information to the service

Attributes

- `orclNetAddrList`
- `orclNetInstanceName`
- `orclNetConnParamList`
- `orclNetFailover`
- `orclNetLoadBalance`
- `orclNetSdu`
- `orclNetServiceName`
- `orclNetSourceRoute`
- `orclSid`
- `orclVersion`

orclNetDescriptionList

Description

Specifies a list of connect descriptors

Attributes

- `orclNetDescList`
- `orclVersion`

orclNetAddress

Description

Specifies a listener protocol address

Attributes

- `orclNetAddressString`
- `orclNetProtocol`
- `orclVersion`

orclNetAddressList

Description

Specifies a list of protocol addresses

Attributes

- `orclNetAddrList`
- `orclNetFailover`
- `orclNetLoadBalance`
- `orclNetSourceRoute`
- `orclVersion`

Attributes

[Table A-1](#) lists the attributes used for the object classes. This list is subject to change.

Table A-1 LDAP Schema Attributes for Oracle Net Services

Attribute	Description
orclNetAddrList	Identifies one or more listener protocol addresses
orclNetAddressString	Defines a listener protocol address
orclNetConnParamList	Placeholder for future 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
orclNetInstanceName	Specifies the instance name to access
orclNetLoadBalance	Turns client load balancing on for a protocol address list
orclNetProtocol	Identifies the protocol used in the <code>orclAddressString</code> attribute
orclNetSdu	Specifies the session data unit (SDU) size
orclNetServiceName	Specifies the Oracle9i or Oracle8i database service name in the <code>CONNECT_DATA</code> 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 <code>CONNECT_DATA</code> portion of a connection descriptor
orclVersion	Specifies the version of software used to create the entry

Commands and Parameters Not Supported in This Release

This appendix describes features no longer supported by Oracle Net Services.

This appendix contains these topics:

- [Overview of Unsupported Features](#)
- [Unsupported Parameters](#)
- [Unsupported Control Utility Commands](#)

Overview of Unsupported Features

In an effort to streamline configuration decisions for the Internet, the following subsections describe the features and the configuration file that are no longer being supported:

- [Identix and SecurID Authentication Methods](#)
- [Novell Directory Services \(NDS\) External Naming and NDS Authentication](#)
- [Net8 OPEN](#)
- [protocol.ora File](#)
- [Prespawned Dedicated Servers](#)
- [Protocols](#)

Identix and SecurID Authentication Methods

If you are using Identix or SecurID **authentication methods**, provided by **Oracle Advanced Security**, Oracle Corporation recommends migrating to one of the following authentication methods:

- CyberSafe
- RADIUS
- Kerberos
- SSL

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

Novell Directory Services (NDS) External Naming and NDS Authentication

Support for NDS as an authentication method and as an external **naming method** is no longer supported. If you are using NDS as an external naming method, Oracle Corporation recommends using **directory naming** instead.

Net8 OPEN

Net8 OPEN, which provided an application program interface (API) that enabled programmers to develop both database and non-database applications, is no longer supported.

protocol.ora File

The `protocol.ora` file is no longer supported.

Parameters in the `protocol.ora` file have been merged into the `sqlnet.ora` file. These parameters enable you to configure access control to the database, as well as no delays in TCP/IP buffer flushing. These parameters include:

- [TCP.EXCLUDED_NODES](#)
- [TCP.INVITED_NODES](#)
- [TCP.NODELAY](#)
- [TCP.VALIDNODE_CHECKING](#)

See Also: [Chapter 6, "Profile Parameters \(sqlnet.ora\)"](#) for a description of these parameters

If you have a `protocol.ora` file in the `$ORACLE_HOME/network/admin` directory on UNIX, and the `ORACLE_HOME\network\admin` directory on Windows operating systems, Oracle Net Manager, when first started, automatically merges the `protocol.ora` parameters into the `sqlnet.ora` file.

There may be operating system-specific parameters in `protocol.ora` that are node specific. For this reason, Oracle Corporation recommends not sharing `sqlnet.ora` with other nodes after merging or adding these parameters.

Prespawnd Dedicated Servers

Prespawnd dedicated server processes are no longer supported. Instead, configure [shared server](#) to improve scalability and system resource usage.

Protocols

Protocol addresses using the SPX or LU6.2 protocol must be replaced. Oracle Net provides support for the following network protocols:

- TCP/IP
- TCP/IP with SSL
- Named Pipes

See Also: ["Protocol Parameters"](#) on page 5-3 for protocol parameter configuration

Unsupported Parameters

Table B-1 describes the networking parameters no longer supported.

Table B-1 *Unsupported Networking Parameters*

Parameter	File	Description	Last Supported Release
COMMUNITY	tnsnames.ora	The parameter was a required part of all network service addresses. Thus, it appears anywhere you might find an address (for example, local naming and listener configuration files).	8.0
AUTOMATIC_IPC	sqlnet.ora	This parameter was used to force sessions through IPC addresses. Due to performance issues, this parameter has been removed. Configure an IPC address instead.	8.0
NAMES.DEFAULT_ZONE	sqlnet.ora	This parameter was used to be included in profiles as slight variants of the NAMES.DEFAULT_DOMAIN and NAMES.PREFERRED_SERVERS parameters.	8.0
NAMES.NDS.NAME.CONTEXT	sqlnet.ora	This parameter was used to configure naming contexts for NDS external naming.	8.1
OSS.SOURCE_MY_WALLET	sqlnet.ora	This parameter's name has changed to WALLET_LOCATION .	8.1
SQLNET.IDENTIX_FINGERPRINT_DATABASE SQLNET.IDENTIX_FINGERPRINT_DATABASE_USER SQLNET.IDENTIX_FINGERPRINT_DATABASE_PASSWORD SQLNET.IDENTIX_FINGERPRINT_METHOD	sqlnet.ora	These parameters supported the Identix authentication method.	8.1
CONNECT_TIMEOUT	listener.ora	This parameter specified the amount of time that the listener waited for a client's request after the transport connection had been established.	8.1

Parameter	File	Description	Last Supported Release
PRESPAWN_DESC PRESPAWN_LIST PRESPAWN_MAX	listener.ora	These parameters were used for prespawnd dedicated server configuration. Prespawnd dedicated servers are no longer supported. Use shared server instead.	8.1
USE_PLUG_AND_PLAY_ <i>listener_name</i>	listener.ora	This parameter instructed the listener to register database information with an Oracle Names server during startup. Use the Oracle Names Control utility REGISTER command to register this information.	8.1
NAMES.USE_PLUG_AND_PLAY	names.ora	This parameter enabled/disabled the Dynamic Discovery Option. Use the Oracle Names Control utility REORDER_NS command to discover other Oracle Names servers.	2.3

Unsupported Control Utility Commands

[Table B-2](#) describes the control utility commands not supported in release 9.0.

Table B-2 *Unsupported Network Control Utility Commands*

Commands	Control Utility	Description
DOMAIN_HINT	Oracle Names Control utility	This command created a domain hint. Configure the NAMES.DOMAIN_HINTS parameter in the <code>names.ora</code> file instead.
DBSNMP_START DBSNMP_STATUS DBSNMP_STOP	Listener Control utility	These commands controlled the Oracle Intelligent Agent for use with Oracle Enterprise Manager. You can now control the Oracle Intelligent Agent through the Oracle Enterprise Manager Console.
SET CONNECT_TIMEOUT SHOW CONNECT_TIMEOUT	Listener Control utility	These commands specified the amount of time that the listener waited for a client's request after the transport connection had been established.
SET USE_PLUGANDPLAY SHOW USE_PLUGANDPLAY	Listener Control utility	These commands instructed the listener to register database information with an Oracle Names server. Use the Oracle Names Control utility REGISTER command to register this information.

Glossary

A

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 \(ACL\)](#).

access control

A feature of Oracle Connection Manager that sets rules for denying or allowing certain clients to access designated servers.

address

See [protocol address](#).

administrative region

An organizational entity for administering Oracle Network Services' components. Each administrative region includes:

- One or more domains
- One or more Oracle Names servers
- One or more databases and listeners

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 Names 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 high 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 Oracle9i, depending on whether or not **Oracle Advanced Security** is installed:

- CyberSafe
- RADIUS
- Kerberos
- **SSL**
- **Windows NT native authentication**

C

cache

Memory that stores recently-accessed data so that subsequent requests to access the same data can be processed quickly.

CDS

See [Cell Directory Services \(CDS\)](#).

Cell Directory Services (CDS)

An [external naming](#) method that enables users to use Oracle tools transparently and applications to access Oracle databases in a Distributed Computing Environment (DCE) environment.

central administration

An Oracle Names network where network management consists of one [administrative region](#) for the entire network. All Oracle Names servers know about one another and about all the services in the network. Contrast with [delegated administration](#).

client

A user, software application, or computer that requests the services, data, or processing of another application or computer. The client is the user process. In a network environment, the client is the local user process and the server may be local or remote.

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 Names](#) and [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 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 process

An **Oracle Connection Manager** administrative process responsible for all administrative functions of Oracle Connection Manager.

CMGW process

An **Oracle Connection Manager** gateway process that receives client connections and evaluates against a set of rules whether to deny or allow access. If access is allowed, the gateway process forwards the requests to the next hop, typically the database server. In addition to allowing or denying access, the CMGW process can also multiplex or funnel multiple client connections through a single protocol connection.

configuration files

Files that are used to identify and characterize the components of a network. Configuration is largely a process of naming network components and identifying relationships among those components.

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.acme.com`:

```
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)
  (CONNECT_DATA=
    (SERVICE_NAME=sales.us.acme.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** for Oracle9i or Oracle8i databases or its **Oracle System Identifier (SID)** for Oracle release 8.0 databases. The network route provides, at a minimum, the location of the listener through use of a network address.

connect identifier

A **connect descriptor** or a name that maps to a connect descriptor. A connect identifier can be a **net service name**, database **service name**, **alias**, or **net service alias**. Users initiate a connect request by passing a username and password along with a connect identifier in a connect string for the service to which they wish to connect:

```
CONNECT username/password@connect_identifier
```

connect string

Information the user passes to a service to connect, such as username, password, and **connect identifier**:

```
CONNECT username/password@net_service_name
```

connect-time failover

A client connect request is forwarded to a another listener if the first listener is not responding. Connect-time failover is enabled by **service registration**, because the listener knows if an instance is running to attempt a connection.

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

Load balancing, whereby the number of active connections among various instances and dispatchers for the same service are balanced. This enables listeners to make their routing decisions based on how many connections each dispatcher has and on how loaded the nodes that the instances run.

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.

D

data packet

See [packet](#).

database administrator (DBA)

(1) A person responsible for operating and maintaining an Oracle Server or a database application. (2) An Oracle username 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 pointer is actually defined as an entry in a data dictionary table. To access the link, you must be connected to the local database that contains the data dictionary entry.

A database link connection is one-way in the sense that a client connected to local database A can use a link stored in database A to access information in remote database B, but 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 they must define a link that is stored in the data dictionary of database B.

The following database links types 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.
- A [global database link](#) defined in an [Oracle Names server](#). Anyone in the network can use it.

decentralized administration

See [delegated administration](#).

dedicated server

A server process that is dedicated to one client connection. Contrast with [shared server process](#).

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 "." character within it.

delegated administration

A network where network management is delegated to one or more administrative regions under the **root administrative region**. Also referred to as distributed or decentralized administration. Contrast with **central administration**.

delegated administrative region

A region hierarchically under the **root administrative region**. Any region other than the root administrative region.

destination

The client that is the endpoint of a connection. The initiator of the connection requires some data or service of the destination.

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 clientside and serverside 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 administration

See [delegated administration](#).

distributed processing

Division of front-end and back-end processing to different computers. Oracle Network 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 hint

A `NAMES.DOMAIN_HINTS` parameter in the `names.ora` file that contains the name of the domain and at least one address of an Oracle Names server in that domain. This enables an Oracle Names server to forward the client requests to a specific address, reducing network traffic.

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 Network Services, DNS translates the host name in a TCP/IP address into an IP address.

domestic domains

The set of domains that are managed within a given administrative region. Domains are domestic only in relation to a region; they are never domestic in any absolute sense. Also referred to as local domains.

DNS

Domain Name System. See [Domain Name System \(DNS\)](#).

E**error message**

A message from a computer program informing you of a potential problem or condition preventing program or command execution.

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 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](#) or [CDS](#).

external procedure

A 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.

F

failover

See [connect-time failover](#).

firewall support

See [access control](#).

flat naming model

An Oracle Names infrastructure in which there is only one domain. All names must be unique within that domain.

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 Names Server caching performance.

G

global database link

A [database link](#) that is registered with an [Oracle Names server](#). A global database link that is the same as the [global database name](#) is automatically registered with an Oracle Names server. You can use a global database link to access an object in the database. For example, if the global database name is `sales.us.acme.com`, a global database link of `sales.us.acme.com` is created and registered with an Oracle Names server. You can then use `sales.us.acme.com` to perform a `SELECT` statement on a table, for example:

```
SQL> SELECT * FROM emp@sales.us.acme.com
```

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.acme.com`.

The database name portion, `sales`, is a simple name you wish to call your database. The database domain portion, `us.acme.com`, specifies the database domain in which the database is located, making the global database name unique. When possible, Oracle Corporation 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](#).

H

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 protocol

Hypertext transfer protocol. A protocol that provides the language that enables Web browsers and application Web servers to communicate.

initialization parameter file

File that contains information to initialize the database and instances.

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's 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

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 144.45.9.22.

IPC

See **Interprocess Communication**.

J

Java Database Connectivity (JDBC) Driver

A driver that provides Java applications and applets access to an Oracle database.

K

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.

L

latency

Networking round-trip time.

Lightweight Directory Access Protocol (LDAP)

A standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate. The framework of design conventions supporting industry-standard **directory servers**.

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 Net Configuration Assistant that contains the following directory server access information:

- Type of directory server
- Location of the directory server
- Default directory entry that the client or server will use to look up or configure connect identifiers for connections to database services

The `ldap.ora` file resides in `$ORACLE_HOME/network/admin` on UNIX and `ORACLE_HOME\network\admin` on Windows operating systems.

link qualifier

A qualifier appended to a global database link to provide alternate settings for the database username and password credentials. For example, a link qualifier of `fieldrep` can be appended to a global database link of `sales.us.acme.com`.

```
SQL> SELECT * FROM emp@sales.us.acme.com@fieldrep
```

listener

A process that resides on the database 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 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.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 `$ORACLE_HOME/network/admin` on UNIX and `ORACLE_HOME\network\admin` on Windows NT.

Oracle9i does not require identification of the database service because of **service registration**. However, static service configuration is required for if you plan to use Oracle Enterprise Manager.

Listener Control utility

A utility included with Oracle Network Services to control various listener functions, such as to starting, stopping, and getting the status of the listener.

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 Network 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.

M**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 the 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.

N

names.ora file

A configuration file that contains parameter settings for an **Oracle Names server**.

Named Pipes protocol

A high-level interface protocol providing interprocess communications between clients and servers (distributed applications). Named Pipes enables client/server conversation over a network using Named Pipes.

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 five naming methods:

- **local naming**
- **directory naming**
- **Oracle Names**
- **host naming**
- **external naming**

naming model

The set and structure of domains within which names can be allocated.

In a flat naming model, there is a single domain.

In a hierarchical naming model, the highest level is the root domain, and all other domains are hierarchically related.

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 username and password along with a net service name in a connect string for the service to which they wish 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
- Oracle Names server
- External naming service, such as **NIS** or **CDS**

network

A group of two or more computers linked together through hardware and software to allow the sharing of data or 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 upper- 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 or a Oracle Names server.

network protocol

See [Oracle protocol support](#).

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 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

Network Interface

NIS

See [Network Information Service \(NIS\)](#).

NN

Network Naming (Oracle Names)

node

A computer or terminal that is part of a network

NPI

See [Network Program Interface \(NPI\)](#).

NR

Network Routing

NS

Network Session. See [Network Session \(NS\)](#).

NT

Network Transport. See [transport](#).

O**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 to that entry the object classes that hold those attributes.

All objects associated with the same object class share the attributes of that object class.

OCI

Oracle Call Interface. See [Oracle Call Interface \(OCI\)](#).

OPI

See [Oracle Program Interface \(OPI\)](#).

Open Systems Interconnection (OSI)

A model of network architecture developed by ISO as a framework for international standards in heterogeneous computer network architecture.

The OSI architecture is split between 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

Each layer uses the layer immediately following it and provides a service to the layer preceding.

Oracle Advanced Security

A product that provides a comprehensive suite of security features to protect enterprise networks and securely extend 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 datatypes, 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 then take advantage of the **session multiplexing**, **access control**, or **protocol conversion** features configured on that Oracle Connection Manager.

Oracle Connection Manager Control utility

A utility included with Oracle Network Services to control various functions, such as starting, stopping, and getting the status of the Oracle Connection Manager.

Oracle Context

A **RDN** of `cn=OracleContext` in a **directory information tree (DIT)** that is located under a **naming context** or an unpublished directory entry. An 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 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 Names

An Oracle directory service made up of a system of Oracle Names servers that provide name-to-address resolution for each service on the network.

Oracle Names Control utility

A utility included with Oracle Names to control various functions for Oracle Names servers, such as starting, stopping, and getting the status of an Oracle Names server.

Oracle Names LDAP Proxy server

Oracle Names servers that have been configured to proxy for **directory servers**. Upon startup, Oracle Names LDAP Proxy servers obtain network object information from a directory server. This provides a single point of definition for all data in a directory server and does not require that both Oracle Names servers and directory servers be maintained separately and simultaneously.

Oracle Names server

A computer that uses Oracle Names software to store network address information for a service along with its simple name so that client applications can request connections with simple names rather than lengthy addresses.

Oracle Net

Communication software that enables a network session from a client application to an Oracle database server. Once 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 is able to perform these jobs because it is located on each computer in the network.

Oracle Net Configuration Assistant

A post-installation tool that configures basic network components after installation, including:

- Listener names and protocol addresses
- Naming methods the client will use 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 graphical user interface tool that combines configuration abilities with component control to provide an integrated environment for configuring and managing Oracle Net Services. It can be used on either the client or server.

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, centralized LDAP-compliant directory server, or an Oracle Names server.
- Naming Methods
Configure the different 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 once 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 pre-release 8.1 Oracle database. For any database, there is at least one instance referencing the database.

For pre-release 8.1 databases, 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_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)**.

P

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: connect, accept, redirect, data, and so on. Packet information can be useful in troubleshooting.

PMON process

A process monitor database process that performs process recovery when a user process fails. PMON is responsible for cleaning up 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.

preferred Oracle Names server

The Oracle Names servers preferred by a client for names resolution; usually the Oracle Names Server that is physically closest to the client, or available over the least expensive network link.

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 his or her exclusive use.

See also **database link**, **global 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**, **Oracle Connection Manager**, or **Oracle Names server**, 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 to 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 the 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](#), [global database link](#), and [private database link](#).

R

RDBMS

Relational Database Management System

RDN

See [relative distinguished name \(RDN\)](#).

relative distinguished name (RDN)

The local, most granular level entry name. It has no other qualifying entry names that would serve to address the entry uniquely. In the example, `cn=sales,dc=us,dc=acme,dc=com`, `cn=sales` is the RDN.

region

See [administrative region](#).

region database

Tables in an Oracle database that store Oracle Names information.

root administrative region

The highest level administrative region in a distributed installation. The root administrative region contains the root domain.

root domain

The highest level domain in a hierarchical naming model.

RPC

Remote Procedure Call

S

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

Work done for others. 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 comprised of the database name and domain name, entered during installation or database creation. If you are not sure what the global database name is, 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 (**dispatchers** or **dedicated servers**) 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, the listener can spawn a dedicated server for the connection.

service replication

A process that fully replicates a directory system on the network. New services need to register with only one **Oracle Names server**. The service replication process automatically distributes the new registration to all other active Oracle Names servers on the network.

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 dialogue 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 in order 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 version 8.0 Oracle databases, as Oracle9*i* and Oracle8*i* instance information is automatically registered with the listener. Static configuration is also required for other services, such as [external procedure](#) calls and [Heterogeneous Services](#), and some management tools, including Oracle Enterprise Manager.

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
- Preferred Oracle Names servers
- External naming parameters
- Oracle Advanced Security parameters

The `sqlnet.ora` file typically resides in `$ORACLE_HOME/network/admin` on UNIX and `ORACLE_HOME\network\admin` on Windows operating systems.

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](#).

system or topology data

Data used by the Oracle Names server to control regular functioning or communicate with other Oracle Names servers. Includes interchanges, Oracle Names servers in a root region, and Oracle Names servers in any delegated region.

T

TCP/IP protocol

Transmission Control Protocol/Internet Protocol. The de facto 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 protocol** 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

Thin JDBC Driver

Thin JDBC driver is Oracle's Type 4 driver designed for Java applet and Java application developers. The JDBC driver establishes a direct connection to the Oracle database server over Java sockets. Access to the database is assisted with a lightweight implementation of Oracle Net and **Two-Task Common (TTC)**.

TNS

See **Transparent Network Substrate (TNS)**.

tnsnames.ora file

A configuration file that contains one or more **domains** mapped to **connect descriptors**. This file is used for the **local naming** method. The `tnsnames.ora` file typically resides in `$ORACLE_HOME/network/admin` on UNIX and `ORACLE_HOME\network\admin`.

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 executed. Administrators use the trace facility for diagnosing an abnormal condition; it is not normally turned on.

See also [logging](#).

Transparent Application Failover (TAF)

A runtime failover for high-availability environments, such as Oracle9i 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.

U

UPI

User Program Interface

V

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.

W

well-known Oracle Names server

Addresses for one or more Oracle Names servers hard coded into both the Oracle Names server and its clients. Oracle Names servers then become available at these well known addresses, so that clients do not need to be told, by way of configuration files, where to find the server.

Windows NT native authentication

An **authentication method** that enables a client single login access to a Windows NT server and a database running on the server.

Index

Symbols

- " (quotation mark) symbol
 - reserved in configuration files, 4-4
- # (quotation mark) symbol
 - reserved in configuration files, 4-4
- () (parenthesis) symbol
 - reserved in configuration files, 4-4
- = (equals sign) symbol
 - reserved in configuration files, 4-4
- ' (single quote) symbol
 - reserved in configuration files, 4-4
- / (slash) symbol
 - reserved in configuration files, 4-4

Numerics

- 1024 port, 5-6
- 1521 port, 5-5
- 1575 port, 5-5, 10-3
- 1630 port, 5-5
- 1646 port, 6-35
- 1830 port, 5-5
- 2482 port, 5-5
- 2484 port, 5-5

A

- ACT networking parameter, 9-5
- ADDRESS networking parameter, 5-2, 7-8, 8-4
- ADDRESS_LIST networking parameter, 5-2, 7-9
- ADMIN_RESTRICTIONS_listener_name networking parameter, 8-11
- ANSWER_TIMEOUT networking parameter, 9-8

A.SMD Oracle Names record, 3-31, 3-42

attributes

- orclDescList, A-5
- orclDescName, A-5
- orclLoadBalance, A-5
- orclNetAddrList, A-5
- orclNetAddrString, A-5
- orclNetConnParamList, A-5
- orclNetFailover, A-5
- orclNetInstanceName, A-5
- orclNetProtocol, A-5
- orclNetSdu, A-5
- orclNetServiceName, A-5
- orclNetSourceRoute, A-5
- orclSid, A-5
- orclVersion, A-5

AUTHENTICATION_LEVEL networking parameter, 9-8

AUTOMATIC_IPC networking parameter, B-4

B

BACKUP networking parameter, 7-16

BASE networking parameter, 10-6

C

CDS. See Cell Directory Services (CDS)

Cell Directory Services (CDS) external naming configuring, 6-8

CHANGE_PASSWORD command, 1-4, 1-5

character sets

- for net service name, 4-4
- network, for keyword values, 4-4

- client load balancing
 - configuring, 7-11
 - with Oracle Connection Manager, 7-5
- CLOSE_RELAY command, 2-6
- CMAN networking parameter, 9-3
- CMAN_ADMIN networking parameter, 9-4
- CMAN_PROFILE networking parameter, 9-6
- CMAN_RULES networking parameter, 9-4
- cmn.ora file
 - allowed values of parameters, 9-7
 - configuration parameter reference, 9-3 to 9-12
 - default values of parameters, 9-6
 - parameters
 - ACT, 9-5
 - ANSWER_TIMEOUT, 9-8
 - AUTHENTICATION_LEVEL, 9-8
 - CMAN, 9-3
 - CMAN_ADMIN, 9-4
 - CMAN_PROFILE, 9-6
 - CMAN_RULES, 9-4
 - DST, 9-5
 - LOG_LEVEL, 9-8
 - MAX_FREELIST_BUFFERS, 9-8
 - MAXIMUM_CONNECT_DATA, 9-9
 - MAXIMUM_RELAYS, 9-9
 - RELAY_STATISTICS, 9-9
 - REMOTE_ADMIN, 9-10
 - SHOW_TNS_INFO, 9-10
 - SRC, 9-5
 - SRV, 9-5
 - TRACE_DIRECTORY, 9-11
 - TRACE_FILELEN, 9-11
 - TRACE_TIMESTAMP, 9-11
 - TRACING, 9-10
 - USE_ASYNC_CALL, 9-12
- CNAME.SMD Oracle Names record, 3-31, 3-42
- comments in configuration files, 4-3
- COMMUNITY networking parameter, B-4
- configuring
 - protocol addresses, 5-1
- connect descriptors, 7-2
- CONNECT_DATA networking parameter, 7-15
- CONNECT_TIMEOUT_listener_name networking parameter, B-4

- connections
 - adjusting listener queue size to avoid errors, 8-5
- connect-time failover
 - configuring, 7-10
 - with Oracle Connection Manager, 7-5
- control utilities
 - Listener Control utility, 1-5 to 1-36
 - Oracle Connection Manager Control utility, 2-5 to 2-33
 - Oracle Names Control utility, 3-2 to 3-95

D

- DBSNMP_START command, B-6
- DBSNMP_STATUS command, B-6
- DBSNMP_STOP command, B-6
- DEFAULT_ADMIN_CONTEXT networking parameter, 11-3
- DELAY networking parameter, 7-17
- DELEGATE_DOMAIN command, 3-6
- DESCRIPTION networking parameter, 7-7, 8-4, 10-5
- DESCRIPTION_LIST networking parameter, 7-8
- directory naming
 - configuring, 6-8
 - exporting
 - aliases from Oracle Names to net service aliases in a directory, 3-9
 - objects from Oracle Names, 3-13
 - migrating
 - aliases from Oracle Names to net service aliases in a directory, 3-9
 - objects from Oracle Names, 3-13
- DIRECTORY_SERVER_TYPE networking parameter, 11-3
- DIRECTORY_SERVERS networking parameter, 11-2
- DISABLE_OOB networking parameter, 6-4
- DLCR.RDBMS.OMD Oracle Names record, 3-31, 3-42
- DL.RDBMS.OMD Oracle Names record, 3-31, 3-42
- DOMAIN networking parameter, 10-13
- DOMAIN_HINT command, 3-7, B-6
- DOMAIN_LIST networking parameter, 10-15
- DST networking parameter, 9-5

DUMP_ALIAS command, 3-9
DUMP_LDAP command, 3-13
DUMP_TNSNAMES command, 3-17

E

ENVS networking parameter, 8-7
error messages
 NNL-00005, 3-4
 NNL-00406, 3-4
EXIT command
 of Listener Control utility, 1-7
 of Oracle Connection Manager Control utility, 2-7
 of Oracle Names Control utility, 3-18
EXPIRE networking parameter, 10-5
exporting
 aliases from Oracle Names to net service aliases in a directory, 3-9
 objects from Oracle Names to a directory server, 3-13
external naming
 Cell Directory Services (CDS), 6-8
 Network Information Service (NIS), 6-8

F

failover
 connect-time, 7-10
 Transparent Application Failover, 7-16
FAILOVER networking parameter, 7-10, 7-11, 10-17, 10-18, 10-20, 10-23
FAILOVER_MODE networking parameter, 7-16
FLUSH command, 3-19
FLUSH_NAME command, 3-20

G

global database name
 configuring on the listener, 8-9
 described, 8-9
GLOBAL_DBNAME networking parameter, 8-9
GLOBAL_NAME networking parameter, 7-17

H

HELP command
 of Listener Control utility, 1-8
 of Oracle Connection Manager Control utility, 2-8
 of Oracle Names Control utility, 3-21
HINT_LIST networking parameter, 10-11, 10-15
host naming
 configuring, 6-8
HOST networking parameter, 5-3, 5-4, 10-5
HS networking parameter, 7-18

I

Identix authentication, B-2
INSTANCE_NAME networking parameter, 7-18
IPC protocol
 KEY parameter, 5-3
 PROTOCOL parameter, 5-3
IPC, parameters for addresses, 5-3

K

KEY networking parameter, 5-3
keyword syntax rules, for configuration files, 4-3
keyword values, network character sets for, 4-4

L

LDAP schema
 attributes, A-1 to A-5
 object classes, A-1 to A-5
ldap.ora file
 configuration parameter reference, 11-2 to 11-3
 DEFAULT_ADMIN_CONTEXT parameter, 11-3
 DIRECTORY_SERVER_TYPE parameter, 11-3
 DIRECTORY_SERVERS parameter, 11-2
LIST_DELEGATED command, 3-23
LIST_DOMAINS command, 3-24
LIST_OBJECTS command, 3-25

- Listener Control utility, 1-4
 - command reference, 1-5 to 1-36
 - commands
 - CHANGE_PASSWORD, 1-5
 - EXIT, 1-7
 - HELP, 1-8
 - QUIT, 1-9
 - RELOAD, 1-10
 - SAVE_CONFIG, 1-11
 - SERVICES, 1-12
 - SET, 1-14
 - SET CONNECT_TIMEOUT, 1-16
 - SET CURRENT_LISTENER, 1-16
 - SET DISPLAYMODE, 1-17
 - SET LOG_DIRECTORY, 1-18
 - SET LOG_FILE, 1-19
 - SET LOG_STATUS, 1-20
 - SET PASSWORD, 1-21
 - SET STARTUP_WAITTIME, 1-23
 - SET TRC_DIRECTORY, 1-24
 - SET TRC_FILE, 1-25
 - SET TRC_LEVEL, 1-26
 - SET USE_PLUGANDPLAY, 1-27
 - SHOW, 1-27
 - SHOW CURRENT_LISTENER, 1-28
 - SHOW DISPLAYMODE, 1-28
 - SHOW LOG_DIRECTORY, 1-28
 - SHOW LOG_FILE, 1-28
 - SHOW LOG_STATUS, 1-28
 - SHOW RAWMODE, 1-28
 - SHOW SAVE_CONFIG_ON_STOP, 1-28
 - SHOW STARTUP_WAITTIME, 1-28
 - SHOW TRC_DIRECTORY, 1-28
 - SHOW TRC_FILE, 1-28
 - SHOW TRC_LEVEL, 1-28
 - SPAWN, 1-29
 - START, 1-30
 - STATUS, 1-32
 - STOP, 1-34
 - TRACE, 1-35
 - VERSION, 1-36
 - distributed operation, 1-4
 - function of and syntax format, 1-2
 - remote administration, 1-4
 - SET commands, 1-3
 - SET CONNECT_TIMEOUT, B-6
 - SHOW commands, 1-3
 - SHOW CONNECT_TIMEOUT, B-6
 - unsupported commands
 - DBSNMP_START, B-6
 - DBSNMP_STATUS, B-6
 - DBSNMP_STOP, B-6
 - SET USE_PLUGANDPLAY, B-6
 - SHOW USE_PLUGANDPLAY, B-6
- listener.ora file
 - configuration parameter reference, 8-3 to 8-20
 - parameters
 - ADDRESS, 8-4
 - ADMIN_RESTRICTIONS_listener_name, 8-11
 - DESCRIPTION, 8-4
 - ENVS, 8-7
 - GLOBAL_DBNAME, 8-9
 - LOG_DIRECTORY_listener_name, 8-12
 - LOG_FILE_listener_name, 8-12
 - LOGGING_listener_name, 8-13
 - ORACLE_HOME, 8-9
 - PASSWORDS_listener_name, 1-4, 8-13
 - PRESPAWN_DESC, 8-11
 - PRESPAWN_MAX, 8-11
 - PROGRAM, 8-10
 - QUEUESIZE, 8-5
 - SAVE_CONFIG_ON_STOP_listener_name, 8-14
 - SID_DESC, 8-7
 - SID_LIST_listener_name, 8-6, 8-11
 - SID_NAME, 8-10
 - SQLNET.CLIENT_
 - AUTHENTICATION, 8-14
 - STARTUP_WAITTIME_listener_name, 8-15
 - TRACE_DIRECTORY_listener_name, 8-15
 - TRACE_FILE_listener_name, 8-16
 - TRACE_FILEN_listener_name, 8-16
 - TRACE_FILENO_listener_name, 8-17
 - TRACE_LEVEL_listener_name, 8-17
 - TRACE_TIMESTAMP_listener_name, 8-18
 - WALLET_LOCATION, 8-18
 - unsupported parameters
 - CONNECT_TIMEOUT_listener_name, B-4
 - PRESPAWN_DESC, B-5
 - PRESPAWN_LIST, B-5
 - PRESPAWN_MAX, B-5
 - USE_PLUG_AND_PLAY_listener_name, B-5

- listeners
 - adjusting queue size for, 8-5
 - multiple, 8-2
- load balancing
 - client, 7-11
- LOAD_BALANCE networking parameter, 7-11
- LOAD_TNSNAMES command, 3-26
- local naming
 - configuring, 6-8
- LOG_DIRECTORY_CLIENT networking parameter, 6-4
- LOG_DIRECTORY_listener_name networking parameter, 8-12
- LOG_DIRECTORY_SERVER networking parameter, 6-5
- LOG_FILE_CLIENT networking parameter, 6-5
- LOG_FILE_listener_name networking parameter, 8-12
- LOG_FILE_SERVER networking parameter, 6-5
- LOG_LEVEL networking parameter, 9-8
- LOG_STATS command, 3-28
- LOGGING_listener_name networking parameter, 8-13
- LU6.2 protocol, B-3

M

- MAX_FREELIST_BUFFERS networking parameter, 9-8
- MAXIMUM_CONNECT_DATA networking parameter, 9-9
- MAXIMUM_RELAYS networking parameter, 9-9
- METHOD networking parameter, 7-17
- migrating
 - aliases from Oracle Names to net service aliases in a directory, 3-9
 - objects from Oracle Names to a directory server, 3-13
- MIN_TTL networking parameter, 10-13
- multiple listeners, 8-2

N

- Named Pipes protocol
 - parameters for addresses, 5-3
 - PIPE parameter, 5-3
 - PROTOCOL parameter, 5-3
 - SERVER parameter, 5-3
- NAMES.ADDRESSES networking parameter, 10-3
- NAMES.ADMIN_REGION networking parameter, 10-3, 10-11
- NAMES.AUTHORITY_REQUIRED networking parameter, 10-7
- NAMES.AUTO_REFRESH_EXPIRE networking parameter, 10-7
- NAMES.AUTO_REFRESH_RETRY networking parameter, 10-8
- NAMES.CACHE_CHECKPOINT_FILE networking parameter, 10-8
- NAMES.CACHE_CHECKPOINT_INTERVAL networking parameter, 10-9
- NAMES.CONFIG_CHECKPOINT_FILE networking parameter, 10-9
- NAMES.CONNECT_TIMEOUT networking parameter, 6-6, 10-10
- NAMESCTL.ECHO networking parameter, 6-12
- NAMESCTL.INTERNAL_ENCRYPT_PASSWORD networking parameter, 6-13
- NAMESCTL.INTERNAL_USE networking parameter, 6-13
- NAMESCTL.NO_INITIAL_SERVER networking parameter, 6-14
- NAMESCTL.NOCONFIRM networking parameter, 6-14
- NAMESCTL.SERVER_PASSWORD networking parameter, 6-15
- NAMESCTL.TRACE_DIRECTORY networking parameter, 6-16
- NAMESCTL.TRACE_FILE networking parameter, 6-16
- NAMESCTL.TRACE_LEVEL networking parameter, 6-15
- NAMESCTL.TRACE_UNIQUE networking parameter, 6-17
- NAMES.DCE.PREFIX networking parameter, 6-6

NAMES.DEFAULT_DOMAIN networking parameter, B-4
 NAMES.DEFAULT_FORWARDERS networking parameter, 10-11
 NAMES.DEFAULT_FORWARDERS_ONLY networking parameter, 10-12
 NAMES.DEFAULT_ZONE networking parameter, B-4
 NAMES.DEFAULT.DOMAIN networking parameter, 6-7
 NAMES.DIRECTORY_PATH networking parameter, 6-7
 cds, 6-8
 hostname, 6-8
 ldap, 6-8
 nis, 6-8
 onames, 6-8
 tnsnames, 6-8
 NAMES.DOMAIN_CHECKPOINT_FILE networking parameter, 10-24
 NAMES.DOMAIN_HINTS networking parameter, 10-15
 NAMES.DOMAINS networking parameter, 10-13
 NAMES.FORWARDING_AVAILABLE networking parameter, 10-17
 NAMES.FORWARDING_DESIRED networking parameter, 10-17
 NAMES.INITIAL_RETRY_TIMEOUT networking parameter, 6-9
 NAMES.KEEP_DB_OPEN networking parameter, 10-18
 NAMES.LOG_DIRECTORY networking parameter, 10-18
 NAMES.LOG_FILE networking parameter, 10-19
 NAMES.LOG_STATS_INTERVAL networking parameter, 10-19
 NAMES.MAX_OPEN_CONNECTIONS networking parameter, 6-9, 10-20
 NAMES.MAX_REFORWARDS networking parameter, 10-21
 NAMES.MESSAGE_POOL_START_SIZE networking parameter, 6-10, 10-21
 NAMES.NDS.NAME.CONTEXT networking parameter, B-4
 NAMES.NIS.META_MAP networking parameter, 6-10
 NAMES.NO_MODIFY_RESPONSE networking parameter, 10-22
 NAMES.NO_REGION_DATABASE networking parameter, 10-23
 names.ora file
 configuration parameter reference, 10-3 to 10-29
 parameters
 BASE, 10-6
 DESCRIPTION, 10-5
 DOMAIN, 10-13
 DOMAIN_LIST, 10-15
 EXPIRE, 10-5
 HINT_LIST, 10-11, 10-15
 HOST, 10-5
 MIN_TTL, 10-13
 NAMES.ADDRESSES, 10-3
 NAMES.ADMIN_REGION, 10-3, 10-11
 NAMES.AUTHORITY_REQUIRED, 10-7
 NAMES.AUTO_REFRESH_EXPIRE, 10-7
 NAMES.AUTO_REFRESH_RETRY, 10-8
 NAMES.CACHE_CHECKPOINT_FILE, 10-8
 NAMES.CACHE_CHECKPOINT_INTERVAL, 10-9
 NAMES.CONFIG_CHECKPOINT_FILE, 10-9
 NAMES.CONNECT_TIMEOUT, 10-10
 NAMES.DEFAULT_FORWARDERS, 10-11
 NAMES.DEFAULT_FORWARDERS_ONLY, 10-12
 NAMES.DOMAIN_CHECKPOINT_FILE, 10-24
 NAMES.DOMAIN_HINTS, 10-15
 NAMES.DOMAINS, 10-13
 NAMES.FORWARDING_AVAILABLE, 10-17
 NAMES.FORWARDING_DESIRED, 10-17
 NAMES.KEEP_DB_OPEN, 10-18
 NAMES.LOG_DIRECTORY, 10-18
 NAMES.LOG_FILE, 10-19
 NAMES.LOG_STATS_INTERVAL, 10-19
 NAMES.MAX_OPEN_CONNECTIONS, 10-20
 NAMES.MAX_REFORWARDS, 10-21

NAMES.MESSAGE_POOL_START_SIZE, 10-21
 NAMES.NO_MODIFY_REQUESTS, 10-22
 NAMES.NO_REGION_DATABASE, 10-23
 NAMES.PASSWORD, 10-23
 NAMES.RESET_STATS_INTERVAL, 10-24
 NAMES.SAVE_CONFIG_ON_STOP, 10-24
 NAMES.SERVER_NAME, 10-25
 NAMES.TOPOLOGY_CHECKPOINT_FILE, 10-26
 NAMES.TRACE_DIRECTORY, 10-26
 NAMES.TRACE_FILE, 10-27
 NAMES.TRACE_FUNC, 10-27
 NAMES.TRACE_LEVEL, 10-28
 NAMES.TRACE_UNIQUE, 10-28
 PASSWORD, 10-5
 PORT, 10-5
 REFRESH, 10-5
 RETRY, 10-5
 SCOPE, 10-6
 SUBTREE, 10-6
 SUBTREE_LIST, 10-6
 TIMEOUT, 10-5
 TYPE, 10-5
 USERID, 10-5
 VERSION, 10-5
 unsupported parameters
 NAMES.USE_PLUG_AND_PLAY, B-5
 NAMES.PASSWORDS networking
 parameter, 10-23
 NAMES.PREFERRED_SERVERS networking
 parameter, 6-11
 NAMES.REQUEST_RETRIES networking
 parameter, 6-11
 NAMES.RESET_STATS_INTERVAL networking
 parameter, 10-24
 NAMES.SAVE_CONFIG_ON_STOP networking
 parameter, 10-24
 NAMES.SERVER_NAMES networking
 parameter, 10-25
 NAMES.TOPOLOGY_CHECKPOINT_FILE
 networking parameter, 10-26
 NAMES.TRACE_DIRECTORY networking
 parameter, 10-26
 NAMES.TRACE_FILE networking
 parameter, 10-27
 NAMES.TRACE_FUNC networking
 parameter, 10-27
 NAMES.TRACE_LEVEL networking
 parameter, 10-28
 NAMES.TRACE_UNIQUE networking
 parameter, 10-28
 NAMES.USE_PLUG_AND_PLAY networking
 parameter, B-5
 NDS. See Novell Directory Services (NDS)
 net service aliases
 exporting aliases from Oracle Names, 3-9
 migrating aliases from Oracle Names, 3-9
 Net8 OPEN, B-2
 network character sets, keyword values, 4-4
 network configuration files
 cman.ora, 9-3 to 9-12
 ldap.ora, 11-2 to 11-3
 listener.ora, 8-3 to 8-20
 names.ora, 10-3 to 10-29
 protocol addresses, 5-1
 sqlnet.ora, 6-3 to 6-52
 syntax rules, 4-2
 tnsnames.ora, 7-7 to 7-24
 Network Information Service external naming
 configuring, 6-8
 network performance, improving
 client load balancing, 7-11
 networking parameters
 cman.ora configuration reference, 9-3 to 9-12
 ldap.ora configuration reference, 11-2 to 11-3
 listener.ora configuration reference, 8-3 to 8-20
 names.ora configuration reference, 10-3 to 10-29
 sqlnet.ora configuration reference, 6-3 to 6-52
 tnsnames.ora configuration
 reference, 7-7 to 7-24
 >NNL-00005 error message, 3-4
 >NNL-00406 error message, 3-4
 Novell Directory Services (NDS)
 authentication, B-2
 Novell Directory Services (NDS) external
 naming, B-2
 NS.SMD Oracle Names record, 3-31, 3-42

O

object classes

- orclDBServer, A-2
- orclNetAddress, A-4
- orclNetAddressList, A-4
- orclNetDescription, A-3
- orclNetDescriptionList, A-4
- orclNetService, A-2
- orclNetServiceAlias, A-3

obsolete parameters, B-4

Oracle Connection Manager

- client load balancing, 7-5
- connect-time failover, 7-5
- SOURCE_ROUTE networking parameter, 7-13

Oracle Connection Manager Control utility

- command reference, 2-5 to 2-33

commands

- CLOSE_RELAY, 2-6
- EXIT, 2-7
- HELP, 2-8
- QUIT, 2-9
- SET, 2-10
- SET AUTHENTICATION_LEVEL, 2-11
- SET DISPLAYMODE, 2-12
- SET LOG_LEVEL, 2-13
- SET RELAY_STATISTICS, 2-14
- SHOW, 2-15
- SHOW ADDRESS, 2-16
- SHOW ALL, 2-17
- SHOW DISPLAYMODE, 2-19
- SHOW PROFILE, 2-20
- SHOW RELAY, 2-21
- SHOW RULES, 2-23
- SHUTDOWN, 2-24
- START, 2-25
- STATS, 2-27
- STATUS, 2-29
- STOP, 2-31
- STOPNOW, 2-32
- VERSION, 2-33

- distributed operation, 2-4

- remote administration, 2-4

- SET commands, 2-3

Oracle Names

- configuring, 6-8

exporting

- aliases to net service aliases in a directory server, 3-9
- objects to a directory server, 3-13

migrating

- aliases to net service aliases in a directory server, 3-9
- objects to a directory server, 3-13

- .sdns.ora file, 3-2

- sdns.ora file, 3-2

Oracle Names Control utility

- command reference, 3-2 to 3-95

commands

- DELEGATE_DOMAIN, 3-6
- DOMAIN_HINT, 3-7
- DUMP_ALIAS, 3-9
- DUMP_LDAP, 3-13
- DUMP_TNSNAMES, 3-17
- EXIT, 3-18
- FLUSH, 3-19
- FLUSH_NAME, 3-20
- HELP, 3-21
- LIST_DELEGATED, 3-23
- LIST_DOMAINS, 3-24
- LIST_OBJECTS, 3-25
- LOAD_TNSNAMES, 3-26
- LOG_STATS, 3-28
- PASSWORD, 3-29
- PING, 3-30
- QUERY, 3-31
- QUIT, 3-33
- REGISTER, 3-34
- REGISTER_NS, 3-37
- RELOAD, 3-39
- REORDER_NS, 3-40
- REPEAT, 3-42
- RESET_STATS, 3-44
- RESTART, 3-45
- SAVE_CONFIG, 3-46
- SET, 3-47
- SET CACHE_CHECKPOINT_INTERVAL, 3-48
- SET DEFAULT_DOMAIN, 3-49

- SET FORWARDING_AVAILABLE, 3-50
- SET LOG_FILE_NAME, 3-52
- SET LOG_STATS_INTERVAL, 3-53
- SET NAMESCTL_TRACE_LEVEL, 3-54
- SET PASSWORD, 3-56
- SET REQUESTS_ENABLED, 3-57
- SET RESET_STATS_INTERVAL, 3-58
- SET SAVE_CONFIG_INTERVAL, 3-59
- SET SAVE_CONFIG_ON_STOP, 3-60
- SET SERVER, 3-61
- SET TRACE_FILE_NAME, 3-62
- SET TRACE_LEVEL, 3-63
- SHOW, 3-64
- SHOW CACHE CHECKPOINT_INTERVAL, 3-66
- SHOW DEFAULT_DOMAIN, 3-67
- SHOW FORWARDING_AVAILABLE, 3-68
- SHOW LOG_FILE_NAME, 3-69
- SHOW LOG_STATS_INTERVAL, 3-70
- SHOW NAMESCTL_TRACE_LEVEL, 3-71
- SHOW REQUEST_ENABLED, 3-72
- SHOW RESETS_STATS_INTERVAL, 3-73
- SHOW SAVE_CONFIG_INTERVAL, 3-74
- SHOW SAVE_CONFIG_ON_STOP, 3-75
- SHOW SERVER, 3-76
- SHOW STATS, 3-77
- SHOW SYSTEM_QUERIES, 3-78
- SHOW TRACE_FILE_NAME, 3-79
- SHOW TRACE_LEVEL, 3-80
- SHOW VERSION, 3-81
- SHUTDOWN, 3-82
- START, 3-83
- START_CLIENT_CACHE, 3-85
- STARTUP, 3-86
- STATUS, 3-87
- STOP, 3-88
- TIMED_QUERY, 3-89
- UNREGISTER, 3-91
- UNREGISTER_NS command, 3-93
- VERSION, 3-95
- confirmation mode, 3-6
- distributed operation, 3-3
- function of and syntax format, 3-2
- remote administration, 3-3
- security, 3-5
- SET commands, 3-3
- SHOW commands, 3-3
- unsupported commands
 - DOMAIN_HINT, B-6
- Oracle protocol support
 - configuring addresses, 5-3
 - IPC, 5-3
 - Named Pipes, 5-3
 - TCP/IP, 5-3
 - TCP/IP with SSL, 5-4
- Oracle schema
 - attributes, A-1 to A-5
 - object classes, A-1 to A-5
- ORACLE_HOME networking parameter, 8-9
- Oracle9i Real Application Clusters
 - connect-time failover, 7-10, 7-11, 10-17, 10-18, 10-20, 10-23
 - FAILOVER networking parameter, 7-10, 7-11, 10-17, 10-18, 10-20, 10-23
 - FAILOVER_MODE networking parameter, 7-16
 - INSTANCE_NAME networking parameter, 7-19
 - LOAD_BALANCE networking parameter, 7-11
- orclDBServer object class, A-2
- orclDescList attribute, A-5
- orclDescName attribute, A-5
- orclLoadBalance attribute, A-5
- orclNetAddress object class, A-4
- orclNetAddressList object class, A-4
- orclNetAddrList attribute, A-5
- orclNetAddrString attribute, A-5
- orclNetConnParamList attribute, A-5
- orclNetDescription object class, A-3
- orclNetDescriptionList object class, A-4
- orclNetFailover attribute, A-5
- orclNetInstanceName attribute, A-5
- orclNetSdu attribute, A-5
- orclNetService object class, A-2
- orclNetServiceAlias object class, A-3
- orclNetServiceName attribute, A-5
- orclNetSourceRoute attribute, A-5
- orclProtocol attribute, A-5
- orclSid attribute, A-5
- orclVersion attribute, A-5

OSS.MY.WALLET networking parameter, B-4

P

password

- for Listener Control utility access, 1-4
- for Oracle Names Control utility access, 3-5

PASSWORD command, 3-29

PASSWORD networking parameter, 10-5

PASSWORDS_listener_name networking parameter, 8-13

PASSWORDS_listener_name parameter, 1-4

PING utility, 3-30

PIPE networking parameter, 5-3

port 1024, 5-6

port 1521, 5-5

port 1575, 5-5, 10-3

port 1630, 5-5

port 1646, 6-35

port 1830, 5-5

port 2483, 5-5

port 2484, 5-5

PORT networking parameter, 5-3, 5-4, 10-5

port numbers, allowed, 5-6

ports

- privileged, 5-6

PRESPAWN_DESC networking parameter, 8-11, B-5

PRESPAWN_LIST networking parameter, B-5

PRESPAWN_MAX networking parameter, 8-11, B-5

prespawned dedicated servers, B-3

privileged ports, 5-6

PROGRAM networking parameter, 8-10

PROTOCOL networking parameter, 5-3, 5-4

protocol.ora file, B-3

protocols, 5-3

- configuring addresses, 5-3

- IPC, 5-3

- Named Pipes, 5-3

- TCP/IP, 5-3

- TCP/IP with SSL, 5-4

Q

QUERY command

- A.SMD Oracle Names record, 3-31, 3-42

- CNAME.SMD Oracle Names record, 3-31, 3-42

- DLCR.TDBMS.OMD Oracle Names

 - record, 3-31, 3-42

- DL.RDBMS.OMD Oracle Names record, 3-31, 3-42

- NS.SMD Oracle Names record, 3-31, 3-42

- of Oracle Names Control utility, reference, 3-31

- V1ADD.NPO.OMD Oracle Names record, 3-31, 3-42

QUEUESIZE networking parameter, 8-5

QUEUESIZE parameter

- for adjusting listener queue size, 8-5

QUIT command

- of Listener Control utility, 1-9

- of Oracle Connection Manager Control utility, 2-9

- of Oracle Names Control utility, 3-33

R

randomizing requests among listeners, 7-11

RDB_DATABASE networking parameter, 7-19

reference

- for cman.ora, 9-3 to 9-12

- for ldap.ora, 11-2 to 11-3

- for Listener Control utility

 - commands, 1-5 to 1-36

- for listener.ora, 8-3 to 8-20

- for names.ora, 10-3 to 10-29

- for Oracle Connection Manager Control utility

 - commands, 2-5 to 2-33

- for Oracle Names Control utility

 - commands, 3-2 to 3-95

- for sqlnet.ora, 6-3 to 6-52

- for tnsnames.ora, 7-7 to 7-24

REFRESH networking parameter, 10-5

REGISTER command, 3-34

REGISTER_NS command, 3-37

RELAY_STATISTICS networking parameter, 9-9

RELOAD command

- of Listener Control utility, 1-10

- of Oracle Names Control utility, 3-39

REMOTE_ADMIN networking parameter, 9-10
REORDER_NS command, 3-40
REPEAT command, 3-42
RESET_STATS command, 3-44
RESTART command, 3-45
RETRIES networking parameter, 7-17
RETRY networking parameter, 10-5
rules, syntax for network configuration files, 4-2

S

SAVE_CONFIG command
 of Listener Control utility, 1-11
 of Oracle Names Control utility, 3-46
SAVE_CONFIG_ON_STOP_listener_name
 networking parameter, 8-14
SCOPE networking parameter, 10-6
.sdns.ora file, 3-2
sdns.ora file, 3-2
SDU networking parameter, 7-12
SecurID authentication, B-2
security, 1-4
 Listener Control utility, 1-4
 Oracle Names Control utility, 3-5
 restricting runtime administration of the
 listener, 8-11
SECURITY networking parameter, 7-23
SERVER networking parameter, 5-3, 7-20
service name
 character set keyword values, 4-4
SERVICE_NAME networking parameter, 7-21
SERVICES command, 1-12
SET AUTHENTICATION_LEVEL command, 2-11
SET CACHE_CHECKPOINT_INTERVAL
 command, 3-48
SET command
 of Listener Control utility, 1-14
 of Oracle Connection Manager Control
 utility, 2-10
 of Oracle Names Control utility, 3-47
SET CONNECT_TIMEOUT command, 1-16, B-6
SET CURRENT_LISTENER command, 1-16
SET DEFAULT_DOMAIN command, 3-49
SET DISPLAYMODE command
 of Listener Control utility, 1-17
 of Oracle Connection Manager Control
 utility, 2-12
SET FORWARDING_AVAILABLE command, 3-50
SET LOG_DIRECTORY command
 of Listener Control utility, 1-18
SET LOG_FILE command, 1-19
SET LOG_FILE_NAME command, 3-52
SET LOG_LEVEL command, 2-13
SET LOG_STATS_INTERVAL command, 3-53
SET LOG_STATUS command, 1-20
SET NAMESCTL_TRACE_LEVEL command, 3-54
SET PASSWORD command
 of Listener Control utility, 1-21
 of Oracle Names Control utility, 3-56
SET RELAY_STATISTICS command, 2-14
SET REQUESTS_ENABLED command, 3-57
SET RESET_STATS_INTERVAL command, 3-58
SET SAVE_CONFIG_INTERVAL command, 3-59
SET SAVE_CONFIG_ON_STOP command, 1-22
 of Listener Control utility, 1-22
 of Oracle Names Control utility, 3-60
SET SERVER command, 3-61
SET STARTUP_WAITTIME command, 1-23
SET TRACE_FILE_NAME command, 3-62
SET TRACE_LEVEL command, 3-63
SET TRC_DIRECTORY command, 1-24
SET TRC_FILE command, 1-25
SET TRC_LEVEL command, 1-26
SET USE_PLUGANDPLAY command, 1-27, B-6
SHOW ADDRESS command, 2-16
SHOW ALL command, 2-17
SHOW CACHE_CHECKPOINT_INTERVAL
 command, 3-66
SHOW command
 of Listener Control utility, 1-27
 of Oracle Connection Manager Control
 utility, 2-15
 of Oracle Names Control utility, 3-64
SHOW CONNECT_TIMEOUT command, B-6
SHOW CURRENT_LISTENER command, 1-28
SHOW DEFAULT_DOMAIN command, 3-67

SHOW DISPLAYMODE command
 of Listener Control utility, 1-28
 of Oracle Connection Manager Control utility, 2-19
 SHOW FORWARDING_AVAILABLE command, 3-68
 SHOW LOG_DIRECTORY command, 1-28
 SHOW LOG_FILE command, 1-28
 SHOW LOG_FILE_NAME command, 3-69
 SHOW LOG_STATS_INTERVAL command, 3-70
 SHOW LOG_STATS command, 1-28
 SHOW NAMESCTL_TRACE_LEVEL command, 3-71
 SHOW PROFILE command, 2-20
 SHOW RAWMODE command, 1-28
 SHOW RELAY command, 2-21
 SHOW REQUESTS_ENABLED command, 3-72
 SHOW RESETS_STATS_INTERVAL command, 3-73
 SHOW RULES command, 2-23
 SHOW SAVE_CONFIG_INTERVAL command, 3-74
 SHOW SAVE_CONFIG_ON_STOP command, 1-28, 3-75
 SHOW SERVER command, 3-76
 SHOW STARTUP_WAITTIME command, 1-28
 SHOW STATUS command, 3-77
 SHOW SYSTEM_QUERIES command, 3-78
 SHOW TRACE_FILE_NAME command, 3-79
 SHOW TRACE_LEVEL command, 3-80
 SHOW TRC_DIRECTORY command, 1-28
 SHOW TRC_FILE command, 1-28
 SHOW TRC_LEVEL command, 1-28
 SHOW VERSION command, 3-81
 SHOW_TNS_INFO networking parameter, 9-10
 SHUTDOWN command
 of Oracle Connection Manager Control utility, 2-24
 of Oracle Names Control utility, 3-82
 SID networking parameter, 7-22
 SID_DESC networking parameter, 8-7
 SID_LIST_listener_name networking parameter, 8-11
 Oracle Enterprise Manager requirements, 8-6
 SID_NAME networking parameter, 8-10
 SOURCE_ROUTE networking parameter, 7-13
 SPAWN command, 1-29
 SPX protocol, B-3
 SQLNET.ALTERNATE_PORT networking parameter, 6-30
 SQLNET.AUTHENTICATION_GSSAPI_SERVICE networking parameter, 6-18
 SQLNET.AUTHENTICATION_KERBEROS5_SERVICE networking parameter, 6-17
 SQLNET.AUTHENTICATION_SERVICES networking parameter, 6-18
 SQLNET.CLIENT_AUTHENTICATION networking parameter, 8-14
 SQLNET.CLIENT_REGISTRATION networking parameter, 6-19
 SQLNET.CRYPTO_CHECKSUM_CLIENT networking parameter, 6-19
 SQLNET.CRYPTO_CHECKSUM_SERVER networking parameter, 6-20
 SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT networking parameter, 6-21
 SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER networking parameter, 6-21
 SQLNET.CRYPTO_SEED networking parameter, 6-22
 SQLNET.ENCRYPTION_CLIENT networking parameter, 6-22
 SQLNET.ENCRYPTION_SERVER networking parameter, 6-23
 SQLNET.ENCRYPTION_TYPES_CLIENT networking parameter, 6-24
 SQLNET.ENCRYPTION_TYPES_SERVER networking parameter, 6-25
 SQLNET.EXPIRE_TIME networking parameter, 6-26
 SQLNET.IDENTIX_FINGERPRINT_DATABASE networking parameter, B-4
 SQLNET.IDENTIX_FINGERPRINT_DATABASE_PASSWORD networking parameter, B-4
 SQLNET.IDENTIX_FINGERPRINT_DATABASE_USER networking parameter, B-4
 SQLNET.IDENTIX_FINGERPRINT_METHOD networking parameter, B-4
 SQLNET.KERBEROS5_CC_NAME networking parameter, 6-27

SQLNET.KERBEROS5_CLOCKSKEW networking parameter, 6-27

SQLNET.KERBEROS5_CONF networking parameter, 6-28

SQLNET.KERBEROS5_KEYTAB networking parameter, 6-28

SQLNET.KERBEROS5_REALMS networking parameter, 6-29

sqlnet.ora file

- configuration parameter reference, 6-3 to 6-52
- parameters
 - DISABLE_OOB, 6-4
 - LOG_DIRECTORY_CLIENT, 6-4
 - LOG_DIRECTORY_SERVER, 6-5
 - LOG_FILE_CLIENT, 6-5
 - LOG_FILE_SERVER, 6-5
 - NAMES_DIRECTORY_PATH, 6-7
 - NAMES.CONNECT_TIMEOUT, 6-6
 - NAMESCTL.ECHO, 6-12
 - NAMESCTL.INTERNAL_ENCRYPT_PASSWORD, 6-13
 - NAMESCTL.INTERNAL_USE, 6-13
 - NAMESCTL.NO_INITIAL_SERVER, 6-14
 - NAMESCTL.NOCONFIRM, 6-14
 - NAMESCTL.SERVER_PASSWORD, 6-15
 - NAMESCTL.TRACE_DIRECTORY, 6-16
 - NAMESCTL.TRACE_FILE, 6-16
 - NAMESCTL.TRACE_LEVEL, 6-15
 - NAMESCTL.TRACE_UNIQUE, 6-17
 - NAMES.DCE_PREFIX, 6-6
 - NAMES.DEFAULT_DOMAIN, 6-7, B-4
 - NAMES.INITIAL_RETRY_TIMEOUT, 6-9
 - NAMES.MAX_OPEN_CONNECTIONS, 6-9
 - NAMES.MESSAGE_POOL_START_SIZE, 6-10
 - NAMES.NIS.META_MAP, 6-10
 - NAMES.PREFERRED_SERVERS, 6-11
 - NAMES.REQUEST_RETRIES, 6-11
 - SQLNET.ALTERNATE_PORT, 6-30
 - SQLNET.AUTHENTICATION_GSSAPI_SERVICE, 6-18
 - SQLNET.AUTHENTICATION_KERBEROS5_SERVICE, 6-17
 - SQLNET.AUTHENTICATION_SERVICES, 6-18
 - SQLNET.CLIENT_REGISTRATION, 6-19
 - SQLNET.CRYPTO_CHECKSUM_CLIENT, 6-19
 - SQLNET.CRYPTO_CHECKSUM_SERVER, 6-20
 - SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT, 6-21
 - SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER, 6-21
 - SQLNET.CRYPTO_SEED, 6-22
 - SQLNET.ENCRYPTION_SERVER, 6-23
 - SQLNET.ENCRYPTION_TYPES_CLIENT, 6-24
 - SQLNET.ENCRYPTION_TYPES_SERVER, 6-25
 - SQLNET.ENCRYPTION_CLIENT, 6-22
 - SQLNET.EXPIRE_TIME, 6-26
 - SQLNET.IDENTIX_FINGERPRINT_DATABASE_PASSWORD, B-4
 - SQLNET.KERBEROS5_CC_NAME, 6-27
 - SQLNET.KERBEROS5_CLOCKSKEW, 6-27
 - SQLNET.KERBEROS5_CONF, 6-28
 - SQLNET.KERBEROS5_KEYTAB, 6-28
 - SQLNET.KERBEROS5_REALMS, 6-29
 - SQLNET.RADIUS_ALTERNATE_RETRIES, 6-30
 - SQLNET.RADIUS_ALTERNATE_RETRIES, 6-31
 - SQLNET.RADIUS_AUTHENTICATION, 6-31
 - SQLNET.RADIUS_AUTHENTICATION_INTERFACE, 6-32
 - SQLNET.RADIUS_AUTHENTICATION_PORT, 6-32
 - SQLNET.RADIUS_AUTHENTICATION_RETRIES, 6-33
 - SQLNET.RADIUS_AUTHENTICATION_TIMEOUT, 6-33
 - SQLNET.RADIUS_CHALLENGE_RESPONSE, 6-34
 - SQLNET.RADIUS_SECRET, 6-34
 - SQLNET.RADIUS_SEND_ACCOUNTING, 6-35
 - SSL_CIPHER_SUITES, 6-35
 - SSL_SERVER_DN_MATCH, 6-36
 - SSL_VERSION, 6-37

SSL.CLIENT_AUTHENTICATION, 6-36
 TCP.EXCLUDED_NODES, 6-38
 TCP.INVITED_NODES, 6-38
 TCP.NODELAY, 6-39
 TCP.VALIDNODE_CHECKING, 6-39
 TNSPING.TRACE_DIRECTORY, 6-40
 TNSPING.TRACE_LEVEL, 6-40
 TRACE_DIRECTORY_CLIENT, 6-41
 TRACE_DIRECTORY_SERVER, 6-41
 TRACE_FILE_CLIENT, 6-42
 TRACE_FILE_SERVER, 6-42
 TRACE_FILEN_CLIENT, 6-43
 TRACE_FILENO_CLIENT, 6-44
 TRACE_FILENO_SERVER, 6-44
 TRACE_LEVEL_CLIENT, 6-45
 TRACE_LEVEL_SERVER, 6-46
 TRACE_TIMESTAMP_CLIENT, 6-46
 TRACE_TIMESTAMP_SERVER, 6-47
 TRACE_UNIQUE_CLIENT, 6-47
 USE_CMAN, 6-48
 USE_DEDICATED_SERVER, 6-49
 WALLET_LOCATION, 6-50, B-4
 unsupported parameters
 AUTOMATIC_IPC, B-4
 NAMES.DEFAULT_ZONE, B-4
 NAMES.NDS.NAME.CONTEXT, B-4
 OSS.MY.WALLET, B-4
 SQLNET.IDENTIX_FINGERPRINT_
 DATABASE, B-4
 SQLNET.IDENTIX_FINGERPRINT_
 DATABASE_PASSWORD, B-4
 SQLNET.IDENTIX_FINGERPRINT_
 DATABASE_USER, B-4
 SQLNET.IDENTIX_FINGERPRINT_
 METHOD, B-4
 SQLNET.RADIUS_ALTERNATE networking
 parameter, 6-30
 SQLNET.RADIUS_ALTERNATE_RETRIES
 networking parameter, 6-31
 SQLNET.RADIUS_AUTHENTICATION
 networking parameter, 6-31
 SQLNET.RADIUS_AUTHENTICATION_
 INTERFACE networking parameter, 6-32
 SQLNET.RADIUS_AUTHENTICATION_PORT
 networking parameter, 6-32
 SQLNET.RADIUS_AUTHENTICATION_RETRIES
 networking parameter, 6-33
 SQLNET.RADIUS_AUTHENTICATION_TIMEOUT
 networking parameter, 6-33
 SQLNET.RADIUS_CHALLENGE_RESPONSE
 networking parameter, 6-34
 SQLNET.RADIUS_SECRET networking
 parameter, 6-34
 SQLNET.RADIUS_SEND_ACCOUNTING
 networking parameter, 6-35
 SRC networking parameter, 9-5
 SRV networking parameter, 9-5
 SSL_CIPHER_SUITES networking parameter, 6-35
 SSL_SERVER_CERT_DN networking
 parameter, 7-24
 SSL_SERVER_DN_MATCH networking
 parameter, 6-36
 SSL_VERSION networking parameter, 6-37
 SSL.CLIENT_AUTHENTICATION networking
 parameter, 6-36
 START command
 of Listener Control utility, 1-30
 of Oracle Connection Manager Control
 utility, 2-25
 of Oracle Names Control utility, 3-83
 START_CLIENT_CACHE command, 3-85
 STARTUP command
 of Oracle Names Control utility, 3-86
 STARTUP_WAITTIME_listener_name networking
 parameter, 8-15
 STATS command, 2-27
 ACTIVE_DELAYS, 2-27
 MOST_RELAYS, 2-28
 OUT_OF_RELAYS, 2-28
 TOTAL_REFUSED, 2-28
 TOTAL_RELAYS, 2-27
 STATUS command
 of Listener Control utility, 1-32
 of Oracle Connection Manager Control
 utility, 2-29
 of Oracle Names Control utility, 3-87

- STOP command
 - of Listener Control utility, 1-34
 - of Oracle Connection Manager Control utility, 2-31
 - of Oracle Names Control utility, 3-88
- STOPNOW command
 - of Oracle Connection Manager Control utility, 2-32
- SUBTREE parameter, 10-6
- SUBTREE_LIST networking parameter, 10-6
- syntax
 - rules for network configuration files, 4-2

T

- TAF
 - see Transparent Application Failover (TAF)
- TCP.EXCLUDED_NODES networking parameter, 6-38
- TCP.INVITED_NODES networking parameter, 6-38
- TCP/IP protocol
 - HOST parameter, 5-3
 - parameters for addresses, 5-3
 - PORT parameter, 5-3
 - PROTOCOL parameter, 5-3
- TCP/IP with SSL protocol
 - HOST parameter, 5-4
 - parameters for addresses, 5-4
 - PORT parameter, 5-4
 - PROTOCOL parameter, 5-4
- TCP.NODELAY networking parameter, 6-39
- TCP.VALIDNODE_CHECKING networking parameter, 6-39
- terminated connection detection
 - limitations, 6-26
 - SQLNET.EXPIRE_TIME parameter, 6-26
- TIMED_QUERY command, 3-89
- TIMEOUT networking parameter, 10-5
- time-stamping
 - client trace files, 6-46
 - database server trace files, 6-47
 - listener trace files, 8-18

- tnsnames.ora file
 - configuration parameter reference, 7-7 to 7-24
 - parameters
 - ADDRESS, 7-8
 - ADDRESS_LIST, 7-9
 - BACKUP, 7-16
 - CONNECT_DATA, 7-15
 - DELAY, 7-17
 - DESCRIPTION, 7-7
 - DESCRIPTION_LIST, 7-8
 - FAILOVER, 7-10, 7-11, 10-17, 10-18, 10-20, 10-23
 - FAILOVER_MODE, 7-16
 - GLOBAL_NAME, 7-17
 - HS, 7-18
 - INSTANCE_NAME, 7-18
 - LOAD_BALANCE, 7-11
 - METHOD, 7-17
 - RDB_DATABASE, 7-19
 - RETRIES, 7-17
 - SDU, 7-12
 - SECURITY, 7-23
 - SERVER, 7-20
 - SERVICE_NAME, 7-21
 - SID, 7-22
 - SOURCE_ROUTE, 7-13
 - SSL_SERVER_CERT_DN, 7-24
 - TYPE, 7-16
 - TYPE_OF_SERVICE, 7-14
 - unsupported parameters
 - COMMUNITY, B-4
- TNSPING.TRACE_DIRECTORY networking parameter, 6-40
- TNSPING.TRACE_LEVEL networking parameter, 6-40
- TRACE command, 1-35
- trace files
 - cycling
 - client, 6-44
 - database server, 6-44
 - listener, 8-17
 - time-stamping
 - client, 6-46
 - database server, 6-47
 - listener, 8-18
- TRACE_DIRECTORY networking parameter, 9-11

TRACE_DIRECTORY_CLIENT networking
parameter, 6-41

TRACE_DIRECTORY_listener_name networking
parameter, 8-15

TRACE_DIRECTORY_SERVER networking
parameter, 6-41

TRACE_FILE_CLIENT networking
parameter, 6-42

TRACE_FILE_listener_name networking
parameter, 8-16

TRACE_FILE_SERVER networking
parameter, 6-42

TRACE_FILELEN networking parameter, 9-11

TRACE_FILELEN_CLIENT networking
parameter, 6-43

TRACE_FILEN_listener_name networking
parameter, 8-16

TRACE_FILENO_CLIENT networking
parameter, 6-44

TRACE_FILENO_listener_name networking
parameter, 8-17

TRACE_FILENO_SERVER networking
parameter, 6-44

TRACE_LEVEL_CLIENT networking
parameter, 6-45

TRACE_LEVEL_listener_name networking
parameter, 8-17

TRACE_LEVEL_SERVER networking
parameter, 6-46

TRACE_TIMESTAMP networking parameter, 9-11

TRACE_TIMESTAMP_CLIENT networking
parameter, 6-46

TRACE_TIMESTAMP_listener_name networking
parameter, 8-18

TRACE_TIMESTAMP_SERVER networking
parameter, 6-47

TRACE_UNIQUE_CLIENT networking
parameter, 6-47

tracing
cycling files
client, 6-44
database server, 6-44
listener, 8-17

time-stamping
client trace files, 6-46
database server trace files, 6-47
listener trace files, 8-18

TRACING networking parameter, 9-10

Transparent Application Failover (TAF)
parameters, 7-16

TYPE networking parameter, 7-16, 10-5

TYPE_OF_SERVICE networking parameter, 7-14

U

UNREGISTER command, 3-91

UNREGISTER_NS command, 3-93

USE_ASYNC_CALL networking parameter, 9-12

USE_CMAN networking parameter, 6-48

USE_DEDICATED_SERVER networking
parameter, 6-49

USE_PLUG_AND_PLAY_listener_name
parameter, B-5

USERID networking parameter, 10-5

V

V1ADD.NPO.OMD Oracle Names record, 3-31,
3-42

VERSION command
of Listener Control utility, 1-36
of Oracle Connection Manager Control
utility, 2-33
of Oracle Names Control utility, 3-95

VERSION networking parameter, 10-5

W

WALLET_LOCATION networking
parameter, 6-50, 8-18, B-4