Contents

Preface

Audience xiv
Documentation Accessibility xiv
Related Documents xiv
Conventions xv

1 Listener Control Utility

1.1 Listener Control Utility Overview 1-1
1.2 SET and SHOW Listener Control Utility Commands 1-2
1.3 Understanding Distributed Operations 1-2
1.4 Understanding Oracle Net Listener Security 1-3
1.5 Listener Control Utility Commands 1-3
  1.5.1 EXIT 1-5
  1.5.2 HELP 1-5
  1.5.3 QUIT 1-6
  1.5.4 RELOAD 1-7
  1.5.5 SAVE_CONFIG 1-8
  1.5.6 SERVICES 1-9
  1.5.7 SET 1-10
  1.5.8 SET CURRENT_LISTENER 1-11
  1.5.9 SET DISPLAYMODE 1-12
  1.5.10 SET INBOUND_CONNECT_TIMEOUT 1-12
  1.5.11 SET LOG_DIRECTORY 1-13
  1.5.12 SET LOG_FILE 1-14
  1.5.13 SET LOG_STATUS 1-15
  1.5.14 SET SAVE_CONFIG_ON_STOP 1-16
  1.5.15 SET TRC_DIRECTORY 1-16
  1.5.16 SET TRC_FILE 1-17
  1.5.17 SET TRC_LEVEL 1-18
  1.5.18 SHOW 1-19
  1.5.19 SPAWN 1-20
  1.5.20 START 1-21
2 Oracle Connection Manager Control Utility

2.1 Connection Manager Control Utility Command Modes and Syntax 2-1
2.2 Oracle Connection Manager Control Utility Overview 2-3
2.3 Oracle Connection Manager Control Utility Commands 2-3
  2.3.1 ADMINISTER 2-6
  2.3.2 CLOSE CONNECTIONS 2-7
  2.3.3 EXIT 2-9
  2.3.4 HELP 2-9
  2.3.5 QUIT 2-10
  2.3.6 RELOAD 2-11
  2.3.7 RESUME GATEWAYS 2-12
  2.3.8 SAVE_PASSWD 2-12
  2.3.9 SET 2-13
  2.3.10 SET ASO_AUTHENTICATION_FILTER 2-14
  2.3.11 SET CONNECTION_STATISTICS 2-15
  2.3.12 SET EVENT 2-16
  2.3.13 SET IDLE_TIMEOUT 2-17
  2.3.14 SET INBOUND_CONNECT_TIMEOUT 2-17
  2.3.15 SET LOG_DIRECTORY 2-18
  2.3.16 SET LOG_LEVEL 2-19
  2.3.17 SET OUTBOUND_CONNECT_TIMEOUT 2-20
  2.3.18 SET PASSWORD 2-21
  2.3.19 SET SESSION_TIMEOUT 2-22
  2.3.20 SET TRACE_DIRECTORY 2-23
  2.3.21 SET TRACE_LEVEL 2-23
  2.3.22 SHOW 2-24
  2.3.23 SHOW ALL 2-25
  2.3.24 SHOW CONNECTIONS 2-26
  2.3.25 SHOW DEFAULTS 2-30
  2.3.26 SHOW EVENTS 2-31
  2.3.27 SHOW GATEWAYS 2-31
  2.3.28 SHOW PARAMETERS 2-32
  2.3.29 SHOW RULES 2-33
  2.3.30 SHOW SERVICES 2-35
  2.3.31 SHOW STATUS 2-36
3 Syntax Rules for Configuration Files

3.1 Overview of Configuration File Syntax 3-1
3.2 Syntax Rules for Configuration Files 3-2
3.3 Network Character Set for Keywords 3-3
3.4 Permitted Listener and Net Service Name Character Set 3-3

4 Protocol Address Configuration

4.1 Protocol Addresses 4-1
4.1.1 ADDRESS 4-1
4.1.2 ADDRESS_LIST 4-2
4.2 Protocol Parameters 4-2
4.3 Recommended Port Numbers 4-4
4.4 Port Number Limitations 4-4

5 Parameters for sqlnet.ora Files

5.1 Overview of Profile Configuration Files 5-1
5.2 Profile Parameters in sqlnet.ora Files 5-2
5.2.1 ACCEPT_MD5_CERTS 5-8
5.2.2 ACCEPT_SHA1_CERTS 5-9
5.2.3 ADD_SSLV3_TO_DEFAULT 5-9
5.2.4 EXADIRECT_FLOW_CONTROL 5-9
5.2.5 EXADIRECT_RECVPOL 5-10
5.2.6 DEFAULT_SDU_SIZE 5-10
5.2.7 DISABLE_OOB 5-11
5.2.8 DISABLE_OOB_AUTO 5-11
5.2.9 HTTPS_SSL_VERSION 5-12
5.2.10 IPC.KEYPATH 5-12
5.2.11 NAMES.DEFAULT_DOMAIN 5-13
5.2.12 NAMES.DIRECTORY_PATH 5-13
5.2.13 NAMES.LDAP_AUTHENTICATE_BIND 5-14
5.2.14 NAMES.LDAP_CONN_TIMEOUT 5-14
5.2.15 NAMES.LDAP_PERSISTENT_SESSION 5-15
5.2.16 NAMES.NIS.META_MAP 5-15
5.2.17 RECV_BUF_SIZE 5-16
<p>| 5.2.18 | SDP.PF_INET_SDP | 5-17 |
| 5.2.19 | SEC_USER_AUDIT_ACTION_BANNER | 5-17 |
| 5.2.20 | SEC_USER_UNAUTHORIZED_ACCESS_BANNER | 5-18 |
| 5.2.21 | SEND_BUF_SIZE | 5-18 |
| 5.2.22 | SQLNET.ALLOWED_LOGON_VERSION_CLIENT | 5-19 |
| 5.2.23 | SQLNET.ALLOWED_LOGON_VERSION_SERVER | 5-20 |
| 5.2.24 | SQLNET.AUTHENTICATION_SERVICES | 5-25 |
| 5.2.25 | SQLNET.CLIENT_REGISTRATION | 5-27 |
| 5.2.26 | SQLNET.CLOUD_USER | 5-27 |
| 5.2.27 | SQLNET.COMPRESSION | 5-29 |
| 5.2.28 | SQLNET.COMPRESSION_ACCELERATION | 5-29 |
| 5.2.29 | SQLNET.COMPRESSION_LEVELS | 5-30 |
| 5.2.30 | SQLNET.COMPRESSION_THRESHOLD | 5-30 |
| 5.2.31 | SQLNET.CRYPTO_CHECKSUM_CLIENT | 5-31 |
| 5.2.32 | SQLNET.CRYPTO_CHECKSUM_SERVER | 5-31 |
| 5.2.33 | SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT | 5-32 |
| 5.2.34 | SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER | 5-33 |
| 5.2.35 | SQLNET.DBFW_PUBLIC_KEY | 5-33 |
| 5.2.36 | SQLNET.DOWN_HOSTS_TIMEOUT | 5-34 |
| 5.2.37 | SQLNET.ENCRYPTION_CLIENT | 5-34 |
| 5.2.38 | SQLNET.ENCRYPTION_SERVER | 5-35 |
| 5.2.39 | SQLNET.ENCRYPTION_TYPES_CLIENT | 5-36 |
| 5.2.40 | SQLNET.ENCRYPTION_TYPES_SERVER | 5-37 |
| 5.2.41 | SQLNET.EXPIRE_TIME | 5-38 |
| 5.2.42 | SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS | 5-39 |
| 5.2.43 | SQLNET.INBOUND_CONNECT_TIMEOUT | 5-39 |
| 5.2.44 | SQLNET.FALLBACK_AUTHENTICATION | 5-40 |
| 5.2.45 | SQLNET.KERBEROS5_CC_NAME | 5-40 |
| 5.2.46 | SQLNET.KERBEROS5_CLOCKSKEW | 5-41 |
| 5.2.47 | SQLNET.KERBEROS5_CONF | 5-42 |
| 5.2.48 | SQLNET.KERBEROS5_CONF_LOCATION | 5-42 |
| 5.2.49 | SQLNET.KERBEROS5_KEYTAB | 5-43 |
| 5.2.50 | SQLNET.KERBEROS5_REALMS | 5-43 |
| 5.2.51 | SQLNET.KERBEROS5_REPLAY_CACHE | 5-44 |
| 5.2.52 | SQLNET.OUTBOUND_CONNECT_TIMEOUT | 5-44 |
| 5.2.53 | SQLNET.RADIUS_ALTERNATE | 5-45 |
| 5.2.54 | SQLNET.RADIUS_ALTERNATE_PORT | 5-46 |
| 5.2.55 | SQLNET.RADIUS_ALTERNATE RETRIES | 5-46 |
| 5.2.56 | SQLNET.RADIUS_AUTHENTICATION | 5-47 |
| 5.2.57 | SQLNET.RADIUS_AUTHENTICATION_INTERFACE | 5-47 |
| 5.2.58 | SQLNET.RADIUS_AUTHENTICATION_PORT | 5-48 |</p>
<table>
<thead>
<tr>
<th>Section</th>
<th>Parameter Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2.59</td>
<td>SQLNET.RADIUS_AUTHENTICATION_RETRIES</td>
<td>5-48</td>
</tr>
<tr>
<td>5.2.60</td>
<td>SQLNET.RADIUS_AUTHENTICATION_TIMEOUT</td>
<td>5-49</td>
</tr>
<tr>
<td>5.2.61</td>
<td>SQLNET.RADIUS_CHALLENGE_RESPONSE</td>
<td>5-49</td>
</tr>
<tr>
<td>5.2.62</td>
<td>SQLNET.RADIUS_SECRET</td>
<td>5-50</td>
</tr>
<tr>
<td>5.2.63</td>
<td>SQLNET.RADIUS_SEND_ACCOUNTING</td>
<td>5-50</td>
</tr>
<tr>
<td>5.2.64</td>
<td>SQLNET.RECV_TIMEOUT</td>
<td>5-51</td>
</tr>
<tr>
<td>5.2.65</td>
<td>SQLNET.SEND_TIMEOUT</td>
<td>5-52</td>
</tr>
<tr>
<td>5.2.66</td>
<td>SQLNET.URI</td>
<td>5-53</td>
</tr>
<tr>
<td>5.2.67</td>
<td>SQLNET.USE_HTTPS_PROXY</td>
<td>5-53</td>
</tr>
<tr>
<td>5.2.68</td>
<td>SQLNET.WALLET_OVERRIDE</td>
<td>5-53</td>
</tr>
<tr>
<td>5.2.69</td>
<td>SSL_CERT_REVOCATION</td>
<td>5-54</td>
</tr>
<tr>
<td>5.2.70</td>
<td>SSL_CRL_FILE</td>
<td>5-55</td>
</tr>
<tr>
<td>5.2.71</td>
<td>SSL_CRL_PATH</td>
<td>5-56</td>
</tr>
<tr>
<td>5.2.72</td>
<td>SSL_CIPHER_SUITES</td>
<td>5-56</td>
</tr>
<tr>
<td>5.2.73</td>
<td>SSL_EXTENDED_KEY_USAGE</td>
<td>5-57</td>
</tr>
<tr>
<td>5.2.74</td>
<td>SSL_SERVER_DN_MATCH</td>
<td>5-58</td>
</tr>
<tr>
<td>5.2.75</td>
<td>SSL_VERSION</td>
<td>5-59</td>
</tr>
<tr>
<td>5.2.76</td>
<td>TCP.CONNECT_TIMEOUT</td>
<td>5-60</td>
</tr>
<tr>
<td>5.2.77</td>
<td>TCP.EXCLUDED_NODES</td>
<td>5-60</td>
</tr>
<tr>
<td>5.2.78</td>
<td>TCP.INVITED_NODES</td>
<td>5-61</td>
</tr>
<tr>
<td>5.2.79</td>
<td>TCP.NODENAME</td>
<td>5-61</td>
</tr>
<tr>
<td>5.2.80</td>
<td>TCP.QUEUESIZE</td>
<td>5-61</td>
</tr>
<tr>
<td>5.2.81</td>
<td>TCP.VALIDNODE_CHECKING</td>
<td>5-62</td>
</tr>
<tr>
<td>5.2.82</td>
<td>TNSPING.TRACE_DIRECTORY</td>
<td>5-63</td>
</tr>
<tr>
<td>5.2.83</td>
<td>TNSPING.TRACE_LEVEL</td>
<td>5-63</td>
</tr>
<tr>
<td>5.2.84</td>
<td>USE_CMAN</td>
<td>5-63</td>
</tr>
<tr>
<td>5.2.85</td>
<td>USE_DEDICATED_SERVER</td>
<td>5-64</td>
</tr>
<tr>
<td>5.2.86</td>
<td>WALLET_LOCATION</td>
<td>5-65</td>
</tr>
<tr>
<td>5.2.87</td>
<td>BEQUEATH_DETACH</td>
<td>5-67</td>
</tr>
</tbody>
</table>

5.3 ADR Diagnostic Parameters in sqlnet.ora

5.3.1 About ADR Diagnostic Parameters

5.3.2 ADR_BASE

5.3.3 DIAG_ADR_ENABLED

5.3.4 TRACE_LEVEL_CLIENT

5.3.5 TRACE_LEVEL_SERVER

5.3.6 TRACE_TIMESTAMP_CLIENT

5.3.7 TRACE_TIMESTAMP_SERVER

5.4 Non-ADR Diagnostic Parameters in sqlnet.ora Files

5.4.1 LOG_DIRECTORY_CLIENT

5.4.2 LOG_DIRECTORY_SERVER

5.4.3 LOG_FILE_CLIENT
Local Naming Parameters in the tnsnames.ora File

6.1 Overview of Local Naming Parameters 6-1
6.2 General Syntax of tnsnames.ora 6-2
6.3 Using Multiple Descriptions in tnsnames.ora Files 6-3
6.4 Multiple Address Lists in tnsnames.ora Files 6-3
6.5 Connect-Time Failover and Client Load Balancing with Oracle Connection Managers 6-4
6.6 Connect Descriptor Descriptions 6-5
   6.6.1 DESCRIPTION_LIST 6-6
   6.6.2 DESCRIPTION 6-6
6.7 Protocol Addresses 6-6
   6.7.1 ADDRESS 6-7
   6.7.2 HTTPS_PROXY 6-7
   6.7.3 HTTPS_PROXY_PORT 6-8
   6.7.4 ADDRESS_LIST 6-9
6.8 Optional Parameters for Address Lists 6-9
   6.8.1 ENABLE 6-10
   6.8.2 EXPIRE_TIME 6-10
   6.8.3 FAILOVER 6-11
   6.8.4 LOAD_BALANCE 6-12
   6.8.5 RECV_BUF_SIZE 6-12
   6.8.6 SDU 6-13
   6.8.7 SEND_BUF_SIZE 6-14
   6.8.8 SOURCE_ROUTE 6-15
   6.8.9 TYPE_OF_SERVICE 6-16
   6.8.10 ENVS Parameter 6-16
6.9 Connection Data Section 6-17
   6.9.1 CONNECT_DATA 6-18
6.9.2  FAILOVER_MODE  6-19
6.9.3  GLOBAL_NAME  6-20
6.9.4  HS  6-20
6.9.5  INSTANCE_NAME  6-21
6.9.6  RDB_DATABASE  6-21
6.9.7  SHARDING_KEY  6-22
6.9.8  SUPER_SHARDING_KEY  6-22
6.9.9  SERVER  6-23
6.9.10  SERVICE_NAME  6-23
6.9.11  COLOCATION_TAG  6-24

6.10  Security Section  6-25
6.10.1  SECURITY  6-25
6.10.2  SSL_SERVER_CERT_DN  6-25
6.10.3  IGNORE_ANO_ENCRYPTION_FOR_TCPS  6-26
6.10.4  WALLET_LOCATION Parameter  6-27

6.11  Timeout Parameters  6-28
6.11.1  CONNECT_TIMEOUT  6-28
6.11.2  RETRY_COUNT  6-29
6.11.3  RETRY_DELAY  6-29
6.11.4  TRANSPORT_CONNECT_TIMEOUT  6-30

6.12  Compression Parameters  6-30
6.12.1  COMPRESSION  6-31
6.12.2  COMPRESSION_LEVELS  6-31

7  Oracle Net Listener Parameters in the listener.ora File

7.1  Overview of Oracle Net Listener Configuration File  7-1
7.2  Protocol Address Parameters  7-3
  7.2.1  ADDRESS  7-3
  7.2.2  DESCRIPTION  7-4
  7.2.3  Firewall  7-4
  7.2.4  IP  7-4
  7.2.5  QUEUESIZE  7-5
  7.2.6  RECV_BUF_SIZE  7-5
  7.2.7  SEND_BUF_SIZE  7-6
7.3  Connection Rate Limiter Parameters  7-7
  7.3.1  CONNECTION_RATE_listener name  7-8
  7.3.2  RATE_LIMIT  7-8
7.4  Control Parameters  7-9
  7.4.1  ADMIN_RESTRICTIONS_listener_name  7-10
  7.4.2  ALLOW_MULTIPLE_REDIRECTS_listener_name  7-11
7.4.3 ENABLE_EXADIRECT_listener_name 7-12
7.4.4 CRS_NOTIFICATION_listener_name 7-12
7.4.5 DEDICATED_THROUGH_BROKER_LISTENER 7-13
7.4.6 DEFAULT_SERVICE_listener_name 7-13
7.4.7 INBOUND_CONNECT_TIMEOUT_listener_name 7-14
7.4.8 LOCAL_REGISTRATION_ADDRESS_listener_name 7-14
7.4.9 MAX_ALL_CONNECTIONS_listener_name 7-15
7.4.10 MAX_REG_CONNECTIONS_listener_name 7-15
7.4.11 REGISTRATION_EXCLUDED_NODES_listener_name 7-16
7.4.12 REGISTRATION_INVITED_NODES_listener_name 7-16
7.4.13 REMOTE_REGISTRATION_ADDRESS_listener_name 7-17
7.4.14 SAVE_CONFIG_ON_STOP_listener_name 7-17
7.4.15 SERVICE_RATE_listener_name 7-18
7.4.16 SSL_CLIENT_AUTHENTICATION 7-18
7.4.17 SSL_VERSION 7-19
7.4.18 SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name 7-19
7.4.19 USE_SID_AS_SERVICE_listener_name 7-20
7.4.20 VALID_NODE_CHECKING_REGISTRATION_listener_name 7-20
7.4.21 WALLET_LOCATION 7-21

7.5 ADR Diagnostic Parameters for Oracle Net Listener 7-23
7.5.1 ADR_BASE_listener_name 7-24
7.5.2 DIAG_ADR_ENABLED_listener_name 7-24
7.5.3 LOG_FILE_NUM_listener_name 7-25
7.5.4 LOG_FILE_SIZE_listener_name 7-25
7.5.5 LOGGING_listener_name 7-26
7.5.6 TRACE_LEVEL_listener_name 7-26
7.5.7 TRACE_TIMESTAMP_listener_name 7-27

7.6 Non-ADR Diagnostic Parameters for Oracle Net Listener 7-27
7.6.1 LOG_DIRECTORY_listener_name 7-27
7.6.2 LOG_FILE_listener_name 7-28
7.6.3 TRACE_DIRECTORY_listener_name 7-28
7.6.4 TRACE_FILE_listener_name 7-28
7.6.5 TRACE_FILEAGE_listener_name 7-29
7.6.6 TRACE_FILELEN_listener_name 7-29
7.6.7 TRACE_FILENO_listener_name 7-29

7.7 Class of Secure Transports Parameters 7-30
7.7.1 SECURE_REGISTER_listener_name 7-30
7.7.2 Using COST Parameters in Combination 7-31
7.7.3 DYNAMIC_REGISTRATION_listener_name 7-32
7.7.4 SECURE_PROTOCOL_listener_name 7-32
# Oracle Connection Manager Parameters

## 8

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1 Overview of Oracle Connection Manager Configuration File</td>
<td>8-1</td>
</tr>
<tr>
<td>8.2 Oracle Connection Manager Parameters</td>
<td>8-3</td>
</tr>
<tr>
<td>8.2.1 ADDRESS</td>
<td>8-5</td>
</tr>
<tr>
<td>8.2.2 ASO_AUTHENTICATION_FILTER</td>
<td>8-6</td>
</tr>
<tr>
<td>8.2.3 COMPRESSION</td>
<td>8-6</td>
</tr>
<tr>
<td>8.2.4 COMPRESSION_LEVELS</td>
<td>8-7</td>
</tr>
<tr>
<td>8.2.5 COMPRESSION_THRESHOLD</td>
<td>8-7</td>
</tr>
<tr>
<td>8.2.6 CONNECTION_STATISTICS</td>
<td>8-7</td>
</tr>
<tr>
<td>8.2.7 EVENT_GROUP</td>
<td>8-8</td>
</tr>
<tr>
<td>8.2.8 EXPIRE_TIME</td>
<td>8-9</td>
</tr>
<tr>
<td>8.2.9 IDLE_TIMEOUT</td>
<td>8-9</td>
</tr>
<tr>
<td>8.2.10 INBOUND_CONNECT_TIMEOUT</td>
<td>8-10</td>
</tr>
<tr>
<td>8.2.11 LOG_DIRECTORY</td>
<td>8-10</td>
</tr>
<tr>
<td>8.2.12 LOG_FILE_NUM</td>
<td>8-10</td>
</tr>
<tr>
<td>8.2.13 LOG_FILE_SIZE</td>
<td>8-11</td>
</tr>
<tr>
<td>8.2.14 LOG_LEVEL</td>
<td>8-11</td>
</tr>
<tr>
<td>8.2.15 MAX_ALL_CONNECTIONS</td>
<td>8-11</td>
</tr>
<tr>
<td>8.2.16 MAX_CMCTL_SESSIONS</td>
<td>8-12</td>
</tr>
<tr>
<td>8.2.17 MAX_CONNECTIONS</td>
<td>8-12</td>
</tr>
<tr>
<td>8.2.18 MAX_GATEWAY_PROCESSES</td>
<td>8-12</td>
</tr>
<tr>
<td>8.2.19 MAX_REG_CONNECTIONS</td>
<td>8-12</td>
</tr>
<tr>
<td>8.2.20 MIN_GATEWAY_PROCESSES</td>
<td>8-13</td>
</tr>
<tr>
<td>8.2.21 OUTBOUND_CONNECT_TIMEOUT</td>
<td>8-13</td>
</tr>
<tr>
<td>8.2.22 PASSWORD_instance_name</td>
<td>8-13</td>
</tr>
<tr>
<td>8.2.23 REGISTRATION_EXCLUDED_NODES</td>
<td>8-13</td>
</tr>
<tr>
<td>8.2.24 REGISTRATION_INVITED_NODES</td>
<td>8-14</td>
</tr>
<tr>
<td>8.2.25 RULE</td>
<td>8-14</td>
</tr>
<tr>
<td>8.2.26 GROUP Parameter</td>
<td>8-16</td>
</tr>
<tr>
<td>8.2.27 SDU</td>
<td>8-17</td>
</tr>
<tr>
<td>8.2.28 SERVICE_RATE</td>
<td>8-17</td>
</tr>
<tr>
<td>8.2.29 SESSION_TIMEOUT</td>
<td>8-18</td>
</tr>
<tr>
<td>8.2.30 TRACE_FILE</td>
<td>8-18</td>
</tr>
<tr>
<td>8.2.31 TRACE_FILELEN</td>
<td>8-18</td>
</tr>
<tr>
<td>8.2.32 TRACE_FILENO</td>
<td>8-18</td>
</tr>
<tr>
<td>8.2.33 TRACE_LEVEL</td>
<td>8-19</td>
</tr>
<tr>
<td>8.2.34 TRACE_TIMESTAMP</td>
<td>8-19</td>
</tr>
<tr>
<td>8.2.35 USE_SID_AS_SERVICE</td>
<td>8-19</td>
</tr>
</tbody>
</table>
8.2.36  VALID_NODE_CHECKING_REGISTRATION  8-20
8.2.37  WALLET_LOCATION  8-20
8.2.38  NEXT_HOP Parameter  8-23
8.2.39  REST_ADDRESS Parameter  8-23
8.2.40  MAX_BANDWIDTH_GROUP Parameter  8-24
8.2.41  BANDWIDTH Parameter  8-24

8.3  Oracle Connection Manager in Traffic Director Mode Parameters  8-24

8.3.1  TDM  8-25
8.3.2  TDM_BIND_THREAD  8-25
8.3.3  TDM_DATATYPE_CHECK  8-26
8.3.4  TDM_PRCP_MAX_CALL_WAIT_TIME  8-26
8.3.5  TDM_PRCP_MAX_TXN_CALL_WAIT_TIME  8-27
8.3.6  TDM_SHARED_THREADS_MAX  8-27
8.3.7  TDM_SHARED_THREADS_MIN  8-27
8.3.8  TDM_THREADING_MODE  8-28

8.4  ADR Diagnostic Parameters for Oracle Connection Manager  8-28

8.4.1  ADR_BASE  8-29
8.4.2  DIAG_ADR_ENABLED  8-29
8.4.3  LOG_LEVEL  8-30
8.4.4  TRACE_LEVEL  8-30
8.4.5  TRACE_TIMESTAMP  8-31

8.5  Non-ADR Diagnostic Parameters for Oracle Connection Manager  8-31

8.5.1  LOG_DIRECTORY  8-32
8.5.2  TRACE_DIRECTORY  8-32
8.5.3  TRACE_FILELEN  8-32
8.5.4  TRACE_FILENO  8-33

9  Directory Usage Parameters in the ldap.ora File

9.1  Overview of Directory Server Usage File  9-1
9.2  Directory Usage Parameters  9-1
  9.2.1  DEFAULT_ADMIN_CONTEXT  9-2
  9.2.2  DIRECTORY_SERVER_TYPE  9-2
  9.2.3  DIRECTORY_SERVERS  9-2

Part I  Appendices

A  Features Not Supported in this Release

A.1  Overview of Unsupported Features  A-1
  A.1.1  Oracle Net Connection Pooling  A-1
## A

<table>
<thead>
<tr>
<th>A.1.2</th>
<th>Oracle Names</th>
<th>A-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1.3</td>
<td>Oracle Net Listener Password</td>
<td>A-2</td>
</tr>
<tr>
<td>A.2</td>
<td>Unsupported Parameters</td>
<td>A-2</td>
</tr>
<tr>
<td>A.3</td>
<td>Unsupported Control Utility Commands</td>
<td>A-2</td>
</tr>
<tr>
<td>A.4</td>
<td>Unsupported or Deprecated Protocols</td>
<td>A-3</td>
</tr>
</tbody>
</table>

## B

### B.1
Anonymous Access to Oracle Internet Directory

### C

### C.1
Structural Object Classes

### C.2
Attributes
Preface

Review this publication to obtain a complete listing and description of control utility commands and configuration file parameters that you can use to manage Oracle Net Services components.

• Audience
• Documentation Accessibility
• Related Documents
• Conventions

Audience

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

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

Documentation Accessibility

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

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For additional information, see the following Oracle resources:

• *Oracle Database Net Services Administrator's Guide*
• Online Help for Oracle Net Services tools and utilities
• Oracle Database 19c documentation set
• *Oracle Database Global Data Services Concepts and Administration Guide*

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

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

If you have a user name and password for OTN, then you can go directly to the documentation section of the OTN website at http://docs.oracle.com

**Conventions**

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

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
This chapter describes the Listener Control Utility commands and syntax. The terms SQL*Net and Net Services are used interchangeably throughout Oracle documentation.

- **Listener Control Utility Overview**
  To perform basic management functions on one or more listeners, you can use the Listener Control utility commands. You can also use these commands to view and change parameter settings.

- **SET and SHOW Listener Control Utility Commands**
  The SET and SHOW commands enable you to alter and view listener configuration parameters.

- **Understanding Distributed Operations**
  Listener Control utility can perform operations on local or remote listeners.

- **Understanding Oracle Net Listener Security**
  Authentication for listener administration depends on whether you access the listener locally or remotely.

- **Listener Control Utility Commands**
  Use Listener Control Utility commands to manage and configure your listeners.

### 1.1 Listener Control Utility Overview

To perform basic management functions on one or more listeners, you can use the Listener Control utility commands. You can also use these commands to view and change parameter settings.

The Listener Control utility enables you to administer listeners. The syntax of Listener Control utility commands is as follows:

```
lsnrctl command listener_name
```

In the preceding command, `listener_name` is the name of the listener that you want to administer. If you do not specify a specific listener in the command string, then the command is directed to the default listener name, LISTENER.

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 commands from the program prompt. The basic syntax of issuing commands from LSNRCTL> program prompt is:

```
lsnrctl
LSNRCTL> command listener_name
```

You can combine commands in a standard text file and then run them as a sequence of commands. To run in batch mode, use this format:
lsnrctl @file_name

To identify comments in the batch script, you can use either REM or #. All other lines are considered commands. Commands that require confirmation do not require confirmation during batch processing.

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

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

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

The Listener Control utility supports the following types of commands:

- Operational commands, such as START and STOP.
- Modifier commands, such as SET TRC_LEVEL.
- Informational commands, such as STATUS and SHOW LOG_FILE.

1.2 SET and SHOW Listener Control Utility Commands

The SET and SHOW commands enable you to alter and view listener configuration parameters.

Use the SET command to alter parameter values for a specified listener. You set the name of the listener to administer using the SET CURRENT_LISTENER command. Parameter values remain in effect until the listener is shut down. If you want these settings to persist, then use the SAVE_CONFIG command to save changes to the listener.ora file.

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

1.3 Understanding Distributed Operations

Listener Control utility can perform operations on local or remote listeners.

This example explains how to configure listeners for remote administration.

Set Up a Computer to Remotely Administer a Listener

Ensure that the Listener Control utility (lsnrctl) executable is installed in the ORACLE_HOME/bin directory. You can resolve the name of the listener that you want to administer either through a listener.ora file, or by a naming method.

When you administer a listener remotely, you can issue all of the commands except START. However, Listener Control utility only starts the listener on the same computer from which the utility runs.
When issuing commands, specify the listener name as an argument. If you omit the listener name in the command, then the listener name set with the command `SET CURRENT_LISTENER` is used. If you do not set the listener name with that command, then the command is directed to the default listener name, `LISTENER`.

**Example 1-1   Issuing Commands Using the Listener Control Utility**

`LSNRCTL> SERVICES lsnr`

1.4 Understanding Oracle Net Listener Security

Authentication for listener administration depends on whether you access the listener locally or remotely.

Local listener administration security is administered using local operating system authentication. Local authentication restricts listener administration to the user account that started the listener or to the super user. By default, remote listener administration is disabled.

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

1.5 Listener Control Utility Commands

Use Listener Control Utility commands to manage and configure your listeners.

- **EXIT**
  Use the Listener Control utility command `EXIT` to exit from the Listener Control utility and return to the operating system prompt.

- **HELP**
  Use the Listener Control utility command `HELP` to list the Listener Control utility commands and to obtain syntax help for particular Listener Control utility commands.

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

- **RELOAD**
  Use the Listener Control utility command `RELOAD` to reload the `listener.ora` file so that you can add or change statically configured services without stopping the listener.

- **SAVE_CONFIG**
  Use the Listener Control utility command `SAVE_CONFIG` to save the current configuration state of the listener to the `listener.ora` file.

- **SERVICES**
  Use the Listener Control utility command `SERVICES` to return details about database services, instances, and service handlers to which listeners forward client connection requests.

- **SET**
  Use the Listener Control utility command `SET` to alter listener parameter values.
• **SET CURRENT_LISTENER**
  Use the Listener Control utility command `SET CURRENT_LISTENER` to set the name of the listener to administer.

• **SET DISPLAYMODE**
  Use the Listener Control utility command `SET DISPLAYMODE` to change the format and detail level for the `SERVICES` and `STATUS` commands.

• **SET INBOUND_CONNECT_TIMEOUT**
  Use the Listener Control utility command `SET INBOUND_CONNECT_TIMEOUT` to specify the duration in which clients must complete connection requests to listeners after establishing network connections.

• **SET LOG_DIRECTORY**
  Use the Listener Control utility command `SET LOG_DIRECTORY` to set the destination directory to which the listener log file is written.

• **SET LOG_FILE**
  Use the Listener Control utility command `SET LOG_FILE` to set the listener log file name.

• **SET LOG_STATUS**
  Use the Listener Control utility command `SET LOG_STATUS` to turn listener logging on or off.

• **SET SAVE_CONFIG_ON_STOP**
  Use the Listener Control utility command `SET SAVE_CONFIG_ON_STOP` to specify whether listener parameter value changes that you make with the `SET` command are saved to the `listener.ora` file when you stop the listener with the `STOP` command.

• **SET TRC_DIRECTORY**
  Use the Listener Control utility command `SET TRC_DIRECTORY` to set the destination directory into which Oracle writes listener trace files.

• **SET TRC_FILE**
  Use the Listener Control utility command `SET TRC_FILE` to set the names of listener trace files.

• **SET TRC_LEVEL**
  Use the Listener Control utility command `SET TRC_LEVEL` to set a specific listener tracing level.

• **SHOW**
  Use the Listener Control utility command `SHOW` to list current listener parameter values.

• **SPAWN**
  Use the Listener Control utility command `SPAWN` to start a program that is stored on the computer on which the listener is running and that is listed with an alias in the `listener.ora` file.

• **START**
  Use the Listener Control utility command `START` to start a named listener.

• **STATUS**
  Use the Listener Control utility command `STATUS` to show listener status information.

• **STOP**
  Use the Listener Control utility command `STOP` to stop the named listener.
- **TRACE**
  Use the Listener Control utility command `TRACE` to set listener tracing.

- **VERSION**
  Use the Listener Control utility command `VERSION` to show the current version of Listener Control utility.

### 1.5.1 EXIT

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

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

**Prerequisites**
None

**Syntax**
From the Listener Control utility:

```
LSNRCTL> EXIT
```

**Arguments**
None

**Usage Notes**
This command is identical to the `QUIT` command.

**Example**

```
LSNRCTL> EXIT
```

### 1.5.2 HELP

Use the Listener Control utility command `HELP` to list the Listener Control utility commands and to obtain syntax help for particular Listener Control utility commands.

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

**Prerequisites**
None
Syntax

From the operating system:

lsnrctl HELP command

From the Listener Control utility:

LSNRCTL> HELP command

Arguments

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

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

Example

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

1.5.3 QUIT

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

Purpose

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

Prerequisites

None
Syntax
From the Listener Control utility:

LSNRCTL> QUIT

Arguments
None

Usage Notes
This command is identical to the EXIT command.

Example

LSNRCTL> QUIT

1.5.4 RELOAD

Use the Listener Control utility command RELOAD to reload the listener.ora file so that you can add or change statically configured services without stopping the listener.

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

When you run this command, the database services, instances, service handlers, and listening endpoints previously registered dynamically with the listener are unregistered, and subsequently registered again.

To obtain a lightweight reload without dropping registration, use the option -with_ha.
Using this option ensures that registered services remain available to clients during reload.

Prerequisites
None

Syntax
From the operating system:

lsnrctl RELOAD [-with_ha] listener_name

From the Listener Control utility:

LSNRCTL> RELOAD [-with_ha] listener_name

Arguments

listener_name: The listener name, if the default name of LISTENER is not used.
-with_ha: command option used with RELOAD that indicates that the reload of listener.ora is completed without dropping existing registrations.

Example

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

1.5.5 SAVE_CONFIG

Use the Listener Control utility command SAVE_CONFIG to save the current configuration state of the listener to the listener.ora file.

Purpose

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

Syntax

From the operating system:

lsnrctl SAVE_CONFIG listener_name

From the Listener Control utility:

LSNRCTL> SAVE_CONFIG listener_name

Arguments

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

Usage Notes

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

Example

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

Use the Listener Control utility command `SERVICES` to return details about database services, instances, and service handlers to which listeners forward client connection requests.

**Purpose**

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

**Prerequisites**

None

**Syntax**

**Arguments**

From the operating system:

```bash
lsnrctl SERVICES listener_name
```

From the Listener Control utility:

```bash
LSNRCTL> SERVICES listener_name
```

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

**Usage Notes**

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

**See Also:**

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

**Example**

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

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

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

1.5.7 SET

Use the Listener Control utility command SET to alter listener parameter values.

**Purpose**

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

**Prerequisites**

None

**Syntax**

From the operating system:

```
lsnrctl SET parameter
```

From the Listener Control utility:

```
LSNRCTL> SET parameter
```

**Arguments**

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

When you enter SET without an argument, the Listener Control utility lists all of the parameters.
Usage Notes

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

Example

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

1.5.8 SET CURRENT_LISTENER

Use the Listener Control utility command SET CURRENT_LISTENER to set the name of the listener to administer.

Purpose

To set the name of the listener that you want to administer. After you set the listener name with this command, you can issue subsequent commands that normally require listener_name without specifying the listener.

Syntax

From the Listener Control utility:

LSNRCTL> SET CURRENT_LISTENER listener_name

Arguments

listener_name: The listener name, if you are not using the default name LISTENER.

Usage Notes

When you specify a listener name using SET CURRENT_LISTENER, the Listener Control utility commands act on the listener name that you specify with this command. You do not have to continue to specify the name of the listener.
Example

LSNRCTL> SET CURRENT_LISTENER lsnr
Current Listener is lsnr

1.5.9 SET DISPLAYMODE

Use the Listener Control utility command SET DISPLAYMODE to change the format and detail level for the SERVICES and STATUS commands.

Purpose
To change the format and level of detail for the SERVICES and STATUS commands.

Syntax
From the Listener Control utility:

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

Arguments
Specify one of the following modes:

compat: Output that is compatible with earlier releases of the listener.

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

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

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

Example

LSNRCTL> SET DISPLAYMODE normal
Service display mode is NORMAL

1.5.10 SET INBOUND_CONNECT_TIMEOUT

Use the Listener Control utility command SET INBOUND_CONNECT_TIMEOUT to specify the duration in which clients must complete connection requests to listeners after establishing network connections.

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

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

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

Syntax

From the Listener Control utility:

LSNRCTL> SET INBOUND_CONNECT_TIMEOUT time

Arguments

time: The duration of time in seconds. Default setting is 60 seconds.

Example

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

1.5.11 SET LOG_DIRECTORY

Use the Listener Control utility command SET LOG_DIRECTORY to set the destination directory to which the listener log file is written.

Purpose

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

Note:

This command works only if Automatic Diagnostic Repository (ADR) is disabled. The default is for ADR to be enabled and to use the log directory ORACLE_HOME/log/diag/product_type.

Prerequisites

None

Syntax

From the operating system:

lsnrctl SET LOG_DIRECTORY directory
From the Listener Control utility:

LSNRCTL> SET LOG_DIRECTORY directory

Arguments

directory: The directory path of the listener log file.

Example

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

1.5.12 SET LOG_FILE

Use the Listener Control utility command SET LOG_FILE to set the listener log file name.

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

Note:
This command works only if Automatic Diagnostic Repository (ADR) is disabled. The default is for ADR to be enabled and to use the log directory ORACLE_HOME/log/diag/product_type.

Prerequisites
None

Syntax
From the operating system:

lsnrctl SET LOG_FILE file_name

From the Listener Control utility:

LSNRCTL> SET LOG_FILE file_name

Arguments

file_name: The file name of the listener log.
1.5.13 SET LOG_STATUS

Use the Listener Control utility command SET LOG_STATUS to turn listener logging on or off.

Purpose
To turn listener logging on or off.

Prerequisites
None

Syntax
From the operating system:

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

From the Listener Control utility:

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

Arguments

- **on**: To turn logging on.
- **off**: To turn logging off.

Example

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

Use the Listener Control utility command SET SAVE_CONFIG_ON_STOP to specify whether listener parameter value changes that you make with the SET command are saved to the listener.ora file when you stop the listener with the STOP command.

Purpose

To specify whether changes made to parameter values for the listener by the SET command are saved to the listener.ora file when the listener is stopped with the STOP command.

When you save changes, the Listener Control utility tries to preserve formatting, comments, and letter case. Before the command modifies the listener.ora file, it creates a backup of the file called listener.bak.

To have all parameters saved immediately, use the SAVE_CONFIG command.

Syntax

From the operating system:

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

From the Listener Control utility:

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

Arguments

- **on**: To save configuration to listener.ora.
- **off**: To not save configuration to listener.ora.

Example

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

1.5.15 SET TRC_DIRECTORY

Use the Listener Control utility command SET TRC_DIRECTORY to set the destination directory into which Oracle writes listener trace files.

Purpose

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

None

**Syntax**

From the operating system:

`lsnrctl SET TRC_DIRECTORY directory`

From the Listener Control utility:

`LSNRCTL> SET TRC_DIRECTORY directory`

**Arguments**

directory: The directory path of the listener trace files.

**Example**

`LSNRCTL> SET TRC_DIRECTORY /usr/oracle/admin`

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

LISTENER parameter "trc_directory" set to /usr/oracle/admin

The command completed successfully

### 1.5.16 SET TRC_FILE

Use the Listener Control utility command `SET TRC_FILE` to set the names of listener trace files.

**Purpose**

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

---

**Note:**

This command works only if Automatic Diagnostic Repository (ADR) is disabled. The default is for ADR to be enabled and to use the log directory `ORACLE_HOME/log/diag/product_type`.

---
Prerequisites
None

Syntax
From the operating system:

```sh
lsnrctl SET TRC_FILE file_name
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_FILE file_name
```

Arguments

- `file_name`: The file name of the listener trace.

Example

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

1.5.17 SET TRC_LEVEL

Use the Listener Control utility command `SET TRC_LEVEL` to set a specific listener tracing level.

Purpose

To set a specific level of tracing for listeners.

Prerequisites
None

Syntax

From the operating system:

```sh
lsnrctl SET TRC_LEVEL level
```

From the Listener Control utility:

```
LSNRCTL> SET TRC_LEVEL level
```

Arguments

- `level`: One of the following trace levels:
  - `off` for no trace output
• user for user trace information
• admin for administration trace information
• support for Oracle Support Services trace information

Usage Notes
This command has the same functionality as the TRACE command.

Example

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

1.5.18 SHOW

Use the Listener Control utility command SHOW to list current listener parameter values.

Purpose
To view the current parameter values for the listener.
All of the SET parameters have equivalent SHOW parameters.

Prerequisites
None

Syntax
From the operating system:

lsnrctl SHOW parameter

From the Listener Control utility:

LSNRCTL> SHOW parameter

Arguments

parameter: A parameter whose settings you want to review. Parameters are shown in the example output.

When you enter SHOW without an argument, the Listener Control utility lists all the parameters.

Example

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

1.5.19 SPAWN

Use the Listener Control utility command SPAWN to start a program that is stored on the computer on which the listener is running and that is listed with an alias in the listener.ora file.

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

Prerequisites
None

Syntax
From the operating system:

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

From the Listener Control utility:

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

Arguments

- `listener_name`: The listener name, if the default name of LISTENER is not used.
- `alias`: The alias of the program to be spawned is specified by a listener.ora file entry that is similar to the following:

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

For example:

```sql
nstest = (PROGRAM=(NAME=nstest)(ARGS=test1)(ENVS='ORACLE_HOME=/usr/oracle'))
```
Example
The nstest program, shown in the preceding section, can then be spawned using the following command:

```bash
lsnrctl SPAWN listener_name nstest
```

1.5.20 START

Use the Listener Control utility command `START` to start a named listener.

Purpose
To start the named listener.

Prerequisites
The listener must not be running.

Syntax
From the operating system:

```bash
lsnrctl START listener_name
```

From the Listener Control utility:

```
LSNRCTL> START listener_name
```

Note:
On Microsoft Windows, if the database was installed with the Oracle Home User, then the utility can prompt for a password. The password is the operating system password for the Oracle Home User. The prompt is displayed only if the listener service does not exist and if it needs to be created as part of starting the listener.

Arguments

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

Usage Notes
To start a listener that you configured in the `listener.ora` file whose name does not contain the string `LISTENER`.

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

```bash
lsnrctl START tcp_lsnr
```
From the Listener Control utility:

LSNRCTL> START tcp_lsnr

Example

LSNRCTL> START

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

TNSLSNR for Linux: Version 18.0.0.0.0
System parameter file is $ORACLE_HOME/network/admin/listener.ora
Log messages written to $ORACLE_BASE/diag/tnslsnr/node_name/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 18.0.0.0.0
Start Date                21-JAN-2018 21:50:49
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   $ORACLE_HOME/network/admin/listener.ora
Listener Log File         $ORACLE_BASE/diag/tnslsnr/node_name/listener/alert/log.xml
Listening Endpoints Summary...
    (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)))
The listener supports no services
The command completed successfully

See Also:
Oracle Database Administrator’s Reference for Microsoft Windows for information about the Oracle Home User

1.5.21 STATUS

Use the Listener Control utility command STATUS to show listener status information.

Purpose

To display basic status information about a listener, including a summary of listener configuration settings, listening protocol addresses, and a summary of services that are registered with the listener.
Note:
You can also obtain the status of the listener through the Oracle Enterprise Manager Cloud Control console.

Prerequisites
None

Syntax
From the operating system:

```
lsnrctl STATUS listener_name
```

From the Listener Control utility:

```
LSNRCTL> STATUS listener_name
```

Arguments

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

Usage Notes

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

See Also:

`Oracle Database Net Services Administrator's Guide` for a complete description of STATUS output.

Example

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

- Listener configuration settings
- Listening endpoints summary
- Services summary, which is an abbreviated version of the SERVICES command output

```
LSNRCTL> STATUS
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=net)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 18.0.0.0.0 -
Production
Start Date                12-JAN-2018 12:02:00
Uptime                    0 days 0 hr. 5 min. 29 sec
Trace Level               support
Security                  OFF
SNMP                      OFF
Listener Parameter File   /oracle/network/admin/listener.ora
Listener Log File         /oracle/network/log/listener.log
Listener Trace File       /oracle/network/trace/listener.trc

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

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

1.5.22 STOP

Use the Listener Control utility command STOP to stop the named listener.

Purpose
To stop the named listener.

Prerequisites
The named listener must be running.

Syntax
From the operating system:
lsnrctl STOP listener_name

From the Listener Control utility:
LSNRCTL> STOP listener_name

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

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

Use the Listener Control utility command `TRACE` to set listener tracing.

**Purpose**
To set tracing for the listener.

**Syntax**
From the operating system:

```
lsnrctl trace level listener_name
```

From the Listener Control utility:

```
LSNRCTL> trace level listener_name
```

**Arguments**
- `level`: One of the following trace levels:
  - `off` for no trace output
  - `user` for user trace information
  - `admin` for administration trace information
  - `support` for Oracle Support Services trace information
- `listener_name`: Specify the listener name, if the default name of `LISTENER` is not used.

**Usage Notes**
This command has the same functionality as the `SET TRC_LEVEL` command.

**Example**

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

1.5.24 VERSION

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

**Purpose**
To display the current version of Listener Control utility.

**Prerequisites**
None
Syntax

From the operating system:

lsnrctl VERSION listener_name

From the Listener Control utility:

LSNRCTL> VERSION listener_name

Arguments

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

Example

LSNRCTL> version listener
Connecting to ADDRESS=(PROTOCOL=TCP)(HOST=sales-server)(PORT=1521))
TNSLSNR for Linux: Version 19.0.0.0
    TNS for Linux: Version 19.0.0.0
    Oracle Bequeath NT Protocol Adapter for Linux: Version 19.0.0.0
    Unix Domain Socket IPC NT Protocol Adaptor for Linux: Version 19.0.0.0
    TCP/IP NT Protocol Adapter for Linux: Version 19.0.0.0
The command completed successfully
Oracle Connection Manager Control Utility

This chapter describes Oracle Connection Manager Control Utility commands and syntax.

• **Connection Manager Control Utility Command Modes and Syntax**
  Oracle Connection Manager Control utility (CMCTL) enables you to start up, configure, and alter how client connection requests are managed.

• **Oracle Connection Manager Control Utility Overview**

• **Oracle Connection Manager Control Utility Commands**
  Use Oracle Connection Manager Control utility commands to manage and configure Oracle Connection Manager instances.

### 2.1 Connection Manager Control Utility Command Modes and Syntax

Oracle Connection Manager Control utility (CMCTL) enables you to start up, configure, and alter how client connection requests are managed.

The syntax of the Oracle Connection Manager Control utility is:

```
cmctl command [argument]
```

Oracle Connection Manager Control utility supports the following command types:

• **Initialization and termination commands** such as `STARTUP` and `SHUTDOWN`

• **Alter commands** such as `SET LOG_LEVEL` and `SET EVENT`

• **Display commands**, such as `SHOW STATUS` and `SHOW RULES`

• **Gateway commands** such as `SHOW GATEWAYS` and `RESUME GATEWAYS`
**Note:**

Use *SET* commands to dynamically alter configuration parameters that control how the listener receives client connections. Changes only remain in effect until Oracle Connection Manager shuts down. You cannot save changes to the `cman.ora` file. One exception is the Oracle Connection Manager password, which you save using the `SAVE_PASSWD` command.

Use the Oracle Connection Manager Control utility in command mode or batch mode.

- **Using command mode:**
  - From the Oracle Connection Manager Control utility:
    
    Enter `cmctl` at the command line to obtain the program prompt and run the command:
    
    ```
    cmctl
    CMCTL> command
    ```
    
  - From the operating system:
    
    Enter the command from the operating system command prompt:
    
    ```
    cmctl [command] [argument1 .. argumentN] [-c instance_name]
    ```
    
    You can append an Oracle Connection Manager instance name as an argument to all commands that you run in this manner. If you do not include an Oracle Connection Manager instance name, then the default instance name is assumed. The default name is `cman_hostname`. If you set password in a previous CMCTL session, then you can be prompted for a password. If a password was set, then to issue commands from an Oracle Connection Manager Control utility session of Oracle Connection Manager, you must enter a password once, at the beginning of the session.

  - **Caution:**
    
    There is an option to specify the password on the command line. However, doing this exposes the password on the screen; this is a security risk. Oracle recommends that you not use the password option (`-p`) on the command line.

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

```cmctl @input_file```

See Also:

*Oracle Database Net Services Administrator's Guide* for more information about Oracle Connection Manager architecture

---

### 2.2 Oracle Connection Manager Control Utility Overview

Oracle Connection Manager Control utility (CMCTL) enables you to administer Oracle Connection Manager. Use Oracle Connection Manager Control utility commands to administer one or more Oracle Connection Manager instances. Additionally, you can view and change parameter settings.

### 2.3 Oracle Connection Manager Control Utility Commands

Use Oracle Connection Manager Control utility commands to manage and configure Oracle Connection Manager instances.

- **ADMINISTER**
  - Use the Oracle Connection Manager Control Utility command `ADMINISTER` to select Oracle Connection Manager instances to administer.

- **CLOSE CONNECTIONS**
  - Use the Oracle Connection Manager Control Utility command `CLOSE CONNECTIONS` to terminate connections.

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

- **HELP**
  - Use the Oracle Connection Manager Control Utility `HELP` command to display all of the Oracle Connection Manager Control utility commands or to show the syntax of a particular command.

- **QUIT**
  - Use the Oracle Connection Manager Control Utility command `QUIT` to exit Oracle Connection Manager Control utility.

- **RELOAD**
  - Use the Oracle Connection Manager Control utility `RELOAD` command to make the utility dynamically reread parameters and rules.

- **RESUME GATEWAYS**
  - Use the Oracle Connection Manager Control Utility command `RESUME GATEWAYS` to resume suspended gateway processes.
• **SAVE_PASSWD**
  Use the Oracle Connection Manager Control Utility command `SAVE_PASSWD` to save passwords to the `cman.ora` file.

• **SET**
  Use the Oracle Connection Manager Control Utility `SET` command to list the parameters that you can modify using this command.

• **SET ASO_AUTHENTICATION_FILTER**
  Use the Oracle Connection Manager Control Utility command `SET ASO_AUTHENTICATION_FILTER` to indicate whether clients must use Oracle Database security authentication.

• **SET CONNECTION_STATISTICS**
  Use the Oracle Connection Manager Control Utility `SET CONNECTION_STATISTICS` command to specify whether gateway processes collect connection statistics.

• **SET EVENT**
  Use the Oracle Connection Manager Control Utility `SET EVENT` command to log information for a particular event.

• **SET IDLE_TIMEOUT**
  Use the Oracle Connection Manager Control Utility `SET IDLE_TIMEOUT` command to specify the amount of time that clients can be idle without transmitting data.

• **SET INBOUND_CONNECT_TIMEOUT**
  Use the Oracle Connection Manager Control Utility `SET INBOUND_CONNECT_TIMEOUT` command to specify the maximum amount of time that Oracle Connection Manager listeners wait for client connection requests before timing out.

• **SET LOG_DIRECTORY**
  Use the Oracle Connection Manager Control Utility `SET LOG_DIRECTORY` command to designate where Oracle Connection Manager log files are written.

• **SET LOG_LEVEL**
  Use the Oracle Connection Manager Control Utility `SET LOG_LEVEL` command to set Oracle Connection Manager log levels.

• **SET OUTBOUND_CONNECT_TIMEOUT**
  Use the Oracle Connection Manager Control Utility `SET OUTBOUND_CONNECT_TIMEOUT` command to specify the time limit that Oracle Connection Manager instances wait for server connections before timing out.

• **SET PASSWORD**
  Use the Oracle Connection Manager Control Utility `SET PASSWORD` command to assign Oracle Connection Manager instance passwords.

• **SET SESSION_TIMEOUT**
  Use the Oracle Connection Manager Control utility `SET SESSION_TIMEOUT` command to specify the maximum duration of Oracle Connection Manager sessions.

• **SET TRACE_DIRECTORY**
  Use the Oracle Connection Manager Control utility `SET TRACE_DIRECTORY` command to designate where Oracle Connection Manager instance trace files are written.

• **SET TRACE_LEVEL**
  Use the Oracle Connection Manager Control utility `SET TRACE_LEVEL` command to set Oracle Connection Manager instance trace levels.
• **SHOW**
  Use the Oracle Connection Manager Control utility **SHOW** command to display the parameters that you can use as arguments for this command.

• **SHOW ALL**
  Use the Oracle Connection Manager Control utility **SHOW ALL** command to combine and display **SHOW PARAMETERS** and **SHOW RULES** command output.

• **SHOW CONNECTIONS**
  Use the Oracle Connection Manager Control utility **SHOW CONNECTIONS** command to display connection information.

• **SHOW DEFAULTS**
  Use the Oracle Connection Manager Control utility **SHOW DEFAULTS** command to display default parameter settings.

• **SHOW EVENTS**
  Use the Oracle Connection Manager Control utility **SHOW EVENTS** command to display events that are currently operating.

• **SHOW GATEWAYS**
  Use the Oracle Connection Manager Control utility **SHOW GATEWAYS** command to display the statuses of gateway processes.

• **SHOW PARAMETERS**
  Use the Oracle Connection Manager Control utility **SHOW PARAMETERS** command to display the parameter settings for an instance.

• **SHOW RULES**
  Use the Oracle Connection Manager Control Utility **SHOW RULES** command to display an instance access control list.

• **SHOW SERVICES**
  Use the Oracle Connection Manager Control utility **SHOW SERVICES** command to display Oracle Connection Manager instance information.

• **SHOW STATUS**
  Use the Oracle Connection Manager Control utility **SHOW STATUS** command to display Oracle Connection Manager instance information.

• **SHOW VERSION**
  Use the Oracle Connection Manager Control utility **SHOW VERSION** command

• **SHUTDOWN**
  Use the Oracle Connection Manager Control utility **SHUTDOWN** command to shut down gateway processes or an entire Oracle Connection Manager instance.

• **STARTUP**
  Use the Oracle Connection Manager Control utility **STARTUP** command to start Oracle Connection Manager.

• **SUSPEND GATEWAY**
  Use the Oracle Connection Manager Control utility **SUSPEND GATEWAY** command to specify the gateway processes than cannot accept new client connections.
2.3.1 ADMINISTER

Use the Oracle Connection Manager Control Utility command `ADMINISTER` to select Oracle Connection Manager instances to administer.

**Purpose**

To select an Oracle Connection Manager instance.

**Prerequisites**

None

**Syntax**

From the Oracle Connection Manager Control utility:

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

**Arguments**

*instance_name*: The Oracle Connection Manager instance name that you want to administer. Instances are defined in the `cman ora` file.

**Usage Notes**

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

`ADMINISTER` enables you to choose Oracle Connection Manager instances to administer. To start the Oracle Connection Manager instance, run the `STARTUP` command.

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

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

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

**Example**

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

Use the Oracle Connection Manager Control Utility command `CLOSE CONNECTIONS` to terminate connections.

**Purpose**

To terminate connections, using specific qualifiers to select the connections to close.

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

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

**Arguments**

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

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

If you do not specify a state, then `CLOSE CONNECTIONS` defaults to all possible states. If the time qualifier is included under these conditions, then the time specified is the amount of time that has elapsed since a client initiated a connection.

*time*: The time format. Use the following format to specify connections that have a duration greater than the time indicated:

```
gt[hh:mm:ss]
```

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

- **from IP**
- **from hostname**
- **from subnet**
destination: The destination address. Use one of the following formats to specify the destination address:

- to IP
- to hostname
- to subnet

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

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

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

Usage Notes

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

The IDLE state qualifier always requires a time qualifier.

Issuing CLOSE CONNECTIONS without an argument closes all connections.

Examples

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

CMCTL> CLOSE CONNECTIONS gt 1:30:00 from 192.0.2.32/24 to host1

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

CMCTL> CLOSE idle CONNECTIONS gt 30:00 using 0

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

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

REST API for CLOSE CONNECTIONS Command

POST /close/connections

JSON Payload

```json
{
    "in" : [ "idle" | "connecting" | "established" | "terminating"],
    "gt" : "[hh:mm:ss]",
    "from" : ["source ip " | "hostname " | "subnet"],
    "to" : ["destination ip" | "hostname" | "subnet"],
    "for" : "service name",
    "using" : "gateway process identifier",
    "connect_id_list" : [id1, id2, .. ]
}
```
2.3.3 EXIT

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

**Purpose**

To exit from Oracle Connection Manager Control utility.

**Prerequisites**

None

**Syntax**

From the operating system:

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

From Oracle Connection Manager Control utility:

```
CMCTL> EXIT
```

**Usage Notes**

This command is identical to the QUIT command.

**Example 2-1 Example**

```
CMCTL> EXIT
```

2.3.4 HELP

Use the Oracle Connection Manager Control Utility HELP command to display all of the Oracle Connection Manager Control utility commands or to show the syntax of a particular command.

**Purpose**

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

**Prerequisites**

None

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

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

From the operating system:

```
cmctl HELP [command] [-c instance_name]
```
From the Oracle Connection Manager Control utility:

CMCTL> HELP [command]

Arguments

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

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

<table>
<thead>
<tr>
<th>administer</th>
<th>close*</th>
<th>exit</th>
<th>reload</th>
</tr>
</thead>
<tbody>
<tr>
<td>resume*</td>
<td>save_passwd</td>
<td>set*</td>
<td>show*</td>
</tr>
<tr>
<td>shutdown</td>
<td>sleep</td>
<td>startup</td>
<td>suspend*</td>
</tr>
<tr>
<td>show_version</td>
<td>quit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.3.5 QUIT

Use the Oracle Connection Manager Control Utility command QUIT to exit Oracle Connection Manager Control utility.

Purpose

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

Prerequisites

None

Syntax

CMCTL> QUIT

From the Oracle Connection Manager Control utility:

cmctl QUIT

From the operating system:

Usage Notes

This command is identical to the EXIT command.

Example

CMCTL> QUIT
2.3.6 RELOAD

Use the Oracle Connection Manager Control utility `RELOAD` command to make the utility dynamically reread parameters and rules.

**Purpose**
To dynamically reread parameters and rules.

**Prerequisites**
Oracle Connection Manager must be running.

**Syntax**
From the operating system:
```
cmctl RELOAD [-with_ha] [-c instance_name]
```
From the Oracle Connection Manager Control utility:
```
CMCTL> RELOAD [-with_ha]
```

**Arguments**
- `-with_ha`: It is used to reload `cman.ora` without dropping registrations

**Usage Notes**
Configuration information that you modify using the `RELOAD` command applies only to new connections. Existing connections are unaffected. The `SET RELOAD` command restores configurations set in `cman.ora`, and overrides the `SET` command.

The `RELOAD` command reregisters gateways with the Oracle Connection Manager listener during which some new connections might be refused until the registration completes.

You can use the `-with_ha` option with `RELOAD` to not drop registrations, thus providing high service availability during reload.

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

**REST API for RELOAD Command**

```
POST /reload
```
2.3.7 RESUME GATEWAYS

Use the Oracle Connection Manager Control Utility command RESUME GATEWAYS to resume suspended gateway processes.

**Purpose**
To resume gateway processes that have been suspended.

**Prerequisites**
Oracle Connection Manager must be running.

**Syntax**
From the operating system:
```
cmctl RESUME GATEWAYS [gateway_process_id] [cman_name]
```
From the Oracle Connection Manager Control utility:
```
CMCTL> RESUME GATEWAYS [gateway_process_id]
```

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

**Usage Notes**
Running the RESUME GATEWAYS command without an argument reopens all closed gateway processes.

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

**REST API for RESUME GATEWAYS Command**
```
POST /resume/gateways
JSON Payload
{
  "gateway_id_list" : [id1, id2, .. ]
}
```

2.3.8 SAVE_PASSWD

Use the Oracle Connection Manager Control Utility command SAVE_PASSWD to save passwords to the cman.ora file.

**Purpose**
To save the current password to the cman.ora file, the configuration file for Oracle Connection Manager.
Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:
```
cmctl SAVE_PASSWD [-c instance_name]
```
From the Oracle Connection Manager Control utility:
```
CMCTL> SAVE_PASSWD
```

Usage Notes
If you run this command, then the next session of Oracle Connection Manager uses the password that you saved to the file. The password is stored in an encrypted format in the cman.ora file.

Example
```
CMCTL> SAVE_PASSWD
```

2.3.9 SET

Use the Oracle Connection Manager Control Utility SET command to list the parameters that you can modify using this command.

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

Prerequisites
None

Syntax
From the operating system:
```
cmctl SET
```
From the Oracle Connection Manager Control utility:
```
CMCTL> SET
```

Example
```
CMCTL> SET
The following operations are available after set
An asterisk (*) denotes a modifier or extended command:

aso_authentication_filter outbound_connect_timeout
connection_statistics password
event session_timeout
idle_timeout trace_directory
inbound_connect_timeout
trace_level
```
2.3.10 SET ASO_AUTHENTICATION_FILTER

Use the Oracle Connection Manager Control Utility command `SET ASO_AUTHENTICATION_FILTER` to indicate whether clients must use Oracle Database security authentication.

**Purpose**
To indicate whether the client must use Oracle Database security to authenticate.

**Prerequisites**
Oracle Connection Manager must be running.

**Syntax**
From the operating system:
```
  cmctl SET ASO_AUTHENTICATION_FILTER {on | off}[-c instance_name]
```

From the Oracle Connection Manager Control utility:
```
  CMCTL> SET ASO_AUTHENTICATION_FILTER {on | off}
```

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

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

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

**REST API for SET ASO_AUTHENTICATION_FILTER Command**

```
POST /set/authlevel
JSON Payload
{
  "authlevel": "true" | "false"
}
```
2.3.11 SET CONNECTION_STATISTICS

Use the Oracle Connection Manager Control Utility SET_CONNECTION_STATISTICS command to specify whether gateway processes collect connection statistics.

Purpose
To specify whether gateway processes collect connection statistics.

Prerequisites
To specify whether gateway processes collect connection statistics.

Syntax
From the operating system:

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

From the Oracle Connection Manager Control utility:

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

Arguments

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

Usage Notes
If you SET_CONNECTION_STATISTICS to yes, then you can obtain statistics by running the SHOW CONNECTIONS command.

Example

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

REST API for SET_CONNECTION_STATISTICS Command

```
POST /set/connstats/
JSON Payload
{
  "connection_statistics" : "yes"|"no"
}
```
2.3.12 SET EVENT

Use the Oracle Connection Manager Control Utility SET EVENT command to log information for a particular event.

**Purpose**

To log information for a particular event.

**Syntax**

From the operating system:

```cmct1 SET EVENT event_group [-c instance_name]```

From the Oracle Connection Manager Control utility:

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

**Arguments**

`event_group`: Specify one of the following event groups:

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

`on`: To enable an event group.

`off`: To disable an event group.

**Usage Notes**

The SET EVENT command accepts only one argument at a time. To log multiple events, run the command for each event separately.

**Example**

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

Use the Oracle Connection Manager Control Utility SET IDLE_TIMEOUT command to specify the amount of time that clients can be idle without transmitting data.

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

Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:
```
cmctl SET IDLE_TIMEOUT [time] [-c instance_name]
```

From the Oracle Connection Manager Control utility:
```
CMCTL> SET IDLE_TIMEOUT [time]
```

Arguments
- **time**: Specify the idle timeout in seconds. The default is 0 (zero), which disables this feature.

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

REST API for SET IDLE_TIMEOUT Command

POST /set/maxidletime

JSON Payload
```
{
  "idle_timeout" : "time"
}
```

2.3.14 SET INBOUND_CONNECT_TIMEOUT

Use the Oracle Connection Manager Control Utility SET INBOUND_CONNECT_TIMEOUT command to specify the maximum amount of time that Oracle Connection Manager listeners wait for client connection requests before timing out.

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

Syntax
From the operating system:
```bash
cmctl SET INBOUND_CONNECT_TIMEOUT [time] [-c instance_name]
```
From Oracle Connection Manager Control:
```bash
CMCTL> SET INBOUND_CONNECT_TIMEOUT [time]
```

Arguments
- `time`: The inbound connect timeout in seconds. The default is 0 (zero), which disables this feature.

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

REST API for SET INBOUND_CONNECT_TIMEOUT Command

```json
POST /set/maxcntdtime
JSON Payload
{
  "inbound_connect_timeout" : "time"
}
```

2.3.15 SET LOG_DIRECTORY

Use the Oracle Connection Manager Control Utility `SET LOG_DIRECTORY` command to designate where Oracle Connection Manager log files are written.

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

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

Prerequisites
Oracle Connection Manager must be running.
Syntax
From the operating system:
```
cmctl SET LOG_DIRECTORY [directory_path] [-c instance_name]
```
From the Oracle Connection Manager Control utility:
```
CMCTL> SET LOG_DIRECTORY [directory_path]
```
Arguments
directory_path: The location of the log directory. The default path is as follows:
- Linux and UNIX:
  ORACLE_HOME/network/log directory
- Microsoft Windows:
  ORACLE_HOME\network\log directory

Usage Notes
Use the `SHOW PARAMETERS` command to determine the location of the log files.

Example
```
CMCTL>
SET LOG_DIRECTORY /disk1/user_cman_test/oracle/network/admin
```
CMA user.us.example.com parameter log_directory set to
/disk1/user_cman_test/oracle/network/admin

The command completed successfully

2.3.16 SET LOG_LEVEL

Use the Oracle Connection Manager Control Utility `SET LOG_LEVEL` command to set Oracle Connection Manager log levels.

Purpose
To set the log level for Oracle Connection Manager.

Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:
```
cmctl SET LOG_LEVEL [level] [-c instance_name]
```
```
CMCTL> SET LOG_LEVEL [level]
```
From Oracle Connection Manager Control utility:
Arguments

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

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

Usage Notes

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

Example

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

REST API for SET LOG_LEVEL Command

```json
POST /set/loglevel
JSON Payload
{
   "log_level" : "level"
}
```

2.3.17 SET OUTBOUND_CONNECT_TIMEOUT

Use the Oracle Connection Manager Control Utility **SET OUTBOUND_CONNECT_TIMEOUT** command to specify the time limit that Oracle Connection Manager instances wait for server connections before timing out.

**Purpose**

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

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the **From the Oracle Connection Manager Control utility**:

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

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

Example

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

REST API for SET OUTBOUND_CONNECT_TIMEOUT Command

POST /set/octo
JSON Payload

```json
{
  "outbound_connect_timeout" : "time"
}
```

2.3.18 SET PASSWORD

Use the Oracle Connection Manager Control Utility *SET PASSWORD* command to assign Oracle Connection Manager instance passwords.

Purpose

To assign a password to the Oracle Connection Manager instance.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

```
cmctl SET PASSWORD
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SET PASSWORD
```

Arguments

None.

Usage Notes

Use this command to either set a password for the first time or to change an existing password.

This command does not save the password to `cman.ora`. As a result, the password that you set with this command is valid only for the current session. To save the password after you have set it, run the *SAVE_PASSWD* command.
Example

CMCTL> SET PASSWORD

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

The command completed successfully

2.3.19 SET SESSION_TIMEOUT

Use the Oracle Connection Manager Control utility SET SESSION_TIMEOUT command to specify the maximum duration of Oracle Connection Manager sessions.

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

Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:

cmctl SET SESSION_TIMEOUT [time] [-c instance_name]

From the Oracle Connection Manager Control utility:

CMCTL> SET SESSION_TIMEOUT [time]

Arguments

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

Example

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

REST API for SET SESSION_TIMEOUT Command

POST /set/mct
JSON Payload

```json
{
  "session_timeout" : "time"
}
```
**2.3.20 SET TRACE_DIRECTORY**

Use the Oracle Connection Manager Control utility `SET TRACE_DIRECTORY` command to designate where Oracle Connection Manager instance trace files are written.

**Note:**

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

**Purpose**

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

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

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

**Arguments**

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

**Usage Notes**

Use the `SHOW PARAMETERS` command to determine the trace file locations.

**Example**

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

---

**2.3.21 SET TRACE_LEVEL**

Use the Oracle Connection Manager Control utility `SET TRACE_LEVEL` command to set Oracle Connection Manager instance trace levels.

**Purpose**

To set the trace level for an Oracle Connection Manager instance.
Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:

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

From the Oracle Connection Manager Control utility:

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

Arguments

`level`: Specify one of the following log levels:
- `off`: No tracing. This is the default.
- `user`: User trace information.
- `admin`: Administrative trace information.
- `support`: Oracle Support Services trace information.

Usage Notes

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

Use the `SHOW PARAMETERS` command to determine the current trace level.

Example

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

REST API for SET TRACE_LEVEL Command

```
POST /set/tracelevel
JSON Payload
{
  "trace_level" : "level"
}
```

2.3.22 SHOW

Use the Oracle Connection Manager Control utility `SHOW` command to display the parameters that you can use as arguments for this command.

Purpose

To display a list of parameters that you can use as arguments for the `SHOW` command. Entering one of these parameters with the command displays the parameter value or values.
Prerequisites

None

Syntax

From the operating system:

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

From the Oracle Connection Manager Control utility:

```bash
CMCTL> SHOW
```

Example

```bash
CMCTL> SHOW
```

The following operations are available after show
An asterisk (*) denotes a modifier or extended command:

<table>
<thead>
<tr>
<th>all</th>
<th>gateways</th>
<th>status</th>
</tr>
</thead>
<tbody>
<tr>
<td>connections</td>
<td>parameters</td>
<td>version</td>
</tr>
<tr>
<td>defaults</td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td>events</td>
<td>services</td>
<td></td>
</tr>
</tbody>
</table>

### 2.3.23 SHOW ALL

Use the Oracle Connection Manager Control utility `SHOW ALL` command to combine and display `SHOW PARAMETERS` and `SHOW RULES` command output.

Purpose

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

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

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

From the Oracle Connection Manager Control utility:

```bash
CMCTL> SHOW ALL
```

Example

```bash
CMCTL> SHOW ALL
```

<table>
<thead>
<tr>
<th>listener_address</th>
<th>(address=(protocol=tcp)(host=users.us.example.com)(port=1630))</th>
</tr>
</thead>
<tbody>
<tr>
<td>aso_authentication_filter</td>
<td>OFF</td>
</tr>
<tr>
<td>connection_statistics</td>
<td>OFF</td>
</tr>
<tr>
<td>event_group</td>
<td>OFF</td>
</tr>
<tr>
<td>log_directory</td>
<td>/disk1/user_cman_test/oracle/network/log/</td>
</tr>
<tr>
<td>log_level</td>
<td>SUPPORT</td>
</tr>
<tr>
<td>max_connections</td>
<td>256</td>
</tr>
</tbody>
</table>
idle_timeout | 0
inbound_connect_timeout | 0
session_timeout | 0
outbound_connect_timeout | 0
max_gateway_processes | 16
min_gateway_processes | 2
max_cmctl_sessions | 4
password | OFF
trace_directory | /disk1/user_cman_test/oracle/network/trace/
trace_level | OFF
trace_timestamp | OFF
trace_filelen | 0
trace_fileno | 0
(rule_list=
(rule=
(src=*)
(dst=*)
(srv=*)
(act=accept)
)
)
The command completed successfully

REST API for SHOW ALL Command

GET /show/all

2.3.24 SHOW CONNECTIONS

Use the Oracle Connection Manager Control utility SHOW CONNECTIONS command to display connection information.

Purpose

To display information about specific connections or all connections.

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

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

From the Oracle Connection Manager Control utility:

CMCTIL> SHOW CONNECTIONS [information][in state] [gt time] [from source]
[to destination] [for service] [using gateway_process_id]
[connect_identifier_list]
Arguments

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

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

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

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

If you do not specify a state, then `SHOW CONNECTIONS` defaults to all possible states. If the time qualifier is included under these conditions, then the time specified is the amount of time that has elapsed since a client initiated a connection.

**Note:**

This argument is not supported with Oracle Connection Manager in Traffic Director mode.

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

```
> [hh:mm:ss]
```

**Note:**

This argument is not supported with Oracle Connection Manager in Traffic Director mode.

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

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

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

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

**service**: Use the `service_name` format to request a service:
**gateway_process_id**: Use the following format to specify connections that are proxied by the gateway process indicated:

```
using gateway_process_id
```

**connect_identifier_list**: Separate multiple connection identifiers using a space.

**Usage Notes**

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

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

**Examples**

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

```
CMCTL> SHOW CONNECTIONS gt 1:30:00 from 192.0.2.32/24 to host1
```

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

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

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

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

**REST API for SHOW CONNECTIONS Command**

```
POST /show/connections
JSON Payload
{
    "count" : "[yes | no]",
    "in" : "[ idle | connecting | established | terminated]",
    "gt" : " time elapsed since client connection, [hh:mm:ss] format",
    "from" : "[ IP | hostname | subnet]",
    "to" : "[ IP | hostname | subnet]",
    "for" : "service name",
    "using" : " gateway process id",
    "connect_ids" : [id1, id2]
}
```

An example to show the established connection details for `sales.us.example.com` using the json schema is:

```
{
    "count": "no",
    "state" : "established",
```
Additional Statistics Shown in Traffic Director Mode

Each connection to Oracle Connection Manager in Traffic Director mode displays these additional statistics:

- **Source Host Name**: Host name of the client connection.
- **Source Process Id**: Process Id of the connected client.
- **Source Program Name**: The name of the connected client program.
- **Destination Hostname**: Host name of the database server to which the client is connected through Oracle Connection Manager.
- **State**: State of the inbound connection with one of the following values:
  - **THREAD WAIT**: Connection is waiting for a worker thread, not seen in dedicated threads mode.
  - **ACTIVE**: Connection is transferring data, occupying the thread.
  - **IDLE**: Connection is established but inactive, can still occupy the thread if `tdm_bind_thread=true` in `cman.ora`.
- **Idle time**: Cumulative time in µs the connection is in IDLE state.
- **Thread Wait time**: Cumulative time in µs the connection is in THREAD WAIT state. It is always 0 in dedicated threads mode.
- **Active time**: Cumulative time in µs the connection is in ACTIVE state.
- **PRCP State**: State of the inbound connection with respect to the Proxy Resident Connection Pool (PRCP) and can be one of the following values:
  - **WAIT**: Connection is waiting for a session from the PRCP.
  - **CHECKED-OUT**: Connection is holding an outbound session from PRCP but not making any OCI calls.
  - **ACTIVE**: Connection is holding an outbound session from PRCP and busy with OCI calls.
  - **CHECKED-IN**: Connection released the CHECKED-OUT session back to the PRCP.
  - **NO STATE**: Clients to a service without a configured PRCP configured have this state.
- **PRCP Wait time, PRCP Checked-out time, and PRCP Active time**: Cumulative time in µs the connection is in PRCP WAIT, CHECKED-OUT, and ACTIVE states. All these three states are zero in case of non-PRCP service.
- **Total Session Gets**: Total count of PRCP session get requests from this connection. It is always 1 if PRCP is not configured.
- **Session Get Hits**: Number of times a session is found existing in the PRCP out of all the requests. It is always 0 if PRCP is not configured.
2.3.25 SHOW DEFAULTS

Use the Oracle Connection Manager Control utility \texttt{SHOW DEFAULTS} command to display default parameter settings.

**Purpose**

To display default parameter settings.

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

\texttt{cmctl SHOW DEFAULTS [-c instance_name]}

From the Oracle Connection Manager Control utility:

\texttt{CMCTL> SHOW DEFAULTS}

**Example**

\texttt{CMCTL> SHOW DEFAULTS}

\begin{verbatim}
listener_address          | (address=(protocol=tcp)(host=users.us.example.com)(port=1521))
aso_authentication_filter | OFF
connection_statistics     | OFF
event_group               | OFF
log_directory             | /disk1/user_cman_test/oracle/network/log/
log_level                 | SUPPORT
max_connections           | 256
idle_timeout              | 0
inbound_connect_timeout   | 0
session_timeout           | 0
outbound_connect_timeout  | 0
max_gateway_processes     | 16
min_gateway_processes     | 2
max_cmctl_sessions        | 4
password                  | OFF
trace_directory           | /disk1/user_cman_test/oracle/network/trace/
trace_level               | OFF
trace_timestamp           | OFF
trace_filelen             | 0
trace_fileno              | 0
\end{verbatim}

The command completed successfully

**REST API for SHOW DEFAULTS Command**

\texttt{GET /show/defaults}
2.3.26 SHOW EVENTS

Use the Oracle Connection Manager Control utility SHOW EVENTS command to display events that are currently operating.

**Purpose**

To display the events that are in operation.

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

```bash
CMCTL> SHOW EVENTS
```

**Example**

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

2.3.27 SHOW GATEWAYS

Use the Oracle Connection Manager Control utility SHOW GATEWAYS command to display the statuses of gateway processes.

**Purpose**

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

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

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

**Arguments**

`gateway`: The identifier of the gateway or gateways whose status to display.
Issuing `SHOW GATEWAYS` without an argument displays the status of all gateway processes.

**Usage Notes**

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

**Example**

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

**REST API for SHOW GATEWAYS Command**

```
POST /show/gateways
JSON Payload
{
  "gateway_ids" : [id1, id2, ... ]
}
```

### 2.3.28 SHOW PARAMETERS

Use the Oracle Connection Manager Control utility `SHOW PARAMETERS` command to display the parameter settings for an instance.

**Purpose**

To display current parameter settings for an instance.

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW PARAMETERS
```

**Usage Notes**

Several configuration parameters can be dynamically modified using the `SET` command. Therefore, the information that `SHOW PARAMETERS` displays might be different from what appears in the `cman.ora` file.
Example

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

REST API for SHOW PARAMETERS Command

GET /show/parameters

2.3.29 SHOW RULES

Use the Oracle Connection Manager Control Utility SHOW RULES command to display an instance access control list.

Purpose

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

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

cmctl SHOW RULES [-c instance_name]

From the Oracle Connection Manager Control utility:

CMCTL> SHOW RULES

cmctl SHOW RULES [-c instance_name]

From the Oracle Connection Manager Control utility:
CMCTL> SHOW RULES

**Usage Notes**

You can update the rules list by issuing the `RELOAD` command.

**Example**

CMCTL> SHOW RULES
Number of filtering rules currently in effect: 5
(rule_list=
  (rule=
    (src=usunnae12)
    (dst=usunnae13)
    (srv=*)
    (act=accept)
    (action_list=(mit=120) (mct=1800) (conn_stats=on) (aut=off)))
  )
  (rule=
    (src=usunnae12)
    (dst=usunnae14)
    (srv=service2)
    (act=accept)
  )
  (rule=
    (src=*)
    (dst=usunnae15)
    (srv=*)
    (act=accept)
    (action_list=(mit=120) (mct=3000) (moct=200) (aut=on))
  )
  )
  (rule=
    (src=*)
    (dst=usunnae16)
    (srv=*)
    (act=reject)
    (action_list=(moct=20) (aut=on))
  )
  )
  (rule=
    (src=users.us.example.com)
    (dst=users.us.example.com)
    (srv=cmon)
    (act=accept)
    (action_list=(mit=100) (mct=1130) (moct=200) (aut=on))
  )
  )
)

**REST API for SHOW RULES Command**

GET /show/rules
2.3.30 SHOW SERVICES

Use the Oracle Connection Manager Control utility `SHOW SERVICES` command to display Oracle Connection Manager instance information.

**Purpose**

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

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW SERVICES
```

**Example**

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

The command completed successfully

**REST API for SHOW SERVICES Command**

```
GET /show/services
```
2.3.31 SHOW STATUS

Use the Oracle Connection Manager Control utility SHOW STATUS command to display Oracle Connection Manager instance information.

**Purpose**

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

**Prerequisites**

Oracle Connection Manager must be running.

**Syntax**

From the operating system:

```
cmctl SHOW STATUS
```

From the Oracle Connection Manager Control utility:

```
CMCTL> SHOW STATUS
```

**Example**

```
CMCTL> SHOW STATUS
Status of the Instance
----------------------
Instance name             CMAN_user.us.example.com
Version                   CMAN for Linux: Version 18.0.0.0.0
Start date                12-JAN-2018 14:50:35
Uptime                    0 days 1 hr. 25 min. 24 sec
Num of gateways started   2
Average Load level        0
Log Level                 SUPPORT
Trace Level               OFF
Instance Config file      /disk1/user_cman_test/oracle/network/admin/cman.ora
Instance Log directory    /disk1/user_cman_test/oracle/network/log/
Instance Trace directory  /disk1/user_cman_test/oracle/network/trace/
```

The command completed successfully

**REST API for SHOW STATUS Command**

```
GET /show/status
```

---

2.3.32 SHOW VERSION

Use the Oracle Connection Manager Control utility SHOW VERSION command

**Purpose**

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

None

Syntax

From the operating system:

cmctl SHOW VERSION [-c instance_name]

From the Oracle Connection Manager Control utility:

CMCTL> SHOW VERSION

Examples

CMCTL> SHOW VERSION
CMAN for Linux: Version 18.0.0.0.0
The command completed successfully

REST API for SHOW VERSION Command

GET /show/version

2.3.33 SHUTDOWN

Use the Oracle Connection Manager Control utility SHUTDOWN command to shut down gateway processes or an entire Oracle Connection Manager instance.

Purpose

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

Prerequisites

Oracle Connection Manager must be running.

Syntax

From the operating system:

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

From the Oracle Connection Manager Control utility:

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

Arguments

gateways: To shut down a specific gateway.

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

abort: To shut down Oracle Connection Manager immediately, and close all open connections.
To specify more than one gateway, separate gateways using a space.

**Usage Notes**

Running the `SHUTDOWN` command without an argument shuts down all of the gateways.

**Example**

CMCTL> SHUTDOWN GATEWAYS 0
The command completed successfully

### 2.3.34 STARTUP

Use the Oracle Connection Manager Control utility `STARTUP` command to start Oracle Connection Manager.

**Purpose**

To start Oracle Connection Manager.

**Prerequisites**

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

**Syntax**

From the operating system:

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

From the Oracle Connection Manager Control utility:

```
CMCTL> STARTUP
```

**Usage Notes**

Before running this command, you must use the `ADMINISTER` command to select an instance to start.

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

The utility may prompt you for a password if you installed Oracle Connection Manager with secure installation option.

**Example**

CMCTL> STARTUP
Starting Oracle Connection Manager instance cman_1. Please wait...
CMAN for Linux: Version 18.0.0.0.0
Status of the Instance
----------------------
<table>
<thead>
<tr>
<th>Instance name</th>
<th>cman_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>CMAN for Linux: Version 18.0.0.0.0</td>
</tr>
<tr>
<td>Start date</td>
<td>22-JAN-2018 01:16:55</td>
</tr>
<tr>
<td>Uptime</td>
<td>0 days 0 hr. 0 min. 9 sec</td>
</tr>
<tr>
<td>Num of gateways started</td>
<td>8</td>
</tr>
<tr>
<td>Average Load level</td>
<td>0</td>
</tr>
</tbody>
</table>

---

Chapter 2

Oracle Connection Manager Control Utility Commands
STARTUP -MIGRATE

Use the STARTUP -MIGRATE parameter to start Oracle Connection Manager (CMAN) in migration mode. You can start a new instance of CMAN in migration mode, and migrate connected sessions from the already running instance of Oracle CMAN.

Prerequisites

A CMAN instance with the same configuration as the new instance must be running in a different ORACLE_HOME on the same host where the new CMAN is being started.

Syntax

From the operating system:

cmctl STARTUP -MIGRATE [-c instance_name]

Usage Notes

This command starts new instance components, such as, the listener, CMADMIN, and the gateway processes.

The new listener inherits the listening endpoints and the listen queue from the old listener. It also accepts new connection requests.

The old gateway processes migrate the connected sessions to new gateways. This migration happens without the client or the server intervention.

The old listener exits after processing the pending connections. The old instance, CMADMIN, and the gateway processes will exit as soon as migration is complete or after 7 minutes timeout.

Use instance_name_old as the instance name to monitor the old instance.

Example

CMCTL STARTUP -MIGRATE -C cman_1
CMCTL for Linux: Version 20.0.0.0.0

Copyright (c) 1996, 2019, Oracle. All rights reserved.

Current instance cman_1_ is already started
Connecting to (DESCRIPTION=(address=(protocol=tcp)(host=localhost)(port=2556)))
CMAN Session Migration Stats

| No of Gateways | 1 |
| Total Connections | 0 |
| TCP Connections | 0 |
| TCPS Connections (Migratable) | 0 |
Starting CMAN Session Migration....
----------------------------------
Old CMADMIN address alias parameter cman_1_old set to
(configuration=(ADDRESS=(PROTOCOL=ipc) (KEY=#124470.1) (KEYPATH=/var/tmp/.oracle_754500)))

Starting Oracle Connection Manager instance cman_1. Please wait...
CMAN for Linux: Version 20.0.0.0.0 - Development
Status of the Instance
---------------------
Instance name             cman_1
Version                   CMAN for Linux: Version 20.0.0.0.0
Start date                29-AUG-2019 05:31:03
Uptime                    0 days 0 hr. 0 min. 9 sec
Num of gateways started   1
Average Load level        0
Log Level                 SUPPORT
Trace Level               SUPPORT
Instance Config file      /network/admin/cman.ora
Instance Log directory $ORACLE_BASE/diag/netcman/node_name/cman_1/alert
Instance Trace directory $ORACLE_BASE/diag/netcman/node_name/cman_1/trace
The command completed successfully.
Now session migration will be initiated by gateways separately....

2.3.35 SUSPEND GATEWAY

Use the Oracle Connection Manager Control utility SUSPEND GATEWAY command to specify the gateway processes than cannot accept new client connections.

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

Prerequisites
Oracle Connection Manager must be running.

Syntax
From the operating system:

cmctl SUSPEND GATEWAY [gateway_process_id] [-c instance_name]

From the Oracle Connection Manager Control utility:

CMCTL> SUSPEND GATEWAY [gateway_process_id]

Arguments

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

Issuing SUSPEND GATEWAY without an argument suspends all gateway processes.
Usage Notes

Use the RESUME GATEWAYS command to enable gateway processes to accept new connections.

Example

CMCTL> SUSPEND GATEWAY 1
The command completed successfully

REST API for SUSPEND GATEWAY Command

POST /suspend
{
  "gateway" : "gateway id"
}
Syntax Rules for Configuration Files

Learn the syntax rules for configuring Oracle Net Services parameters, keywords, addresses, and naming methods.

• Overview of Configuration File Syntax
  Create Oracle Net Services configuration files using syntax rules and standard conventions.

• Syntax Rules for Configuration Files
  Follow the structure, hierarchy, and character requirements for configuration files.

• Network Character Set for Keywords
  Use the permitted character set for keyword values and network character sets.

• Permitted Listener and Net Service Name Character Set
  Create listener names and net service names for clients that comply with Oracle Net Services character set requirements.

3.1 Overview of Configuration File Syntax

Create Oracle Net Services configuration files using syntax rules and standard conventions.

The Oracle Net Services configuration files contain parameters that include keyword-value pairs. Keyword-value pairs are surrounded by parentheses:

\[ \text{parameter} = (\text{keyword} = \text{value}) \]

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

\[ (\text{keyword} = \\
  (\text{keyword1} = \text{value1}) \\
  (\text{keyword2} = \text{value2})) \]

For example, the address portion of the local naming configuration file (tnsnames.ora) can include lines such as:

\[ (\text{ADDRESS} = \\
  (\text{PROTOCOL} = \text{tcp}) \\
  (\text{HOST} = \text{sales-server}) \\
  (\text{PORT} = 1521)) \]

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

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

The following syntax is not acceptable:

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

### 3.2 Syntax Rules for Configuration Files

Follow the structure, hierarchy, and character requirements for configuration files.

The following rules apply to the configuration file syntax:

- Any keyword in a configuration file that begins a parameter that also includes one or more keyword-value pairs must be in the far-left column of a line. If you indent the keyword by one or more spaces, then Oracle interprets the indented keyword as a continuation of the previous line.
- All characters must be part of the network character set.
- Keywords are not case-sensitive. However, values can be case-sensitive depending on your operating system and protocol.
- In keyword-value pairs, spaces around the equal sign (\=) are optional.
- There is a hierarchy of keywords that requires that some keywords are always followed by others. At any level in 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 the values with spaces are enclosed within double quotation marks (""') or single quotation marks (').
- If the keyword-value pair consists of a single word, or a concatenation of words on either side of the equal sign, then no parentheses are needed.
- The maximum length of a connect descriptor is 4KB.
- You can include comments by using the number sign (#) at the beginning of a line. Anything following the number sign to the end of the line is considered a comment.
3.3 Network Character Set for Keywords

Use the permitted character set for keyword values and network character sets.

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

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

Within this character set, the following symbols are reserved:

( ) = \ " ' 

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

You can use the following characters within a connect descriptor, but not in a keyword or value:

- Space
- Tab
- Carriage return
- Newline

3.4 Permitted Listener and Net Service Name Character Set

Create listener names and net service names for clients that comply with Oracle Net Services character set requirements.

Listener names and net service names are limited to the following character set:

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

The first character in the listener name or net service name must be an alphanumeric character. In general, names up to 64 characters are acceptable. In addition, a database service name must match the global database name that is defined by the database administrator, which consists of a database name and the database domain. Both net service names and global database names are not case-sensitive.
Learn how to configure connections for Oracle Database instances and clients.

A network object is identified by a protocol address. When a connection is made, the client and the receiver of the request (a listener or Oracle Connection Manager) are configured with identical protocol addresses. The client uses this address to send the connection request to a particular network object location. The recipient "listens" for requests on this address, and grants connections based on its address information matching the client's information.

- **Protocol Addresses**
  The protocol address comprises ADDRESS and ADDRESS_LIST elements.

- **Protocol Parameters**
  The listener and Oracle Connection Manager are identified by protocol addresses.

- **Recommended Port Numbers**
  Oracle recommends that you use the default port numbers for client and Oracle Connection Manager connections.

- **Port Number Limitations**
  Use this procedure to configure listeners to use a system port number in the 1 to 1024 range.

### 4.1 Protocol Addresses

The protocol address comprises ADDRESS and ADDRESS_LIST elements.

- **ADDRESS**
  The ADDRESS networking parameter specifies the protocol address under the ADDRESS_LIST or DESCRIPTION parameter.

- **ADDRESS_LIST**
  The ADDRESS_LIST networking parameter specifies the number of protocol addresses sharing common characteristics.

#### 4.1.1 ADDRESS

The ADDRESS networking parameter specifies the protocol address under the ADDRESS_LIST or DESCRIPTION parameter.

**Purpose**

To define a protocol address.

**Usage Notes**

Put this parameter under an ADDRESS_LIST or DESCRIPTION parameter. A DESCRIPTION is used in a tnsnames.ora or a listener.ora file.
4.1.2 ADDRESS_LIST

The ADDRESS_LIST networking parameter specifies the number of protocol addresses sharing common characteristics.

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

Usage Notes
This parameter is not mandatory when specifying multiple addresses.

Example

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

ADDRESS=
 (PROTOCOL=tcp)
 (HOST=hr-server)
 (PORT=1521))

4.2 Protocol Parameters

The listener and Oracle Connection Manager are identified by protocol addresses.

The following table lists the parameters that Oracle protocol support uses:
Table 4-1  Protocol-Specific Parameters

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPC</td>
<td>PROTOCOL</td>
<td>Specify <code>ipc</code> as the value.</td>
</tr>
<tr>
<td>IPC</td>
<td>KEYPATH</td>
<td>On UNIX variants, the IPC protocol uses the UNIX domain socket and this socket creates an internal file for client/server communication. The parameter <code>keypath</code> specifies the location where this file is created. If you use <code>keypath</code>, then use the same value of a version greater than 18 on the client and listener sides. Example: <code>(PROTOCOL=ipc) (KEY=sales)</code></td>
</tr>
<tr>
<td>IPC</td>
<td>KEY</td>
<td>Specify a unique name for the service. Oracle recommends using the service name or the Oracle system identifier (SID) of the service. Example: <code>(PROTOCOL=ipc) (KEY=sales)</code></td>
</tr>
<tr>
<td>Named Pipes</td>
<td>PROTOCOL</td>
<td>Specify <code>nmp</code> as the value.</td>
</tr>
<tr>
<td>Named Pipes</td>
<td>SERVER</td>
<td>Specify the Oracle server name.</td>
</tr>
<tr>
<td>Named Pipes</td>
<td>PIPE</td>
<td>Specify the pipe name used to connect to the database server. This is the same <code>PIPE</code> keyword specified on the server with Named Pipes. This name can be any name. Example: <code>(PROTOCOL=nmp) (SERVER=sales) (PIPE=dbpipe0)</code></td>
</tr>
<tr>
<td>SDP</td>
<td>PROTOCOL</td>
<td>Specify <code>sdp</code> as the value. Example: <code>(PROTOCOL=sdp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>SDP</td>
<td>HOST</td>
<td>Specify the host name or IP address of the computer. Example: <code>(PROTOCOL=sdp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>SDP</td>
<td>PORT</td>
<td>Specify the listening port number. Example: <code>(PROTOCOL=sdp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>TCP/IP</td>
<td>PROTOCOL</td>
<td>Specify <code>tcp</code> as the value. Example: <code>(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>TCP/IP</td>
<td>HOST</td>
<td>Specify the host name or IP address of the computer. Example: <code>(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>TCP/IP</td>
<td>PORT</td>
<td>Specify the listening port number. Example: <code>(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521)</code></td>
</tr>
<tr>
<td>TCP/IP with SSL</td>
<td>PROTOCOL</td>
<td>Specify <code>tcps</code> as the value. Example: <code>(PROTOCOL=tcps) (HOST=sales-server) (PORT=2484)</code></td>
</tr>
<tr>
<td>TCP/IP with SSL</td>
<td>HOST</td>
<td>Specify the host name or IP address of the computer. Example: <code>(PROTOCOL=tcps) (HOST=sales-server) (PORT=2484)</code></td>
</tr>
<tr>
<td>TCP/IP with SSL</td>
<td>PORT</td>
<td>Specify the listening port number. Example: <code>(PROTOCOL=tcps) (HOST=sales-server) (PORT=2484)</code></td>
</tr>
<tr>
<td>Exadirect</td>
<td>PROTOCOL</td>
<td>Specify <code>exadirect</code> as the value.</td>
</tr>
<tr>
<td>Exadirect</td>
<td>HOST</td>
<td>Specify the IP address of the InfiniBand interface.</td>
</tr>
</tbody>
</table>
Table 4-1  (Cont.) Protocol-Specific Parameters

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exadirect</td>
<td>PORT</td>
<td>Specify the listening port number. Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(PROTOCOL=exadirect) (HOST=sales-server) (PORT=2484)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(PROTOCOL=tcp) (HOST=192.0.2.204) (PORT=1522)</td>
</tr>
<tr>
<td>Websocket</td>
<td>PROTOCOL</td>
<td>Specify <em>ws</em> as the value; use this protocol only as a web server back-end</td>
</tr>
<tr>
<td>Websocket</td>
<td>HOST</td>
<td>Specify the host name or IP address of the computer.</td>
</tr>
<tr>
<td>Websocket</td>
<td>PORT</td>
<td>Specify the listening port number. Example:</td>
</tr>
<tr>
<td>Secure Websocket</td>
<td>PROTOCOL</td>
<td>Specify <em>ws</em> as the value; use this protocol on the client side to connect to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a web server with websocket protocol support. Configure the web server to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>make a websocket connection to the database listener. Configure the wallet</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in sqlnet.ora. Use SQLNET.URI for mapping on web server.</td>
</tr>
<tr>
<td>Secure Websocket</td>
<td>HOST</td>
<td>Specify the host name or IP address of the web server with websocket support.</td>
</tr>
<tr>
<td>Secure Websocket</td>
<td>PORT</td>
<td>Specify the listening port number. Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(protocol=wss) (host=sales-server) (port=1524)</td>
</tr>
</tbody>
</table>

4.3 Recommended Port Numbers

Oracle recommends that you use the default port numbers for client and Oracle Connection Manager connections.

Table 4-2  Recommended Port Numbers

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1521</td>
<td>Default listening port for client connections to the listener. This port</td>
</tr>
<tr>
<td></td>
<td>number can change to the officially registered port number of 2483 for TCP/</td>
</tr>
<tr>
<td></td>
<td>IP, and 2484 for TCP/IP with SSL.</td>
</tr>
<tr>
<td>1521</td>
<td>Default and officially registered listening port for client connections</td>
</tr>
<tr>
<td>1830</td>
<td>Default and officially registered listening port for administrative</td>
</tr>
<tr>
<td></td>
<td>commands to Oracle Connection Manager.</td>
</tr>
</tbody>
</table>

4.4 Port Number Limitations

Use this procedure to configure listeners to use a system port number in the 1 to 1024 range.
Oracle accepts port numbers from 1 to 65535. However, port numbers below 1024 are typically reserved. Only privileged processes can listen for TCP connections on ports below 1024.

To configure a listener to listen on a port number lower than 1024, complete the following procedure:

**Note:**

This procedure is a guideline. Your operating system can require a different procedure.

1. Use Oracle Net Configuration Assistant or Oracle Net Manager to configure the listener with protocol addresses and other configuration parameters.
2. Log in as the root user on the machine that has the listener.
3. Set file ownership and access permissions for the listener executable (`tnslsnr`) and the dependent shared libraries, so that these files can be modified only by the root user.
4. Starting with the root directory, ensure that the permissions of the individual directories in the path names to these files share the same ownership and access permissions.
5. Start the listener as the root user.
6. Enter the following command at the prompt:
   ```
   tnslsnr listener_name -user user -group group
   ```

   In the preceding command, the following options are used:

   **Table 4-3 tnslsnr Utility Options**

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>listener_name</code></td>
<td>Specify the name of the listener to configure. If omitted, then the default name LISTENER is used.</td>
</tr>
<tr>
<td><code>user</code></td>
<td>Specify the user whose privileges you want the listener to use when super user (root) privileges are not needed. After performing the privileged operations, the listener gives up root privileges irreversibly.</td>
</tr>
<tr>
<td><code>group</code></td>
<td>Specify the group whose privileges you want the listener to use when super user (root) group privileges are not needed. After performing the privileged operations, the listener gives up root group privileges irreversibly.</td>
</tr>
</tbody>
</table>

   During this step, the listener changes from root to the user and group privileges that you specify. All operations are done with the specified user and group privileges, except for the issuing of the system calls that are needed to listen on configured endpoints. The listener reverts to the root user to listen on reserved addresses, such as TCP port numbers that are lower than 1024.
After the listener starts listening on all of its endpoints that you configured in the listener.ora file, it permanently switches to the specified user and group. At that point, the listener gives up the root privilege that it initially had. The -user and -group command line arguments only accept user and group identifiers specified in numeric form.

For example, to run a listener called mylsnr with root privileges, and to have it use privileges of the Oracle user with the user identifier (UID) of 37555, and with OSDBA group dba membership, with a group identifier (GID) of 16, enter the following command at the prompt:

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

7. After the listener starts, you can administer it with Listener Control utility.

⚠️ Caution:

- Oracle recommends that the user under whose privileges the listener process runs is the oracle user, or a similarly privileged user with whose privileges the listener process normally runs on the operating system.
- Do not leave the listener process running as the root user. Running processes as the super user is a security risk.
This chapter describes the sqlnet.ora file parameters.

- **Overview of Profile Configuration Files**
  Learn about profile configuration files.

- **Profile Parameters in sqlnet.ora Files**
  This section describes sqlnet.ora file parameters that you use to administer listeners.

- **ADR Diagnostic Parameters in sqlnet.ora**
  Diagnostic data for critical errors is stored in the sqlnet.ora Automatic Diagnostic Repository (ADR).

- **Non-ADR Diagnostic Parameters in sqlnet.ora Files**
  Learn about sqlnet.ora parameters that you use when you disable ADR.

### 5.1 Overview of Profile Configuration Files

Learn about profile configuration files.

The sqlnet.ora file is the Net Services profile configuration file. The sqlnet.ora file resides on clients and databases. You store and implement profiles using this file. You can also configure the database with access control parameters in the sqlnet.ora file.

These parameters specify whether clients are allowed or denied access to a database based on the parameter settings.

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

Oracle Net searches for the sqlnet.ora file in the following locations and in the following order:

- In the directory specified in the TNS_ADMIN environment variable, if it is set.
- In the ORACLE_BASE_HOME/network/admin directory.
- In the ORACLE_HOME/network/admin directory.
5.2 Profile Parameters in sqlnet.ora Files

This section describes sqlnet.ora file parameters that you use to administer listeners.

This section describes the following sqlnet.ora file parameters:

- **ACCEPT_MD5_CERTS**
  The sqlnet.ora profile parameter ACCEPT_MD5_CERTS accepts MD5 signed certificates.

- **ACCEPT_SHA1_CERTS**
  Use the sqlnet.ora profile parameter ACCEPT_SHA1_CERTS to determine whether SQL Net accepts SHA1 signed certificates.

- **ADD_SSLV3_TO_DEFAULT**
  The sqlnet.ora profile parameter ADD_SSLV3_TO_DEFAULT sets the secure sockets layer versions that your server accepts.

- **EXADIRECT_FLOW_CONTROL**
  The sqlnet.ora profile parameter EXADIRECT_FLOW_CONTROL enables or disables Exadirect flow control.

- **EXADIRECT_RECVPOLL**
  Use the sqlnet.ora parameter EXADIRECT_RECVPOLL to specify the amount of time that a receiver polls for incoming data.

- **DEFAULT_SDU_SIZE**
  Use the sqlnet.ora profile parameter to specify the session data unit size (SDU) for connections.
• **DISABLE_OOB**
  Use the sqlnet.ora profile parameter DISABLE_OOB to enable or disable Oracle Net to send or receive out-of-band break messages using urgent data from the underlying protocol.

• **DISABLE_OOB_AUTO**
  Use the sqlnet.ora profile parameter DISABLE_OOB_AUTO to disable server path checks for out-of-band break messages at the time of the connection.

• **HTTPS_SSL_VERSION**
  Use the sqlnet.ora profile parameter HTTPS_SSL_VERSION to control the Secure Sockets Later (SSL) that XDB HTTPS connections use.

• **IPC.KEYPATH**
  Use the sqlnet.ora profile parameter IPC.KEYPATH to specify the destination directory where the internal file is created for UNIX domain sockets.

• **NAMES.DEFAULT_DOMAIN**
  Use the sqlnet.ora profile parameter NAMES.DEFAULT_DOMAIN to set the name of the domain in which clients most often look up names resolution requests.

• **NAMES.DIRECTORY_PATH**
  Use the sqlnet parameter NAMES.DIRECTORY_PATH to specify the order of the naming methods for client name resolution lookups.

• **NAMES.LDAP_AUTHENTICATE_BIND**
  Use the sqlnet parameter NAMES.LDAP_AUTHENTICATE_BIND to specify whether the LDAP naming adapter should authenticate using a specified wallet when it connects to the LDAP directory to resolve connect string names.

• **NAMES.LDAP_CONN_TIMEOUT**
  Use the sqlnet parameter NAMES.LDAP_CONN_TIMEOUT to specify the number of seconds that indicates that a non-blocking connect timeout to the LDAP server occurred.

• **NAMES.LDAP_PERSISTENT_SESSION**
  Use the sqlnet parameter NAMES.LDAP_PERSISTENT_SESSION to specify whether the LDAP naming adapter should leave the session with the LDAP server open after name lookups are complete.

• **NAMES.NIS.META_MAP**
  Use the sqlnet parameter NAMES.NIS.META_MAP to specify the map file to use to map Network Information Service (NIS) attributes to an NIS mapname.

• **RECV_BUF_SIZE**
  Use the sqlnet parameter RECV_BUF_SIZE to specify the buffer space limit for session receive operations.

• **SDP.PF_INET_SDP**
  Use the sqlnet parameter SDP.PF_INET_SDP to specify the protocol family or address family constant for the SDP protocol on your system.

• **SEC_USER_AUDIT_ACTION_BANNER**
  Use the sqlnet parameter SEC_USER_AUDIT_ACTION_BANNER to specify a text file that contains the banner contents that warn users about user action auditing.

• **SEC_USER_UNAUTHORIZED_ACCESS_BANNER**
  Use the sqlnet parameter SEC_USER_UNAUTHORIZED_ACCESS_BANNER to specify the file that contains the banner contents that warn users about unauthorized database access.
- **SEND_BUF_SIZE**
  Use the `sqlnet` parameter `SEND_BUF_SIZE` to specify the buffer space limit for session send operations.

- **SQLNET.ALLOWED_LOGON_VERSION_CLIENT**
  Use the `sqlnet` parameter `SQLNET.ALLOWED_LOGON_VERSION_CLIENT` to define minimum authentication protocols that servers acting as clients to other servers can use to connect to Oracle Database instances.

- **SQLNET.ALLOWED_LOGON_VERSION_SERVER**
  Use the `sqlnet.ora` parameter `SQLNET.ALLOWED_LOGON_VERSION_SERVER` to set the minimum authentication protocol that is permitted when connecting to Oracle Database instances.

- **SQLNET.AUTHENTICATION_SERVICES**
  Use the `sqlnet.ora` parameter `SQLNET.AUTHENTICATION_SERVICES` to enable one or more authentication services.

- **SQLNET.CLIENT_REGISTRATION**
  Use the `sqlnet.ora` parameter `SQLNET.CLIENT_REGISTRATION` to set a unique identifier for the client computer.

- **SQLNET.CLOUD_USER**
  Use the `sqlnet.ora` parameter `SQLNET.CLOUD_USER` to specify a user name for web server HTTP basic authentication.

- **SQLNET.COMPRESSION**
  Use the `sqlnet.ora` parameter `SQLNET.COMPRESSION` to enable or disable data compression.

- **SQLNET.COMPRESSION_ACCELERATION**
  Use the `sqlnet.ora` parameter `SQLNET.COMPRESSION_ACCELERATION` to specify the use of hardware accelerated version of compression using this parameter if it is available for that platform.

- **SQLNET.COMPRESSION_LEVELS**
  Use the `sqlnet.ora` parameter `SQLNET.COMPRESSION_LEVELS` to specify the compression level.

- **SQLNET.COMPRESSION_THRESHOLD**
  Use the `sqlnet.ora` parameter `SQLNET.COMPRESSION_THRESHOLD` to specify the minimum data size for which compression is needed.

- **SQLNET.CRYPTO_CHECKSUM_CLIENT**
  Use the `sqlnet.ora` parameter `SQLNET.CRYPTO_CHECKSUM_CLIENT` to specify the checksum behavior for clients.

- **SQLNET.CRYPTO_CHECKSUM_SERVER**
  Use the `sqlnet.ora` parameter `SQLNET.CRYPTO_CHECKSUM_SERVER` to specify the checksum behavior for the database.

- **SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT**
  Use the `sqlnet.ora` parameter `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT` to specify a list of crypto-checksum algorithms for the client to use.

- **SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER**
  Use the `sqlnet.ora` parameter `SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER` to specify a list of crypto-checksum algorithms for the database to use.
• **SQLNET.DBFW_PUBLIC_KEY**
  Use the `sqlnet.ora` parameter `SQLNET.DBFW_PUBLIC_KEY` to provide Oracle Database Firewall public keys to the Advanced Security Option (ASO) by specifying the file that stores the public keys.

• **SQLNET.DOWN_HOSTS_TIMEOUT**
  Use the `sqlnet.ora` parameter `SQLNET.DOWN_HOSTS_TIMEOUT` to specify the amount of time in seconds that server hosts down state information remains in the client cache.

• **SQLNET.ENCRYPTION_CLIENT**
  Use the `sqlnet.ora` parameter `SQLNET.ENCRYPTION_CLIENT` to enable encryption for clients.

• **SQLNET.ENCRYPTION_SERVER**
  Use the `sqlnet.ora` parameter `SQLNET.ENCRYPTION_SERVER` to enable database encryption.

• **SQLNET.ENCRYPTION_TYPES_CLIENT**
  Use the `sqlnet.ora` parameter `SQLNET.ENCRYPTION_TYPES_CLIENT` to list encryption algorithms for clients to use.

• **SQLNET.ENCRYPTION_TYPES_SERVER**
  Use the `sqlnet.ora` parameter `SQLNET.ENCRYPTION_TYPES_SERVER` to list the encryption algorithms for the database to use.

• **SQLNET.EXPIRE_TIME**
  Use the `sqlnet.ora` parameter `SQLNET.EXPIRE_TIME` to specify how often, in minutes, to verify that client and server connections are active.

• **SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS**
  Use the `sqlnet.ora` parameter `SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS` to ignore the value that is set for the parameter `SQLNET.ENCRYPTION_SERVER` for TCPS connections. This disables ANO encryption on the TCPS listener.

• **SQLNET.INBOUND_CONNECT_TIMEOUT**
  Use the `sqlnet.ora` parameter `SQLNET.INBOUND_CONNECT_TIMEOUT` to specify the amount of time that clients have to connect with the database and authenticate.

• **SQLNET.FALLBACK_AUTHENTICATION**
  Use the `sqlnet.ora` parameter `SQLNET.FALLBACK_AUTHENTICATION` to specify whether to attempt password-based authentication if Kerberos authentication fails.

• **SQLNET.KERBEROS5_CC_NAME**
  Use the `sqlnet.ora` parameter `SQLNET.KERBEROS5_CC_NAME` to specify the complete path name to the Kerberos credentials cache file.

• **SQLNET.KERBEROS5_CLOCKSKEW**
  Use the `sqlnet.ora` parameter `SQLNET.KERBEROS5_CLOCKSKEW` to specify how much time elapses before a Kerberos credential is considered out-of-date.

• **SQLNET.KERBEROS5_CONF**
  Use the `sqlnet.ora` parameter `SQLNET.KERBEROS5_CONF` to specify the path name to the Kerberos configuration file that contains the realm for the default Key Distribution Center (KDC) and that maps realms to KDC hosts.

• **SQLNET.KERBEROS5_CONF_LOCATION**
  Use the `sqlnet.ora` parameter `SQLNET.KERBEROS5_CONF_LOCATION` to specify the directory for the Kerberos configuration file. The `SQLNET.KERBEROS5_CONF_LOCATION` parameter also specifies that the file is created by the system and not by the client.
• **SQLNET.KERBEROS5_KEYTAB**
  Use the sqlnet.ora parameter SQLNET.KERBEROS5_KEYTAB to specify the path name to the Kerberos principal or, secret, key mapping file that extracts keys and decrypts incoming authentication information.

• **SQLNET.KERBEROS5_REALMS**
  Use the sqlnet.ora parameter SQLNET.KERBEROS5_REALMS to specify the complete path name to the Kerberos realm translation file that maps a host name or domain name to a realm.

• **SQLNET.KERBEROS5_REPLAY_CACHE**
  Use the sqlnet.ora parameter SQLNET.KERBEROS5_REPLAY_CACHE to specify that the replay cache is stored in operating system-managed memory on the server, and that file-based replay cache is not used.

• **SQLNET.OUTBOUND_CONNECT_TIMEOUT**
  Use the sqlnet.ora parameter SQLNET.OUTBOUND_CONNECT_TIMEOUT to specify the amount of time, in milliseconds, seconds, or minutes, in which clients must establish Oracle Net connections to database instances.

• **SQLNET.RADIUS_ALTERNATE**
  Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE to specify an alternate RADIUS server if the primary server is unavailable.

• **SQLNET.RADIUS_ALTERNATE_PORT**
  Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE_PORT to specify the listening port of the alternate RADIUS server.

• **SQLNET.RADIUS_ALTERNATE_RETRIES**
  Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE_RETRIES to specify the number of times that the database resends messages to alternate RADIUS servers.

• **SQLNET.RADIUS_AUTHENTICATION**
  Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION to specify a primary RADIUS server location, either by its host name or its IP address.

• **SQLNET.RADIUS_AUTHENTICATION_INTERFACE**
  Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_INTERFACE to specify the class that contains the user interface for interacting with users.

• **SQLNET.RADIUS_AUTHENTICATION_PORT**
  Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_PORT to specify the listening port of a primary RADIUS server.

• **SQLNET.RADIUS_AUTHENTICATION_RETRIES**
  Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_RETRIES to specify the number of times the database should resend messages to a primary RADIUS server.

• **SQLNET.RADIUS_AUTHENTICATION_TIMEOUT**
  Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_TIMEOUT to specify the amount of time that the database should wait for a response from a primary RADIUS server.

• **SQLNET.RADIUS_CHALLENGE_RESPONSE**
  Use the sqlnet.ora parameter SQLNET.RADIUS_CHALLENGE_RESPONSE to enable or disable challenge responses.
• **SQLNET.RADIUS_SECRET**
  Use the `sqlnet.ora` parameter `SQLNET.RADIUS_SECRET` to specify the location of a RADIUS secret key.

• **SQLNET.RADIUS_SEND_ACCOUNTING**
  Use the `sqlnet.ora` parameter `SQLNET.RADIUS_SEND_ACCOUNTING` to enable and disable accounting.

• **SQLNET.RECV_TIMEOUT**
  Use the `sqlnet.ora` parameter `SQLNET.RECV_TIMEOUT` to specify the duration of time that a database client or server should wait for data from a peer after establishing a connection.

• **SQLNET.SEND_TIMEOUT**
  Use the `sqlnet.ora` parameter `SQLNET.SEND_TIMEOUT` to specify the duration of time in which a database must complete send operations to clients after establishing connections.

• **SQLNET.URI**
  Use the `sqlnet.ora` parameter `SQLNET.URI` to specify a database client URI mapping on a web server.

• **SQLNET.USE_HTTPS_PROXY**
  Use the `sqlnet.ora` parameter `SQLNET.USE_HTTPS_PROXY` to enable forward HTTP proxy tunneling for client connections.

• **SQLNET.WALLET_OVERRIDE**
  Use the `sqlnet.ora` parameter `SQLNET.WALLET_OVERRIDE` to determine whether a client should override strong authentication credentials with the password credential from the stored wallet.

• **SSL_CERT_REVOCATION**
  Use the `sqlnet.ora` parameter `SSL_CERT_REVOCATION` to configure revocation checks for certificates.

• **SSL_CRL_FILE**
  Use the `sqlnet.ora` parameter `SSL_CRL_FILE` to specify the name of the file in which you assemble the CRL for client authentication.

• **SSL_CRL_PATH**
  Use the `sqlnet.ora` parameter `SSL_CRL_PATH` to

• **SSL_CIPHER_SUITES**
  Use the `sqlnet.ora` parameter `SSL_CIPHER_SUITES` to control which combination of encryption and data integrity that the Secure Sockets Layer (SSL) uses.

• **SSL_EXTENDED_KEY_USAGE**
  Use the `sqlnet.ora` parameter `SSL_EXTENDED_KEY_USAGE` to specify the purpose certificate keys.

• **SSL_SERVER_DN_MATCH**
  Use the `sqlnet.ora` parameter `SSL_SERVER_DN_MATCH` to enforce server-side certification validation through distinguished name (DN) matching.

• **SSL_VERSION**
  Use the `sqlnet.ora` parameter `SSL_VERSION` to limit the valid SSL or TLS versions that Oracle uses for connections.

• **TCP.CONNECT_TIMEOUT**
  Use the `sqlnet.ora` parameter `TCP.CONNECT_TIMEOUT` to specify the amount of time in which a client must establish TCP connections to database servers.
5.2.1 ACCEPT_MD5_CERTS

The sqlnet.ora profile parameter ACCEPT_MD5_CERTS accepts MD5 signed certificates.

Purpose

To enable sqlnet to accept MD5 signed certificates. In addition to sqlnet.ora, you must also set this parameter must in listener.ora.

Default

FALSE

Values

- TRUE to accept MD5 signed certificates
5.2.2 ACCEPT_SHA1_CERTS

Use the sqlnet.ora profile parameter ACCEPT_SHA1_CERTS to determine whether SQL Net accepts SHA1 signed certificates.

**Purpose**

To determine whether sqlnet accepts SHA1 signed certificates. In addition to setting this parameter in sqlnet.ora, you must also set this parameter in listener.ora.

**Default**

TRUE

**Values**

- TRUE to accept SHA1 signed certificates
- FALSE to not accept SHA1 signed certificates

5.2.3 ADD_SSLV3_TO_DEFAULT

The sqlnet.ora profile parameter ADD_SSLV3_TO_DEFAULT sets the secure sockets layer versions that your server accepts.

**Purpose**

To set the secure sockets layer versions that your server accepts.

To use SSL_VERSION=3.0 in your SSL_VERSION default list, set the value to TRUE. In addition to setting this parameter in sqlnet.ora, you must also set this parameter in listener.ora.

**Default**

FALSE

**Values**

- If set to TRUE and if SSL_VERSION is not specified or is set to "undetermined", then SSL_VERSION includes versions 3.0, 1.2, 1.1, and 1.0.
- If set to FALSE and if SSL_VERSION is not specified or is set to "undetermined", then SSL_VERSION includes versions 1.2, 1.1, and 1.0

5.2.4 EXADIRECT_FLOW_CONTROL

The sqlnet.ora profile parameter EXADIRECT_FLOW_CONTROL enables or disables Exadirect flow control.

**Purpose**

To enable or disable Exadirect flow control.
Usage Notes
Set to on, the parameter enables Oracle Net to broadcast the available receive window to the sender. The sender limits the sends based on the receiver broadcast window.

Default
off

Example
EXADIRECT_FLOW_CONTROL=on

5.2.5 EXADIRECT_RECV POLL

Use the sqlnet.ora parameter EXADIRECT_RECV POLL to specify the amount of time that a receiver polls for incoming data.

Purpose
To specify the amount of time that a receiver polls for incoming data.

Usage Notes
You can set the parameter to a fixed value or set the parameter to AUTO to automatically tune the polling value.

Default
0

Example
EXADIRECT_RECV POLL = 10
EXADIRECT_RECV POLL = AUTO

5.2.6 DEFAULT_SDU_SIZE

Use the sqlnet.ora profile parameter to specify the session data unit size (SDU) for connections.

Purpose
To specify the session data unit (SDU) size, in bytes, for connections.

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

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

Use the `sqlnet.ora` profile parameter `DISABLE_OOB` to enable or disable Oracle Net to send or receive out-of-band break messages using urgent data from the underlying protocol.

Purpose

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

Usage Notes

Set to `off`, the parameter enables Oracle Net to send and receive break messages. Set to `on`, the parameter disables the ability to send and receive break messages. Once enabled, this feature applies to all protocols that the client uses.

Default

default

Example 5-2  Example

`DISABLE_OOB=on`

5.2.8 DISABLE_OOB_AUTO

Use the `sqlnet.ora` profile parameter `DISABLE_OOB_AUTO` to disable server path checks for out-of-band break messages at the time of the connection.

Purpose

To disable `sqlnet.ora` from checking for out-of-band (OOB) break messages in the server path at connection time.

Usage Notes

By default, the client determines if the server path supports out-of-band break messages at the time of establishing the connection. If `DISABLE_OOB_AUTO` is set to `TRUE`, then the client does not perform this check at connection time.

Default

false
5.2.9 HTTPS_SSL_VERSION

Use the sqlnet.ora profile parameter HTTPS_SSL_VERSION to control the Secure Sockters Later (SSL) that XDB HTTPS connections use.

Purpose

Use HTTPS_SSL_VERSION to control the Secure Sockets Layer (SSL) version that XDB HTTPS connections use.

Usage Notes

Note that the SSL_VERSION parameter no longer controls the SSL version that HTTPS uses. You can set this parameter to any valid HTTPS_SSL_VERSION values.

Default

1.1 or 1.2, meaning TLSv1.1 or TLSv1.2.

Values

Any valid HTTPS_SSL_VERSION value.

5.2.10 IPC.KEYPATH

Use the sqlnet.ora profile parameter IPC.KEYPATH to specify the destination directory where the internal file is created for UNIX domain sockets.

Purpose

To specify the destination directory where the internal file is created for UNIX domain sockets.

Usage Notes

This parameter applies only to Oracle Net usage of UNIX domain sockets and does not apply to other uses of UNIX domain sockets in Oracle Database, such as in Oracle Clusterware. If you use the IPC.KEYPATH parameter, then you should use the same value for IPC_KEYPATH on both the client and the listener on Oracle Database versions that are greater than Oracle Database 18c.

Default

The directory path is either /var/tmp/.oracle for Oracle Linux, Oracle Solaris or /tmp/.oracle for other UNIX variants.

Example

ipc.keypath=/home/oracleuser.
5.2.11 NAMES.DEFAULT_DOMAIN

Use the sqlnet.ora profile parameter NAMES.DEFAULT_DOMAIN to set the name of the domain in which clients most often look up names resolution requests.

Purpose
To set the domain from which the client most often looks up names resolution requests.

Usage Notes
When you set NAMES.DEFAULT_DOMAIN, the default domain name is automatically appended to any unqualified net service name or service name.

For example, if you set the default domain to www.example.com, then Oracle searches the connect string CONNECT scott@sales as www.example.com. If the connect string includes the domain extension, such as CONNECT scott@sales.www.example.com, then the domain is not appended to the string.

Default
None

Example
NAMES.DEFAULT_DOMAIN=example.com

5.2.12 NAMES.DIRECTORY_PATH

Use the sqlnet parameter NAMES.DIRECTORY_PATH to specify the order of the naming methods for client name resolution lookups.

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

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

Values
The following table shows the NAMES.DIRECTORY_PATH values for the naming methods.

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

**Example**

NAMES.DIRECTORY_PATH=(tnsnames)

### 5.2.13 NAMES.LDAP_AUTHENTICATE_BIND

Use the sqlnet parameter NAMES.LDAP_AUTHENTICATE_BIND to specify whether the LDAP naming adapter should authenticate using a specified wallet when it connects to the LDAP directory to resolve connect string names.

**Purpose**

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

**Usage Notes**

The parameter value is Boolean.

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

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

**Default**

false

**Example**

NAMES.LDAP_AUTHENTICATE_BIND=true

### 5.2.14 NAMES.LDAP_CONN_TIMEOUT

Use the sqlnet parameter NAMES.LDAP_CONN_TIMEOUT to specify the number of seconds that indicates that a non-blocking connect timeout to the LDAP server occurred.

**Purpose**

The parameter value -1 is for infinite timeout.

**Default**

15 seconds
Values

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

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

Usage Notes

Example

names.ldap_conn_timeout = -1

5.2.15 NAMES.LDAP_PERSISTENT_SESSION

Use the sqlnet parameter NAMES.LDAP_PERSISTENT_SESSION to specify whether the LDAP naming adapter should leave the session with the LDAP server open after name lookups are complete.

Purpose

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

Usage Notes

The parameter value is Boolean.

If you set the parameter to TRUE, then the connection to the LDAP server is left open after the name lookup is complete. The connection remains open for the duration of the process. If the connection is lost, then it is re-established as needed.

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

Default

false

Example

NAMES.LDAP_PERSISTENT_SESSION=true

5.2.16 NAMES.NIS.META_MAP

Use the sqlnet parameter NAMES.NIS.META_MAP to specify the map file to use to map Network Information Service (NIS) attributes to an NIS mapname.

Purpose

To specify the map file to be used to map Network Information Service (NIS) attributes to an NIS mapname.
5.2.17 RECV_BUF_SIZE

Use the sqlnet parameter RECV_BUF_SIZE to specify the buffer space limit for session receive operations.

Purpose
To specify the buffer space limit for receive operations of sessions.

Usage Notes
You can override this parameter for a particular client connection by specifying the RECV_BUF_SIZE parameter in the connect descriptor for a client.

This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

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

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

**Default**
The default value for this parameter is operating system specific. The default for Linux 2.6 operating system is 87380 bytes.

**Example**
RECV_BUF_SIZE=11784
5.2.18 SDP.PF_INET_SDP

Use the sqlnet parameter SDP.PF_INET_SDP to specify the protocol family or address family constant for the SDP protocol on your system.

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

**Default**
27

**Values**
Any positive integer

**Example**
SDP.PF_INET_SDP=30

5.2.19 SEC_USER_AUDIT_ACTION_BANNER

Use the sqlnet parameter SEC_USER_AUDIT_ACTION_BANNER to specify a text file that contains the banner contents that warn users about user action auditing.

**Purpose**
To specify a text file containing the banner contents that warn users about possible user action auditing.

**Usage Notes**
You must specify the complete path of the text file in the sqlnet.ora file on the server. Oracle Call Interface (OCI) applications can use OCI features to retrieve this banner and display it to users.

**Default**
None

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

**Example**
SEC_USER_AUDIT_ACTION_BANNER=/opt/oracle/admin/data/auditwarning.txt
5.2.20 SEC_USER_UNAUTHORIZED_ACCESS_BANNER

Use the sqlnet parameter SEC_USER_UNAUTHORIZED_ACCESS_BANNER to specify the file that contains the banner contents that warn users about unauthorized database access.

Purpose

To specify the name of a text file containing the banner contents that warn users about unauthorized access to the database.

Usage Notes

You must specify the complete path of the text file in the sqlnet.ora file on the server. OCI applications can use OCI features to retrieve this banner and display it to users.

Default

None

Values

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

Example

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

5.2.21 SEND_BUF_SIZE

Use the sqlnet parameter SEND_BUF_SIZE to specify the buffer space limit for session send operations.

Purpose

To specify the buffer space limit for send operations of sessions.

Usage Notes

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

This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Note:

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

The default value for this parameter is operating system specific. The default for Linux 2.6 operating systems is 16 KB.

Example

```
SEND_BUF_SIZE=11784
```

### 5.2.22 SQLNET.ALLOWED_LOGON_VERSION_CLIENT

Use the `sqlnet` parameter `SQLNET.ALLOWED_LOGON_VERSION_CLIENT` to define minimum authentication protocols that servers acting as clients to other servers can use to connect to Oracle Database instances.

**Purpose**

To set the minimum authentication protocol allowed for clients when a server is acting as a client, such as connecting over a database link, when connecting to Oracle Database instances.

**Usage Notes**

The term `VERSION` in the parameter name refers to the version of the authentication protocol, not the version of the Oracle Database release.

If the version does not meet or exceed the value defined by this parameter, then authentication fails with an ORA-28040: No matching authentication protocol error.

**Values**

- 12 for the critical patch updates CPUOct2012 and later Oracle Database 11g authentication protocols (stronger protection)

**Note:**

Using this setting, the clients can only authenticate using a de-optimized password version. For example, the 12C password version.
Note:

Using this setting, the clients can only authenticate using a password hash value that uses salt. For example, the 11G or 12C password versions.

- 11 for Oracle Database 11g authentication protocols (default)
- 10 for Oracle Database 10g authentication protocols

Default

11

Example

If an Oracle Database 12c database hosts a database link to an Oracle Database 10g database, then set the SQLNET.ALLOWED_LOGON_VERSION_CLIENT parameter as follows for the database link connection to proceed:

```
SQLNET.ALLOWED_LOGON_VERSION_CLIENT=10
```

See Also:

Oracle Database Reference

5.2.23 SQLNET.ALLOWED_LOGON_VERSION_SERVER

Use the sqlnet.ora parameter SQLNET.ALLOWED_LOGON_VERSION_SERVER to set the minimum authentication protocol that is permitted when connecting to Oracle Database instances.

Purpose

To set the minimum authentication protocol for connecting to Oracle Database instances.

Usage Notes

The term VERSION in the parameter name refers to the version of the authentication protocol, not the Oracle Database release.

The authentication fails with the error ORA-28040: No matching authentication protocol error or an ORA-03134: Connections to this server version are no longer supported. This error appears if the client does not have the ability listed in the "Ability Required of the Client" column that corresponds to the row that matches the value of the SQLNET.ALLOWED_LOGON_VERSION_SERVER parameter in Table 1.
A setting of 8 enables all password versions, and enables any combination of the 
DBA_USERS.PASSWORD_VERSIONS values 10G, 11G, and 12C.

A setting of 12a enables only the 12C password version.

A greater value means that the server is less compatible in terms of the protocol that 
clients must understand in order to authenticate. The server is also more restrictive in 
terms of the password version that must exist to authenticate any specific account. 
Whether a client can authenticate to a specific account depends on both the server’s 
setting of its SQLNET.ALLOWED_LOGON_VERSION_SERVER parameter, as well as on the 
password versions that exist for the specified account. You can see the list of 
password versions in file DBA_USERS.PASSWORD_VERSIONS.

Note the following implications of setting the value to 12 or 12a:

- Do not use the value of FALSE for the SEC_CASE_SENSITIVE_LOGON Oracle instance 
  initialization parameter because password case insensitivity requires the use of 
  the 10G password version. If you set the SEC_CASE_SENSITIVE_LOGON Oracle 
  instance initialization parameter to FALSE, then user accounts and secure roles 
  become unusable because exclusive mode excludes the use of the 10G password 
  version. The SEC_CASE_SENSITIVE_LOGON Oracle instance initialization parameter 
  enables or disables password case sensitivity. However, because exclusive mode 
  is enabled by default in this release, disabling the password case sensitivity is not 
  supported.

Note:

- The use of the Oracle instance initialization parameter 
  SEC_CASE_SENSITIVE_LOGON is deprecated in favor of setting the 
  SQLNET.ALLOWED_LOGON_VERSION_SERVER parameter to 12 to ensure 
  that passwords are treated as case-sensitive.

- Disabling password case sensitivity is not supported in exclusive 
  mode (when SQLNET.ALLOWED_LOGON_VERSION_SERVER is set to 12 or 
  12a).

- Releases of OCI clients earlier than Oracle Database 10g cannot authenticate to 
  the Oracle database using password-based authentication.

- If the client uses Oracle Database 10g, then the client will receive an ORA-03134: 
  Connections to this server version are no longer supported error 
  message. To enable the connection, set the 
  SQLNET.ALLOWED_LOGON_VERSION_SERVER value to 8. Ensure the 
  DBA_USERS.PASSWORD_VERSIONS value for the account contains the value 10G. You 
  may need to reset the password for that account.

Note the following implication of setting the value to 12a:
To take advantage of the new 12C password version introduced in Oracle Database release 12.2, user passwords should be expired to encourage users to change their passwords and cause the new 12C password version to be generated for their account. By default in this release, new passwords are treated in a case-sensitive fashion. When an account password is changed, the earlier 10g case-insensitive password version is automatically removed, and the new 12c password version is generated.

When an account password is changed, the earlier 10g case-insensitive password version and the 11g password version are both automatically removed.

**JDBC Thin Client Support:**

In Oracle Database release 12.1.0.2 and later, if you set the `sqlnet.ora` parameter `SQLNET.ALLOWED_LOGON_VERSION_SERVER` to 12a and you create a new account or change the password of an existing account, then only the new 12c password version is generated. The 12c password version is based on a SHA-2 (Secure Hash Algorithm) SHA-512 salted cryptographic hash deoptimized using the PBKDF2 (Password-Based Key Derivation Function 2) algorithm. When the database server is running with `ALLOWED_LOGON_VERSION_SERVER` set to 12a, it is running in exclusive mode. In this mode, to log in using a JDBC client, the JRE version must be at least version 8. The JDBC client enables its O7L_MR capability flag only when it is running with at least version 8 of the JRE.

**Note:**

Check the `PASSWORD_VERSIONS` column of the `DBA_USERS` catalog view to see the list of password versions for any given account.

If you set the `sqlnet.ora` parameter `SQLNET.ALLOWED_LOGON_VERSION_SERVER` to 12, then the server runs in exclusive mode and only the 11g and 12C password versions (the SHA-1 and PBKDF2 SHA-2 based hashes of the password, respectively) are generated and allowed to be used. In such cases, fully-patched JDBC clients having the CPUOct2012 patch can connect because these JDBC clients provide the O5L_NP client ability.

Older JDBC clients that do not have the CPUOct2012 containing the fix for the stealth password cracking vulnerability CVE-2012-3132, do not provide the O5L_NP client ability. Therefore, ensure that all of the JDBC clients are patched properly.

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

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

- **O7L_MR**: The ability to perform the Oracle Database 10g authentication protocol using the 12c password version. For JDBC clients, only those running on at least JRE version 8 offer the O7L_MR capability.
- **O5L_NP**: The ability to perform Oracle Database 10g authentication protocols using the 11g password version, and generating a session key encrypted for critical patch update CPUOct2012.
• **O5L**: The ability to perform the Oracle Database 10g authentication protocol using the 10G password version.

• **O4L**: The ability to perform the Oracle9i database authentication protocol using the 10G password version.

• **O3L**: The ability to perform the Oracle8i database authentication protocol using the 10G password version.

An ability that appears at the top in the above list is more recent and secure than an ability that appears lower toward the bottom in the list. Clients that are more recent have all of the capabilities of the older clients.

The following table describes:

• The allowed settings of the `SQLNET.ALLOWED_LOGON_VERSION_SERVER` parameter

• Its effect on the generated password versions when an account is created or a password is changed

• The ability flag required of the client to authenticate while the server has this setting

• Also indicates whether the setting is considered to be an exclusive mode.

**Table 5-1  SQLNET.ALLOWED_LOGON_VERSION_SERVER Settings**

<table>
<thead>
<tr>
<th>Value of the ALLOWED_LOGON_VERSION_SERVER Parameter</th>
<th>Generated Password Version</th>
<th>Ability Required of the Client</th>
<th>Meaning for Clients</th>
<th>Server Runs in Exclusive Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>12a</td>
<td>12C</td>
<td>O7L_MR</td>
<td>Only Oracle Database 12c release 1 (12.1.0.2 or later) clients can connect to the server.</td>
<td>Yes because it excludes the use of both 10g and 11g password versions.</td>
</tr>
<tr>
<td>Value of the ALLOWED_LOGON_VERSION_SERVER Parameter</td>
<td>Generated Password Version</td>
<td>Ability Required of the Client</td>
<td>Meaning for Clients</td>
<td>Server Runs in Exclusive Mode</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>---------------------------</td>
<td>-------------------------------</td>
<td>--------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>12</td>
<td>11g, 12c</td>
<td>O5L_NP</td>
<td>Yes because it excludes the use of the 10g password version.</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>10g, 11g, 12c</td>
<td>O5L</td>
<td>Clients using Oracle Database 10g and later can connect to the server.</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>10g, 11g, 12c</td>
<td>O5L</td>
<td>It has the same meaning as the earlier row.</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>10g, 11g, 12c</td>
<td>O4L</td>
<td>It has the same meaning as the earlier row.</td>
<td>No</td>
</tr>
</tbody>
</table>
Table 5-1 (Cont.) SQLNET.ALLOWED_LOGON_VERSION_SERVER Settings

<table>
<thead>
<tr>
<th>Value of the ALLOWED_LOGON_VERSION_SERVER Parameter</th>
<th>Generated Password Version</th>
<th>Ability Required of the Client</th>
<th>Meaning for Clients</th>
<th>Server Runs in Exclusive Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>10g, 11g, 12c</td>
<td>O3L</td>
<td>It has the same meaning as the earlier row.</td>
<td>No</td>
</tr>
</tbody>
</table>

Values

- 12a for Oracle Database 12c release 12.1.0.2 or later authentication protocols (strongest protection)
- 12 for Oracle Database 12c release 12.1 authentication protocols (default and recommended value)
- 11 for Oracle Database 11g authentication protocols
- 10 for Oracle Database 10g authentication protocols
- 9 for Oracle9i Database authentication protocol
- 8 for Oracle8i Database authentication protocol

Note:

- Starting with Oracle Database 12c Release 2 (12.2), the default value is 12.
- For earlier releases, the value 12 can be used after the critical patch updates CPUOct2012 and later are applied.

Default

12

Example

SQLNET.ALLOWED_LOGON_VERSION_SERVER=12

5.2.24 SQLNET.AUTHENTICATION_SERVICES

Use the sqlnet.ora parameter SQLNET.AUTHENTICATION_SERVICES to enable one or more authentication services.

Purpose

To enable one or more authentication services. If you installed authentication, then it is recommended that you set SQLNET.AUTHENTICATION_SERVICES to either none or to one of the listed authentication methods.
Usage Notes

When using the `SQLNET.AUTHENTICATION_SERVICES` value `all`, the server attempts to authenticate using each of the following methods. The server falls back to the authentication methods that appear further down on the list if attempts to use the authentication methods appearing higher on the list were unsuccessful.

- Authentication based on a service external to the database, such as a service on the network layer, Kerberos, or RADIUS.
- Authentication based on the operating system user's membership in an administrative operating system group. Group names are platform-specific. This authentication applies to administrative connections only.
- Authentication performed by the database.
- Authentication based on credentials stored in a directory server.

Operating system authentication enables access to the database using any user name and any password when an administrative connection is attempted, such as using the `AS SYSDBA` clause when connecting using SQL*Plus. An example of a connection is as follows.

```
sqlplus ignored_username/ignored_password AS SYSDBA
```

When the operating-system user who issued the preceding command is already a member of the appropriate administrative operating system group, then the connection is successful. This is because the user name and password are ignored by the server because Oracle checks the group membership first.

See Also:

*Oracle Database Security Guide* for additional information about authentication methods

Default

`all`

Note:

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

Values

Authentication methods that are available with Oracle Net Services:

- `none` for no authentication methods, including Microsoft Windows native operating system authentication. When you set `SQLNET.AUTHENTICATION_SERVICES` to `none`, then you can use a valid user name and password to access the database.
- `all` for all authentication methods
- `beq` for native operating system authentication for operating systems other than Microsoft Windows
- `kerberos5` for Kerberos authentication
- `nts` for Microsoft Windows native operating system authentication
- `radius` for Remote Authentication Dial-In User Service (RADIUS) authentication
- `tcps` for SSL authentication

**Example**

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

### See Also:

*Oracle Database Security Guide*

---

#### 5.2.25 SQLNET.CLIENT_REGISTRATION

Use the `sqlnet.ora` parameter `SQLNET.CLIENT_REGISTRATION` to set a unique identifier for the client computer.

**Purpose**

To set a unique identifier for the client computer.

**Usage Notes**

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
```

#### 5.2.26 SQLNET.CLOUD_USER

Use the `sqlnet.ora` parameter `SQLNET.CLOUD_USER` to specify a user name for web server HTTP basic authentication.

**Purpose**

To specify a user name for web server HTTP basic authentication.

**Usage Notes**

When you use a secure websocket protocol, the client uses this user as the user name for authentication. The password for this user should be stored in a wallet using `mkstore` commands.
Perform the following configuration steps to use HTTP basic authentication with secure websockets:

1. Create a wallet using the `orapki` utility.
   
   ```
   orapki wallet create -wallet wallet_directory
   Example
   orapki wallet create -wallet /app/wallet
   ```

2. Add a web server public certificate.
   
   ```
   orapki wallet -wallet wallet_directory -trusted_cert -cert
   web_server_public_certificate_in_pem_format
   Example
   orapki wallet -wallet /app/wallet -trusted_cert -cert
   server_cert.txt
   ```

3. Add the web server user name to `sqlnet.ora`. This user name is only used for authenticating the web server. This is not a database user name. After web server authentication, the web server connects to the back-end database server and database authentication is completed.
   
   ```
   sqlnet.cloud_user = dbuser1
   ```

4. Add a web server user password to the wallet.
   
   ```
   mkstore -wrl wallet_location -createEntry username password
   Example
   mkstore -wrl /app/wallet -createEntry dbuser1 Secretdb#
   ```

5. Make the wallet automatically log in and protect this wallet directory using operating system file permissions or any other means. Do this so that only the database client can have read access to it. Refer to the operating system utilities for information about changing the file permissions.
   
   ```
   orapki wallet create -wallet wallet_directory -auto_login
   Example
   orapki wallet create -wallet /app/wallet -auto_login
   ```

6. Update the `sqlnet.ora` file with the wallet entry.
   
   ```
  (wallet_location=(SOURCE= (METHOD=file) (METHOD_DATA=(DIRECTORY=/app/wallet)))
   ```

   **Default**
   None
5.2.27 SQLNET.COMPRESSION

Use the sqlnet.ora parameter SQLNET.COMPRESSION to enable or disable data compression.

Purpose

To enable or disable data compression. If both the server and client have this parameter set to ON, then compression is used for the connection.

Note:

The SQLNET.COMPRESSION parameter applies to all database connections, except for Oracle Data Guard streaming redo and SecureFiles LOBs (Large Objects).

Default

off

Values

• on to enable data compression.
• off to disable data compression.

Example

SQLNET.COMPRESSION=on

5.2.28 SQLNET.COMPRESSION_ACCELERATION

Use the sqlnet.ora parameter SQLNET.COMPRESSION_ACCELERATION to specify the use of hardware accelerated version of compression using this parameter if it is available for that platform.

Purpose

To specify the use of hardware accelerated version of compression using this parameter if it is available for that platform.

Usage Notes

You can set this parameter in the Oracle Connection Manager alias description.

Default

on

Values

• on
• off
Example 5-4  Example
compression_acceleration = on

5.2.29 SQLNET.COMPRESSION_LEVELS

Use the sqlnet.ora parameter SQLNET.COMPRESSION_LEVELS to specify the compression level.

Purpose
To specify the compression level.

Usage Notes
The compression levels are used at the time of negotiation to verify which levels are used at both ends, and to select one level.

For Database Resident Connection Pooling (DRCP), only the compression level low is supported.

Default
low

Values
- low to use low CPU usage and low compression ratio
- high to use high CPU usage and high compression ratio

Example
SQLNET.COMPRESSION_LEVELS=(high)

5.2.30 SQLNET.COMPRESSION_THRESHOLD

Use the sqlnet.ora parameter SQLNET.COMPRESSION_THRESHOLD to specify the minimum data size for which compression is needed.

Purpose
To specify the minimum data size, in bytes, for which compression is needed.

Usage Notes
Compression is not to be performed if the size of the data you are sending is less than this value.

Default
1024 bytes
Example
SQLNET.COMPRESSION_THRESHOLD=1024

5.2.31 SQLNET.CRYPTO_CHECKSUM_CLIENT

Use the sqlnet.ora parameter SQLNET.CRYPTO_CHECKSUM_CLIENT to specify the checksum behavior for clients.

Purpose
To specify the checksum behavior for the client.

See Also:
Oracle Database Security Guide

Default
accepted

Values
• accepted to enable the security service if required or requested by the other side
• rejected to disable the security service, even if 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

5.2.32 SQLNET.CRYPTO_CHECKSUM_SERVER

Use the sqlnet.ora parameter SQLNET.CRYPTO_CHECKSUM_SERVER to specify the checksum behavior for the database.

Purpose
To specify the checksum behavior for the database.

Default
accepted

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

Example

SQLNET.CRYPTO_CHECKSUM_SERVER=accepted

See Also:
Oracle Database Security Guide

5.2.33 SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT

Use the sqlnet.ora parameter SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT to specify a list of crypto-checksum algorithms for the client to use.

Purpose

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

Default

All available algorithms

Values

- MD5 for the RSA Data Security MD5 algorithm
- SHA1 for the Secure Hash Algorithm
- SHA256 for SHA-2 uses 256 bits with the hashing algorithm
- SHA384 for SHA-2 uses 384 bits with the hashing algorithm
- SHA512 for SHA-2 uses 512 bits with the hashing algorithm

Example

SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT=(SHA256, MD5)

See Also:
Oracle Database Security Guide
5.2.34 SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER

Use the sqlnet.ora parameter SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER to specify a list of crypto-checksum algorithms for the database to use.

**Purpose**

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

**Default**

All available algorithms

**Values**

- **MD5** for the RSA Data Security's MD5 algorithm
- **SHA1** for the Secure Hash algorithm
- **SHA256** for SHA-2 uses 256 bits with the hashing algorithm
- **SHA384** for SHA-2 uses 384 bits with the hashing algorithm
- **SHA512** for SHA-2 uses 512 bits with the hashing algorithm

**Example**

```
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER=(SHA256, MD5)
```

**See Also:**

*Oracle Database Security Guide*

5.2.35 SQLNET.DBFW_PUBLIC_KEY

Use the sqlnet.ora parameter SQLNET.DBFW_PUBLIC_KEY to provide Oracle Database Firewall public keys to the Advanced Security Option (ASO) by specifying the file that stores the public keys.

**Purpose**

To provide Oracle Database Firewall public keys to Advanced Security Option (ASO) by specifying the name of the file that stores the Oracle Database Firewall public keys.

**Default**

None

**Values**

Full path name of the operating system file that has the public keys
Example

SQLNET.DBFW_PUBLIC_KEY="/path_to_file/dbfw_public_key_file.txt"  

See Also:
"SQLNET.ENCRYPTION_TYPES_SERVER"

5.2.36 SQLNET.DOWN_HOSTS_TIMEOUT

Use the sqlnet.ora parameter SQLNET.DOWN_HOSTS_TIMEOUT to specify the amount of time in seconds that server hosts down state information remains in the client cache.

Purpose

To specify the amount of time in seconds that information about the down state of server hosts is kept in the client process cache.

Usage Notes

Clients discover the down state of server hosts when attempting connections. When a connection attempt fails, the information about the down state of the server host is added to the client process cache. Subsequent connection attempts by the same client process move the addresses of the down hosts to the end of the address list, thereby reducing the priority of down hosts. When the duration of time that is specified by the SQLNET.DOWN_HOSTS_TIMEOUT parameter has elapsed, the host is purged from the process cache and its priority in the address list is restored.

Default

600 seconds (10 minutes)

Values

Any positive integer

Example

SQLNET.DOWN_HOSTS_TIMEOUT=60

5.2.37 SQLNET.ENCRYPTION_CLIENT

Use the sqlnet.ora parameter SQLNET.ENCRYPTION_CLIENT to enable encryption for clients.

Purpose

To enable encryption for clients. Setting the tnsnames.ora parameter IGNORE_ANO_ENCRYPTION_FOR_TCPS to TRUE disables SQLNET.ENCRYPTION_CLIENT.

Default

accepted
Values

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

Example

SQLNET.ENCRYPTION_CLIENT=accepted

See Also:
Oracle Database Security Guide

5.2.38 SQLNET.ENCRYPTION_SERVER

Use the sqlnet.ora parameter SQLNET.ENCRYPTION_SERVER to enable database encryption.

Purpose

To enable encryption for the database. Setting SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS to FALSE disables SQLNET.ENCRYPTION_SERVER.

Default

accepted

Values

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

Example

SQLNET.ENCRYPTION_SERVER=accepted

See Also:
Oracle Database Security Guide
5.2.39 SQLNET.ENCRYPTION_TYPES_CLIENT

Use the sqlnet.ora parameter SQLNET.ENCRYPTION_TYPES_CLIENT to list encryption algorithms for clients to use.

Purpose

To specify a list of encryption algorithms for clients to use. You can use SQLNET.ENCRYPTION_TYPES_CLIENT to enable the ARIA, SEED, and GOST encryption algorithms for encrypting SQLNet traffic.

Default

All available algorithms.

Values

One or more of the following:

- 3des112 for triple DES with a two-key (112-bit) option
- 3des168 for triple DES with a three-key (168-bit) option
- aes128 for AES (128-bit key size)
- aes192 for AES (192-bit key size)
- aes256 for AES (256-bit key size)
- des for standard DES (56-bit key size)
- des40 for DES (40-bit key size)
- rc4_40 for RSA RC4 (40-bit key size)
- rc4_56 for RSA RC4 (56-bit key size)
- rc4_128 for RSA RC4 (128-bit key size)
- rc4_256 for RSA RC4 (256-bit key size)
- ARIA128 for ARIA (128-bit key size)
- ARIA192 for ARIA (192-bit key size)
- ARIA256 for ARIA (256-bit key size)
- SEED128 for SEED (128-bit key size)
- GOST256 for GOST (256-bit key size)

Example

SQLNET.ENCRYPTION_TYPES_CLIENT=(rc4_56)

See Also:

Oracle Database Security Guide
5.2.40 SQLNET.ENCRYPTION_TYPES_SERVER

Use the sqlnet.ora parameter SQLNET.ENCRYPTION_TYPES_SERVER to list the encryption algorithms for the database to use.

Purpose

To specify a list of encryption algorithms for the database to use. You can use SQLNET.ENCRYPTION_TYPES_SERVER to enable the ARIA, GOST, and SEED encryption algorithms for encrypting SQLNet traffic.

Default

All available algorithms

Values

One or more of the following:

- 3des112 for triple DES with a two-key (112-bit) option
- 3des168 for triple DES with a three-key (168-bit) option
- aes128 for AES (128-bit key size)
- aes192 for AES (192-bit key size)
- aes256 for AES (256-bit key size)
- des for standard DES (56-bit key size)
- des40 for DES40 (40-bit key size)
- rc4_40 for RSA RC4 (40-bit key size)
- rc4_56 for RSA RC4 (56-bit key size)
- rc4_128 for RSA RC4 (128-bit key size)
- rc4_256 for RSA RC4 (256-bit key size)
- ARIA128 for ARIA (128-bit key size)
- ARIA192 for ARIA (192-bit key size)
- ARIA256 for ARIA (256-bit key size)
- SEED128 for SEED (128-bit key size)
- GOST256 for GOST (256-bit key size)

Example

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

See Also:

Oracle Database Security Guide
5.2.41 SQLNET.EXPIRE_TIME

Use the SQLNET.EXPIRE_TIME parameter in the sqlnet.ora file to specify how often, in minutes, to verify that client and server connections are active.

Purpose
To specify time intervals, in minutes, for how often to verify that client and server connections are active.

Usage Notes
Setting a value greater than 0 ensures that connections are not left open indefinitely due to an abnormal client termination. If your environment supports TCP keepalive tuning, then Oracle Net Services automatically uses the enhanced detection model and tunes the TCP keepalive parameters.

If the verification check identifies a terminated connection or a connection that is no longer in use, then the check returns an error, causing the server process to exit.

The SQLNET.EXPIRE_TIME parameter is primarily intended for the database server, which typically handles multiple connections simultaneously.

You can also use this parameter for database clients to verify if the server connection is active.

Limitations on using the terminated connection detection feature are:
- You cannot use it on bequeathed connections.
- Though very small, a probe packet generates additional traffic that may degrade your network performance.
- Depending on your operating system, the server may need to perform additional processing to distinguish the connection probing event from other events. This can also result in degraded network performance.

Default
0

Minimum Value
0

Recommended Value
10

Example
SQLNET.EXPIRE_TIME=10
5.2.42 SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS

Use the sqlnet.ora parameter SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS to ignore the value that is set for the parameter SQLNET.ENCRYPTION_SERVER for TCPS connections. This disables ANO encryption on the TCPS listener.

Purpose

Use SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS on your server to ignore the value that is set for SQLNET.ENCRYPTION_SERVER for TCPS connections. Doing this disables ANO encryption on the TCPS listener.

Default

FALSE

Example 5-5 Example

SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCPS=TRUE

5.2.43 SQLNET.INBOUND_CONNECT_TIMEOUT

Use the sqlnet.ora parameter SQLNET.INBOUND_CONNECT_TIMEOUT to specify the amount of time that clients have to connect with the database and authenticate.

Purpose

Use the parameter SQLNET.INBOUND_CONNECT_TIMEOUT to specify the time limit in ms, sec, or min, within which a client must connect with the database and provide authentication information.

Usage Notes

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

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

• Set both parameters to a low value initially.
• Set the value of the INBOUND_CONNECT_TIMEOUT_listener_name parameter to a lower value than the value that you have set for the SQLNET.INBOUND_CONNECT_TIMEOUT parameter.

It accepts different timeouts with or without space between the value and the unit. If you do not set a unit of measurement for SQLNET.INBOUND_CONNECT_TIMEOUT, then the default unit is sec. For example, you can set
INBOUND_CONNECT_TIMEOUT_listener_name to 2 seconds and set the SQLNET.INBOUND_CONNECT_TIMEOUT parameter to 3 seconds. If clients are unable to complete the connections within the specified time due to system or network delays that are normal for the particular environment, then increase the value for SQLNET.INBOUND_CONNECT_TIMEOUT as needed.

Default
60 seconds

Example
SQLNET.INBOUND_CONNECT_TIMEOUT=3ms

5.2.44 SQLNET.FALLBACK_AUTHENTICATION

Use the sqlnet.ora parameter SQLNET.FALLBACK_AUTHENTICATION to specify whether to attempt password-based authentication if Kerberos authentication fails.

Purpose
To specify whether to attempt to use password-based authentication if Kerberos authentication fails. This is relevant for direct connections as well as database link connections.

Default
FALSE

Example
SQLNET.FALLBACK_AUTHENTICATION=TRUE

See Also:
Oracle Database Security Guide

5.2.45 SQLNET.KERBEROS5_CC_NAME

Use the sqlnet.ora parameter SQLNET.KERBEROS5_CC_NAME to specify the complete path name to the Kerberos credentials cache file.

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

Usage Notes
The MSLSA option specifies that the file is on Microsoft Windows and is running Microsoft KDC.
The `OS_MEMORY` option specifies that an operating system-managed memory credential is used for the credential cache file. This option is supported for all operating systems with such a feature.

**Default**

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

**Examples**

```
SQLNET.KERBEROS5_CC_NAME=/usr/tmp/krbcache
SQLNET.KERBEROS5_CC_NAME=MSLSA
SQLNET.KERBEROS5_CC_NAME=OS_MEMORY
```

---

**See Also:**

`Oracle Database Security Guide`

---

### 5.2.46 SQLNET.KERBEROS5_CLOCKSKEW

Use the `sqlnet.ora` parameter `SQLNET.KERBEROS5_CLOCKSKEW` to specify how much time elapses before a Kerberos credential is considered out-of-date.

**Purpose**

To specify how many seconds elapse before a Kerberos credential is considered out-of-date.

**Default**

300

**Example**

```
SQLNET.KERBEROS5_CLOCKSKEW=1200
```

---

**See Also:**

`Oracle Database Security Guide`
5.2.47 SQLNET.KERBEROS5_CONF

Use the sqlnet.ora parameter SQLNET.KERBEROS5_CONF to specify the path name to the Kerberos configuration file that contains the realm for the default Key Distribution Center (KDC) and that maps realms to KDC hosts.

Purpose

To specify the complete path name to the Kerberos configuration file that contains the realm for the default Key Distribution Center (KDC) and that also maps realms to KDC hosts.

Usage Notes

KDC maintains a list of user principals and is contacted through the kinit program for the user's initial ticket.

The AUTO_DISCOVER option enables the automatic discovery of KDC and its realms. It is the default configuration for Kerberos clients. If there are multiple realms to specify, then Oracle recommends creating configuration files instead of using the AUTO_DISCOVER option. This option is supported for all operating systems with such a feature.

Default

/krb5/krb.conf on Linux and UNIX operating systems

\c:\\krb5\\krb.conf on Microsoft Windows operating systems

Values

- Directory path to krb.conf file
- AUTO_DISCOVER

Example

SQLNET.KERBEROS5_CONF=/krb5/krb.conf

See Also:

Oracle Database Security Guide

5.2.48 SQLNET.KERBEROS5_CONF_LOCATION

Use the sqlnet.ora parameter SQLNET.KERBEROS5_CONF_LOCATION to specify the directory for the Kerberos configuration file. The SQLNET.KERBEROS5_CONF_LOCATION parameter also specifies that the file is created by the system and not by the client.

Purpose

To specify the directory for the Kerberos configuration file. The parameter also specifies that the file is created by the system, and not by the client.
Usage Notes

The configuration file uses DNS look-up to obtain the realm for the default KDC, and it maps realms to the KDC hosts. This option is supported for all operating systems that support this feature.

Default

/krb5 on Linux and UNIX operating systems

\c:\krb5 on Microsoft Windows operating systems

Example

SQLNET.KERBEROS5_CONF_LOCATION=/krb5

5.2.49 SQLNET.KERBEROS5_KEYTAB

Use the sqlnet.ora parameter SQLNET.KERBEROS5_KEYTAB to specify the path name to the Kerberos principal or, secret, key mapping file that extracts keys and decrypts incoming authentication information.

Purpose

To specify the complete path name to the Kerberos principal or, secret, key mapping file that extracts keys and decrypts incoming authentication information.

Default

/etc/v5srvtab on Linux and UNIX operating systems

\c:\krb5\v5srvtab on Microsoft Windows operating systems

Example

SQLNET.KERBEROS5_KEYTAB=/etc/v5srvtab

See Also:

Oracle Database Security Guide

5.2.50 SQLNET.KERBEROS5_REALMS

Use the sqlnet.ora parameter SQLNET.KERBEROS5_REALMS to specify the complete path name to the Kerberos realm translation file that maps a host name or domain name to a realm.

Purpose

To specify the complete path name to the Kerberos realm translation file that maps a host name or domain name to a realm.
Default

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

Example

SQLNET.KERBEROS5_REALMS=/krb5/krb.realms

See Also:

Oracle Database Security Guide

5.2.51 SQLNET.KERBEROS5_REPLAY_CACHE

Use the sqlnet.ora parameter SQLNET.KERBEROS5_REPLAY_CACHE to specify that the replay cache is stored in operating system-managed memory on the server, and that file-based replay cache is not used.

Purpose

To specify that replay cache is stored in operating system-managed memory on the server and that file-based replay cache is not used.

Usage Notes

The OS_MEMORY option specifies that the replay cache is stored in operating system-managed memory on the server, and file-based replay cache is not used.

Example

SQLNET_KERBEROS5_REPLAY_CACHE=OS_MEMORY

5.2.52 SQLNET.OUTBOUND_CONNECT_TIMEOUT

Use the sqlnet.ora parameter SQLNET.OUTBOUND_CONNECT_TIMEOUT to specify the amount of time, in milliseconds, seconds, or minutes, in which clients must establish Oracle Net connections to database instances.

Purpose

To specify the time in ms, sec, or min for clients to establish an Oracle Net connection to the database instance.

Usage Notes

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

The outbound connect timeout interval is a superset of the TCP connect timeout interval that specifies a limit on the time needed to establish a TCP connection.
Additionally, the outbound connect timeout interval includes the time taken to be connected to an Oracle instance that is providing the service. It accepts different timeouts with or without space between the value and the unit.

Without this parameter, a client connection request to the database server may be blocked for the default TCP connect timeout duration (60 seconds) when the database server host system is unreachable. In this case, no unit is mentioned and the default unit is sec.

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

This parameter is overridden by the CONNECT_TIMEOUT parameter in the address description.

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

The outbound connect timeout interval is a superset of the TCP connect timeout interval, which specifies a limit on the time that is needed to establish a TCP connection. Additionally, the outbound connect timeout interval includes the time taken to be connected to an Oracle instance that is providing the requested service. It accepts different timeouts with or without space between the value and the unit.

Without this parameter, a client connection request to the database server may block for the default TCP connect timeout duration (60 seconds) when the database server host system is unreachable. In this case, no unit is mentioned and the default unit is sec.

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

This parameter is overridden by the CONNECT_TIMEOUT parameter in the address description.

Default
None

Example
SQLNET.OUTBOUND_CONNECT_TIMEOUT=10 ms

5.2.53 SQLNET.RADIUS_ALTERNATE

Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE to specify an alternate RADIUS server if the primary server is unavailable.

Purpose
To specify an alternate RADIUS server to use in case the primary server is unavailable.

Usage Notes
The value can be either the IP address or the host name of the server.
Default
None

Example
SQLNET.RADIUS_ALTERNATE=radius2

See Also:
Oracle Database Security Guide

5.2.54 SQLNET.RADIUS_ALTERNATE_PORT

Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE_PORT to specify the listening port of the alternate RADIUS server.

Purpose
To specify the listening port of the alternate RADIUS server.

Default
1645

Example
SQLNET.RADIUS_ALTERNATE_PORT=1667

See Also:
Oracle Database Security Guide

5.2.55 SQLNET.RADIUS_ALTERNATE_RETRIES

Use the sqlnet.ora parameter SQLNET.RADIUS_ALTERNATE_RETRIES to specify the number of times that the database resends messages to alternate RADIUS servers.

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

Default
3
5.2.56 SQLNET.RADIUS_AUTHENTICATION

Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION to specify a primary RADIUS server location, either by its host name or its IP address.

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

Default
Local host

Example
SQLNET.RADIUS_AUTHENTICATION=officeacct

See Also:
Oracle Database Security Guide

5.2.57 SQLNET.RADIUS_AUTHENTICATION_INTERFACE

Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_INTERFACE to specify the class that contains the user interface for interacting with users.

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

Default
DefaultRadiusInterface

Example
SQLNET.RADIUS_AUTHENTICATION_INTERFACE=DefaultRadiusInterface

See Also:
Oracle Database Security Guide
5.2.58 SQLNET.RADIUS_AUTHENTICATION_PORT

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

**Purpose**
To specify the listening port of a primary RADIUS server.

**Default**
1645

**Example**
```
SQLNET.RADIUS_AUTHENTICATION_PORT=1667
```

See Also:
Oracle Database Security Guide

5.2.59 SQLNET.RADIUS_AUTHENTICATION_RETRIES

Use the `sqlnet.ora` parameter `SQLNET.RADIUS_AUTHENTICATION_RETRIES` to specify the number of times the database should resend messages to a primary RADIUS server.

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

**Default**
3

**Example**
```
SQLNET.RADIUS_AUTHENTICATION_RETRIES=4
```

See Also:
Oracle Database Security Guide
5.2.60 SQLNET.RADIUS_AUTHENTICATION_TIMEOUT

Use the sqlnet.ora parameter SQLNET.RADIUS_AUTHENTICATION_TIMEOUT to specify the amount of time that the database should wait for a response from a primary RADIUS server.

**Purpose**

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

**Default**

5

**Example**

SQLNET.RADIUS_AUTHENTICATION_TIMEOUT=10

**See Also:**

Oracle Database Security Guide

5.2.61 SQLNET.RADIUS_CHALLENGE_RESPONSE

Use the sqlnet.ora parameter SQLNET.RADIUS_CHALLENGE_RESPONSE to enable or disable challenge responses.

**Purpose**

To turn the challenge responses on or off.

**Default**

off

**Values**

on | off

**Example**

SQLNET.RADIUS_CHALLENGE_RESPONSE=on

**See Also:**

Oracle Database Security Guide
5.2.62 SQLNET.RADIUS_SECRET

Use the sqlnet.ora parameter SQLNET.RADIUS_SECRET to specify the location of a RADIUS secret key.

Purpose:
To specify the location of a RADIUS secret key.

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

Example:
SQLNET.RADIUS_SECRET=oracle/bin/admin/radiuskey

See Also:
Oracle Database Security Guide

5.2.63 SQLNET.RADIUS_SEND_ACCOUNTING

Use the sqlnet.ora parameter SQLNET.RADIUS_SEND_ACCOUNTING to enable and disable accounting.

Purpose:
To turn accounting on and off. When you enable accounting, packets are sent to the active RADIUS server at the listening port number's value plus one.

Usage Notes:
The default port is 1646.

Default:
off

Values:
on  |  off

Example:
SQLNET.RADIUS_SEND_ACCOUNTING=on
5.2.64 SQLNET.RECV_TIMEOUT

Use the sqlnet.ora parameter SQLNET.RECV_TIMEOUT to specify the duration of time that a database client or server should wait for data from a peer after establishing a connection.

Purpose

To specify the time, in ms, sec, or min, for a database client or server to wait for data from the peer after establishing a connection. The peer must send data within the time interval that you specify.

Usage Notes

Setting this parameter for clients ensures that receive operations are not left in a wait state indefinitely or for a long period due to an abnormal termination of the server process or server busy state. If a client does not receive response data in the time specified, then the client logs ORA-12535: TNS:operation timed out and ORA-12609: TNS: Receive timeout occurred messages to the sqlnet.log file. If you set the value, then set the value initially to a low value and adjust the value according to the system and network capacity. If necessary, use this parameter with the SQLNET.SEND_TIMEOUT parameter.

You can also set this parameter on the server-side to specify the time, in ms, sec, or min, for a server to wait for client data after a connection is established. If a client does not send data in time specified, then the database server logs ORA-12535: TNS:operation timed out and ORA-12609: TNS: Receive timeout occurred messages to the sqlnet.log file. Without this parameter, the database server might continue to wait for data from clients that may be down or are experiencing problems. The server usually blocks input from the client and gets these timeouts frequently if you set it to a low value. If you do not specify a unit of measurement, then the default unit is sec.

Default

None

Example

SQLNET.RECV_TIMEOUT=10ms
Use the `sqlnet.ora` parameter `SQLNET.SEND_TIMEOUT` to specify the duration of time in which a database must complete send operations to clients after establishing connections.

**Purpose**

To specify the time in ms, sec, or min, for a database to complete send operations to clients after establishing connections.

**Usage Notes**

Setting this parameter is recommended for environments in which clients shut down occasionally or abnormally.

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

You can also set this parameter on the client-side to specify the duration of time in ms, sec, or min, in which client must complete send operations to the database server after connections are established. It accepts different timeouts with or without space between the value and the unit. If you do not specify a unit of measure, then the default unit is sec. Without this parameter, the client might continue to send requests to a database server that is saturated with requests. If you choose to set the value, then set the value initially to a low value and adjust the value according to system and network capacity. If necessary, then use this parameter with the `SQLNET.RECV_TIMEOUT` parameter.

**Default**

None

**Example**

`SQLNET.SEND_TIMEOUT=3 ms`
5.2.66 SQLNET.URI

Use the sqlnet.ora parameter SQLNET.URI to specify a database client URI mapping on a web server.

Purpose
To specify a database client URI mapping on a web server.

Usage Notes
Use this parameter to customize a URI for mapping the database websocket requests that come into a web server to the back-end database server. Secure websocket handshaking requests are sent with this URI.

Default
/sqnet

Example 5-6  Example
sqlnet.uri="/my_uri_prefix/database/"

5.2.67 SQLNET.USE_HTTPS_PROXY

Use the sqlnet.ora parameter SQLNET.USE_HTTPS_PROXY to enable forward HTTP proxy tunneling for client connections.

Purpose
To enable forward HTTP proxy tunneling for client connections.

Usage Notes
If set to on, then clients can tunnel secure connections over forward HTTP proxy using the HTTP CONNECT method. This helps access the public cloud database service because it eliminates the requirement to open an outbound port on a client-side firewall.

This parameter is applicable with Oracle Connection Manager on the server side.

Default
on

Example
SQLNET.USE_HTTPS_PROXY=on

5.2.68 SQLNET.WALLET_OVERRIDE

Use the sqlnet.ora parameter SQLNET.WALLET_OVERRIDE to determine whether a client should override strong authentication credentials with the password credential from the stored wallet.

OracleMetaLink note 340559.1.
Purpose
To determine whether a client should override strong authentication credentials with
the password credential from the stored wallet to log in to a database.

Usage Notes
When you use wallets for authentication, the database credentials for user name and
password are securely stored in an Oracle wallet. The auto-login feature of the wallet
is enabled so that the database does not need a password to open the wallet. From
the wallet, the database gets the credentials to access the database for the user.

Wallet use can simplify large-scale deployments that rely on password credentials for
connecting to databases. When this feature is configured, application code, batch jobs,
and scripts do not need embedded user names and passwords. Risk is reduced
because such passwords are no longer exposed, and password management policies
are more easily enforced without changing application code whenever user names or
passwords change.

Users connect using the `connect /@database_name` command instead of specifying a
user name and password explicitly. This simplifies the maintenance of the scripts and
secures the password management for the applications.

Middle-tier applications create an Oracle Applications wallet during installation to store
an application's identity. The password may be randomly generated rather than
hardcoded. When an Oracle application accesses the database, it sets appropriate
values for `SQLNET.AUTHENTICATION_SERVICES` and `WALLET_LOCATION`. The new wallet-based password authentication code uses the password credential in the Oracle
Applications wallet to log in to the database.

Values
ture | fasle

Examples
`SQLNET.WALLET OVERRIDE=true`

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

5.2.69 SSL_CERT_REVOCATION

Use the `sqlnet.ora` parameter `SSL_CERT_REVOCATION` to configure revocation checks
for certificates.

Purpose
To configure a revocation check for a certificate.
See Also:
Oracle Database Security Guide

Default
none

Values

• none disables certificate revocation checking; this is the default setting.
• requested to perform certificate revocation if a Certificate Revocation List (CRL) is available. Reject an SSL connection if the certificate is revoked. If no appropriate CRL is found to determine the revocation status of the certificate and the certificate is not revoked, then accept the SSL connection.
• required to perform certificate revocation when a certificate is available. If a certificate is revoked and no appropriate CRL is found, then reject the SSL connection. If no appropriate CRL is found to ascertain the revocation status of the certificate and the certificate is not revoked, then accept the SSL connection.

Example
SSL_CERT_REVOCATION=required

5.2.70 SSL_CRL_FILE

Use the sqlnet.ora parameter SSL_CRL_FILE to specify the name of the file in which you assemble the CRL for client authentication.

Purpose
To specify the name of the file where you can assemble the CRL for client authentication.

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

Default
None

Example
SSL_CRL_FILE=
5.2.71 SSL_CRL_PATH

Use the `sqlnet.ora` parameter `SSL_CRL_PATH` to

**Purpose**
To specify the destination directory of the CRL of CA.

**Usage Notes**
The files in this directory are hashed symbolic links that Oracle Wallet Manager creates.
This parameter is only valid if you set `SSL_CERT_REVOCATION` to either `requested` or `required`.

**Default**
None

**Example**
```
SSL_CRL_PATH=
```

5.2.72 SSL_CIPHER_SUITES

Use the `sqlnet.ora` parameter `SSL_CIPHER_SUITES` to control which combination of encryption and data integrity that the Secure Sockets Layer (SSL) uses.

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

Starting with Oracle Database 20c, Transport Layer Security protocol version 1.0 (TLS 1.0) is deprecated.
In accordance with security best practices, Oracle has deprecated the use of TLS 1.0. To meet your security requirements, Oracle strongly recommends that you use TLS 1.2 instead.

**Default**
None

**Values**
- `SSL_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`
- `SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA`
- `SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256`
- `SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA`
- `SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384`
- `SSL_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384`
• SSL_RSA_WITH_AES_128_CBC_SHA256
• SSL_RSA_WITH_AES_128_GCM_SHA256
• SSL_RSA_WITH_AES_128_CBC_SHA
• SSL_RSA_WITH_AES_256_CBC_SHA
• SSL_RSA_WITH_AES_256_CBC_SHA256
• SSL_RSA_WITH_AES_256_GCM_SHA384
• SSL_RSA_WITH_RC4_128_MD5
• SSL_RSA_WITH_RC4_128_SHA
• SSL_RSA_WITH_3DES_EDE_CBC_SHA
• SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
• SSL_DH_anon_WITH_RC4_128_MD5

**Note:**

The SSL_DH_anon_WITH_3DES_EDE_CBC_SHA and SSL_DH_anon_WITH_RC4_128_MD5 parameters do not provide authentication of the communicating parties, and can be vulnerable to man-in-the-middle attacks. Oracle recommends that you do not use these cipher suites to protect sensitive data. However, they are useful if the communicating parties want to remain anonymous or simply do not want the overhead caused by mutual authentication.

**Example**

SSL_CIPHER_SUITES=(ssl_rsa_with_aes_128_cbc_sha256)

**See Also:**

*Oracle Database Security Guide* for additional information about cipher suite values.

### 5.2.73 SSL_EXTENDED_KEY_USAGE

Use the `sqlnet.ora` parameter `SSL_EXTENDED_KEY_USAGE` to specify the purpose of certificate keys.

**Purpose**

To specify the purpose of the key in a certificate.

**Usage Notes**

When you specify this parameter, Oracle uses the certificate with the matching extended key.
Values
client authentication

Example
SSL_EXTENDED_KEY_USAGE="client authentication"

5.2.74 SSL_SERVER_DN_MATCH

Use the sqlnet.ora parameter SSL_SERVER_DN_MATCH to enforce server-side certification validation through distinguished name (DN) matching.

Purpose
To enforce server-side certification validation through distinguished name (DN) matching.

Usage Notes
If you enforce the DN matching, then in addition to verifying the server’s certificate chain, the client performs another check through DN matching. There are two types of DN matching. Partial DN matching occurs if the server’s DN contains its host name. Complete DN matching happens against the server’s complete DN. Not enforcing the match enables the server to potentially fake its identity. You must set this parameter to TRUE to do both full or partial DN matching.

In addition to the sqlnet.ora file, configure the tnsnames.ora parameter SSL_SERVER_CERT_DN to enable full DN matching.

Default
no

Values
- yes | on | true 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 not enforce a match. If the DN does not match the service name, then the connection is successful, but an error is logged to the sqlnet.log file.

Example
SSL_SERVER_DN_MATCH=yes

See Also:
Oracle Database Security Guide
5.2.75 SSL_VERSION

Use the sqlnet.ora parameter SSL_VERSION to limit the valid SSL or TLS versions that Oracle uses for connections.

Purpose
To limit the valid SSL or TLS versions that are used for connections.

Usage Notes
Clients and database servers must use compatible versions. You should only use this parameter when necessary for backward compatibility. The current default uses TLS version 1.2, which is the version required for multiple security compliance requirements.

If you set SSL_VERSION to undetermined, then by default it uses 3.0.

Default
1.2

Values

Note:
The sqlnet.ora parameter ADD_SSLV3_TO_DEFAULT has no impact on this parameter.

undetermined | 3.0 | 1.0 | 1.1 | 1.2

To specify one version or another version, then use the or operator. The following values are permitted:

1.0 or 3.0 | 1.2 or 3.0 | 1.1 or 1.0 | 1.2 or 1.0 | 1.2 or 1.1 | 1.1 or 1.0 or 3.0 | 1.2 or 1.0 or 3.0 | 1.2 or 1.1 or 1.0 | 1.2 or 1.1 or 3.0 | 1.2 or 1.1 or 1.0 or 3.0

Example
SSL_VERSION=1.2

The remaining version numbers correspond to the TLS versions, such as, TLSv1.0, TLSv1.1, and TLSv1.2.

See Also:
Oracle Database Security Guide
5.2.76 TCP.CONNECT_TIMEOUT

Use the `sqlnet.ora` parameter `TCP.CONNECT_TIMEOUT` to specify the amount of time in which a client must establish TCP connections to database servers.

**Purpose**

To specify the time in ms, sec, or min, for a client to establish a TCP connection (`PROTOCOL=tcp` in the TNS connect address) to the database server.

**Usage Notes**

If a TCP connection to the database is not established in the specified amount of time, then the connection attempt ends. The client receives an `ORA-12170: TNS:Connect timeout occurred` error.

The timeout applies to each IP address that resolves to a host name. It accepts different timeouts with or without space between the value and the unit. For example, if a host name resolves to an IPv6 and an IPv4 address, and if the host is not reachable through the network, then the connection request times out twice because there are two IP addresses. In this example, the default timeout setting of 60 causes a timeout in 120 seconds. If you do not specify a unit of measure, then the default unit is sec.

**Default**

60

**Example**

TCP.CONNECT_TIMEOUT=10 ms

5.2.77 TCP.EXCLUDED_NODES

Use the `sqlnet.ora` parameter `TCP.EXCLUDED_NODES` to specify which clients are denied access to the database.

**Purpose**

To specify which clients are denied access to the database.

**Usage Notes**

This parameter is only valid when you set the `TCP.VALIDNODE_CHECKING` parameter to `yes`.

You can use wildcards in this parameter for IPv4 addresses and CIDR notation for IPv4 and IPv6 addresses.

**Syntax**

`TCP.EXCLUDED.NODES=(hostname | ip_address, hostname | ip_address, ...)`

**Example**

TCP.EXCLUDED_NODES=(finance.us.example.com, mktg.us.example.com, 192.0.2.25, 172.30.*, 2001:DB8:200C:417A/32)
5.2.78 TCP.INVITED_NODES

Use the sqlnet.ora parameter TCP.INVITED_NODES to specify which clients are allowed access to the database.

Purpose

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

Syntax

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

Usage Notes

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

Example

TCP.INVITED_NODES=(sales.us.example.com, hr.us.example.com, 192.0.*, 2001:DB8:200C:433B/32)

5.2.79 TCP.NODELAY

Use the sqlnet.ora parameter TCP.NODELAY to preempt delays in buffer flushing within the TCP/IP protocol stack.

Purpose

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

Default

yes

Values

yes | no

Example

TCP.NODELAY=yes

5.2.80 TCP.QUEUESIZE

Use the sqlnet.ora parameter TCP.QUEUESIZE to configure the maximum length of queues for pending connections on TCP listening sockets.

Purpose

To configure the maximum length of the queue for pending connections on a TCP listening socket.
Default
System-defined maximum value. The defined maximum value for Linux is 128.

Values
Any integer value up to the system-defined maximum.

Examples
TCP.QUEUESIZE=100

5.2.81 TCP.VALIDNODE_CHECKING

Use the sqlnet.ora parameter TCP.VALIDNODE_CHECKING to enable and disable valid node checking for incoming connections.

Purpose
To enable and disable valid node checking for incoming connections.

Usage Notes
If you set this parameter to yes, then incoming connections are allowed only if the connections originate from a node that conforms to a list that you specified in the TCP.INVITED_NODES or TCP.EXCLUDED_NODES parameters.

The TCP.INVITED_NODES and TCP.EXCLUDED_NODES parameters are valid only when you set the TCP.VALIDNODE_CHECKING parameter to yes.

You must set this parameter and the dependent parameters, TCP.INVITED_NODES and TCP.EXCLUDED_NODES, in the sqlnet.ora file of the listener. This is important in Oracle RAC environments where listeners run from the Oracle Grid Infrastructure home. Setting the parameter in the database home does not have an effect in Oracle RAC environments. In such environments, you must include the address of all Single Client Access Name (SCANS), Virtual IPs (VIPs), local IP in the TCP.INVITED_NODES list.

In VLAN environments, the sqlnet.ora file present in the Oracle Grid Infrastructure homes should include all of the addresses of all of the VLANs. The VLANs perform the network segregation, whereas the values that are set for INVITED_NODES enables or restricts access to databases within the VLANs.

If multiple databases within the same VLAN require different INVITED_NODE lists, then you must configure separate listeners.

Default
no

Values
yes | no

Example
TCP.VALIDNODE_CHECKING=yes
5.2.82 TNSPING.TRACE_DIRECTORY

Use the sqlnet.ora parameter TNSPING.TRACE_DIRECTORY to specify the destination directory for the TNSPING utility trace file, tnsping.trc.

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

Default
The ORACLE_HOME/network/trace directory.

Example
TNSPING.TRACE_DIRECTORY=/oracle/traces

5.2.83 TNSPING.TRACE_LEVEL

Use the sqlnet.ora parameter TNSPING.TRACE_LEVEL to enable or disable TNSPING utility tracing at a specified level.

Purpose
To enable or disable TNSPING utility tracing at a specified level.

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

5.2.84 USE_CMAN

Use the sqlnet.ora parameter USE_CMAN to specify client routing to Oracle Connection Manager.

Purpose
To specify client routing to Oracle Connection Manager.

Usage Notes
When set to true, the parameter routes the client to a protocol address for Oracle Connection Manager.
When set to false, the client picks one of the address lists at random and fails over to the other address list if the chosen ADDRESS_LIST fails. With USE_CMAN=true, the client always uses the first address list.

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

Default
false

Values
ture | false

Example
USE_CMAN=true

5.2.85 USEDEDICATEDSERVER

Use the sqlnet.ora parameter USEDEDICATEDSERVER to append (SERVER=dedicated) to the CONNECT_DATA section of the connect descriptor that the client uses.

Purpose
To append (SERVER=dedicated) to the CONNECT_DATA section of the connect descriptor used by the client.

Usage Notes
The value for this parameter overrides the current value of the SERVER parameter in the tnsnames.ora file.

When set to on, the parameter USEDEDICATEDSERVER automatically appends (SERVER=dedicated) to the connect data for a connect descriptor. This enables connections from this client use a dedicated server process, even if shared server is configured.

Default
off

Values
• on to append (SERVER=dedicated)
• off to send requests to existing server processes

Example
USEDEDICATEDSERVER=on

Related Topics
• Oracle Database Net Services Administrator’s Guide
5.2.86 WALLET_LOCATION

Use the sqlnet.ora parameter WALLET_LOCATION to specify the location of wallets.

Purpose

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

Usage Notes

The key/value pair for Microsoft certificate store (MCS) omits the METHOD_DATA parameter because MCS does not use wallets. Instead, Oracle PKI (public key infrastructure) applications obtain certificates, trustpoints and private keys directly from a user's profile.

If an Oracle wallet is stored in the Microsoft Windows registry and the wallet's key (KEY) is SALESAPP, then the storage location of the encrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12. The storage location of the decrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO.

Syntax

The syntax depends on the wallet, as follows:

- **Oracle wallets on the file system:**
  
  WALLET_LOCATION=
  
  {SOURCE=}
  
  {METHOD=file}
  
  {METHOD_DATA=}
  
  {DIRECTORY=directory}
  
  {[PKCS11=TRUE/FALSE]}

- **Microsoft certificate store:**
  
  WALLET_LOCATION=
  
  {SOURCE=}
  
  {METHOD=mcs}

- **Oracle wallets in the Microsoft Windows registry:**
  
  WALLET_LOCATION=
  
  {SOURCE=}
  
  {METHOD=reg}
  
  {METHOD_DATA=}
  
  {KEY=registry_key}

- **Entrust wallets:**
  
  WALLET_LOCATION=
  
  {SOURCE=}
  
  {METHOD=entr}
  
  {METHOD_DATA=}
  
  {PROFILE=file.epf}
  
  {INIFILE=file.ini}

Additional Parameters

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

**Default**

None

**Values**

true | false

**Examples**

**Oracle wallets on file system:**

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

**Microsoft certificate store:**

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

**Oracle Wallets in the Microsoft Windows registry:**

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

**Entrust Wallets:**

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

**See Also:**

*Oracle Database Enterprise User Security Administrator’s Guide*
5.2.87 BEQUEATH_DETACH

Use the sqlnet.ora parameter to enable and disable handling signals on Linux and UNIX systems.

Purpose
To enable or disable signal handling on Linux and UNIX systems

Default
no

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

Example
BEQUEATH_DETACH=yes

5.3 ADR Diagnostic Parameters in sqlnet.ora

Diagnostic data for critical errors is stored in the sqlnet.ora Automatic Diagnostic Repository (ADR).

- **About ADR Diagnostic Parameters**
  You can use Automatic Diagnostic Repository (ADR) diagnostic parameters when ADR is enabled, which is the default. Oracle ignores non-ADR parameters in the sqlnet.ora file when you enable ADR.

- **ADR_BASE**
  Use the sqlnet.ora parameter ADR_BASE to specify the base location of the ADR files.

- **DIAG_ADR_ENABLED**
  Use the sqlnet.ora parameter DIAG_ADR_ENABLED to enable and disable ADR tracing.

- **TRACE_LEVEL_CLIENT**
  Use the sqlnet.ora parameter TRACE_LEVEL_CLIENT to enable and disable client tracing at a specific level.

- **TRACE_LEVEL_SERVER**
  Use the sqlnet.ora parameter TRACE_LEVEL_SERVER to enable and disable server tracing at a specific level.

- **TRACE_TIMESTAMP_CLIENT**
  Use the sqlnet.ora parameter TRACE_TIMESTAMP_CLIENT to add time stamps to trace events in client trace files.

- **TRACE_TIMESTAMP_SERVER**
  Use the sqlnet.ora parameter TRACE_TIMESTAMP_CLIENT to add time stamps to trace events in database trace files.
5.3.1 About ADR Diagnostic Parameters

You can use Automatic Diagnostic Repository (ADR) diagnostic parameters when ADR is enabled, which is the default. Oracle ignores non-ADR parameters in the sqlnet.ora file when you enable ADR.

Since Oracle Database 11g, Oracle Database includes an advanced fault diagnostic infrastructure to prevent, detect, diagnose, and resolve problems. The problems might be critical errors such as those that are caused by database code bugs, metadata corruption, or customer data corruption.

When critical errors occur, they are assigned incident numbers. Diagnostic data for the errors, such as traces and dumps, are captured and tagged with the incident number. The data is then stored in ADR, which is a file-based repository outside the database.

The following sqlnet.ora parameters are used when you enable ADR (when DIAG_ADR_ENABLED is set to on):

5.3.2 ADR_BASE

Use the sqlnet.ora parameter ADR_BASE to specify the base location of the ADR files.

Purpose
To specify the base directory in which Oracle stores tracing and logging incidents when ADR is enabled.

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

Values
Any valid directory path to a directory with write permission.

Example
ADR_BASE=/oracle/network/trace

5.3.3 DIAG_ADR_ENABLED

Use the sqlnet.ora parameter DIAG_ADR_ENABLED to enable and disable ADR tracing.

Purpose
To specify whether ADR tracing is enabled.

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

Default
on
Values
on | off

Example 5-7  Example
DIAG_ADR_ENABLED=on

5.3.4 TRACE_LEVEL_CLIENT

Use the sqlnet.ora parameter TRACE_LEVEL_CLIENT to enable and disable client tracing at a specific level.

Purpose
To enable client tracing at a specified level or to disable it.

Usage Notes
This parameter is also applicable when non-ADR tracing is used.

Default
off or 0

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

Example
TRACE_LEVEL_CLIENT=user

5.3.5 TRACE_LEVEL_SERVER

Use the sqlnet.ora parameter TRACE_LEVEL_SERVER to enable and disable server tracing at a specific level.

Purpose
To turn server tracing on at a specified level or to turn it off.

Usage Notes
This parameter is also applicable when non-ADR tracing is used.

Default
off or 0

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

Example
TRACE_LEVEL_SERVER=admin

5.3.6 TRACE_TIMESTAMP_CLIENT

Use the sqlnet.ora parameter TRACE_TIMESTAMP_CLIENT to add time stamps to trace events in client trace files.

Purpose
To add a time stamp in the form of dd-mmm-yyyy hh:mm:ss:mil to every trace event in the client trace file, which has a default name of sqlnet.trc.

Usage Notes
This parameter is also applicable when non-ADR tracing is used.

Default
on

Values
on or true | off or false

Example
TRACE_TIMESTAMP_CLIENT=true

5.3.7 TRACE_TIMESTAMP_SERVER

Use the sqlnet.ora parameter TRACE_TIMESTAMP_CLIENT to add time stamps to trace events in database trace files.

Purpose
To add a time stamp in the form of dd-mmm-yyyy hh:mm:ss:mil to every trace event in the database server trace file, which has a default name of svr_pid.trc.

Usage Notes
This parameter is also applicable when non-ADR tracing is used.

Default
on

Values
on or true | off or false
5.4 Non-ADR Diagnostic Parameters in sqlnet.ora Files

Learn about sqlnet.ora parameters that you use when you disable ADR.

This section lists the sqlnet.ora parameters that are used when you disable ADR.

- **LOG_DIRECTORY_CLIENT**
  Use the sqlnet.ora non-ADR diagnostic parameter LOG_DIRECTORY_CLIENT to specify the destination directory for client log files.

- **LOG_DIRECTORY_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter LOG_DIRECTORY_SERVER to specify the destination directory for database log files.

- **LOG_FILE_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter LOG_FILE_CLIENT to specify the name of log files for clients.

- **LOG_FILE_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter LOG_FILE_SERVER to specify log file names for the database.

- **TRACE_DIRECTORY_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter TRACE_DIRECTORY_CLIENT to specify the destination directory for client trace files.

- **TRACE_DIRECTORY_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter TRACE_DIRECTORY_SERVER to specify the destination directory for database trace files.

- **TRACE_FILE_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILE_CLIENT to specify the names of client trace files.

- **TRACE_FILE_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILE_SERVER to specify the destination directory for database trace output.

- **TRACE_FILEAGE_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILEAGE_CLIENT to specify the maximum age of client trace files in minutes.

---

**Example**

```
TRACE_TIMESTAMP_SERVER=true
```
• **TRACE_FILEAGE_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_FILEAGE_SERVER` to specify the maximum age of database trace files in minutes.

• **TRACE_FILELEN_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_FILELEN_CLIENT` to specify the size of client trace files in kilobytes.

• **TRACE_FILELEN_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_FILELEN_SERVER` to specify the size of database trace files in kilobytes.

• **TRACE_FILENO_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_FILENO_CLIENT` to specify the number of trace files for client tracing.

• **TRACE_FILENO_SERVER**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_FILENO_SERVER` to specify the number of trace files for database tracing.

• **TRACE_UNIQUE_CLIENT**
  Use the non-ADR diagnostic sqlnet.ora parameter `TRACE_UNIQUE_CLIENT` to specify whether Oracle creates a unique trace file for each client trace session.

5.4.1 LOGDIRECTORY_CLIENT

Use the sqlnet.ora non-ADR diagnostic parameter `LOGDIRECTORY_CLIENT` to specify the destination directory for client log files.

**Purpose**
To specify the destination directory for the client log file.

**Usage Notes**
Use this parameter when ADR is not enabled.

**Default**

`ORACLE_HOME/network/log`

**Values**
Any valid directory path.

**Example**

`LOGDIRECTORY_CLIENT=/oracle/network/log`
5.4.2 LOG_DIRECTORY_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter LOG_DIRECTORY_SERVER to specify the destination directory for database log files.

Purpose
To specify the destination directory for database log files.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/trace

Values
Any valid directory path to a directory with write permission.

Example
LOG_DIRECTORY_SERVER=/oracle/network/trace

5.4.3 LOG_FILE_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter LOG_FILE_CLIENT to specify the name of log files for clients.

Purpose
To specify the name of the log file for the client.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/log/sqlnet.log

Values
The default value cannot be changed.

5.4.4 LOG_FILE_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter LOG_FILE_SERVER to specify log file names for the database.

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

Default
sqlnet.log

Values

Example
LOG_FILE_SERVER=svr.log

5.4.5 TRACE_DIRECTORY_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_DIRECTORY_CLIENT to specify the destination directory for client trace files.

Purpose
To specify the destination directory for the client trace file.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/trace

Values
Any valid directory path to a directory with write permission.

Example
TRACE_DIRECTORY_CLIENT=/oracle/traces

5.4.6 TRACE_DIRECTORY_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_DIRECTORY_SERVER to specify the destination directory for database trace files.

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

Default
ORACLE_HOME/network/trace

Values
Any valid directory path to a directory with write permission.
Example

TRACE_DIRECTORY_SERVER=/oracle/traces

5.4.7 TRACE_FILE_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILE_CLIENT to specify the names of client trace files.

Purpose
To specify the name of a client trace file.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/trace/cli.trc

Values
Any valid file name.

Example
TRACCE_FILE_CLIENT=clientsqlnet.trc

5.4.8 TRACE_FILE_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILE_SERVER to specify the destination directory for database trace output.

Purpose
To specify the destination directory for the database server trace output.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/trace/svr_pid.trc

Values
Any valid file name. The process identifier (pid) is appended to the name automatically.

Example
TRACCE_FILE_SERVER=svrsq1net.trc
5.4.9 TRACE_FILEAGE_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILEAGE_CLIENT to specify the maximum age of client trace files in minutes.

Purpose
To specify the maximum age of client trace files in minutes.

Usage Notes
When the age limit is reached, the trace information is written to the next file. The number of files is specified with the TRACE_FILENO_CLIENT parameter. Use this parameter when ADR is not enabled.

Default
Unlimited
This is the same as setting the parameter to 0.

Example 5-8 Example
TRACE_FILEAGE_CLIENT=60

5.4.10 TRACE_FILEAGE_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILEAGE_SERVER to specify the maximum age of database trace files in minutes.

Purpose
To specify the maximum age of database server trace files in minutes.

Usage Notes
When the age limit is reached, the trace information is written to the next file. The number of files is specified with the TRACE_FILENO_SERVER parameter. Use this parameter when ADR is not enabled.

Default
Unlimited
This is the same as setting the parameter to 0.

Example 5-9 Example
TRACE_FILEAGE_SERVER=60
5.4.11 TRACE_FILELEN_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILELEN_CLIENT to specify the size of client trace files in kilobytes.

Purpose

When the file grows to the specified size, Oracle writes the trace information to the next file. The number of files is specified with the TRACE_FILENO_CLIENT parameter. Use this parameter when ADR is not enabled.

To specify the size of the client trace files in kilobytes (KB).

Usage Notes

Example

TRACE_FILELEN_CLIENT=100

5.4.12 TRACE_FILELEN_SERVER

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILELEN_SERVER to specify the size of database trace files in kilobytes.

Purpose

To specify the size of the database server trace files in kilobytes (KB).

Usage Notes

When the file grows to the specified size, Oracle writes the trace information to the next file. The number of files is specified with the TRACE_FILENO_SERVER parameter. Use this parameter when ADR is not enabled.

Example

TRACE_FILELEN_SERVER=100

5.4.13 TRACE_FILENO_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_FILENO_CLIENT to specify the number of trace files for client tracing.

Purpose

To specify the number of trace files for client tracing.

Usage Notes

When this parameter is set with the TRACE_FILELEN_CLIENT parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, then the first file is re-used, and so on.

When this parameter is set with the TRACE_FILEAGE_CLIENT parameter, trace files are cycled based on their age. The first file is used until the age limit is reached, then
the second file is used, and so on. When the last file’s age limit is reached, the first file is re-used.

When you set this parameter with both the `TRACE_FILELEN_CLIENT` and `TRACE_FILEAGE_CLIENT` parameters, trace files are replaced when either the size limit or the age limit is reached.

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

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

**Default**

None

**Example**

`TRACE_FILENO_CLIENT=3`

### 5.4.14 TRACE_FILENO_SERVER

Use the non-ADR diagnostic `sqlnet.ora` parameter `TRACE_FILENO_SERVER` to specify the number of trace files for database tracing.

**Purpose**

To specify the number of trace files for database server tracing.

**Usage Notes**

When you set this parameter 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, then the first file is re-used.

When you set this parameter with the `TRACE_FILEAGE_SERVER` parameter, trace files are cycled based on the age of the trace file. The first file is used until the age limit is reached, then the second file is used, and so on. When the last file’s age limit is reached, the first file is re-used.

When this parameter is set with both the `TRACE_FILELEN_SERVER` and `TRACE_FILEAGE_SERVER` parameters, trace files are cycled when either the size limit or the age limit is reached.

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

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

**Default**

None
Example
TRACE_FILENO_SERVER=3

5.4.15 TRACE_UNIQUE_CLIENT

Use the non-ADR diagnostic sqlnet.ora parameter TRACE_UNIQUE_CLIENT to specify whether Oracle creates a unique trace file for each client trace session.

Purpose
To specify whether a unique trace file is created for each client trace session.

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

Default
on

Values
on or off

Example
TRACE_UNIQUE_CLIENT=on
6

Local Naming Parameters in the tnsnames.ora File

This chapter describes the tnsnames.ora file configuration parameters.

- **Overview of Local Naming Parameters**
  This section provides an overview of Oracle Net service names local naming parameters.

- **General Syntax of tnsnames.ora**
  This section explains the general tnsnames.ora file syntax.

- **Using Multiple Descriptions in tnsnames.ora Files**
  Learn about tnsnames.ora file connect descriptors.

- **Multiple Address Lists in tnsnames.ora Files**
  Learn how to configure multiple address lists in tnsnames.ora files.

- **Connect-Time Failover and Client Load Balancing with Oracle Connection Managers**
  When tnsnames.ora file connect descriptors have at least two protocol addresses for Oracle Connection Manager, you can also include parameters for connect-time failover and load balancing in the file.

- **Connect Descriptor Descriptions**
  Specify connect descriptors using the DESCRIPTION parameter. Identify multiple connect descriptors with the DESCRIPTION_LIST parameter.

- **Protocol Addresses**
  Learn about Oracle Net Services protocol address parameters.

- **Optional Parameters for Address Lists**
  For multiple addresses, you can use the optional parameters to configure address lists.

- **Connection Data Section**
  Learn how to configure network connections with protocol addresses.

- **Security Section**

- **Timeout Parameters**
  The timeout section of the tnsnames.ora file provides the ability to specify timeout and retry configuration through the TNS connect string.

- **Compression Parameters**
  The compression section of the tnsnames.ora file provides the ability to enable compression and specify compression levels. These parameters can be set at the DESCRIPTION level of a connect string.

### 6.1 Overview of Local Naming Parameters

This section provides an overview of Oracle Net service names local naming parameters.
The `tnsnames.ora` file is a configuration file that contains network service names that are mapped to connect descriptors for the local naming method or net service names that are mapped to listener protocol addresses.

A net service name is an alias that is mapped to a database network address that is contained in a connect descriptor. A connect descriptor contains the location of the listener that is accessed 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 connecting with applications.

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

- The directory specified by the `TNS_ADMIN` environment variable. If the file is not found in the directory specified, then it is assumed that the file does not exist.
- If you do not set the `TNS_ADMIN` environment variable, then Oracle Net first checks the `ORACLE_BASE_HOME/network/admin` directory.
- If the file is not found in the `ORACLE_BASE_HOME/network/admin` directory, then Oracle Net checks for the file in the `ORACLE_HOME/network/admin` directory.

**Note:**
- On Microsoft Windows, the `TNS_ADMIN` environment variable is used if it is set in the environment of the process. If you do not define the `TNS_ADMIN` environment variable in the environment, or if the process is a service that does not have an environment, then Microsoft Windows scans the registry for a `TNS_ADMIN` parameter.
- With Oracle Instant Client, the `tnsnames.ora` file is located in the subdirectory of the Oracle Instant Client software. For example, in the `/opt/oracle/instantclient_release_number/network/admin` directory.

### 6.2 General Syntax of tnsnames.ora

This section explains the general `tnsnames.ora` file syntax.

The basic syntax for a `tnsnames.ora` file is shown in **Example 6-1**.

**Example 6-1  Basic Format of tnsnames.ora File**

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

In the preceding example, `DESCRIPTION` contains the connect descriptor, `ADDRESS` contains the protocol address, and `CONNECT_DATA` contains database service identification information.
6.3 Using Multiple Descriptions in tnsnames.ora Files

Learn about tnsnames.ora file connect descriptors.

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. The following example shows two connect descriptors with multiple addresses. Use the tnsnames.ora parameter DESCRIPTION_LIST to define the list of connect descriptors.

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

```sql
net_service_name=
(DESCRIPTION_LIST=
 (DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
  (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521))
  (CONNECT_DATA=
   (SERVICE_NAME=sales.us.example.com)))
 (DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=hr1-svr)(PORT=1521))
  (ADDRESS=(PROTOCOL=tcp)(HOST=hr2-svr)(PORT=1521))
  (CONNECT_DATA=
   (SERVICE_NAME=hr.us.example.com))))
```

Note:
Oracle Net Manager does not support multiple connect descriptors for a net service name if you use Oracle Connection Manager.

6.4 Multiple Address Lists in tnsnames.ora Files

Learn how to configure multiple address lists in tnsnames.ora files.

The tnsnames.ora file supports connect descriptors with multiple lists of addresses, each with its own characteristics. The following example shows two address lists. The first address list features client load balancing and no connect-time failover. These setting apply only to protocol addresses that are within its ADDRESS_LIST. The second protocol address list does not enable client load balancing, but the list does enable connect-time failover. These settings affect only protocol addresses that are included in its ADDRESS_LIST. The client first tries the first or second protocol address at random, then it tries protocol addresses number three and four, in that order, and so on.

Example 6-3  Multiple Address Lists in tnsnames.ora Files

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

**Note:**

- Oracle Net Manager supports only the creation of one protocol address list for a connect descriptor.
- Oracle Net Services supports the IFILE parameter in the `tnsnames.ora` file, with up to three levels of nesting. You must add the parameter manually to the file. The following is an example of the syntax:

```
IFILE=/tmp/listener_em.ora
IFILE=/tmp/listener_cust1.ora
IFILE=/tmp/listener_cust2.ora
```

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

When `tnsnames.ora` file connect descriptors have at least two protocol addresses for Oracle Connection Manager, you can also include parameters for connect-time failover and load balancing in the file.

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

This example illustrates the failover of multiple Oracle Connection Manager protocol addresses.

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

The previous syntax does the following:

1. The client connects to the protocol address of the first Oracle Connection Manager as indicated by:
2. Oracle Connection Manager connects to the first protocol address of another Oracle Connection Manager. If the first protocol address fails, then it tries to connect to 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. Oracle Connection Manager connects to the database service using the following protocol address:

\[
(ADDRESS=(PROTOCOL=tcp) (HOST=host3) (PORT=1521))
\]

Example 6-5  Client Load Balancing in tnsnames.ora

This example illustrates client load balancing among two Oracle Connection Managers and two protocol addresses:

\[
sample2=
(DESCRIPTION=
 (LOAD\_BALANCE=on)                                    # 1
 (FAILOVER=on)
 (ADDRESS\_LIST=
 (SOURCE\_ROUTE=yes)
 (ADDRESS=(PROTOCOL=tcp) (HOST=host1) (PORT=1630))    # 2
 (ADDRESS=(PROTOCOL=tcp) (HOST=host2) (PORT=1521)))
 (ADDRESS\_LIST=
 (SOURCE\_ROUTE=yes)
 (ADDRESS=(PROTOCOL=tcp) (HOST=host3) (port=1630))
 (ADDRESS=(PROTOCOL=tcp) (HOST=host4) (port=1521)))
 (CONNECT\_DATA=(SERVICE\_NAME=sales.us.example.com)))  # 3
\]

The previous syntax does the following:

1. The client selects an ADDRESS\_LIST at random and fails over to the other address if the chosen ADDRESS\_LIST fails. This is indicated if you set the LOAD\_BALANCE and FAILOVER parameters to on.

2. When an ADDRESS\_LIST is chosen, the client first connects to Oracle Connection Manager using the Oracle Connection Manager protocol address that uses port 1630 as is indicated for the ADDRESS\_LIST.

3. Oracle Connection Manager then connects to the database service using the protocol address that is indicated for the ADDRESS\_LIST.

6.6 Connect Descriptor Descriptions

Specify connect descriptors using the DESCRIPTION parameter. Identify multiple connect descriptors with the DESCRIPTION\_LIST parameter.
• DESCRIPTION_LIST
The DESCRIPTION_LIST parameter of the tnsnames.ora file defines a list of connect descriptors for a particular net service name.

• DESCRIPTION
Use the tnsnames.ora DESCRIPTION parameter to specify connect descriptor containers.

6.6.1 DESCRIPTION_LIST

The DESCRIPTION_LIST parameter of the tnsnames.ora file defines a list of connect descriptors for a particular net service name.

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

Example 6-6  Example

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

6.6.2 DESCRIPTION

Use the tnsnames.ora DESCRIPTION parameter to specify connect descriptor containers.

Purpose
To specify a container for a connect descriptor.

Usage Notes
When using more than one DESCRIPTION parameter, place the parameters under the DESCRIPTION_LIST parameter.

Example 6-7  DESCRIPTION Parameter Example

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

6.7 Protocol Addresses

Learn about Oracle Net Services protocol address parameters.

The protocol address section of a tnsnames.ora file specifies listener protocol addresses. If there is only one listener protocol address, then use the ADDRESS parameter. If there is more than one address, then use the ADDRESS_LIST parameter.
• **ADDRESS**
  The *tnsnames.ora* parameter ADDRESS specifies protocol addresses with the ADDRESS_LIST for multiple addresses or with the DESCRIPTION parameter for one listener.

• **HTTPS_PROXY**
  Learn to use the *tnsnames.ora* parameter HTTPS_PROXY to specify HTTP proxy host names to tunnel SSL client connections.

• **HTTPS_PROXY_PORT**
  Learn how to use the *tnsnames.ora* parameter HTTPS_PROXY_PORT to specify forward HTTP proxy host ports for tunneling SSL client connections.

• **ADDRESS_LIST**
  The ADDRESS_LIST networking parameter specifies the number of protocol addresses.

### 6.7.1 ADDRESS

The *tnsnames.ora* parameter ADDRESS specifies protocol addresses with the ADDRESS_LIST for multiple addresses or with the DESCRIPTION parameter for one listener.

**Purpose**
To specify one listener protocol address.

**Usage Notes**
Put this parameter under either the ADDRESS_LIST parameter or the DESCRIPTION parameter.

**ADDRESS Parameter Example**

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

### 6.7.2 HTTPS_PROXY

Learn to use the *tnsnames.ora* parameter HTTPS_PROXY to specify HTTP proxy host names to tunnel SSL client connections.

**Purpose**
To specify HTTP proxy host names for tunneling your SSL client connections.

**Usage Notes**
Your clients can tunnel secure connections over HTTP proxy using the HTTP CONNECT method. This helps access the public cloud database service because it eliminates the need to open an outbound port on a client-side firewall. This parameter is applicable only to the connect descriptors where PROTOCOL=TCPS. This is similar to the web browser setting for intranet users who want to connect to internet hosts. You can increase the forward web proxy read timeout for requests to a higher value depending
on client queries. Otherwise, the forward web proxy closes the connection assuming that no requests are made from the client.

A successful connection depends on your specific proxy configurations. The performance of your data transfers depend on the proxy capacity. Oracle recommends against using this feature in production environments where performance is critical.

Configuring tnsnames.ora for an HTTP proxy may not be secure enough, depending your organization's network configuration and security policies. For example, some networks require a username and password for the HTTP proxy.

Oracle Client versions that are earlier than Oracle Database 18c do not support connections through HTTP proxy.

Contact your network administrator to open outbound connections to hosts that are in the oraclecloud.com domain by using the relevant port, without going through an HTTP proxy. For example, use port 1522.

Default
None

Values
An HTTP proxy host name that can make an outbound connection to internet hosts.

Example
HTTPS_PROXY=www-proxy.mycompany.com

6.7.3 HTTPS_PROXY_PORT

Learn how to use the tnsnames.ora parameter HTTPS_PROXY_PORT to specify forward HTTP proxy host ports for tunneling SSL client connections.

Purpose
To specify forward HTTP proxy host port for tunneling SSL client connections.

Usage Notes
It forwards the HTTP proxy host port that receives the HTTP CONNECT method. Use this parameter with HTTPS_PROXY_PORT. The value for the HTTPS_PROXY_PORT parameter takes effect only when you set SQLNET.USE_HTTPS_PROXY=1 set in your sqlnet.ora file.

Default
none

Values
port number

Example
HTTPS_PROXY_PORT=80
6.7.4 ADDRESS_LIST

The **ADDRESS_LIST** networking parameter specifies the number of protocol addresses.

**Purpose**

To define a list of protocol addresses.

**Usage Notes**

If there is only one listener protocol address, then **ADDRESS_LIST** is not necessary.

Put this parameter either under the **DESCRIPTION** parameter or the **DESCRIPTION_LIST** parameter.

**Example**

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

6.8 Optional Parameters for Address Lists

For multiple addresses, you can use the optional parameters to configure address lists.

- **ENABLE**
- **EXPIRE_TIME**
  Use the **EXPIRE_TIME** parameter to specify how often, in minutes, to verify that the remote server connection is active.
- **FAILOVER**
- **LOAD_BALANCE**
- **RECV_BUF_SIZE**
- **SDU**
- **SEND_BUF_SIZE**
- **SOURCE_ROUTE**
- **TYPE_OF_SERVICE**
- **ENVS Parameter**
  Use this parameter to specify environment variables for the listener.
6.8.1 ENABLE

**Purpose**
To allow the caller to detect a terminated remote server, typically it takes 2 hours or more to notice.

**Usage Notes**
The keepalive feature on the supported TCP transports can be enabled for a net service client by putting `ENABLE=broken` under the DESCRIPTION parameter in the connect string. On the client side, the default for tcp_keepalive is off. Operating system TCP configurables, which vary by platform, define the actual keepalive timing details.

**Values**
broken

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

Although the preceding example has multiple addresses, the ADDRESS_LIST parameter was not used. This is because the ADDRESS_LIST parameter is not mandatory.

6.8.2 EXPIRE_TIME

Use the EXPIRE_TIME parameter to specify how often, in minutes, to verify that the remote server connection is active.

**Purpose**
To specify time intervals, in minutes, for how often to verify that the remote server connection is active.

**Usage Notes**
Oracle Net Services tunes the TCP keepalive parameters so that probes are sent after an idle activity.

Limitations on using the terminated connection detection feature are:

- You cannot use it on bequeathed connections.
- Though very small, a probe packet generates additional traffic that may degrade your network performance.
- Depending on your operating system, the server may need to perform additional processing to distinguish the connection probing event from other events. This can also result in a degraded network performance.
Default
0

Minimum Value
0

Recommended Value
10

Example

net_service_name=
  (DESCRIPTION=
    (EXPRESS_TIME=10)
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com))

6.8.3 FAILOVER

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

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

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

Note:
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 the DESCRIPTION_LIST, DESCRIPTION, and ADDRESS_LIST parameters

Values
• yes | on | true
• no | off | false
6.8.4 LOAD_BALANCE

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

Usage Notes
When you set the parameter to on, yes, or true, Oracle Net goes 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.

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

Default
on for DESCRIPTION_LIST

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

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

6.8.5 RECV_BUF_SIZE

Purpose
To specify, in bytes, the buffer space for receive operations of sessions.

Usage Notes
This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

Put this parameter under the DESCRIPTION parameter or at the end of the protocol address.
Setting this parameter in the connect descriptor for a client overrides the `RECV_BUF_SIZE` parameter at the client-side `sqlnet.ora` file.

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

**Default**
The default value for this parameter is specific to the operating system. The default for the Linux 2.6 operating system is 87380 bytes.

**Example**

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

**See Also:**

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

### 6.8.6 SDU

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

**Usage Notes**
Put this parameter under the `DESCRIPTION` parameter.
Setting this parameter in the connect descriptor for a client overrides the `DEFAULT_SDU_SIZE` parameter at client-side `sqlnet.ora` file.

**Default**

8192 bytes (8 KB)

**Values**

512 to 2097152 bytes.

**Example**

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

### 6.8.7 SEND_BUF_SIZE

**Purpose**

To specify, in bytes, the buffer space for send operations of sessions.

**Usage Notes**

This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.

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

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

**Note:**

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

**Default**

The default value for this parameter is operating system specific. The default for the Linux 2.6 operating system is 16 KB.

**Example**

```plaintext
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-server)(PORT=1521)
        (SEND_BUF_SIZE=11784)))
```

6.8.8 SOURCE_ROUTE

Purpose
To enable routing through multiple protocol addresses.

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

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

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

Default
off

Values
• yes | on
• no | off

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

Purpose
To specify the type of service to use for an Oracle Rdb database.

Usage Notes
This parameter should only be used if the application supports both an Oracle Rdb and Oracle database service, and you want the application to load balance between the two.

Put this parameter under the DESCRIPTION parameter.

Example
net_service_name=
  (DESCRIPTION_LIST=
   (DESCRIPTION=
    (ADDRESS=...)
    (CONNECT_DATA=
      (SERVICE_NAME=generic)
      (RDB_DATABASE=[.mf]mf_personal.rdb)
      (GLOBAL_NAME=alpha5))
      (TYPE_OF_SERVICE=rdb_database))
  (DESCRIPTION=
    (ADDRESS=...)
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com))
      (TYPE_OF_SERVICE=oracle11_database))

6.8.10 ENVS Parameter

Use this parameter to specify environment variables for the listener.

Purpose
Use the ENVS parameter 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.

Usage Notes
Enclose an environment variable and its value within double quotes ("):

(ENVS="variable=value")
You can specify a list of environment variables and their values. Enclose the list within double quotes, from end to end, and separate environment variable definitions with commas. Do not use a space to separate environment variable definitions.

For example:

(ENVS="variable=value,variable=value")

**Note:**

Single quotes (') are supported for backward compatibility.

You cannot use the following characters within the environment variable or its value definition:

- Comma (,)
- Single quotes (')
- Double quotes ("")
- Equal sign (=)

**Example**

```plaintext
SID_LIST_listener_name=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=plsextproc)
    (ORACLE_HOME=/oracle20c)
    (PROGRAM=extproc)
    (ENVS="LD_LIBRARY_PATH=/u01/app/product/oracle/lib:/u01/app/product/oracle/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"))
```

**Note:**

This parameter is not supported on Microsoft Windows. Any process started by the listener will inherit the listener’s environment.

### 6.9 Connection Data Section

Learn how to configure network connections with protocol addresses.

A network object is identified by a protocol address. When a connection is made, the client and the receiver of the request (listener or Oracle Connection Manager) are
configured with identical protocol addresses. The client uses this address to send the connection request to a particular network object location, and the recipient "listens" for requests on this address, and grants a connection based on its address information matching the client information.

- CONNECT_DATA
- FAILOVER_MODE
- GLOBAL_NAME
- HS
- INSTANCE_NAME
- RDB_DATABASE
- SHARDING_KEY
- SUPER_SHARDING_KEY
- SERVER
- SERVICE_NAME
- COLOCATION_TAG

6.9.1 CONNECT_DATA

Purpose
To define the service to which to connect, such as SERVICE_NAME.

Usage Notes
Put this parameter under the DESCRIPTION parameter.

CONNECT_DATA permits the following additional parameters:

- FAILOVER_MODE
- GLOBAL_NAME
- HS
- INSTANCE_NAME
- RDB_DATABASE
- SHARDING_KEY
- SUPER_SHARDING_KEY
- SERVER
- SERVICE_NAME
- COLOCATION_TAG

Example

net_service_name=
(DESCRIPTION=
 (ADDRESS_LIST=
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521))
))
6.9.2 FAILOVER_MODE

Purpose
To instruct Oracle Net to fail over to a different listener if the first listener fails during run time.

Usage Notes
Depending upon the configuration, the session or any SELECT statements which were in progress are automatically failed over.

This type of failover is called Transparent Application Failover (TAF) and should not be confused with the connect-time failover FAILOVER parameter.

Put this parameter under the CONNECT_DATA parameter.

Additional Parameters
FAILOVER_MODE supports the following parameters:

• **BACKUP**: Specifies the failover node by its net service name. A separate net service name must be created for the failover node.

• **TYPE**: Specifies the type of failover. Three types of Oracle Net failover functionality are available by default to Oracle Call Interface (OCI) applications:
  – **SESSION**: Fails over the session. For example, if a user's connection is lost, then a new session is automatically created for the user on the backup. This type of failover does not attempt to recover selects.
  – **SELECT**: Allows users with open cursors to continue fetching them after failure. However, this mode involves overhead on the client side in normal select operations.
  – **NONE**: This is the default, in which no failover functionality is used. This can also be explicitly specified to prevent failover from happening.

• **METHOD**: Specifies 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.

• **TRANSACTION**: Allows the database to complete the current database transaction following a recoverable error. This parameter is used with the COMMIT_OUTCOME=TRUE parameter.

• **RETRIES**: Specifies the number of times to attempt to connect after a failover. If DELAY is specified, then RETRIES defaults to five retry attempts.

• **DELAY**: Specifies the amount of time in seconds to wait between connect attempts. If RETRIES is specified, then DELAY defaults to one second.
Note:

If a callback function is registered, then RETRIES and DELAY parameters are ignored.

Related Topics

• Oracle Database Net Services Administrator’s Guide

6.9.3 GLOBAL_NAME

Purpose

To identify the Oracle RDB database.

Usage Notes

Put this parameter under the CONNECT_DATA parameter.

Example

```plaintext
net_service_name=
 (DESCRIPTION=
   (ADDRESS_LIST=
     (ADDRESS=...) (ADDRESS=...))
   (CONNECT_DATA=
     (SERVICE_NAME=generic)
     (RDB_DATABASE=[.mf]mf_personal.rdb)
     (GLOBAL_NAME=alpha5)))
```

6.9.4 HS

Purpose

To direct Oracle Net to connect to a non-Oracle system through Heterogeneous Services.

Usage Notes

Put this parameter under the CONNECT_DATA parameter.

Default

None

Values

ok

Example

```plaintext
net_service_name=
 (DESCRIPTION=
   (ADDRESS_LIST=
     (ADDRESS=...))
```
6.9.5 INSTANCE_NAME

Purpose
To identify the database instance to access.

Usage Notes
Set the value to the value specified by the INSTANCE_NAME parameter in the initialization parameter file.

Put this parameter under the CONNECT_DATA parameter.

Example

```plaintext
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=...)
      (ADDRESS=...))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)
      (INSTANCE_NAME=sales1)))
```

See Also:
Oracle Database Net Services Administrator's Guide for additional information about the use of INSTANCE_NAME

6.9.6 RDB_DATABASE

Purpose
To specify the file name of an Oracle Rdb database.

Usage Notes
Put this parameter under the CONNECT_DATA parameter.

Example

```plaintext
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=...))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)
      (RDB_DATABASE=example))
```
6.9.7 SHARDING_KEY

**Purpose**
To route the database request to a particular shard.

**Usage Notes**
Put this parameter under the CONNECT_DATA parameter.

**Example**
```
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=...)
      (ADDRESS=...))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)
      (RDB_DATABASE= [.mf]mf_personal.rdb)))
```

**See Also:**
*Oracle Database Net Services Administrator's Guide* for additional information about the use of the SHARDING_KEY parameter

6.9.8 SUPER_SHARDING_KEY

**Purpose**
To route the database request to a collection of shards.

**Usage Notes**
Put this parameter under the CONNECT_DATA parameter.

**Example**
```
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=...)
      (ADDRESS=...))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)
      ((SHARDING_KEY=40598230)
      (SUPER_SHARDING_KEY=gold)))
```
6.9.9 SERVER

Purpose
To direct the listener to connect the client to a specific type of service handler.

Usage Notes
Put this parameter under the CONNECT_DATA parameter.

Values
- dedicated to specify whether client requests be served by dedicated server.
- shared to specify whether client requests be served by a dispatcher or shared server.
- pooled to get a connection from the connection pool if database resident connection pooling is enabled on the 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.
- The USE_DEDICATED_SERVER parameter in the sqlnet.ora file overrides this parameter.

Example

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

6.9.10 SERVICE_NAME

Purpose
To identify the Oracle Database database service to access.
Usage Notes

Set the value to a value specified by the SERVICE_NAMES parameter in the initialization parameter file.

Put this parameter under the CONNECT_DATA parameter.

Example

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

Related Topics

- Oracle Database Net Services Administrator's Guide

6.9.11 COLOCATION_TAG

Purpose

To direct the listener to route all connections with the same colocation_tag to the same database instance.

Usage Notes

Use this parameter with the CONNECT_DATA parameter.

The parameter value must be an alphanumeric string.

Example

```snip
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=...)
      (ADDRESS=...))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)
      (COLOCATION_TAG=abc)))
```

Note:

Under certain conditions, such as, when maximum load of an instance is reached or when new instances are added or deleted for a service, the colocation of client connections that have the same colocation_tag to the same database instance may not be consistent.
6.10 Security Section

The security section of the tnsnames.ora file specifies the following security-related parameters for use with Oracle security features:

- **SECURITY**
- **SSL_SERVER_CERT_DN**
- **IGNORE_ANO_ENCRYPTION_FOR_TCPS**
  The **IGNORE_ANO_ENCRYPTION_FOR_TCPS** parameter specifies if the SQLNET.ENCRYPTION_CLIENT parameter should be ignored for this specific TNS alias.
- **WALLET_LOCATION Parameter**
  Use the **WALLET_LOCATION** parameter to specify different wallet locations where the TLS certificates are stored.

### 6.10.1 SECURITY

**Purpose**

To change the security properties of the connection. Put this parameter under the DESCRIPTION parameter.

**Usage Notes**

The parameters permitted under SECURITY are **SSL_SERVER_CERT_DN** and AUTHENTICATION_SERVICE.

**Example**

```plaintext
net_service_name=
 (DESCRIPTION=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
  (CONNECT_DATA=
    (SERVICE_NAME=sales.us.example.com))
  (SECURITY=
    (SSL_SERVER_CERT_DN="cn=sales,cn=OracleContext,dc=us,dc=acme,dc=com")))
```

### 6.10.2 SSL_SERVER_CERT_DN

**Purpose**

To specify the distinguished name (DN) of the database server.

**Usage Notes**

The server DN must be known by the client ahead of time. Otherwise, the client cannot specify the server's DN in SSL_SERVER_CERT_DN. The client uses this information to obtain the list of DNs it expects for each of the servers, enforcing the database server DN to match its service name. This parameter must be set to the server DN (for example, SSL_SERVER_CERT_DN="finance, cn=OracleContext,c=us,o=example") to use full DN matching. For partial DN matching, do not include this parameter.
Use this parameter with the sqlnet.ora parameter SSL_SERVER_DN_MATCH to enable full DN matching.

Example

finance=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL = tcps) (HOST = finance)
       (PORT = 1575)))
    (CONNECT_DATA=
      (SERVICE_NAME=finance.us.example.com)
      (SECURITY=
       (SSL_SERVER_CERT_DN="cn=finance,cn=OracleContext,c=us,o=example")))

Related Topics
•  Oracle Database Security Guide

6.10.3 IGNORE_ANO_ENCRYPTION_FOR_TCPS

The IGNORE_ANO_ENCRYPTION_FOR_TCPS parameter specifies if the SQLNET.ENCRYPTION_CLIENT parameter should be ignored for this specific TNS alias.

Purpose

To specify if the SQLNET.ENCRYPTION_CLIENT parameter should be ignored for this specific TNS alias.

Usage Notes

If your requirements are that SQLNET.ENCRYPTION_SERVER be set to required, then you can set the IGNORE_ANO_ENCRYPTION_FOR_TCPS parameter in both SQLNET.ENCRYPTION_CLIENT and SQLNET.ENCRYPTION_SERVER to TRUE. This forces the client to ignore the value that is set for the SQLNET.ENCRYPTION_CLIENT parameter for all outgoing TCPS connections.

Default

FALSE

Example 6-8  Example

test_ssl=
  (DESCRIPTION =
    (ADDRESS= (PROTOCOL=tcps) (HOST=) (PORT=1750))
    (CONNECT_DATA= (SID=^ORACLE_SID^))
    (SECURITY= (IGNORE_ANO_ENCRYPTION_FOR_TCPS=TRUE))
  )
6.10.4 WALLET_LOCATION Parameter

Use the **WALLET_LOCATION** parameter to specify different wallet locations where the TLS certificates are stored.

**Purpose**

**WALLET_LOCATION** enables a client connection to have distinct Transport Layer Security (TLS) connections that use certificates. This means that the clients will initiate multiple TLS connections using different TLS certificates, in the same client process.

Use this feature if you have a single client that must rely on more than one TLS session. An example would be for a client that requires access to multiple pluggable databases (PDBs), each with its own identity (certificate). This feature enables you to configure the client to connect to the correct identity for each PDB. After the configuration is complete, multi-threaded clients will be able to access more than one wallet with different certificates in simultaneous TLS sessions.

**Usage Notes**

This parameter denotes a connection specific wallet. You can use this parameter when different connections need to use different wallets on the clients side.

**Example**

```sql
net_service_name=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com))
    (SECURITY=(wallet_location=/home/oracle/wallets/databases)))
```

You can use **WALLET_LOCATION** in both sqlnet.ora file and tnsnames.ora file. Use of **WALLET_LOCATION** in tnsnames.ora will override the **WALLET_LOCATION** in sqlnet.ora for the specific tnsnames.ora service.

For example:

```sql
ssl_cert1 =
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=tcps)(HOST=shobeen.us.example.com) (PORT=1750))
    (CONNECT_DATA=(SID=sales_pdb))
    (SECURITY=(WALLET_LOCATION=/oracle/wallets/certificates/sales_cert))
  )
ssl_cert2 =
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=tcps)(HOST=shobeen.us.example.com) (PORT=1750))
    (CONNECT_DATA=(SID=marketing_pdb))
    (SECURITY=(WALLET_LOCATION=/oracle/wallets/certificates/
Related Topics

- *Oracle Database Security Guide*

### 6.11 Timeout Parameters

The timeout section of the *tnsnames.ora* file provides the ability to specify timeout and retry configuration through the TNS connect string.

The following parameters can be set at the *DESCRIPTION* level of a connect string:

- **CONNECT_TIMEOUT**
- **RETRY_COUNT**
- **RETRY_DELAY**
- **TRANSPORT_CONNECT_TIMEOUT**

#### 6.11.1 CONNECT_TIMEOUT

**Purpose**

To specify the timeout duration in ms, sec, or min for a client to establish an Oracle Net connection to an Oracle database.

**Usage Notes**

Put this parameter under the *DESCRIPTION* parameter.

The timeout interval specified by **CONNECT_TIMEOUT** is a superset of the TCP connect timeout interval. It includes the time to be connected to the database instance providing the requested service, not just the duration of the TCP connection. It accepts different timeouts with or without space between the value and the unit. In case, no unit is mentioned, the default unit is sec.

The timeout interval is applicable for each **ADDRESS** in an **ADDRESS_LIST**, and each IP address to which a host name is mapped.

The **CONNECT_TIMEOUT** parameter is equivalent to the *sqlnet.ora* parameter **SQLNET.OUTBOUND_CONNECT_TIMEOUT**, and overrides it.

**Example**

```sql
net_service_name=
  (DESCRIPTION=
    (CONNECT_TIMEOUT=10 ms) (RETRY_COUNT=3)
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-svr) (PORT=1521))
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-svr) (PORT=1521)))
    (CONNECT_DATA=
      (SERVICE_NAME=sales.us.example.com)))
```
6.11.2 RETRY_COUNT

**Purpose**

To specify the number of times an ADDRESS list is traversed before the connection attempt is terminated.

**Usage Notes**

Put this parameter under the DESCRIPTION parameter.

When a DESCRIPTION_LIST is specified, each DESCRIPTION is traversed multiple times based on the specified number of retries.

**Example**

```plaintext
net_service_name =
(DESCRIPTION_LIST=
 (DESCRIPTION=  
  (CONNECT_TIMEOUT=10) (RETRY_COUNT=3)  
  (ADDRESS_LIST=  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales1a-svr)(PORT=1521))  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales1b-svr)(PORT=1521)))  
  (CONNECT_DATA=(SERVICE_NAME=sales1.example.com)))
 (DESCRIPTION=  
  (CONNECT_TIMEOUT=60) (RETRY_COUNT=1)  
  (ADDRESS_LIST=  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales2a-svr)(PORT=1521))  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales2b-svr)(PORT=1521)))  
  (CONNECT_DATA=(SERVICE_NAME=sales2.us.example.com)))
```

6.11.3 RETRY_DELAY

**Purpose**

To specify the delay in seconds between subsequent retries for a connection. This parameter works in conjunction with RETRY_COUNT parameter.

**Usage Notes**

Put this parameter under the DESCRIPTION parameter.

When a DESCRIPTION_LIST is specified, each DESCRIPTION is traversed multiple times based on the specified number of retries, and the specific delay for the description.

**Example**

```plaintext
net_service_name =
(DESCRIPTION_LIST=
 (DESCRIPTION=  
  (CONNECT_TIMEOUT=10) (RETRY_COUNT=3) (RETRY_DELAY=2)  
  (ADDRESS_LIST=  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales1a-svr)(PORT=1521))  
   (ADDRESS=(PROTOCOL=tcp)(HOST=sales1b-svr)(PORT=1521)))  
  (CONNECT_DATA=(SERVICE_NAME=sales1.example.com)))
 (DESCRIPTION=  
  (CONNECT_TIMEOUT=60) (RETRY_COUNT=2) (RETRY_DELAY=1)
```
6.11.4 TRANSPORT_CONNECT_TIMEOUT

**Purpose**

To specify the transport connect timeout duration in ms, sec, or min for a client to establish an Oracle Net connection to an Oracle database.

**Usage Notes**

This parameter is put under the DESCRIPTION parameter.

The TRANSPORT_CONNECT_TIMEOUT parameter specifies the time, in ms, sec, or min, for a client to establish a TCP connection to the database server. It accepts different timeouts with or without space between the value and the unit. The default value is 60 seconds. In case, no unit is mentioned, the default unit is sec.

The timeout interval is applicable for each ADDRESS in an ADDRESS_LIST description, and each IP address that a host name is mapped. The TRANSPORT_CONNECT_TIMEOUT parameter is equivalent to the sqlnet.ora parameter TCP.CONNECT_TIMEOUT, and overrides it.

**Example**

```plaintext
net_service_name =
  (DESCRIPTION=
    (TRANSPORT_CONNECT_TIMEOUT=10 ms)
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales1-svr)(PORT=1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=sales2-svr)(PORT=1521)))
    (CONNECT_DATA=(SERVICE_NAME=sales.us.example.com)))
```

6.12 Compression Parameters

The compression section of the tnsnames.ora file provides the ability to enable compression and specify compression levels. These parameters can be set at the DESCRIPTION level of a connect string.

- **COMPRESSION**
  The tnsnames.ora file's compression parameter enables or disables the data compression.

- **COMPRESSION_LEVELS**
  The COMPRESSION_LEVELS parameter of the tnsnames.ora file specifies the compression level.
6.12.1 COMPRESSION

The tnsnames.ora file’s compression parameter enables or disables the data compression.

Purpose
To enable or disable data compression.

Usage Notes
Put this parameter under the DESCRIPTION parameter.

Setting this parameter in the connect descriptor for a client overrides the SQLNET.COMPRESSION parameter in the client-side sqlnet.ora file.

Default
off

Values
• on to enable data compression.
• off to disable data compression.

Example

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

Related Topics
• SQLNET.COMPRESSION
Use the sqlnet.ora parameter SQLNET.COMPRESSION to enable or disable data compression.

6.12.2 COMPRESSION_LEVELS

The COMPRESSION_LEVELS parameter of the tnsnames.ora file specifies the compression level.

Purpose
To specify the compression level.
Usage Notes

The compression levels are used at the time of negotiation to verify which levels are used at both ends, and select one level. Put this parameter under the DESCRIPTION parameter.

This parameter is used with the COMPRESSION parameter. Setting this parameter in the connect descriptor for a client overrides the SQLNET.COMPRESSION_LEVELS parameter in the client-side sqlnet.ora file.

Default

low

Values

- low for low CPU usage and a low compression ratio.
- high for high CPU usage and a high compression ratio.

Example

```sql
net_service_name=
 (DESCRIPTION=
   (COMPRESSION=on)
   (COMPRESSION_LEVELS=(LEVEL=low) (LEVEL=high))
   (ADDRESS_LIST=
     (ADDRESS=(PROTOCOL=tcp) (HOST=sales1-server) (PORT=1521))
     (ADDRESS=(PROTOCOL=tcp) (HOST=sales2-server) (PORT=1521)))
 (CONNECT_DATA=
   (SERVICE_NAME=sales.us.example.com)))
```

Related Topics

- SQLNET.COMPRESSION_LEVELS
  Use the sqlnet.ora parameter SQLNET.COMPRESSION_LEVELS to specify the compression level.
Oracle Net Listener Parameters in the listener.ora File

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

- Overview of Oracle Net Listener Configuration File
- Protocol Address Parameters
- Connection Rate Limiter Parameters
  The connection rate limiter feature in Oracle Net Listener enables a database administrator to limit the number of new connections handled by the listener. When this feature is enabled, Oracle Net Listener imposes a user-specified maximum limit on the number of new connections handled by the listener every second. Depending on the configuration, the rate can be applied to a collection of endpoints, or to a specific endpoint.
- Control Parameters
  This section describes the following parameters that control the behavior of the listener:
  - ADR Diagnostic Parameters for Oracle Net Listener
    The diagnostic data for the critical errors is quickly captured and stored in the ADR for Oracle Net listener.
  - Non-ADR Diagnostic Parameters for Oracle Net Listener
    This section lists the parameters used when ADR is disabled. The default value of DIAG_ADR_ENABLED_listener_name is on. Therefore, the DIAG_ADR_ENABLED_listener_name parameter must explicitly be set to off to use non-ADR tracing.
  - Class of Secure Transports Parameters
    The class of secure transports (COST) parameters specify a list of transports that are considered secure for administration and registration of a particular listener.

7.1 Overview of Oracle Net Listener Configuration File

Oracle Net Listener configuration, stored in the listener.ora file, consists of the following elements:

- Name of the listener
- Protocol addresses that the listener is accepting connection requests on
- Valid nodes that the listener allows to register with the database
- Database services
- Control parameters
Dynamic service registration, eliminates the need for static configuration of supported services. However, static service configuration is required if you plan to use Oracle Enterprise Manager Cloud Control.

By default, the `listener.ora` file is located in the `ORACLE_HOME/network/admin` directory. You can also store the `listener.ora` in the following locations:

- The directory specified by the `TNS_ADMIN` environment variable or registry value.
- On Linux and UNIX operating systems, it is the global configuration directory. For example, on the Oracle Solaris operating system, the directory is `/var/opt/oracle`.

**See Also:**

- Oracle Database Global Data Services Concepts and Administration Guide for information about management of global services
- Oracle operating system-specific documentation

In the read-only Oracle home mode, the default location for the `listener.ora` file is `ORACLE_BASE_HOME/network/admin`. If the `listener.ora` file is not present in the `ORACLE_BASE_HOME/network/admin` directory, then search for the file in the `ORACLE_HOME/network/admin` directory.

In the read-only Oracle home mode, the parameters are stored in the `ORACLE_BASE_HOME` location by default.

It is possible to configure multiple listeners, each with a unique name, in one `listener.ora` file. Multiple listener configurations are possible because each of the top-level configuration parameters has a suffix of the listener name or is the listener name itself.

**Note:**

- It is often useful to configure multiple listeners in one `listener.ora` file. However, Oracle recommends running only one listener for each node in most customer environments.
- Oracle Net Services supports the `IFILE` parameter in the `listener.ora` file, with up to three levels of nesting. The parameter is added manually to the file. The following is an example of the syntax:

```plaintext
IFILE=/tmp/listener_em.ora
IFILE=/tmp/listener_cust1.ora
IFILE=/tmp/listener_cust2.ora
```

Refer to Oracle Database Reference for additional information.

The following example shows a `listener.ora` file for a listener named `LISTENER`, which is the default name of the listener.
Example 7-1 listener.ora File

LISTENER=
(DESCRIPTION=
 (ADDRESS_LIST=
  (ADDRESS=(PROTOCOL=tcp) (HOST=sale-server) (PORT=1521))
  (ADDRESS=(PROTOCOL=ipc) (KEY=extproc)))

7.2 Protocol Address Parameters

The protocol address section of the listener.ora file defines the protocol addresses on which the listener is accepting connection requests. This section describes the most common parameters used in protocol addresses. The ADDRESS_LIST parameter is also supported. This section lists and describes the following parameters:

- **ADDRESS**
  The protocol ADDRESS parameter's networking parameter is in the listener.ora file. It specifies the protocol address under the DESCRIPTION parameter for one listener.

- **DESCRIPTION**
  DESCRIPTION networking parameter of the listener.ora file contains listener protocol addresses.

- **Firewall**

- **IP**
  The protocol address parameter IP determine which IP address the listener listens on when a host name is specified

- **QUEUESIZE**

- **RECV_BUF_SIZE**

- **SEND_BUF_SIZE**

7.2.1 ADDRESS

The protocol ADDRESS parameter's networking parameter is in the listener.ora file. It specifies the protocol address under the DESCRIPTION parameter for one listener.

**Purpose**

Specifies a single listener protocol address in the DESCRIPTION parameter

**Usage Notes**

Use this parameter to define the protocol, the host, and the port number for the listener.

**Example**

listener_name=
 (DESCRIPTION=
  (ADDRESS_LIST=
7.2.2 DESCRIPTION

DESCRIPTION networking parameter of the listener.ora file contains listener protocol addresses.

Purpose
To contain listener protocol addresses.

Example 7-2  Example

listener_name

7.2.3 Firewall

Purpose
It can be set in endpoint to enable firewall functionality.

Related Topics
• Oracle Database Net Services Administrator's Guide

7.2.4 IP

The protocol address parameter IP determine which IP address the listener listens on when a host name is specified

Purpose
To determine which IP address the listener listens on when a host name is specified.

Usage Notes
This parameter is only applicable when the HOST parameter specifies a host name.

Values
• first
  Listen on the first IP address returned by the DNS resolution of the host name. If the user wants the listener to listen on the first IP to which the specified host name resolves, then the address must be qualified with (IP=first).
• v4_only
  Listen only on IPv4 addresses.
• v6_only
  Listen only on IPv6 addresses.

Default
This feature is disabled by default.
Example

\[
\text{listener\_name} = \\
(DESCRIPTION= \\
(ADDRESS=(PROTOCOL=tcp)(HOST=rancode1-vip)(PORT=1522)(IP=v6\_only)))
\]

7.2.5 QUEUESIZE

Purpose

To specify the number of concurrent connection requests that the listener can accept on a TCP/IP or IPC listening endpoint (protocol address).

Usage Notes

The number of concurrent connection requests is dependent on the platform and listener usage scenarios. If the listener is heavily-loaded, then set the parameter to a higher number.

Put this parameter at the end of the protocol address with its value set to the expected number of concurrent connection requests.

Default

The default number of concurrent connection requests is operating system specific.

Example

\[
\text{listener\_name} = \\
(DESCRIPTION= \\
(ADDRESS=(PROTOCOL=tcp)(HOST=hr\_server)(PORT=1521)(QUEUESIZE=20)))
\]

See Also:

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

7.2.6 RECV_BUF_SIZE

Purpose

To specify, in bytes, the buffer space for receive operations of sessions.

Usage Notes

Put this parameter under the DESCRIPTION parameter or at the end of the protocol address with its value set to the expected number of bytes.

This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.
7.2.7 SEND_BUF_SIZE

**Purpose**
To specify, in bytes, the buffer space for send operations of sessions.

**Usage Notes**
Put this parameter under the DESCRIPTION parameter or at the end of the protocol address.

This parameter is supported by the TCP/IP, TCP/IP with SSL, and SDP protocols.
7.3 Connection Rate Limiter Parameters

The connection rate limiter feature in Oracle Net Listener enables a database administrator to limit the number of new connections handled by the listener. When this feature is enabled, Oracle Net Listener imposes a user-specified maximum limit on the number of new connections handled by the listener every second. Depending on the configuration, the rate can be applied to a collection of endpoints, or to a specific endpoint.

This feature is controlled through the following `listener.ora` configuration parameters:

- **CONNECTION_RATE_listener name**
  
  The `CONNECTION_RATE_listener name` configuration parameter of the `listener.ora` file specifies a global rate that is enforced across all listening endpoints that are rate-limited.
• **RATE_LIMIT**
  The RATE_LIMIT configuration parameter of the listener.ora file indicates that a particular listening endpoint is rate-limited.

### 7.3.1 CONNECTION_RATE_listener name

The CONNECTION_RATE_listener name configuration parameter of the listener.ora file specifies a global rate that is enforced across all listening endpoints that are rate-limited.

**Purpose**

To specify a global rate that is enforced across all listening endpoints that are rate-limited.

**Usage Notes**

When this parameter is specified, it overrides any endpoint-level numeric rate values that might be specified.

**Syntax**

```
CONNECTION_RATE_listener_name=number_of_connections_per_second
```

### 7.3.2 RATE_LIMIT

The RATE_LIMIT configuration parameter of the listener.ora file indicates that a particular listening endpoint is rate-limited.

**Purpose**

To indicate that a particular listening endpoint is rate-limited.

**Usage Notes**

The parameter is specified in the ADDRESS section of the listener endpoint configuration.

**Syntax**

```
LISTENER=
  (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521) (RATE_LIMIT=yes))
```

- When the RATE_LIMIT parameter is set to yes for an endpoint, that endpoint is included in the enforcement of the global rate configured by the CONNECTION_RATE_listener_name parameter. The global rate limit is enforced individually at each endpoint that has RATE_LIMIT set to yes.
- Dynamic endpoints for listeners managed by Oracle Clusterware have the RATE_LIMIT parameter set to yes.
- When the RATE_LIMIT parameter is set to a value greater than 0, then the rate limit is enforced at that endpoint level.
Examples

The following examples use the `CONNECTION_RATE_listener_name` and `RATE_LIMIT` parameters.

Example 1

`CONNECTION_RATE_LISTENER=10`

```plaintext
LISTENER=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521)(RATE_LIMIT=yes))
    (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1522)(RATE_LIMIT=yes))
    (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1523))
  )
```

In the preceding example, the global rate of new connections is enforced separately for each endpoint. Connections through port 1521 are limited at 10 every second, and the connections through port 1522 are also separately limited at 10 every second. Connections through port 1523 are not limited.

Example 2

```plaintext
LISTENER= (ADDRESS_LIST=
  (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521)(RATE_LIMIT=5))
  (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1522)(RATE_LIMIT=10))
  (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1523))
)
```

In the preceding example, the connection rates are enforced at the endpoint level. A maximum of 5 connections are processed through port 1521 every second. The limit for connections through port 1522 is 10 every second. Connections through port 1523 are not limited.

Note:

The global `CONNECTION_RATE_listener_name` parameter is not specified in the preceding configuration. If it is specified, then the limits on ports 1521 and 1522 are ignored, and the global value is used instead.

7.4 Control Parameters

This section describes the following parameters that control the behavior of the listener:

- `ADMIN_RESTRICTIONS_listener_name`
  The `listener.ora` control parameter `ADMIN_RESTRICTIONS_listener_name` restricts runtime administration of the listener.
• **ALLOW_MULTIPLE_REDIRECTS_listener_name**

  The `listener.ora` control parameter `ALLOW_MULTIPLE_REDIRECTS_listener_name` enables multiple redirects of the client.

• **ENABLE_EXADIRECT_listener_name**

• **CRS_NOTIFICATION_listener_name**

  `CRS_NOTIFICATION_listener_name` control parameter of the `listener.ora` file sets notification to allow or disallow Cluster Ready Services (CRS) to manage the listener in an Oracle Real Application Clusters environment.

• **DEDICATED_THROUGH_BROKER_LISTENER**

  `DEDICATED_THROUGH_BROKER_LISTENER` networking parameter of the `listener.ora` file enables the server to spawn a thread or process when a connection to the database is requested through the listener.

• **DEFAULT_SERVICE_listener_name**

  `DEFAULT_SERVICE_listener_name` control parameter of the `listener.ora` file enables users to connect to the database without having to specify a service name from the client side.

• **INBOUND_CONNECT_TIMEOUT_listener_name**

• **LOCAL_REGISTRATION_ADDRESS_listener_name**

• **MAX_ALL_CONNECTIONS_listener_name**

• **MAX_REG_CONNECTIONS_listener_name**

• **REGISTRATION_EXCLUDED_NODES_listener_name**

• **REGISTRATION_INVITED_NODES_listener_name**

• **REMOTE_REGISTRATION_ADDRESS_listener_name**

• **SAVE_CONFIG_ON_STOP_listener_name**

• **SERVICE_RATE_listener_name**

  The `SERVICE_RATE_listener_name` control parameter specifies incoming connection rate that is allowed per service for an instance.

• **SSL_CLIENT_AUTHENTICATION**

• **SSL_VERSION**

• **SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name**

• **USE_SID_AS_SERVICE_listener_name**

• **VALID_NODE_CHECKING_REGISTRATION_listener_name**

  The `VALID_NODE_CHECKING_REGISTRATION_listener_name` control parameter determines if valid node checking registration is performed, or if the subnet is allowed.

• **WALLET_LOCATION**

### 7.4.1 ADMIN_RESTRICTIONS_listener_name

The `listener.ora` control parameter `ADMIN_RESTRICTIONS_listener_name` restricts runtime administration of the listener.

**Purpose**

To restrict runtime administration of the listener.
Usage Notes

Setting `ADMIN_RESTRICTIONS_listener_name=on` disables the runtime modification of parameters in `listener.ora`. That is, the listener refuses to accept SET commands that alter its parameters. To change any of the parameters in `listener.ora`, including `ADMIN_RESTRICTIONS_listener_name` itself, modify the `listener.ora` file manually and reload its parameters using the RELOAD command for the new changes to take effect without explicitly stopping and restarting the listener.

Default

off

Example

`ADMIN_RESTRICTIONS_listener=on`

Related Topics

- **SET**
  Use the Listener Control utility command `SET` to alter listener parameter values.

- **RELOAD**
  Use the Listener Control utility command `RELOAD` to reload the `listener.ora` file so that you can add or change statically configured services without stopping the listener.

7.4.2 ALLOW_MULTIPLE_REDIRECTS_listener_name

The `listener.ora` control parameter `ALLOW_MULTIPLE_REDIRECTS_listener_name` enables multiple redirects of the client.

Purpose

To support multiple redirects of the client.

Usage Notes

This parameter should only be set on the SCAN listener on the Oracle Public Cloud. When set to `on`, multiple redirects of the client are allowed.

Do not set this parameter for a node listener if that is used as a SCAN listener.

Default

off

Values

on | off

Example

`ALLOW_MULTIPLE_REDIRECTS_listener=on`
7.4.3 ENABLE_EXADIRECT_listener_name

Purpose
To enable Exadirect protocol.

Usage Notes
The parameter enables Exadirect support.

Default
Off

Values
on | off

Example 7-3 Example
ENABLE_EXADIRECT_listener=on

7.4.4 CRS_NOTIFICATION_listener_name

CRS_NOTIFICATION_listener_name control parameter of the listener.ora file sets notification to allow or disallow Cluster Ready Services (CRS) to manage the listener in an Oracle Real Application Clusters environment.

Purpose
To set notification.

Usage Notes
By default, the Oracle Net listener notifies Cluster Ready Services (CRS) when it is started or stopped. These notifications allow CRS to manage the listener in an Oracle Real Application Clusters environment. This behavior can be prevented by setting the CRS_NOTIFICATION_listener_name parameter to off.

Default
on

Values
on | off
7.4.5 DEDICATED_THROUGH_BROKER_LISTENER

DEDICATED_THROUGH_BROKER_LISTENER networking parameter of the listener.ora file enables the server to spawn a thread or process when a connection to the database is requested through the listener.

Purpose

To enable the server to spawn a thread or process when a connection to the database is requested through the listener.

Default

off

Values

on | off

Example 7-4   Example

(Optional) Enter an example to illustrate your reference here.

7.4.6 DEFAULT_SERVICE_listener_name

DEFAULT_SERVICE_listener_name control parameter of the listener.ora file enables users to connect to the database without having to specify a service name from the client side.

Purpose

To enable users to connect to the database without having to specify a service name from the client side.

Usage Notes

In Oracle Database 12c, when a client tries to connect to the database the connection request passes through the listener. The listener may be servicing several different databases. If a service name is configured in this parameter, then users may not necessarily need to specify a service name in the connect syntax. If a user specifies a service name, then the listener connects the user to that specific database, otherwise the listener connects to the service name specified by the DEFAULT_SERVICE_listener_name parameter. For container databases, the client must explicitly specify the service name.

Default

There is no default value for the DEFAULT_SERVICE_listener_name parameter. If this parameter is not configured and a user does not specify a fully-qualified service name in the connect syntax, then the connection attempt fails. This parameter only accepts one value.

Example 7-5   Example

DEFAULT_SERVICE_listener=sales.us.example.com
7.4.7 INBOUND_CONNECT_TIMEOUT_listener_name

**Purpose**
To specify the time, in seconds, for the client to complete its connect request to the listener after the network connection had been established.

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

To protect both the listener and the database server, Oracle recommends setting this parameter in combination with the SQLNET.INBOUND_CONNECT_TIMEOUT parameter in the sqlnet.ora file. When specifying values for these parameters, consider the following recommendations:

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

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

**Default**
60 seconds

**Example**
INBOUND_CONNECT_TIMEOUT_listener=2

7.4.8 LOCAL_REGISTRATION_ADDRESS_listener_name

**Purpose**
To secure registration requests through dedicated secure registration endpoints for local listeners. Service ACLs are accepted by listener only if LOCAL_REGISTRATION_ADDRESS_listener_name is configured. The parameter specifies the group that is allowed to send ACLs.

**Usage Notes**
The local registration endpoint accepts local registration connections from the specified group. All local registration requests coming on normal listening endpoints are redirected to the local registration endpoint. If the registrar is not a part of the group, then it cannot connect to the endpoint.
Default
OFF

Values
ON, OFF, or IPC endpoint address with group

When set to ON, listener defaults the group to oinstall on UNIX and ORA_INSTALL on Windows.

Example 7-6 Example

LOCAL_REGISTRATION_ADDRESS_lsnr_alias = (address=(protocol=ipc) (group=xyz)) LOCAL_REGISTRATION_ADDRESS_lsnr_alias =ON

Related Topics
• Firewall
• DBMS_SFW_ACL_ADMIN

7.4.9 MAX_ALL_CONNECTIONS_listener_name

Purpose
To specify the maximum number of concurrent registration and client connection sessions that can be supported by Oracle Net Listener.

Usage Notes
This number includes registration connections from databases, and ongoing client connection establishment requests. After a connection is established, the clients do not maintain a connection to the listener. This limit only applies to client connections that are in the initial connection establishment phase from a listener perspective.

Default
Operating system-specific

Example
MAX_ALL_CONNECTIONS_listener=40

7.4.10 MAX_REG_CONNECTIONS_listener_name

Purpose
To specify the maximum number of concurrent registration connection sessions that can be supported by Oracle Net Listener.

Default
512
Example

MAX_REG_CONNECTIONS_listener=20

7.4.11 REGISTRATION_EXCLUDED_NODES_listener_name

Purpose
To specify the list of nodes that cannot register with the listener.

Usage Notes
The list can include host names or CIDR notation for IPv4 and IPv6 addresses. The wildcard format (*) is supported for IPv4 addresses. The presence of a host name in the list results in the inclusion of all IP addresses mapped to the host name. The host name should be consistent with the public network interface.

If the REGISTRATION_INVITED_NODES_listener_name parameter and the REGISTRATION_EXCLUDED_NODES_listener_name parameter are set, then the REGISTRATION_EXCLUDED_NODES_listener_name parameter is ignored.

Values
Valid nodes and subnet IP addresses or names.

Example

REGISTRATION_EXCLUDED_NODES_listener = (10.1.26.*, 10.16.40.0/24, 2001:DB8:3eff:fe38, node2)

7.4.12 REGISTRATION_INVITED_NODES_listener_name

Purpose
To specify the list of nodes that can register with the listener.

Usage Notes
The list can include host names or CIDR notation for IPv4 and IPv6 addresses. The wildcard format (*) is supported for IPv4 addresses. The presence of a host name in the list results in the inclusion of all IP addresses mapped to the host name. The host name should be consistent with the public network interface.

If the REGISTRATION_INVITED_NODES_listener_name parameter and the REGISTRATION_EXCLUDED_NODES_listener_name parameter are set, then the REGISTRATION_EXCLUDED_NODES_listener_name parameter is ignored.

Values
Valid nodes and subnet IP addresses or names.

Example

REGISTRATION_INVITED_NODES_listener = (10.1.35.*, 10.1.34.0/24, 2001:DB8:fe38:7303, node1)
7.4.13 REMOTE_REGISTRATION_ADDRESS_listener_name

Purpose
To secure registration requests through dedicated secure registration endpoints for SCAN listeners.

Usage Notes
The registration endpoint is on a private network within the cluster. All remote registration requests coming in on normal listening endpoints are redirected to the registration endpoint. Any system which is not a part of the cluster cannot connect to the endpoint. This feature is not supported when ADMIN_RESTRICTIONS_listener_name is set to ON as the Cluster Ready Services agent configures the remote_registration_address dynamically at run time.

Default
This parameter is configured internally in the SCAN listener to restrict registrations to the private network. The value of this parameter should not be modified or specified explicitly. The only supported explicit setting is for turning this feature off by setting the value to OFF. The value is OFF in non-SCAN listeners.

Values
off

Example
REMOTE_REGISTRATION_ADDRESS_listener=off

7.4.14 SAVE_CONFIG_ON_STOP_listener_name

Purpose
To specify whether runtime configuration changes are saved to the listener.ora file.

Usage Notes
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
ture | false

Example
SAVE_CONFIG_ON_STOP_listener=true
7.4.15 SERVICE_RATE_listener_name

The SERVICE_NAME_listener_name control parameter specifies incoming connection rate that is allowed per service for an instance.

Purpose
To specify incoming connection rate that is allowed per service for an instance.

Usage Notes
Any user-specified value greater than 0 sets the maximum limit on the number of new connections per service-instance handled by the proxy listener every second. Listener rejects connections after it reaches the maximum limit. Client side connection failure is reported with “TNS:listener: rate limit reached”.

Default
0

Example 7-7  Example

SERVICE_RATE=10

7.4.16 SSL_CLIENT_AUTHENTICATION

Purpose
To specify whether a client is authenticated using the Secure Sockets Layer (SSL).

Usage Notes
The database server authenticates the client. Therefore, this value should be set to false. If this parameter is set to true, then the listener attempts to authenticate the client, which can result in a failure.

Default
true

Values
true | false

Example

SSL_CLIENT_AUTHENTICATION=false

See Also:
Oracle Database Security Guide
7.4.17 SSL_VERSION

Purpose
To limit allowable SSL or TLS versions used for connections.

Usage Notes
Clients and database servers must use a compatible version. This parameter should only be used when absolutely necessary for backward compatibility. The current default uses TLS version 1.2 which is the version required for multiple security compliance requirements.

Default
1.2

Values
undetermined | 3.0 | 1.0 | 1.1 | 1.2

If you want to specify one version or another version, then use “or”. The following values are permitted:
1.0 or 3.0 | 1.2 or 3.0 | 1.1 or 1.0 | 1.2 or 1.0 | 1.2 or 1.1 | 1.1 or 1.0 or 3.0 | 1.2 or 1.0 or 3.0 | 1.2 or 1.1 or 1.0 | 1.2 or 1.1 or 3.0 | 1.2 or 1.1 or 1.0 or 3.0

Example
SSL_VERSION=1.2

The remaining version numbers correspond to the TLS versions, such as, TLSv1.0, TLSv1.1, and TLSv1.2.

See Also:
Oracle Database Security Guide

7.4.18 SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name

Purpose
To subscribe to Oracle Notification Service (ONS) notifications for downed events.

Usage Notes
By default, the listener subscribes to the ONS node down event on startup, if ONS is available. This subscription enables the listener to remove the affected service when it receives node down event notification from ONS. The listener uses asynchronous subscription for the event notification. Alter this behavior by setting SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name=off in listener.ora.
7.4.19 USE_SID_AS_SERVICE_listener_name

**Purpose**
To enable the system identifier (SID) in the connect descriptor to be interpreted as a service name when a user attempts a database connection.

**Usage Notes**
Database clients with earlier releases of Oracle Database that have hard-coded connect descriptors can use this parameter to connect to a container or pluggable database.

When the database is an Oracle Database 12c container database, the client must specify a service name in order to connect to it. Setting this parameter to on instructs the listener to use the SID in the connect descriptor as a service name and connect the client to the specified database.

**Default**
on

**Example**
USE_SID_ASSERVICE_listener=on

7.4.20 VALID_NODE_CHECKING_REGISTRATION_listener_name

The listener.ora control parameter VALID_NODE_CHECKING_REGISTRATION_listener_name determines if valid node checking registration is performed, or if the subnet is allowed.

**Purpose**
To determine whether valid node checking registration is performed, or the subnet is allowed.

**Usage Notes**
When set to on, valid node checking registration is performed at the listener for any incoming registration request, and only local IP addresses are allowed.

**Default**
on
Values

- **off | 0** to specify valid node checking registration is off, and no checking is performed.
- **on | 1 | local** to specify valid node checking registration is on, and all local IP addresses can register. If a list of invited nodes is set, then all IP addresses, host names, or subnets in the list as well as local IP addresses are allowed.
- **subnet | 2** to specify valid node checking registration is on, and all machines in the local subnets are allowed to register. If a list of invited nodes is set, then all nodes in the local subnets as well as all IP addresses, host names and subnets in the list are allowed.

Example

```
VALID_NODE_CHECKING_REGISTRATION_listener=on
```

### 7.4.21 WALLET_LOCATION

#### Purpose
To specify the location of wallets.

#### Usage Notes
WALLETS are certificates, keys, and trustpoints processed by SSL that allow for secure connections.

The key/value pair for Microsoft certificate store (MCS) omits the METHOD_DATA parameter because MCS does not use wallets. Instead, Oracle PKI (public key infrastructure) applications obtain certificates, trustpoints and private keys directly from the user's profile.

If an Oracle wallet is stored in the Microsoft Windows registry and the wallet's key (KEY) is SALESAPP, then the storage location of the encrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12. The storage location of the decrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO.

#### Syntax

*Table 7-1* shows the syntax for the WALLET_LOCATION parameter based on wallet storage location.

<table>
<thead>
<tr>
<th>Wallet Location</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle wallets on file system</td>
<td>WALLET_LOCATION=(SOURCE=(METHOD=file) (METHOD_DATA=(DIRECTORY=directory) [PKCS11=TRUE/FALSE])))</td>
</tr>
</tbody>
</table>
### Table 7-1  (Cont.) Syntax for WALLET_LOCATION

<table>
<thead>
<tr>
<th>Wallet Location</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft certificate store</td>
<td>WALLET_LOCATION=</td>
</tr>
<tr>
<td></td>
<td>(SOURCE=</td>
</tr>
<tr>
<td></td>
<td>(METHOD=mcs))</td>
</tr>
<tr>
<td>Oracle wallets in the Microsoft Windows registry</td>
<td>WALLET_LOCATION=</td>
</tr>
<tr>
<td></td>
<td>(SOURCE=</td>
</tr>
<tr>
<td></td>
<td>(METHOD=reg)</td>
</tr>
<tr>
<td></td>
<td>(METHOD_DATA=</td>
</tr>
<tr>
<td></td>
<td>(KEY=registry_key))</td>
</tr>
<tr>
<td>Entrust wallets</td>
<td>WALLET_LOCATION=</td>
</tr>
<tr>
<td></td>
<td>(SOURCE=</td>
</tr>
<tr>
<td></td>
<td>(METHOD=entr)</td>
</tr>
<tr>
<td></td>
<td>(METHOD_DATA=</td>
</tr>
<tr>
<td></td>
<td>(PROFILE=file.epf)</td>
</tr>
<tr>
<td></td>
<td>(INIFILE=file.ini))</td>
</tr>
</tbody>
</table>

**Additional Parameters**

The following additional parameters are available for WALLET_LOCATION:

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

**Default**

None

**Examples**

**Oracle wallets on file system:**

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

**Microsoft certificate store:**

WALLET_LOCATION=
(SOURCE=
(METHOD=mcs))
Oracle Wallets in the Microsoft Windows registry:

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

Entrust Wallets:

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

See Also:

Oracle Database Enterprise User Security Administrator's Guide

7.5 ADR Diagnostic Parameters for Oracle Net Listener

The diagnostic data for the critical errors is quickly captured and stored in the ADR for Oracle Net listener.

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

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

This section includes the parameters used when ADR is enabled. ADR is enabled by default. Non-ADR parameters listed in the listener.ora file are ignored when ADR is enabled.

The following listener.ora parameters are used when ADR is enabled (when DIAG_ADR_ENABLED is set to on):

- **ADR_BASE_listener_name**
  The ADR_BASE_listener_name parameter is a diagnostic parameter specifies the base directory that stores tracing and logging incidents when ADR is enabled.

- **DIAG_ADR_ENABLED_listener_name**
  The DIAG_ADR_ENABLED_listener_name is a diagnostic parameter of the listener.ora file. It indicates whether ADR is enabled.

- **LOG_FILE_NUM_listener_name**
  The LOG_FILE_NUM_listener_name is a diagnostic parameter of the listener.ora file that specifies the number of log file segments.
• **LOG_FILE_SIZE_listener**
  The LOG_FILE_SIZE_listener diagnostic parameter of the listener.ora file specifies the size of each log file segment.

• **LOGGING_listener**
  The LOGGING_listener diagnostic parameter of the listener.ora file turns logging on or off.

• **TRACE_LEVEL_listener**
  The TRACE_LEVEL_listener diagnostic parameter of the listener.ora file turns listener tracing on, at a specific level, or turns it off.

• **TRACE_TIMESTAMP_listener**
  The TRACE_TIMESTAMP_listener diagnostic parameter of the listener.ora file adds a time stamp to every trace event in the trace file for the listener.

### 7.5.1 ADR_BASE_listener

The ADR_BASE_listener parameter is a diagnostic parameter specifies the base directory that stores tracing and logging incidents when ADR is enabled.

**Purpose**

To specify the base directory that stores tracing and logging incidents when ADR is enabled.

**Default**

The default is ORACLE_BASE, or ORACLE_HOME/log if ORACLE_BASE is not defined.

**Values**

Any valid directory path to a directory with write permission.

**Example**

ADR_BASE_listener=/oracle/network/trace

### 7.5.2 DIAG_ADR_ENABLED_listener

The DIAG_ADR_ENABLED_listener is a diagnostic parameter of the listener.ora file. It indicates whether ADR is enabled.

**Purpose**

To indicate whether ADR tracing is enabled.

**Usage Notes**

When the DIAG_ADR_ENABLED_listener parameter is set to on, then ADR file tracing is used. When the DIAG_ADR_ENABLED_listener parameter is set to off, then non-ADR file tracing is used.

**Default**

on
Values
on|off

Example 7-8   Example
DIAG_ADR_ENABLED_listener=on

7.5.3 LOG_FILE_NUM_listener_name

The LOG_FILE_NUM_listener_name is a diagnostic parameter of the listener.ora file that specifies the number of log file segments.

Purpose
To specify the number of log file segments. At any point of time there can be only \( n \) log file segments where \( n \) is LOG_FILE_NUM_listener_name. If the log grows beyond this number, then the older segments are deleted.

Default
No default. If you don't specify a value, or set the value to zero, then the number of segments grows indefinitely.

Values
Any integer value.

Example 7-9
LOG_FILE_NUM_listener=3

7.5.4 LOG_FILE_SIZE_listener_name

The LOG_FILE_SIZE_listener_name diagnostic parameter of the listener.ora file specifies the size of each log file segment.

Purpose
To specify the size of each log file segment. The size is in MB.

Default
300 MB

Values
Any integer value.

Example 7-10   Example
LOG_FILE_SIZE_listener=10
7.5.5 LOGGING_listener_name

The LOGGING_listener_name diagnostic parameter of the listener.ora file turns logging on or off.

**Purpose**
To turn logging on or off.

**Usage Notes**
This parameter is also applicable when non-ADR tracing is used.

**Default**
on

**Values**
on | off

**Example**
LOGGING_listener=on

7.5.6 TRACE_LEVEL_listener_name

The TRACELEVEL_listener_name diagnostic parameter of the listener.ora file turns listener tracing on, at a specific level, or turns it off.

**Purpose**
To turn listener tracing on, at a specific level, or to turn it off.

**Usage Notes**
This parameter is also applicable when non-ADR tracing is used.

**Default**
off | 0

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

**Example**
TRACE_LEVEL_listener=admin
7.5.7 TRACE_TIMESTAMP_listener_name

The TRACE_TIMESTAMP_listener_name diagnostic parameter of the listener.ora file adds a time stamp to every trace event in the trace file for the listener.

Purpose
To add a time stamp in the form of dd-mmm-yyyy hh:mm:ss:mil to every trace event in the trace file for the listener.

Usage Notes
This parameter is used with the TRACE_LEVEL_listener_name parameter. This parameter is also applicable when non-ADR tracing is used.

Default
on

Values
• on | true
• off | false

Example
TRACE_TIMESTAMP_listener=true

7.6 Non-ADR Diagnostic Parameters for Oracle Net Listener

This section lists the parameters used when ADR is disabled. The default value of DIAG_ADR_ENABLED_listener_name is on. Therefore, the DIAG_ADR_ENABLED_listener_name parameter must explicitly be set to off to use non-ADR tracing.

• LOG_DIRECTORY_listener_name
• LOG_FILE_listener_name
• TRACE_DIRECTORY_listener_name
• TRACE_FILE_listener_name
• TRACE_FILEAGE_listener_name
• TRACE_FILELEN_listener_name
• TRACE_FILENO_listener_name

7.6.1 LOG_DIRECTORY_listener_name

Purpose
To specify the destination directory of the listener log file.
Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/log

Example
LOG_DIRECTORY_listener=/oracle/network/admin/log

7.6.2 LOG_FILE_listener_name

Purpose
To specify the name of the log file for the listener.

Usage Notes
Use this parameter when ADR is not enabled.

Default
listener.log

Example
LOG_FILE_listener=list.log

7.6.3 TRACE_DIRECTORY_listener_name

Purpose
To specify the destination directory of the listener trace file.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_HOME/network/trace

Example
TRACE_DIRECTORY_listener=/oracle/network/admin/trace

7.6.4 TRACE_FILE_listener_name

Purpose
To specify the name of the trace file for the listener.

Usage Notes
Use this parameter when ADR is not enabled.
7.6.5 TRACE_FILEAGE_listener_name

Purpose
To specify the maximum age of listener trace files in minutes.

Usage Notes
When the age limit is reached, the trace information is written to the next file. The number of files is specified with the TRACE_FILENO_listener_name parameter. Use this parameter when ADR is not enabled.

Default
Unlimited

This is the same as setting the parameter to 0.

Example 7-11  Example
TRACE_FILEAGE_listener=60

7.6.6 TRACE_FILELEN_listener_name

Purpose
To specify the size of the listener trace files in kilobytes (KB).

Usage Notes
When the size is met, the trace information is written to the next file. The number of files is specified using the TRACE_FILENO_listener_name parameter. Use this parameter when ADR is not enabled.

Default
Unlimited

Example
TRACE_FILELEN_listener=100

7.6.7 TRACE_FILENO_listener_name

Purpose
To specify the number of trace files for listener tracing.
Usage Notes

When this parameter is set along with the `TRACE_FILELEN_listener_name` parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is re-used, and so on.

The trace file names are distinguished from one another by their sequence number. For example, if the default trace file of `listener.trc` is used, and this parameter is set to 3, then the trace files would be named `listener1.trc`, `listener2.trc` and `listener3.trc`.

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

Default

1

Example

`TRACE_FILENO_listener=3`

7.7 Class of Secure Transports Parameters

The class of secure transports (COST) parameters specify a list of transports that are considered secure for administration and registration of a particular listener.

The COST parameters identify which transports are considered secure for that installation and whether the administration of a listener requires secure transports. Configuring these parameters is optional.

- `SECURE_REGISTER_listener_name`
- Using COST Parameters in Combination
- `DYNAMIC_REGISTRATION_listener_name`
  `DYNAMIC_REGISTRATION_listener_name` is a class of secure transports (COST) parameter and it enables or disables dynamic registration of a listener.
- `SECURE_PROTOCOL_listener_name`
- `SECURE_CONTROL_listener_name`

See Also:

`Oracle Database Net Services Administrator's Guide` for additional information about COST parameters and listener security

7.7.1 SECURE_REGISTER_listener_name

Purpose

To specify the transports on which registration requests are to be accepted.
Usage Notes

If the `SECURE_REGISTER_listener_name` parameter is configured with a list of transport names, then only the connections arriving on the specified transports are able to register the service with the listener. Connections arriving by other transport protocols are refused. The following is an example:

```
SECURE_REGISTER_listener1 = (TCPS,IPC)
```

In the preceding example, registration requests are accepted only on TCPS and IPC transports.

If no values are entered for this parameter, then the listener accepts registration requests from any transport.

Syntax

```
SECURE_REGISTER_listener_name = ([transport1[,transport2, ....,transportn]])
```

In the preceding example, `transport1`, `transport2`, and `transportn` are valid, installed transport protocol names.

If this parameter and `SECURE_CONTROL_listener_name` are configured, then they override the `SECURE_PROTOCOL_listener_name` parameter.

Example

```
LISTENER1=
 (DESCRIPTION=
   (ADDRESS_LIST=
     (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
     (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
     (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))
   )
   SECURE_REGISTER_listener1=tcps
)
```

7.7.2 Using COST Parameters in Combination

COST parameters can also be used in combination to further control which transports accept service registration and control commands.

In Example 7-12, control commands are accepted only on the IPC channel and the TCPS transport, and service registrations are accepted only on an IPC channel.

Example 7-12 Combining COST Parameters

```
LISTENER1=
 (DESCRIPTION=
   (ADDRESS_LIST=
     (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
     (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
     (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522))
   )
   SECURE_CONTROL_listener1=(tcps,ipc)
   SECURE_REGISTER_listener1=ipc
)
```

In Example 7-13, control commands are accepted only on the TCPS transport, and service registrations are accepted only on the IPC channel.
Example 7-13  Combining COST Parameters

LISTENER1=
  (DESCRIPTION=
   (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521))
    (ADDRESS=(PROTOCOL=ipc)(KEY=extproc))
    (ADDRESS=(PROTOCOL=tcps)(HOST=sales-server)(PORT=1522)))
  SECURE_CONTROL_listener1=tcps
  SECURE_PROTOCOL_listener1=ipc

7.7.3 DYNAMIC_REGISTRATION_listener_name

DYNAMIC_REGISTRATION_listener_name is a class of secure transports (COST) parameter and it enables or disables dynamic registration of a listener.

Purpose
To enable or disable dynamic registration.

Usage Notes
Static registrations are not affected by this parameter.

Default
The default value is on. Unless this parameter is explicitly set to off, all registration connections are accepted.

Values
• on: The listener accepts dynamic registration.
• off: The listener refuses dynamic registration.

Example 7-14  Example

DYNAMIC_REGISTRATION_listener_name=on

7.7.4 SECURE_PROTOCOL_listener_name

Purpose
To specify the transports on which administration and registration requests are accepted.

Usage Notes
If this parameter is configured with a list of transport names, then the control commands and service registration can happen only if the connection belongs to the list of transports.

If this parameter is not present and neither SECURE_CONTROL_listener_name or SECURE_REGISTER_listener_name are configured, then all supported transports accept control and registration requests.

If the SECURE_CONTROL_listener_name and SECURE_REGISTER_listener_name parameters are configured, then they override the SECURE_PROTOCOL_listener_name parameter.
7.7.5 SECURE_CONTROL_listener_name

**Purpose**
To specify the transports on which control commands are to be serviced.

**Usage Notes**
If the SECURE_CONTROL_listener_name parameter is configured with a list of transport names, then the control commands are serviced only if the connection is one of the listed transports. Connections arriving by other transport protocols are refused. The following is an example:

SECURE_CONTROL_listener1 = (TCPS, IPC)

In the preceding example, administration requests are accepted only on TCPS and IPC transports.

If no values are entered for this parameter, then the listener accepts any connection on any endpoint.

**Syntax**
SECURE_CONTROL_listener_name = [[transport1[,transport2, ....,transportn]]

In the preceding syntax, transport1, transport2, and transportn are valid, installed transport protocol names.

**Example**
LISTENER1=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp) (HOST=sales-server) (PORT=1521))
      (ADDRESS=(PROTOCOL=ipc) (KEY=extproc))
      (ADDRESS=(PROTOCOL=tcps) (HOST=sales-server) (PORT=1522)))
  SECURE_CONTROL_LISTENER1=tcps
8

Oracle Connection Manager Parameters

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

• Overview of Oracle Connection Manager Configuration File
Oracle Connection Manager configuration information is stored in the cman.ora file.

• Oracle Connection Manager Parameters
This section lists and describes the following cman.ora file parameters:

• Oracle Connection Manager in Traffic Director Mode Parameters

• ADR Diagnostic Parameters for Oracle Connection Manager
The diagnostic data for critical errors is quickly captured and stored in the ADR for Oracle Connection Manager.

• Non-ADR Diagnostic Parameters for Oracle Connection Manager
This section lists the parameters used when ADR is disabled:

8.1 Overview of Oracle Connection Manager Configuration File

Oracle Connection Manager configuration information is stored in the cman.ora file.

Oracle Connection Manager Configuration File
Oracle Connection Manager configuration information consists of the following elements:

• Protocol address of the Oracle Connection Manager listener
• Access control parameters
• Performance parameters

By default, the cman.ora file is located in the ORACLE_HOME/network/admin directory. You can also store the cman.ora file in the following locations:

• The directory specified by the TNS_ADMIN environment variable or registry value.
• On Linux and UNIX operating systems, the global configuration directory. For example, on the Oracle Solaris operating system, this directory is /var/opt/oracle.
• ORACLE_BASE_HOME/network/admin directory.
• ORACLE_HOME/network/admin directory.

Example 8-1 Sample cman.ora File

CMAN=
{CONFIGURATION=
cman.ora File Sections

- **Listening address:** Preceded by `ADDRESS=`, this section contains information pertinent to the listener. The `ADDRESS` parameter is required.

- **Rule list:** Preceded by `RULE_LIST=`, this section contains rule information. The `RULE` parameter is listed in the rule list section of the file. The `RULE` parameter is required.

- **Rule Group:** Preceded by `RULE_GROUP=`, this section contains `rule_list` grouped by service names. You can use either the `rule_group` syntax or the `rule_list` syntax.

- **Parameter list:** Preceded by `PARAMETER_LIST=`, this section contains all other parameters including those listed in "ADR Diagnostic Parameters for Oracle Connection Manager", and "Non-ADR Diagnostic Parameters for Oracle Connection Manager".

The following parameters are allowed in the parameter list section of the `cman.ora` file. The default values are bold. To override the default setting for a parameter, enter the parameter and a nondefault value.

- **ASO_AUTHENTICATION_FILTER**
  - `{off | on}`

- **CONNECTION_STATISTICS**
  - `{no | yes}`

- **EVENT_GROUP**
  - `{init_and_term | memory_ops | conn_hdlg | proc_mgmt | reg_and_load | wake_up | timer | cmd_proc | relay}`

- **IDLE_TIMEOUT**
  - `= 0 or greater`

- **INBOUND_CONNECT_TIMEOUT**
  - `= 0 or greater. The default value is 60.`

- **LOG_DIRECTORY**
  - `= log_directory. The default value is ORACLE_HOME/network/log.`

- **LOG_LEVEL**
  - `{off | user | admin | support}`

- **MAX_CMCTL_SESSIONS**
  - `= Any positive number. The default value is 4.`

- **MAX_CONNECTIONS**
  - `= A value between 1 and 1024. The default value is 256.`

- **MAX_GATEWAY_PROCESSES**
  - `= Any number greater than the minimum number of gateway processes up to 64. The default value is 16.`

- **MIN_GATEWAY_PROCESSES**
  - `= Any positive number less than or equal to 64. Must be less than or equal to the maximum number of gateway processes. The default value is 2.`

- **OUTBOUND_CONNECT_TIMEOUT**
  - `= 0 or greater`

- **PASSWORD_instance_name**
  - `Value is the encrypted instance password, if one has been set. The default value is no value.`
SESSION_TIMEOUT=0 or greater

TRACE_DIRECTORY=trace_directory. The default value is ORACLE_HOME/network/trace.

TRACE_FILELEN= Any positive number. The default value is 0 (zero).

TRACE_FILENO= Any positive number. The default value is 0 (zero).

TRACE_LEVEL={off | user | admin | support}

TRACE_TIMESTAMP={off | on}

Note:

You cannot add the parameter PASSWORD_instance_name directly to the cman.ora file. The parameter is added using the SAVE_PASSWD command.

(PARAMETER_LIST=

(ASO_AUTHENTICATION_FILTER=ON)
(CONNECTION_STATISTICS=NO)
(EVENT_GROUP=INIT_AND_TERM,MEMORY_OPS,PROCESS_MGMT)
(IDLE_TIMEOUT=30)
(INBOUND_CONNECT_TIMEOUT=30)
(LOG_DIRECTORY=/home/user/network/admin/log)
(LOG_LEVEL=SUPPORT)
(MAX_CMCTL_SESSIONS=6)
(MAX_CONNECTIONS=512)
(MAX_GATEWAY_PROCESSES=10)
(MIN_GATEWAY_PROCESSES=4)
(OUTBOUND_CONNECT_TIMEOUT=30)
(SESSION_TIMEOUT=60)
(TRACE_DIRECTORY=/home/user/network/admin/trace)
(TRACE_FILELEN=100)
(TRACE_FILENO=2)
(TRACE_LEVEL=SUPPORT)
(TRACE_TIMESTAMP=ON)
(VALID_NODE_CHECKING_REGISTRATION=ON)
(REGISTRATION_EXCLUDED_NODES = 10.1.26.*)
(REGISTRATION_INVITED_NODES = 10.1.35.*)
)

8.2 Oracle Connection Manager Parameters

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

- ADDRESS
  The ADDRESS networking parameter specifies the protocol address of Oracle Connection Manager.
• **ASO_AUTHENTICATION_FILTER**
  It is a networking parameter for Oracle Connection Manager. It instructs Oracle Connection Manager to check the connection requests for Secure Network Services (SNS).

• **COMPRESSION**

• **COMPRESSION_LEVELS**
The `COMPRESSION_LEVELS` networking parameter of the `cman.ora` file specifies the CPU usage and compression ratio.

• **COMPRESSION_THRESHOLD**

• **CONNECTION_STATISTICS**
The `CONNECTION_STATISTICS` networking parameter of the `cman.ora` file specifies whether the `SHOW_CONNECTIONS` command displays connection statistics.

• **EVENT_GROUP**
The `EVENT_GROUP` networking parameter of the `cman.ora` file specifies which event groups are logged.

• **EXPIRE_TIME**
The `EXPIRE_TIME` networking parameter of `cman.ora` file specifies a time interval, in minutes, to send a check to verify that client/gateway connections are active.

• **IDLE_TIMEOUT**

• **INBOUND_CONNECT_TIMEOUT**

• **LOG_DIRECTORY**

• **LOG_FILE_NUM**
The `LOG_FILE_NUM` networking parameter of the `cman.ora` file specifies the number of log file segments.

• **LOG_FILE_SIZE**
The `LOG_FILE_SIZE` networking parameter of the `cman.ora` file specifies the size of each log file segment.

• **LOG_LEVEL**

• **MAX_ALL_CONNECTIONS**

• **MAX_CMCTL_SESSIONS**

• **MAX_CONNECTIONS**

• **MAX_GATEWAY PROCESSES**

• **MAX_REG_CONNECTIONS**

• **MIN_GATEWAY PROCESSES**

• **OUTBOUND_CONNECT_TIMEOUT**

• **PASSWORD_instance_name**

• **REGISTRATION_EXCLUDED_NODES**
The Oracle Connection Manager parameter file (`cman.ora`) `REGISTRATION_EXCLUDED_NODES` specifies the list of nodes that cannot register with the listener.

• **REGISTRATION_INVITED_NODES**
The Oracle Connection Manager parameter file (`cman.ora`) `REGISTRATION_INVITED_NODES` parameter specifies the list of node that can register with the listener.
• **RULE**
  
  **GROUP Parameter**
  Use the `GROUP` parameter to specify a `rule_list` for a service.
  
• **SDU**

• **SERVICE_RATE**
  The `SERVICE_RATE` parameter of `cman.ora` file specifies incoming connection rate that is allowed per service for an instance.
  
• **SESSION_TIMEOUT**

• **TRACE_FILE**

• **TRACE_FILELEN**

• **TRACE_FILENO**

• **TRACE_LEVEL**

• **TRACE_TIMESTAMP**

• **USE_SID_AS_SERVICE**
  The `USE_SID_AS_SERVICE` Oracle Connection Manager parameter enables the system identifier (SID) in the connect descriptor to be interpreted as a service name when a user attempts a database connection.
  
• **VALID_NODE_CHECKING_REGISTRATION**

• **WALLET_LOCATION**

• **NEXT_HOP Parameter**
  The `NEXT_HOP` parameter provides static routing of client connections from Oracle Connection Manager (Oracle CMAN).
  
• **REST_ADDRESS Parameter**
  Use the `REST_ADDRESS` parameter to configure REST endpoint hostname and port. Oracle CMAN listens to tcps endpoint based on the specified hostname and port.
  
• **MAX_BANDWIDTH_GROUP Parameter**
  The `MAX_BANDWIDTH_GROUP` parameter specifies the maximum number of service that can be configured.
  
• **BANDWIDTH Parameter**
  Use the `BANDWIDTH` parameter to limit all the connections of a service to a specified value in bytes per second.

### 8.2.1 ADDRESS

The `ADDRESS` networking parameter specifies the protocol address of Oracle Connection Manager.

**Purpose**

To specify the protocol address of Oracle Connection Manager.

**Syntax**

```
(ADDRESS=(PROTOCOL=protocol) (HOST=host_name) (PORT=port_number))
```
Example

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

8.2.2 ASO_AUTHENTICATION_FILTER

It is a networking parameter for Oracle Connection Manager. It instructs Oracle Connection Manager to check the connection requests for Secure Network Services (SNS).

Purpose
To specify whether Oracle Database security authentication settings must be used by the client.

Usage Notes
The global setting can be overridden by a rule-level setting in ACTION_LIST.

Values
- **on** to instruct Oracle Connection Manager to reject connection requests that are not using Secure Network Services (SNS). SNS is part of Oracle Database security.
- **off** to instruct Oracle Connection Manager not to check for SNS between the client and server. This is the default.

8.2.3 COMPRESSION

Purpose
To enable or disable data compression. If both the Oracle Connection Manager and the other end (server or client or Oracle Connection Manager) have this parameter set to **ON**, then compression is used for the connection.

Default
**off**

Values
- **on** to enable data compression.
- **off** to disable data compression.

Example

COMPRESSION=on
8.2.4 COMPRESSION_LEVELS

The COMPRESSION_LEVELS networking parameter of the cman.ora file specifies the CPU usage and compression ratio.

Purpose
To specify the compression level.

Usage Notes
The compression levels are used at the time of negotiation to verify which levels are used at both ends, and select one level.

Default
low

Values
- low for low CPU usage and a low compression ratio.
- high for high CPU usage and a high compression ratio.

Example 8-2  Example

COMPRESSION_LEVELS=high,low

8.2.5 COMPRESSION_THRESHOLD

Purpose
To specify the minimum data size, in bytes, for which compression is required.

Usage Notes
Compression is not be done if the size of the data to be sent is less than this value.

Default
1024 bytes

Example

COMPRESSION_THRESHOLD=1024

8.2.6 CONNECTION_STATISTICS

CONNECTION_STATISTICS networking parameter of the cman.ora file specifies whether the SHOW_CONNECTIONS command displays connection statistics.

Purpose
To specify whether the SHOW_CONNECTIONS command displays connection statistics.
Usage Notes
The global setting can be overridden by a rule-level setting in ACTION_LIST.

Values
- **yes** to display statistics.
- **no** to not display statistics. This is the default.

8.2.7 EVENT_GROUP

**EVENT_GROUP** networking parameter of the cman.ora file specifies which event groups are logged.

Purpose
To specify which event groups are logged.

Usage Notes
Multiple events may be designated using a comma-delimited list.

Values
- **alert** for alert notifications.
- **cmd_proc** for command processing.
- **conn_hdlg** for connection handling.
- **init_and_term** for initialization and termination.
- **memory_ops** for memory operations.
- **proc_mgmt** for process management.
- **reg_and_load** for registration and load update.
- **relay** for events associated with connection control blocks.
- **timer** for gateway timeouts.
- **wake_up** for events related to Connection Manager Administration (CMADMIN) wake-up queue.

**Note:**
The event group **ALERT** cannot be turned off.
8.2.8 EXPIRE_TIME

The **EXPIRE_TIME** networking parameter of `cman.ora` file specifies a time interval, in minutes, to send a check to verify that client/gateway connections are active.

**Purpose**

To specify a time interval, in minutes, to send a check to verify that client/server connections are active.

**Usage Notes**

Setting a value greater than 0 ensures that connections are not left open indefinitely, due to an abnormal client termination. If the system supports TCP keepalive tuning, then Oracle Net Services automatically uses the enhanced detection model, and tunes the TCP keepalive parameters.

If the probe finds a terminated connection, or a connection that is no longer in use, then it returns an error, causing the server process to exit.

This parameter is primarily intended for the database server, which typically handles multiple connections at any one time.

Limitations on using this terminated connection detection feature are:

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

**Values**

- 0 to disable dead connection detection. This is the default.
- Any number greater than 0 to enable dead connection detection. The number equals the time interval in minutes.

**Example 8-3**

EXPIRE_TIME=10

8.2.9 IDLE_TIMEOUT

**Purpose**

To specify the amount of time that an established connection can remain active without transmitting data.

**Usage Notes**

The global setting can be overridden by a rule-level setting in `ACTION_LIST`. 
Values

- 0 to disable the timeout. This is the default.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

8.2.10 INBOUND_CONNECT_TIMEOUT

Purpose
To specify how long in seconds the Oracle Connection Manager listener waits for a valid connection from a client or another instance of Oracle Connection Manager.

Values

- 60 sec is the default. Use value 0 to disable timeout.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

8.2.11 LOG_DIRECTORY

Purpose
To specify the directory for the Oracle Connection Manager log files.

Default
ORACLE_BASE_HOME/network/log

8.2.12 LOG_FILE_NUM

LOG_FILE_NUM networking parameter of the cman.ora file specifies the number of log file segments.

Purpose
To specify the number of log file segments. At any point of time there can be only \( n \) log file segments where \( n \) is LOG_FILE_NUM and if the log grows beyond this number, then the older segments are deleted.

Default
No default. Number of segments grow indefinitely, if not specified or set to zero.

Values
Any integer value.

Example 8-4  Example

LOG_FILE_NUM=3
8.2.13 LOG_FILE_SIZE

LOG_FILE_SIZE networking parameter of the `cman.ora` file specifies the size of each log file segment.

**Purpose**
To specify the size of each log file segment. The size is in MB.

**Default**
300 MB

**Values**
Any integer value.

**Example 8-5**  Example

LOG_FILE_SIZE=10

8.2.14 LOG_LEVEL

**Purpose**
To specify the level for log messages.

**Values**
- *off* for no logging. This is the default.
- *user* for user-induced errors log information.
- *admin* for administration log information, such as installation-specific.
- *support* for Oracle Support Services information.

8.2.15 MAX_ALL_CONNECTIONS

**Purpose**
To specify the maximum number of concurrent registration and client connection sessions that can be supported by Oracle Connection Manager.

**Usage Notes**
This number includes registration connections from databases, and ongoing client connection establishment requests. After a connection is established, the clients do not maintain a connection to the listener. This limit only applies to client connections that are in the initial connection establishment phase from a listener perspective.

**Default**
Operating system-specific
Example

MAX_ALL_CONNECTIONS=40

8.2.16 MAX_CMCTL_SESSIONS

Purpose
To specify the maximum number of concurrent local or remote sessions of the Oracle Connection Manager control utility allowable for a given instance.

Usage Notes
One of the sessions must be a local session.

Values
Any number of sessions can be designated.

8.2.17 MAX_CONNECTIONS

Purpose
To specify the maximum number of connection slots that a gateway process can handle.

Values
Any number in the range of 1 to 1024.

8.2.18 MAX_GATEWAY_PROCESSES

Purpose
To specify the maximum number of gateway processes that an instance of Oracle Connection Manager supports.

Values
The number designated must be greater than the minimum number of gateway processes. The maximum is 64.

8.2.19 MAX_REG_CONNECTIONS

Purpose
To specify the maximum number of concurrent registration connection sessions that can be supported by Oracle Connection Manager.

Default
512
Example
MAX_REG_CONNECTIONS=20

8.2.20 MIN_GATEWAY_PROCESSES

Purpose
To specify the minimum number of gateway processes that an instance of Oracle Connection Manager supports.

Values
Any number of sessions can be designated up to 64.

8.2.21 OUTBOUND_CONNECT_TIMEOUT

Purpose
To specify the length of time in seconds that the Oracle Connection Manager instance waits for a valid connection to be established with the database server or with another Oracle Connection Manager instance.

Values
- 60 to disable the timeout. This is the default.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

8.2.22 PASSWORD_instance_name

Purpose
To specify the encrypted instance password, if one has been set.

8.2.23 REGISTRATION_EXCLUDED_NODES

The Oracle Connection Manager parameter file (cman.ora) REGISTRATION_EXCLUDED_NODES specifies the list of nodes that cannot register with the listener.

Purpose
To specify the list of nodes that cannot register with the listener.

Usage Notes
The list can include host names or CIDR notation for IPv4 and IPv6 addresses. The wildcard format (*) is supported for IPv4 addresses. The presence of a host name in the list results in the inclusion of all IP addresses mapped to the host name. The host name should be consistent with the public network interface.
If the `REGISTRATION_INVITED_NODES` parameter and the `REGISTRATION_EXCLUDED_NODES` parameter are set, then the `REGISTRATION_EXCLUDED_NODES` parameter is ignored.

Values
Valid nodes and subnet IP addresses or names.

Example

```
REGISTRATION_EXCLUDED_NODES = 10.1.26.*, 10.16.40.0/24, \
2001:DB8:3eff:fe38, node2
```

8.2.24 REGISTRATION_INVITED_NODES

The Oracle Connection Manager parameter file (`cman.ora`) `REGISTRATION_EXCLUDED_NODES` parameter specifies the list of node that can register with the listener.

Purpose
To specify the list of node that can register with the listener.

Usage Notes
The list can include host names or CIDR notation for IPv4 and IPv6 addresses. The wildcard format (`*`) is supported for IPv4 addresses. The presence of a host name in the list results in the inclusion of all IP addresses mapped to the host name. The host name should be consistent with the public network interface.

If the `REGISTRATION_INVITED_NODES` parameter and the `REGISTRATION_EXCLUDED_NODES` parameter are set, then the `REGISTRATION_EXCLUDED_NODES` parameter is ignored.

Values
Valid nodes and subnet IP addresses or names.

Example

```
REGISTRATION_INVITED_NODES = 10.1.35.*, 10.1.34.0/24, \
2001:DB8:fe38:7303, node1
```

8.2.25 RULE

Purpose
To specify an access control rule list to filter incoming connections.

Usage Notes
A rule list specifies which connections are accepted, rejected, or dropped.
If no rules are specified, then all connections are rejected.
The source and destination can be a host name, IP address, or subnet mask.

There must be at least one rule for client connections and one rule for CMCTL connections. Omitting one or the other results in the rejection of all connections for the rule type omitted. The last rule in the example that follows is a CMCTL rule.

Oracle Connection Manager does not support wildcards for partial IP addresses. If you use a wildcard, then use it in place of a full IP address. The IP address of the client may, for example, be (SRC=*).

Oracle Connection Manager supports only the /nn notation for subnet addresses. In the first rule in Example “Sample cman.ora File”, /27 represents a subnet mask that comprises 27 left-most bits.

Values

This parameter is listed in the rule list section of the cman.ora file preceded by RULE_LIST=.

Syntax

(RULE_LIST=
  (RULE=
    (SRC=host)
    (DST=host)
    (SRV=service_name)
    (ACT=[accept|reject|drop])
    (ACTION_LIST=AUT={on|off}
      (CONN_STATS={yes|no})(MCT=time)(MIT=time)(MOCT=time)))
  (RULE= ...))

Additional Parameters

The RULE parameter filters a connection or group of connections using the following parameters:

SRC: The source host name or IP address of the client.

DST: The destination server host name or IP address of the database server.

SRV: The database service name of Oracle Database obtained from the SERVICE_NAME parameter in the initialization parameter file.

ACT: The action for the connection request. Use accept to accept incoming requests, reject to reject incoming requests, or drop to reject incoming requests without sending an error message.

ACTION_LIST: The rule-level parameter settings for some parameters. These parameters are as follows:

- AUT: Oracle Database security authentication on client side.
- CONN_STATS: Log input and output statistics.
- MCT: Maximum connect time.
- MIT: Maximum idle timeout.
- MOCT: Maximum outbound connect time.

Rule-level parameters override their global counterparts.
Example

RULE_LIST=
RULE=
(SRC=client1-pc)
(DST=sales-server)
(SRV=sales.us.example.com)
(ACT=reject)
(RULE=
(SRC=192.0.2.45)
(DST=192.0.2.200)
(SRV=db1)
(ACT=accept))
(RULE=
(SRC=sale-rep)
(DST=sales1-server)
(SRV=cmob)
(ACT=accept))

8.2.26 GROUP Parameter

Use the GROUP parameter to specify a rule_list for a service.

Purpose

This parameter is listed in the RULE_GROUP section of the cman.ora file preceded by RULE_GROUP=.

Syntax

(GROUP =
 (DESCRIPTION = (NAME = service_name))
 (RULE_LIST =
   (RULE=...)
 )

Usage Notes

The service name (SRV =) in the rule should match the service_name specified in the NAME parameter. Alternatively, you can specify the service name using an asterisk *.

You can configure a DEFAULT_GROUP in RULE_GROUP. The rules that you specify in this section applies to those services that do not have an explicit GROUP. You do not need to specify DESCRIPTION inside a DEFAULT_GROUP.

Example

{RULE_GROUP=
 (GROUP =
   (DESCRIPTION = (NAME = sales.us.example.com))
 (RULE_LIST =
   (RULE=
     (SRC=client1-pc)
     (DST=sales-server)
     (SRV=*)
   )
 )
}
8.2.27 SDU

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

Usage Notes
Oracle Connection Manager can negotiate large SDU with client and server when configured. When the configured values of client, database server, and Oracle Connection Manager do not match for a session, the least value of all the three values is used.

Default
8192 bytes (8 KB)

Values
512 to 2097152 bytes

Example
SDU=32768

8.2.28 SERVICE_RATE

The SERVICE_RATE parameter of cman.ora file specifies incoming connection rate that is allowed per service for an instance.

Purpose
To specify incoming connection rate that is allowed per service for an instance.
Usage Notes

Any user-specified value greater than 0 sets the maximum limit on the number of new connections per service-instance handled by the proxy listener every second. Listener rejects connections after it reaches the maximum limit. Client side connection failure is reported with “TNS:listener: rate limit reached”.

Values

- 0 to disable service rate limit. This is the default.
- Any number greater than 0 to enable service rate limit.

Example 8-6    Example

SERVICE_RATE=10

8.2.29 SESSION_TIMEOUT

Purpose

To specify the maximum time in seconds allowed for a user session.

Usage Notes

The global setting can be overridden by a rule-level setting in ACTION_LIST.

Values

- 0 to disable the timeout. This is the default.
- Any number greater than 0 to enable the timeout. The number equals the timeout period in seconds.

8.2.30 TRACE_FILE

Purpose

To specify the directory for Oracle Connection Manager trace files.

8.2.31 TRACE_FILELEN

Purpose

To specify the size of the trace file in KB.

Usage Notes

When the size is reached, the trace information is written to the next file. The number of files is specified with the TRACE_FILENO parameter.

8.2.32 TRACE_FILENO

Purpose

To specify the number of trace files.
8.2.33 TRACE_LEVEL

Purpose
To specify the level for trace messages.

Values
- **off** for no tracing. This is the default.
- **user** for user-induced errors trace information.
- **admin** for administration trace information, such as installation-specific.
- **support** for Oracle Support Services information.

8.2.34 TRACE_TIMESTAMP

Purpose
To specify the use of a timestamp for the tracing logs.

Usage Notes
If the TRACING parameter is enabled, then a time stamp in the form of `dd-mmm-yyyy hh:mm:ss:mill` for every trace event in the trace file.

Values
- **off** for no timestamp to be included in the file.
- **on** for timestamp to be included in the file.

8.2.35 USE_SID_AS_SERVICE

The **USE_SID_AS_SERVICE** Oracle Connection Manager parameter enables the system identifier (SID) in the connect descriptor to be interpreted as a service name when a user attempts a database connection.

Purpose
To enable the system identifier (SID) in the connect descriptor to be interpreted as a service name when a user attempts a database connection.

Usage Notes
Database clients with earlier releases of Oracle Database that have hard-coded connect descriptors can use this parameter to connect to a container or pluggable database.
When the database is an Oracle Database 12c container database, the client must specify a service name in order to connect to it. Setting this parameter to on instructs the Oracle Connection Manager listener to use the SID in the connect descriptor as a service name and connect the client to the specified database.

Values
- off. This is the default.
- on.

Example 8-7  Example
USE_SID_AS_SERVICE=on.

8.2.36 VALID_NODE_CHECKING_REGISTRATION

Purpose
To determine whether valid node checking registration is performed, and if the subnet is allowed.

Usage Notes
When set to on, valid node checking registration is performed at the listener for any incoming registration request, and only local IP addresses are allowed.

Default
on

Values
- off | 0 to specify valid node checking registration is off, and no checking is performed.
- on | 1 | local to specify valid node checking registration is on, and all local IP addresses can register. If a list of invited nodes is set, then all IP addresses, host names, or subnets in the list as well as local IP addresses are allowed.
- subnet | 2 to specify valid node checking registration is on, and all machines in the local subnets are allowed to register. If a list of invited nodes is set, then all nodes in the local subnets as well as all IP addresses, host names and subnets in the list are allowed.

Example
VALID_NODE_CHECKING_REGISTRATION = on

8.2.37 WALLET_LOCATION

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

The key/value pair for Microsoft certificate store (MCS) omits the METHOD_DATA parameter because MCS does not use wallets. Instead, Oracle PKI (public key infrastructure) applications obtain certificates, trustpoints and private keys directly from the user’s profile.

If an Oracle wallet is stored in the Microsoft Windows registry and the wallet’s key (KEY) is SALESAPP, then the storage location of the encrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12. The storage location of the decrypted wallet is HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO.

Note:
This parameter must be specified outside Oracle Connection Manager alias

Syntax

The syntax depends on the wallet, as follows:

• Oracle wallets on the file system:

  WALLET_LOCATION=
  {SOURCE=}
  {METHOD=file}
  {METHOD_DATA=}
  {DIRECTORY=directory}
  {(PKCS11=TRUE/FALSE})

• Microsoft certificate store:

  WALLET_LOCATION=
  {SOURCE=}
  {METHOD=mcs}

• Oracle wallets in the Microsoft Windows registry:

  WALLET_LOCATION=
  {SOURCE=}
  {METHOD=reg}
  {METHOD_DATA=}
  {KEY=registry_key}

• Entrust wallets:

  WALLET_LOCATION=
  {SOURCE=}
  {METHOD=entr}
  {METHOD_DATA=}
  {PROFILE=file.epf}
  {INIFILE=file.ini}

Additional Parameters

WALLET_LOCATION supports the following parameters:

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

Default
None

Values
ture | false

Examples

Oracle wallets on file system:

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

Microsoft certificate store:

WALLET_LOCATION=
  (SOURCE=
    (METHOD=mcs))

Oracle Wallets in the Microsoft Windows registry:

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

Entrust Wallets:

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

The NEXT_HOP parameter provides static routing of client connections from Oracle Connection Manager (Oracle CMAN).

Purpose
To specify a fixed address for Oracle CMAN to connect and to relay all client connection requests.

Usage Notes
This parameter contains the next hop address to which Oracle CMAN should connect to, whenever there is a client connection to it. This parameter provides static routing of client connections from Oracle CMAN and does not require service registration.

Values
You must specify this parameter in the CONFIGURATION section. Use description or address list to specify multiple addresses along with other characteristics such as load_balance and failover.

Default
Not enabled.

Example

```
CMAN=
  (CONFIGURATION=
    (ADDRESS=(PROTOCOL=tcp)(HOST=proxysvr)(PORT=4555))
    (rule_list=(rule=(src=*)(dst=*)(srv=*)(act=accept)))
    (PARAMETER_LIST=
      (MAX_GATEWAY_PROCESSES=8)
      (MIN_GATEWAY_PROCESSSES=3))
    (NEXT_HOP=(ADDRESS=(PROTOCOL=tcps)(HOST=proxysvr1)(PORT=1555)))
  )
```

8.2.39 REST_ADDRESS Parameter

Use the REST_ADDRESS parameter to configure REST endpoint hostname and port. Oracle CMAN listens to tcps endpoint based on the specified hostname and port.

Usage Notes
Add the REST_ADDRESS attribute under the parameter_list of the cman.ora file.

Syntax

REST_ADDRESS=host name:port

Example

REST_ADDRESS=cman_host:1524
8.2.40 MAX_BANDWIDTH_GROUP Parameter

The `MAX_BANDWIDTH_GROUP` parameter specifies the maximum number of service that can be configured.

Usage Notes

Configure this parameter to a value of maximum services that your system supports. Add this parameter in the `parameter` section of the `cman.ora` file.

You can also configure this parameter with an additional 20% to 100% buffer, depending upon how often the services are created and destroyed in the system.

Example

```
MAX_BANDWIDTH_GROUP = 100
```

8.2.41 BANDWIDTH Parameter

Use the `BANDWIDTH` parameter to limit all the connections of a service to a specified value in bytes per second.

Usage Notes

Specify a limit on the number of bytes transmitted per second. You must include this parameter in the `parameter_list` section of the `cman.ora` file.

Example

```
BANDWIDTH = 524288
```

8.3 Oracle Connection Manager in Traffic Director Mode Parameters

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

- `TDM`
- `TDM_BIND_THREAD`
- `TDM_DATATYPE_CHECK`
- `TDM_PRCP_MAX_CALL_WAIT_TIME`
- `TDM_PRCP_MAX_TXN_CALL_WAIT_TIME`
- `TDM_SHARED_THREADS_MAX`
- `TDM_SHARED_THREADS_MIN`
- `TDM_THREADING_MODE`
8.3.1 TDM

Purpose
To configure Oracle Connection Manager to act as Oracle Connection Manager in Traffic Director Mode.

Default
FALSE

Values
• TRUE
• FALSE

Example
tdm = TRUE

8.3.2 TDM_BIND_THREAD

Purpose
To make the application connection hold on to the TDM thread and has different implications with and without PRCP. This parameter only applies when TDM_THREADING_MODE is set to SHARED.

Usage Notes
Without PRCP, setting this parameter to yes makes the application connection hold on the TDM worker thread as long as there is a transaction in progress.

With PRCP, setting this parameter to yes makes the application connection hold on to the TDM thread from the time OCISessionGet is done by the application till it does an OCISessionRelease.

Default
no

Values
• yes
• no

Example
TDM_BIND_THREAD = yes
8.3.3 TDM_DATATYPE_CHECK

Purpose
To validate all the inbound data to the database, of the data type NUMBER, DATE, 
TIMESTAMP, TIMESTAMP WITH LOCAL TIMEZONE, TIMESTAMP WITH TIMEZONE, BLOB, 
CLOB, BFILE, UROWID and REF. The following error is received by the application if 
there is any problem with the data sent to the Oracle Connection Manager in Traffic 
Director Mode.
ORA-03137: malformed TTC packet from client rejected: [3101]

Usage Notes
Turning ON/OFF this parameter enables or disables the data validation.

Default
OFF

Values
• ON
• OFF

Example
tdm_datatype_check={ON | OFF}

8.3.4 TDM_PRCP_MAX_CALL_WAIT_TIME

Purpose
To record the maximum time of inactivity, in seconds, for a client after obtaining a 
session from the PRCP pool. This parameter is applicable when the Oracle 
Connection Manager in Traffic Director Mode is configured to have Proxy Resident 
Connection Pool.

Usage Notes
After obtaining a session from the PRCP pool, if the client application does not issue a 
database call for the time specified by TDM_PRCP_MAX_CALL_WAIT_TIME parameter, 
then the PRCP session is freed and the client connection is terminated.

Default
30 seconds

Values
Any non negative value. However, Oracle recommends not to use a value of 0 as that 
implies that a connection can acquire a PRCP session for an indefinite amount of time
8.3.5 TDM_PRCP_MAX_TXN_CALL_WAIT_TIME

Purpose

To record the maximum time of inactivity, in seconds, for a client after it obtains a
session from the Proxy Resident Connection Pool and starts a transaction. This
parameter is applicable when the Oracle Connection Manager in Traffic Director Mode
is configured to have PRCP.

Usage Notes

If the client application does not issue a database call for the time specified by
TDM_PRCP_MAX_TXN_CALL_WAIT_TIME parameter while in a transaction, the PRCP
session is freed, the transaction is rolled back, and the client connection is terminated.

Default

0

Values

Any nonnegative value. However, it is recommended not to use a value of 0 as it
implies that a connection can acquire a PRCP session for an indefinite amount of time.

8.3.6 TDM_SHARED_THREADS_MAX

Purpose

To configure the maximum number of threads that an Oracle Connection Manager
process in Traffic Director Mode should have, when tdm_threading_mode is set to
SHARED.

Values

Any number can be designated for the maximum number of threads. For DEDICATED
mode, the maximum number of threads is same as the maximum number of
connections. In SHARED mode, though there is no fixed upper bound, it should ideally
be proportional to the load.

8.3.7 TDM_SHARED_THREADS_MIN

Purpose

To configure the minimum number of threads that an Oracle Connection Manager
process in Traffic Director Mode should have, when tdm_threading_mode is set to
SHARED.

Values

Any number can be designated for the minimum number of threads. For SHARED mode,
there is no limit enforced. However, the number of threads should be proportional to
the load.
8.3.8 TDM_THREADING_MODE

Purpose

To configure the usage of threads by the Oracle Connection Manager in Traffic Director Mode.

Usage Notes

If this parameter is set to DEDICATED, then a worker thread is spawned for each inbound connection and the maximum number of threads is determined by the max_connections parameter.

If this parameter is set to SHARED, then a shared pool of worker threads handle all inbound connections. The minimum number of worker threads is specified by the tdm_shared_threads_min setting and the maximum number of worker threads is specified by the tdm_shared_threads_max setting. The thread pool is internally managed within these bounds.

Default

DEDICATED

Values

• DEDICATED
• SHARED

Example

tdm_threading_mode={DEDICATED | SHARED}

tdm_shared_threads_min = 4

tdm_shared_threads_max = 5

8.4 ADR Diagnostic Parameters for Oracle Connection Manager

The diagnostic data for critical errors is quickly captured and stored in the ADR for Oracle Connection Manager.

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

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error, such as traces and dumps, are immediately captured and tagged with the incident number. The data is then stored in the Automatic Diagnostic Repository (ADR), a file-based repository outside the database.
This section describes the parameters used when ADR is enabled. ADR is enabled by default. Non-ADR parameters listed in the cman.ora file are ignored when ADR is enabled.

- **ADR_BASE**
  It is a diagnostic parameter in the cman.ora file and it specifies the base directory to store tracing and logging incidents when ADR is enabled.

- **DIAG_ADR_ENABLED**
  DIAG_ADR_ENABLED diagnostic parameter of the cman.ora file indicates whether ADR tracing is enabled.

- **LOG_LEVEL**
- **TRACE_LEVEL**
- **TRACE_TIMESTAMP**

### 8.4.1 ADR_BASE

It is a diagnostic parameter in the cman.ora file and it specifies the base directory to store tracing and logging incidents when ADR is enabled.

**Purpose**

To specify the base directory to store tracing and logging incidents when ADR is enabled.

**Default**

The default is `ORACLE_BASE`, or `ORACLE_HOME/log` if `ORACLE_BASE` is not defined.

**Values**

Any valid directory path to a directory with write permission.

**Example 8-8 Example**

ADR_BASE=/oracle/network/trace

### 8.4.2 DIAG_ADR_ENABLED

DIAG_ADR_ENABLED diagnostic parameter of the cman.ora file indicates whether ADR tracing is enabled.

**Purpose**

To indicate whether ADR tracing is enabled.

**Usage Notes**

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

**Values**

`on | off`
Example 8-9  Example

DIAG_ADR_ENABLED=on

8.4.3 LOG_LEVEL

Purpose
To specify the level of logging performed by Oracle Connection Manager.

Usage Notes
This parameter is also applicable when non-ADR logging is used.
The following log files are used with Oracle Connection Manager:
- `instance-name_pid.log` for the listener.
- `instance-name_cmadmin_pid.log` for CMADMIN.
- `instance-name_cmgw_pid.log` for the gateway processes.
The log files are located in the `ORACLE_HOME/network/log` directory.

Default
off or 0

Values
- off or 0 for no log output.
- user or 4 for user log information.
- admin or 10 for administration log information.
- support or 16 for Oracle Support Services log information.

Example
LOG_LEVEL=admin

8.4.4 TRACE_LEVEL

Purpose
To specify the trace level for the Oracle Connection Manager instance.

Usage Notes
This parameter is also applicable when non-ADR tracing is used.
The following trace files are used with Oracle Connection Manager:
- `instance-name_pid.trc` for the listener.
- `instance-name_cmadmin_pid.trc` for CMADMIN.
- `instance-name_cmgw_pid.trc` for the gateway processes.
The log files are located in the `ORACLE_HOME/network/log` directory.
Default
off

Values
• off for no trace output.
• user for user trace information.
• admin for administration trace information.
• support for Oracle Support Services trace information.

Example
TRACE_LEVEL=admin

8.4.5 TRACE_TIMESTAMP

Purpose
To add a time stamp in the form of dd-mmm-yyyy hh:mm:ss:ms to every trace event in the trace file for the listener.

Usage Notes
This parameter is used with the TRACE_LEVEL parameter. This parameter is also applicable when non-ADR tracing is used.

Default
on

Values
• on or true
• off or false

Example
TRACE_TIMESTAMP=true

8.5 Non-ADR Diagnostic Parameters for Oracle Connection Manager

This section lists the parameters used when ADR is disabled:

• LOG_DIRECTORY
• TRACE_DIRECTORY
• TRACE_FILELEN
• TRACE_FILENO
8.5.1 LOG_DIRECTORY

Purpose
To specify the location of Oracle Connection Manager log files.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_BASE_HOME/network/log

Values
Any valid directory path to a directory with write permission.

Example
LOG_DIRECTORY=/oracle/network/log

8.5.2 TRACE_DIRECTORY

Purpose
To specify the location of the Oracle Connection Manager trace files.

Usage Notes
Use this parameter when ADR is not enabled.

Default
ORACLE_BASE_HOME/network/trace

Values
Any valid directory path to a directory with write permission.

Example
TRACE_DIRECTORY=/oracle/network/admin/trace

8.5.3 TRACE_FILELEN

Purpose
To specify the size, in KB, of the trace file.

Usage Notes
When the size is met, the trace information is written to the next file. The number of files is specified with the TRACE_FILENO parameter. Any size can be designated. Use this parameter when ADR is not enabled.
8.5.4 TRACE_FILENO

Purpose
To specify the number of trace files for Oracle Connection Manager tracing.

Usage Notes
When this parameter is set along with the TRACE_FILELEN parameter, trace files are used in a cyclical fashion. The first file is filled first, then the second file, and so on. When the last file has been filled, the first file is reused, and so on. Any number of files can be designated.

The trace file names are distinguished from one another by their sequence number. For example, if this parameter is set to 3, then the gateway trace files would be named instance-name_cmgw1_pid.trc, instance-name_cmgw2_pid.trc and instance-name_cmgw3_pid.trc.

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

Default
1

Example
TRACE_FILENO=3
9

Directory Usage Parameters in the ldap.ora File

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

- **Overview of Directory Server Usage File**
  The ldap.ora file contains directory usage configuration parameters created by Oracle Internet Directory Configuration Assistant, or Oracle Net Configuration Assistant. Do not modify these parameters or their settings.

- **Directory Usage Parameters**
  This section lists and describes the following ldap.ora file configuration parameters.

### 9.1 Overview of Directory Server Usage File

The ldap.ora file contains directory usage configuration parameters created by Oracle Internet Directory Configuration Assistant, or Oracle Net Configuration Assistant. Do not modify these parameters or their settings.

When created with Oracle Internet Directory Configuration Assistant, ldap.ora is located in the ORACLE_HOME/ldap/admin directory. When created with Oracle Net Configuration Assistant, the ldap.ora file is located either in the ORACLE_BASE_HOME/network/admin directory or the ORACLE_HOME/network/admin directory. The ldap.ora file can also be stored in the directory specified by the LDAP_ADMIN or TNS_ADMIN environment variable.

**Related Topics**
- Oracle Internet Directory
- Oracle Net Configuration Assistant

### 9.2 Directory Usage Parameters

This section lists and describes the following ldap.ora file configuration parameters.

- **DEFAULT_ADMIN_CONTEXT**
  DEFAULT_ADMIN_CONTEXT ldap.ora file configuration parameter specifies the default directory for the creation, modification, or search of the connect identifiers.

- **DIRECTORY_SERVER_TYPE**
  DIRECTORY_SERVER_TYPE is a networking parameter of the ldap.ora file and it specifies the type of directory server that is being used.

- **DIRECTORY_SERVERS**
  DIRECTORY_SERVERS is a directory usage parameter and it lists the host names and port number of the primary and alternate LDAP directory servers.
9.2.1 DEFAULT_ADMIN_CONTEXT

DEFAULT_ADMIN_CONTEXT ldap.ora file configuration parameter specifies the default directory for the creation, modification, or search of the connect identifiers.

Purpose
To specify the default directory entry that contains an Oracle Context from which connect identifiers can be created, modified, or looked up.

Values
Valid distinguished name (DN)

Example 9-1 Example

DEFAULT_ADMIN_CONTEXT="o=OracleSoftware,c=US"

9.2.2 DIRECTORY_SERVER_TYPE

DIRECTORY_SERVER_TYPE is a networking parameter of the ldap.ora file and it specifies the type of directory server that is being used.

Purpose
To specify the type of directory server that is being used.

Values
• oid for Oracle Internet Directory
• ad for Microsoft Active Directory

Example 9-2 Example

DIRECTORY_SERVER_TYPE=oid

9.2.3 DIRECTORY_SERVERS

DIRECTORY_SERVERS is a directory usage parameter and it lists the host names and port number of the primary and alternate LDAP directory servers.

Purpose
To list the host names and port number of the primary and alternate LDAP directory servers.

Values

host:port[:sslport]

Example 9-3 Example

DIRECTORY_SERVERS=(ldap-server:389, raffles:400:636)
Appendices

Review information about features no longer supported in this release, upgrade concerns, and information about the Oracle Net Services LDAP schema.

• **Features Not Supported in this Release**
  This appendix describes features no longer supported by Oracle Net Services.

• **Upgrade Considerations for Oracle Net Services**
  This appendix describes the coexistence and upgrade issues for Oracle Net Services.

• **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.
Features Not Supported in this Release

This appendix describes features no longer supported by Oracle Net Services.

- **Overview of Unsupported Features**
  This section describes the features and the configuration file that are no longer being supported in Oracle Database.

- **Unsupported Parameters**
  This section describes the unsupported or obsolete parameters.

- **Unsupported Control Utility Commands**
  This section describes the control utility commands not supported by this release.

- **Unsupported or Deprecated Protocols**
  This section describes the protocols not supported or deprecated since Oracle Database 12c.

### A.1 Overview of Unsupported Features

This section describes the features and the configuration file that are no longer being supported in Oracle Database.

- **Oracle Net Connection Pooling**
  In Oracle Database 12c Release 2 (12.2), Oracle Net connection pooling is no longer supported.

- **Oracle Names**
  Oracle Names is not supported in this release.

- **Oracle Net Listener Password**
  In Oracle Database 12c Release 2 (12.2), the Oracle Net Listener password feature is no longer supported.

### A.1.1 Oracle Net Connection Pooling

In Oracle Database 12c Release 2 (12.2), Oracle Net connection pooling is no longer supported.

**Deprecation of Oracle Net Connection Pooling**

It was deprecated in Oracle Database 11g release. Refer to My Oracle Support note 1469466.1.
A.1.2 Oracle Names

Oracle Names is not supported in this release.

Naming Method

Oracle Names has not been supported as a naming method since Oracle Database 11g. You must migrate to directory naming.

A.1.3 Oracle Net Listener Password

In Oracle Database 12c Release 2 (12.2), the Oracle Net Listener password feature is no longer supported.

Oracle Net Listener Password Support

In Oracle Database 12c Release 2 (12.2), the Oracle Net Listener password feature is no longer supported. This does not cause a loss of security because authentication is enforced through local operating system authentication.

A.2 Unsupported Parameters

This section describes the unsupported or obsolete parameters.

Table A-1  Unsupported Networking Parameters

<table>
<thead>
<tr>
<th>File</th>
<th>Parameter</th>
<th>Description</th>
<th>Last Supported Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>sqlnet.ora</td>
<td>SQLNET.KERBEROS5_CONF_MIT</td>
<td>This parameter was used to specify that MIT Kerberos configuration format was used. Starting with Oracle Database 12c Release 2 (12.2), only the current MIT Kerberos configuration is supported.</td>
<td>11.2</td>
</tr>
<tr>
<td>sqlnet.ora</td>
<td>SQLNET.ALLOWED_LOGON_VERSION</td>
<td>This parameter has been divided into SQLNET.ALLOWED_LOGON_VERSION_CLIENT and SQLNET.ALLOWED_LOGON_VERSION_SERVER.</td>
<td>11.2</td>
</tr>
</tbody>
</table>

A.3 Unsupported Control Utility Commands

This section describes the control utility commands not supported by this release.
Table A-2  Unsupported Network Control Utility Commands

<table>
<thead>
<tr>
<th>Control Utility</th>
<th>Commands</th>
<th>Description</th>
<th>Last Supported Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Names Control Utility</td>
<td>All commands</td>
<td>Oracle Names is no longer supported.</td>
<td>9.2</td>
</tr>
</tbody>
</table>

A.4 Unsupported or Deprecated Protocols

This section describes the protocols not supported or deprecated since Oracle Database 12c.

Table A-3  Unsupported Protocols

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Description</th>
<th>Last Supported Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT LAN Manager (NTLM) protocol for domain authentication</td>
<td>NTLM domain authentication has been deprecated from the Oracle Windows adapter. Only Kerberos authentication is used for the NTS adapter. NTLM is still used for local user authentication, as well as in the case in which the database service runs as a local user.</td>
<td>11.2</td>
</tr>
</tbody>
</table>
B

Upgrade Considerations for Oracle Net Services

This appendix describes the coexistence and upgrade issues for Oracle Net Services.

• Anonymous Access to Oracle Internet Directory
  Typical users of directory naming (LDAP) require anonymous access to the Oracle Internet Directory for name lookup.

B.1 Anonymous Access to Oracle Internet Directory

Typical users of directory naming (LDAP) require anonymous access to the Oracle Internet Directory for name lookup.

Oracle Internet Directory Setting

If you upgrade your Oracle Internet Directory software release 11g or later, then the default setting for Oracle Internet Directory changes to disallow anonymous access to the directory. The directory administrator must configure the directory to enable anonymous binds after upgrading the directory to release 11g. In addition, the way anonymous binds are configured in Oracle Internet Directory changed between Oracle Database 10g and Oracle Database 11g.
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.

- **Structural Object Classes**
  The Oracle schema supports the structural object classes for Oracle Net directory naming lookups.

- **Attributes**
  It lists the attributes used for the object classes. This list is subject to change.

### C.1 Structural Object Classes

The Oracle schema supports the structural object classes for Oracle Net directory naming lookups.

#### Table C-1  Oracle Net Structural Object Classes

<table>
<thead>
<tr>
<th>Object Class</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclDBServer</td>
<td>orclNetDescName, orclVersion</td>
<td>Defines the attributes for database service entries.</td>
</tr>
<tr>
<td>orclNetAddress</td>
<td>orclNetAddressString, orclNetProtocol, orclVersion</td>
<td>Specifies a listener protocol address.</td>
</tr>
<tr>
<td>orclNetAddressAux1</td>
<td>orclNetHostname</td>
<td>Specifies an auxiliary object class to add attributes to an orclNetAddress entry.</td>
</tr>
<tr>
<td>orclNetAddressList</td>
<td>orclNetAddrList, orclNetFailover, orclNetLoadBalance, orclNetSourceRoute, orclVersion</td>
<td>Specifies a list of protocol addresses.</td>
</tr>
<tr>
<td>orclNetDescription</td>
<td>orclNetAddrList, orclNetInstanceName, orclNetConnParamList, orclNetFailover, orclNetLoadBalance, orclNetSdu, orclNetServiceName, orclNetSourceRoute, orclSid, orclVersion</td>
<td>Specifies a connect descriptor containing the protocol address of the listener and the connect information to the service.</td>
</tr>
</tbody>
</table>
Table C-1  (Cont.) Oracle Net Structural Object Classes

<table>
<thead>
<tr>
<th>Object Class</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclNetDescriptionAux1</td>
<td>orclNetSendBufSize, orclNetReceiveBufSize, orclNetFailoverModeString, orclNetInstanceRole</td>
<td>Specifies auxiliary object class to add attributes to an orclNetDescription entry.</td>
</tr>
<tr>
<td>orclNetDescriptionList</td>
<td>orclNetDescList, orclVersion</td>
<td>Specifies a list of connect descriptors.</td>
</tr>
<tr>
<td>orclNetService</td>
<td>orclNetDescName, orclVersion</td>
<td>Defines the attributes for network service name entries.</td>
</tr>
<tr>
<td>orclNetServiceAlias</td>
<td>orclNetDescName, orclVersion</td>
<td>Defines the attributes for network service alias entries.</td>
</tr>
</tbody>
</table>

C.2 Attributes

It lists the attributes used for the object classes. This list is subject to change.

Table C-2  LDAP Schema Attributes for Oracle Net Services

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclCommonContextMap</td>
<td>Allows the mapping of more than one default Oracle Context in the directory server.</td>
</tr>
<tr>
<td>orclNetAddrList</td>
<td>Identifies one or more listener protocol addresses.</td>
</tr>
<tr>
<td>orclNetAddressString</td>
<td>Defines a listener protocol address.</td>
</tr>
<tr>
<td>orclNetConnParamList</td>
<td>Placeholder for connect data parameters.</td>
</tr>
<tr>
<td>orclNetDescList</td>
<td>Identifies one or more connect descriptors.</td>
</tr>
<tr>
<td>orclNetDescName</td>
<td>Identifies a connect descriptor or a list of connect descriptors.</td>
</tr>
<tr>
<td>orclNetFailover</td>
<td>Turns connect-time failover on for a protocol address list.</td>
</tr>
<tr>
<td>orclNetFailoverModeString</td>
<td>Instructs Oracle Net to fail over to a different listener if the first listener fails during runtime. Depending on the configuration, session or any SELECT statements that were in progress are automatically failed over.</td>
</tr>
<tr>
<td>orclNetHostname</td>
<td>Specifies the host name.</td>
</tr>
<tr>
<td>orclNetInstanceName</td>
<td>Specifies the instance name to access.</td>
</tr>
<tr>
<td>orclNetInstanceRole</td>
<td>Specifies a connection to the primary or secondary instance of an Oracle Real Application Clusters (Oracle RAC) configuration.</td>
</tr>
<tr>
<td>orclNetLoadBalance</td>
<td>Turns client load balancing on for a protocol address list.</td>
</tr>
</tbody>
</table>
Table C-2  (Cont.) LDAP Schema Attributes for Oracle Net Services

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclNetProtocol</td>
<td>Identifies the protocol used in the orclAddressString attribute.</td>
</tr>
<tr>
<td>orclNetReceiveBufSize</td>
<td>Specifies the buffer space limit for receive operations of sessions.</td>
</tr>
<tr>
<td>orclNetSdu</td>
<td>Specifies the session data unit (SDU) size.</td>
</tr>
<tr>
<td>orclNetSendBufSize</td>
<td>Specifies the buffer space limit for send operations of sessions.</td>
</tr>
<tr>
<td>orclNetServiceName</td>
<td>Specifies the database service name in the CONNECT_DATA portion.</td>
</tr>
<tr>
<td>orclNetSourceRoute</td>
<td>Instructs Oracle Net to use each address in order until the destination is reached.</td>
</tr>
<tr>
<td>orclSid</td>
<td>Specifies the Oracle system identifier (SID) in the CONNECT_DATA portion of a connection descriptor.</td>
</tr>
<tr>
<td>orclVersion</td>
<td>Specifies the version of software used to create the entry.</td>
</tr>
</tbody>
</table>
Symbols

.IGNORE_ANO_ENCRYPTION_FOR_TCPS networking parameter, 6-26
( ) (parenthesis) symbol reserved in configuration files, 3-3
# (quotation mark) symbol reserved in configuration files, 3-3
= (equal sign) symbol reserved in configuration files, 3-3

Numerics

1024 port, 4-4
1521 port, 4-4
1646 port, 5-50
1830 port, 4-4
2483 port, 4-4
2484 port, 4-4

A

ACCEPT_MD5_CERTS networking parameter, 5-8
ACCEPT_SHA1_CERTS networking parameter, 5-9
ACT networking parameter, 8-15
ACTION_LIST networking parameter, 8-15
ADD_SSLV3_TO_DEFAULT networking parameter, 5-9
ADDRESS networking parameter, 4-1, 6-7, 7-3, 8-5
ADDRESS_LIST networking parameter, 4-2, 6-9
ADMINISTER command, 2-6
ADR, 8-28
ADR diagnostic parameters
sqlnet.ora
ADR_BASE, 5-68
DIAG_ADR_ENABLED, 5-68
ADR_BASE diagnostic parameter, 5-68, 8-29
ADR_BASE_diagnostic parameter, 7-24
ALLOW_MULTIPLE_REDIRECTS_listener_name control parameter, 7-11
anonymous access, 8-1

ASO_AUTHENTICATION_FILTER networking parameter, 8-6
attributes orclCommonContextMap, C-2
orclDescList, C-2
orclDescName, C-2
orclLoadBalance, C-2
orclNetAddrList, C-2
orclNetAddrString, C-2
orclNetConnParamList, C-2
orclNetFailover, C-2
orclNetFailoverModeString, C-2
orclNetHostname, C-2
orclNetInstanceName, C-2
orclNetInstanceRole, C-2
orclNetProtocol, C-2
orclNetReceiveBufSize, C-2
orclNetSdu, C-2
orclNetSendBufSize, C-2
orclNetServiceName, C-2
orclNetSourceRoute, C-2
orclSid, C-2
orclVersion, C-2
authentication ability, 5-20
automatic diagnostic repository, 8-28

B

BEQUEATH_DETACH networking parameter, 5-67

C

class of secure transports parameters
See COST parameters, 7-30
closed client load balancing
configuring, 6-12
CLOSE CONNECTIONS command, 2-7
cman.ora file
control parameters
USE_SID_AS_SERVICE_listener_name, 8-19

anonymous access, 8-1

character sets
for net service name, 3-3
network, for keyword values, 3-3

Index-1
cman.ora file (continued)
diagnostic parameters
ADR_BASE, 8-29
DIAG_ADR_ENABLED, 8-29
LOG_DIRECTORY, 8-32
LOG_LEVEL, 8-30
TRACE_DIRECTORY, 8-32
TRACE_FILELEN, 8-32
TRACE_FILENO, 8-33
TRACE_LEVEL, 8-30
TRACE_TIMESTAMP, 8-31

example, 8-1
listening address section, 8-1
networking parameters
LOG_FILE_NUM, 8-10
LOG_FILE_SIZE, 8-11
parameter list section, 8-1
parameters
ACT, 8-15
ACTION_LIST, 8-15
ADDRESS, 8-5
ASO_AUTHENTICATION_FILTER, 8-6
COMPRESSION, 8-6
COMPRESSION_LEVELS, 8-7
COMPRESSION_THRESHOLD, 8-7
CONNECTION_STATISTICS, 8-7
DST, 8-15
EVENT_GROUP, 8-8
EXPIRE_TIME, 8-9
IDLE_TIMEOUT, 8-9
INBOUND_CONNECT_TIMEOUT, 8-10
LOG_DIRECTORY, 8-10
LOG_LEVEL, 8-11
MAX_ALL_CONNECTIONS, 8-11
MAX_CCMCTL_SESSIONS, 8-12
MAX_CONNECTIONS, 8-12
MAX_GATEWAY_PROCESSES, 8-12
MAX_REG_CONNECTIONS, 8-12
MIN_GATEWAY_PROCESSES, 8-13
OUTBOUND_CONNECT_TIMEOUT, 8-13
PARAMETER_LIST, 8-1
PASSWORD_instance_name, 8-13
RULE, 8-14
SDU, 8-17
SERVICE_RATE, 8-17
SERVICE_RATE_listener_name, 8-18
SESSION_TIMEOUT, 8-18
SRC, 8-15
SRV, 8-15
TDM, 8-25
TDM_BIND_THREAD, 8-25
TDM_DATATYPE_CHECK, 8-26
TDM_PRCP_MAX_CALL_WAIT_TIME, 8-26
TDM_PRCP_MAX_TXN_CALL_WAIT_TIME, 8-27

rule list section, 8-1
COMPRESSION networking parameters, 6-30
COMPRESSION_LEVELS networking parameter, 6-31
COMPRESSION_THRESHOLD networking parameter, 8-7
connect descriptors, 6-1
CONNECT_DATA networking parameter, 6-18
CONNECT_TIMEOUT, 6-28
CONNECT_TIMEOUT networking parameter, 6-28
CONNECTION_RATE_listener_name configuration parameter, 7-8
CONNECTION_STATISTICS networking parameter, 8-7
connections
adjusting listener queue size to avoid errors, 7-5
control parameters
listener.ora
SSL_CLIENT_AUTHENTICATION, 7-18
WALLET_LOCATION, 7-21
control utilities
Listener Control utility, 1-25
Oracle Connection Manager Control utility, 2-3
COST parameters, 7-30
combining, 7-31
DYNAMIC_REGISTRATION_listener_name, 7-32
SECURE_CONTROL_listener_name, 7-33
SECURE_PROTOCOL_listener_name, 7-32
SECURE_REGISTER_listener_name, 7-30
CRS_NOTIFICATION_listener_name control parameter, 7-12
DEDICATED_THROUGH_BROKER_LISTENER networking parameter, 7-13
DEFAULT_ADMIN_CONTEXT networking parameter, 9-2
DEFAULT_SDU_SIZE networking parameter, 5-10
DEFAULT_SERVICE_listener_name control parameter, 7-13
DESCRIPTION networking parameter, 6-6, 7-4
DESCRIPTION_LIST networking parameter, 6-6
DIAG_ADR_ENABLED diagnostic parameter, 5-68, 8-29
DIAG_ADR_ENABLED_listener_name diagnostic parameter, 7-24
diagnostic parameters
cman.ora
ADR_BASE, 8-29
DIAG_ADR_ENABLED, 8-29
LOG_DIRECTORY, 8-32
LOG_LEVEL, 8-30
TRACE_DIRECTORY, 8-32
TRACE_FILELEN, 8-32
TRACE_FILENO, 8-33
TRACE_LEVEL, 8-30
TRACE_TIMESTAMP, 8-31
listener.ora, 7-26
ADR_BASE_listener_name, 7-24
LOG_DIRECTORY_listener_name, 7-27
LOG_FILE_listener_name, 7-28
LOG_FILE_NUM_listener_name, 7-25
LOG_FILE_SIZE_listener_name, 7-25
TRACE_DIRECTORY_listener_name, 7-28
TRACE_FILE_listener_name, 7-28
TRACE_FILEAGE_listener_name, 7-29
TRACE_FILELEN_listener_name, 7-29
TRACE_FILENO_listener_name, 7-29
TRACE_LEVEL_listener_name, 7-26
TRACE_TIMESTAMP_listener_name, 7-27
Oracle Net Listener diagnostic reference, 7-23
sqlnet.ora
ADR_BASE, 5-68
DIAG_ADR_ENABLED, 5-68
LOG_DIRECTORY_CLIENT, 5-72
LOG_DIRECTORY_SERVER, 5-73
LOG_FILE_CLIENT, 5-73
LOG_FILE_SERVER, 5-73
TRACE_DIRECTORY_CLIENT, 5-74
TRACE_DIRECTORY_SERVER, 5-74
TRACE_FILE_CLIENT, 5-75
TRACE_FILE_SERVER, 5-75
diagnostic parameters (continued)
sqlnet.ora (continued)
TRACE_FILEAGE_CLIENT, 5-76
TRACE_FILEAGE_SERVER, 5-76
TRACE_FILELEN_CLIENT, 5-77
TRACE_FILELEN_SERVER, 5-77
TRACE_FILENO_CLIENT, 5-77
TRACE_FILENO_SERVER, 5-78
TRACE_LEVEL_CLIENT, 5-69
TRACE_LEVEL_SERVER, 5-69
TRACE_TIMESTAMP_CLIENT, 5-70
TRACE_TIMESTAMP_SERVER, 5-70
TRACE_UNIQUE_CLIENT, 5-79
sqlnet.ora diagnostic reference, 5-67
directory naming
configuring, 5-13
DIRECTORY_SERVER_TYPE networking parameter, 9-2
DIRECTORYSERVERS, 9-2
DISABLE_OOB networking parameter, 5-11
DISABLE_OOB_AUTO networking parameter, 5-11
DST networking parameter, 8-15
DYNAMIC_REGISTRATION_listener_name
COST parameter, 7-32
E
ENABLE networking parameter, 6-10
ENABLE_EXADIRECT_listener_name control parameter, 7-12
error messages
ORA-12170, 5-60
ORA-12525, 1-12, 7-14
ORA-12535, 5-51, 5-52
EVENT_GROUP networking parameter, 8-8
Exadirect protocol
parameters for addresses, 4-2
EXADIRECT_FLOW_CONTROL networking parameter, 5-9
EXADIRECT_RECV POLL networking parameter, 5-10
EXIT command
Listener Control utility, 1-5
Oracle Connection Manager Control utility, 2-9
EXPRI E_TIME networking parameter, 8-9
external naming
Network Information Service (NIS), 5-13
F
failover
connect-time, 6-11
FAILOVER networking parameter, 6-11, 6-12
Firewall networking parameter, 7-4

G

GLOBAL_NAME networking parameter, 6-20

H

HELP command
  of Listener Control utility, 1-5
  of Oracle Connection Manager Control utility, 2-9
HTTPS_PROXY networking parameter, 6-7
HTTPS_PROXY_PORT networking parameter, 6-8
HTTPS_SSL_VERSION networking parameter, 5-12

I

IDLE_TIMEOUT networking parameter, 8-9
  improving client load balancing, 6-12
INBOUND_CONNECT_TIMEOUT networking parameter, 8-10
INBOUND_CONNECT_TIMEOUT_listener_name control parameter, 7-14
INBOUND_CONNECT_TIMEOUT_listener_name networking parameter, 7-14
INSTANCE_NAME networking parameter, 6-21
IP networking parameter, 7-4
IPC protocol addresses, 4-2
KEY parameter, 4-2
IPC, parameters for addresses, 4-2
IPC_KEYPATH networking parameter, 5-12

K

keepalive feature, 6-10
  keyword syntax rules for configuration files, 3-2
  keyword values and network character sets, 3-3

L

ldap.ora file
  DEFAULT_ADMIN_CONTEXT parameter, 9-2
  DIRECTORY_SERVER_TYPE parameter, 9-2
Listener Control utility
  command reference, 1-25
  commands
    EXIT, 1-5

Listener Control utility (continued)
  commands (continued)
    HELP, 1-5
    QUIT, 1-6, 1-7
    RELOAD, 1-7
    SAVE_CONFIG, 1-8
    SERVICES, 1-9
    SET, 1-2, 1-10
    SETCONNECT_TIMEOUT, 1-11
    SETCURRENT_LISTENER, 1-11
    SETDISPLAYMODE, 1-12
    SETINBOUND_CONNECT_TIMEOUT, 1-12
    SETLOG_DIRECTORY, 1-13
    SETLOG_FILE, 1-14
    SETLOG_STATUS, 1-15
    SETTRC_DIRECTORY, 1-16
    SETTRC_FILE, 1-17
    SETTRC_LEVEL, 1-18
    SETUSE_PLUGINANDPLAY, 1-19
    SHOW, 1-2, 1-19
    SHOWCURRENT_LISTENER, 1-19
    SHOWDISPLAYMODE, 1-19
    SHOWINBOUND_CONNECT_TIMEOUT, 1-19
    SHOWLOG_DIRECTORY, 1-19
    SHOWLOG_FILE, 1-19
    SHOWLOG_STATUS, 1-19
    SHOWRAWMODE, 1-19
    SHOWSAVE_CONFIG_ON_STOP, 1-19
    SHOWTTRC_DIRECTORY, 1-19
    SHOWTTRC_FILE, 1-19
    SHOWTTRC_LEVEL, 1-19
    START, 1-21
    STATUS, 1-22
    STOP, 1-24
    TRACE, 1-25
    VERSION, 1-25
  distributed operation, 1-2
  function of and syntax format, 1-1
  remote administration, 1-2
Listener Control utility access, 1-3
listener.ora file
  ADR diagnostic parameters, 7-23
  configuration parameter reference, 7-23
  configuration parameters
    CONNECTION_RATE_listener_name, 7-8
    RATE_LIMIT, 7-8
  control parameters, 7-10, 7-20
    ALLOW_MULTIPLE_REDIRECTS_listener_name, 7-11
    CRS_NOTIFICATION_listener_name, 7-12
    DEFAULT_SERVICE_listener_name, 7-13
    ENABLE_EXADIRECT_listener_name, 7-12
    INBOUND_CONNECT_TIMEOUT_listener_name, 7-14
    MAX_ALL_CONNECTIONS_listener_name, 7-15
listener.ora file (continued)
control parameters (continued)
MAX_REG_CONNECTIONS_listener_name, 7-15
REGISTRATION_EXCLUDED_NODES_listener_name, 7-16
REGISTRATION_INVITED_NODES_listener_name, 7-16
REMOTE_REGISTRATION_ADDRESS_listener_name, 7-16
SAVE_CONFIG_ON_STOP_listener_name, 7-17
SERVICE_RATE_listener_name, 7-18
SSL_CLIENT_AUTHENTICATION, 7-18
SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name, 7-19
USE_SID_AS_SERVICE_listener_name, 7-20
VALID_NODE_CHECKING_REGISTRATION_listener_name, 7-20
WALLET_LOCATION, 7-21
COST parameters, 7-30
diagnostic parameters, 7-26
ADR_BASE_listener_name, 7-24
DIAG_ADR_ENABLED_listener_name, 7-24
LOG_DIRECTORY_listener_name, 7-27
LOG_FILE_listener_name, 7-28
LOG_FILE_NUM_listener_name, 7-25
LOG_FILE_SIZE_listener_name, 7-25
TRACE_DIRECTORY_listener_name, 7-28
TRACE_FILE_listener_name, 7-28
TRACE_FILEAGE_listener_name, 7-29
TRACE_FILELEN_listener_name, 7-29
TRACE_FILENO_listener_name, 7-29
TRACE_LEVEL_listener_name, 7-26
TRACE_TIMESTAMP_listener_name, 7-27
ADDRESS, 7-3
DEDICATED_THROUGH_BROKER_LISTENER, 7-13
DESCRIPTION, 7-4
Firewall, 7-4
INBOUND_CONNECT_TIMEOUT_listener_name, 7-14
IP, 7-4
QUEUESIZE, 7-5
RECV_BUF_SIZE, 7-5
SEND_BUF_SIZE, 7-6
SSL_VERSION, 7-19
listeners
adjusting queue size, 7-5
connect-request timeouts, 7-14
load balancing
client, 6-12
LOAD_BALANCE networking parameter, 6-12
local naming
configuring, 5-13
LOCAL_REGISTRATION_ADDRESS_listener_name control parameter, 7-14
LOG_DIRECTORY diagnostic parameter, 8-32
LOG_DIRECTORY networking parameter, 8-10
LOG_DIRECTORY_CLIENT diagnostic parameter, 5-72
LOGDIRECTORY_listener_name diagnostic parameter, 7-17
LOG_DIRECTORY_SERVER diagnostic parameter, 5-73
LOG_FILE_CLIENT diagnostic parameter, 5-73
LOG_FILE_NUM networking parameter, 8-10
LOG_FILE_SIZE_listener_name diagnostic parameter, 7-28
LOG_FILE_SERVER diagnostic parameter, 5-73
LOG_FILE_SIZE networking parameter, 8-11
LOG_LEVEL diagnostic parameter, 8-30
LOG_LEVEL networking parameter, 8-11
LOGGING_listener_name diagnostic parameter, 7-26
M
MAX_ALL_CONNECTIONS control parameter, 8-11
MAX_ALL_CONNECTIONS_listener_name control parameter, 7-15
MAX_CMCTL_SESSIONS networking parameter, 8-12
MAX_CONNECTIONS networking parameter, 8-12
MAX_GATEWAY_PROCESSES networking parameter, 8-12
MAX_REG_CONNECTIONS control parameter, 8-12
MAX_REG_CONNECTIONS_listener_name control parameter, 7-15
MIN_GATEWAY_PROCESSES networking parameter, 8-13
N
NAME_DEFAULT.DOMAIN networking parameter, 5-13
NAMES.DIRECTORY_PATH networking parameter, 5-13
ezconnect, 5-13
hostname, 5-13
NAMES.LDAP_AUTHENTICATE_BIND networking parameter, 5-14
NAMES.LDAP_CONN_TIMEOUT networking parameter, 5-14
NAMES.LDAP_PERSISTENT_SESSION networking parameter, 5-15
NAMES.NIS.META_MAP networking parameter, 5-15
NAMESCTL.TRACE_UNIQUE networking parameter, 5-16
network character sets, keyword values, 3-3
network configuration files
  listener.ora, 7-23
  sqlnet.ora, 5-2, 5-67
  syntax rules, 3-1
Network Information Service external naming configuring, 5-13
networking parameters
cman.ora
  LOG_FILE_SIZE, 8-11
listener.ora
  LOG_FILE_NUM, 8-10
listener.ora configuration reference, 7-23
sqlnet.ora configuration reference, 5-2

Oracle Connection Manager Control utility (continued) commands (continued)
  SET ASO_AUTHENTICATION_FILTER, 2-14
  SET CONNECTION_STATISTICS, 2-15
  SET EVENT, 2-16
  SET IDLE_TIMEOUT, 2-17
  SET INBOUND_CONNECT_TIMEOUT, 2-17
  SET LOG_DIRECTORY, 2-18
  SET LOG_LEVEL, 2-19, 2-23
  SET OUTBOUND_CONNECT_TIMEOUT, 2-20
  SET PASSWORD, 2-21
  SET SESSION_TIMEOUT, 2-22
  SET TRACE_DIRECTORY, 2-23
  SET TRACE_LEVEL, 2-23
  SHOW, 2-24
  SHOW ALL, 2-25
  SHOW CONNECTIONS, 2-26
  SHOW DEFAULTS, 2-30
  SHOW EVENTS, 2-31
  SHOW GATEWAYS, 2-31
  SHOW PARAMETERS, 2-32
  SHOW RULES, 2-33
  SHOW SERVICES, 2-35
  SHOW STATUS, 2-36
  SHOW VERSION, 2-36
  SHUTDOWN, 2-37
  STARTUP, 2-38
  SUSPEND GATEWAY, 2-40

Oracle Connection Manager parameters
  ADDRESS, 8-5
  ASO_AUTHENTICATION_FILTER, 8-6
  CONNECTION_STATISTICS, 8-7
  EVENT_GROUP, 8-8
  IDLE_TIMEOUT, 8-9
  INBOUND_CONNECT_TIMEOUT, 8-10
  LOG_DIRECTORY, 8-10
  LOG_LEVEL, 8-11
  MAX_CMCTL_SESSIONS, 8-12
  MAX_CONNECTIONS, 8-12
  MAX_GATEWAY PROCESSES, 8-12
  MIN_GATEWAY PROCESSES, 8-13
  OUTBOUND_CONNECT_TIMEOUT, 8-13
  PASSWORD_instance_name, 8-13
  RULE, 8-14
  SESSION_TIMEOUT, 8-18
  TDM, 8-25
  TDM_BIND THREAD, 8-25
  TDM_DATATYPE_CHECK, 8-26
  TDM_PRCP_MAX_CALL_WAIT_TIME, 8-26
  TDM_PRCP_MAX_TXN_CALL_WAIT_TIME, 8-27
  TDM_SHARED_THREADS_MAX, 8-27
  TDM_SHARED_THREADS_MIN, 8-27

O

object classes
  orclNetAddress, C-1
  orclNetAddressList, C-1
  orclNetDescription, C-1
  orclNetDescriptionList, C-1
  orclNetServiceAlias, C-1
obsolete parameters, A-2
ORA-12170 error message, 5-60
ORA-12525 error message, 1-12, 7-14
ORA-12535 error message, 5-51, 5-52
ORA-12535: TNS:operation timed out and error messages
  ORA-12535, 5-51, 5-52
  ORA-12608, 5-52
  ORA-12535 error message, 5-51, 5-52
  ORA-12608 error message, 5-52
Oracle Connection Manager
  client load balancing, 6-4
  connect-time failover, 6-4
  SOURCE_ROUTE networking parameter, 6-15
Oracle Connection Manager Control utility
  command reference, 2-3
  commands
    ADMINISTER, 2-6
    CLOSE CONNECTIONS, 2-7
    EXIT, 2-9
    HELP, 2-9
    QUIT, 2-10
    RELOAD, 2-11
    RESUME GATEWAYS, 2-12
    SAVE_PASSWD, 2-12
    SET, 2-13
Oracle Connection Manager parameters (continued)
  TDM_THREADING_MODE, 8-28
  TRACE_FILE, 8-18
  TRACE_FILELEN, 8-18
  TRACE_FILENO, 8-18
  TRACE_LEVEL, 8-19
  TRACE_TIMESTAMP, 8-19
Oracle Internet Directory access, B-1
Oracle Names support, A-2
Oracle Net Connection Pooling, A-1
Oracle Net Listener Password, A-2
Oracle protocol support
  configuring addresses, 4-2
    Exadirect, 4-2
    IPC, 4-2
    Named Pipes, 4-2
    SDP, 4-2
    TCP/IP, 4-2
    TCP/IP with SSL, 4-2
Oracle Real Application Clusters
  connect-time failover, 6-12
  FAILOVER networking parameter, 6-12
  INSTANCE_NAME networking parameter, 6-21
  orclCommonContextMap attribute, C-2
  orclDescList attribute, C-2
  orclDescName attribute, C-2
  orclLoadBalance attribute, C-2
  orclNetAddress object class, C-1
  orclNetAddressList object class, C-1
  orclNetAddrList attribute, C-2
  orclNetAddrString attribute, C-2
  orclNetConnParamList attribute, C-2
  orclNetDescription object class, C-1
  orclNetDescriptionList object class, C-1
  orclNetFailover attribute, C-2
  orclNetFailoverModeString attribute, C-2
  orclNetHostname attribute, C-2
  orclNetInstanceName attribute, C-2
  orclNetInstanceRole attribute, C-2
  orclNetReceiveBufSize attribute, C-2
  orclNetSdu attribute, C-2
  orclNetSendBufSize attribute, C-2
  orclNetServiceAlias object class, C-1
  orclNetServiceName attribute, C-2
  orclNetSourceRoute attribute, C-2
  orclProtocol attribute, C-2
  orclSid attribute, C-2
  orclVersion attribute, C-2
  outbound connect timeout interval, 5-44
  OUTBOUND_CONNECT_TIMEOUT networking parameter, 8-13
P
  PARAMETER_LIST networking parameter, 8-1
  PASSWORD_instance_name networking parameter, 8-13
  port 1024, 4-4
  port 1521, 4-4
  port 1646, 5-50
  port 1830, 4-4
  port 2483, 4-4
  port 2484, 4-4
  port numbers, allowed, 4-4
  ports
    privileged, 4-4
    privileged ports, 4-4
  protocols
    authentication ability, 5-20
    configuring addresses, 4-2
    Exadirect, 4-2
    IPC, 4-2
    Named Pipes, 4-2
    SDP, 4-2
    TCP/IP, 4-2
    TCP/IP with SSL, 4-2
Q
  QUEUESIZE networking parameter, 7-5
  QUIT command
    of Listener Control utility, 1-6, 1-7
    of Oracle Connection Manager Control utility, 2-10
R
  RATE_LIMIT configuration parameter, 7-8
  RDB_DATABASE networking parameter, 6-21
  RECV_BUF_SIZE, 5-16
  RECV_BUF_SIZE networking parameter, 6-11, 7-5
  reference
    ADR for Oracle Net Listener, 7-23
    ADR for sqlnet.ora, 5-67
    for Listener Control utility commands, 1-25
    for listener.ora, 7-23
    for Oracle Connection Manager Control utility commands, 2-3
    for sqlnet.ora, 5-2
  REGISTRATION_EXCLUDED_NODES_listener_name control parameter, 7-16, 8-13
  REGISTRATION_INVITED_NODES_listener_name control parameter, 7-16, 8-14
  RELOAD command, 2-11
  of Listener Control utility, 1-7
REMOTE_REGISTRATION_ADDRESS_listener_name control parameter, 7-17
RESUME GATEWAYS command, 2-12
RETRY_COUNT, 6-29
RETRY_COUNT networking parameter, 6-29
RETRY_DELAY, 6-29
RETRY_DELAY networking parameter, 6-29
RULE networking parameter, 8-14
rules, syntax for network configuration files, 3-1

SAVE_CONFIG command
of Listener Control utility, 1-8
SAVE_CONFIG_ON_STOP_listener_name control parameter, 7-17
SAVE_PASSWD command, 2-12
SDP protocol, parameters for addresses, 4-2
SDP.PF_INET_SDP networking parameter, 5-17
SDU networking parameter, 6-13, 8-17
SEC_USER_AUDIT_ACTION_BANNER networking parameter, 5-17
SEC_USER_UNAUTHORIZED_ACCESS_BANNER networking parameter, 5-18
SECURE_CONTROL_listener_name COST parameter, 7-33
SECURE_PROTOCOL_listener_name COST parameter, 7-32
SECURE_REGISTER_listener_name COST parameter, 7-30
SECURITY networking parameter, 6-25
SEND_BUF_SIZE networking parameter, 5-18, 6-14, 7-6
SECURITY networking parameter, 6-25
SEND_BUF_SIZE networking parameter, 5-18, 6-14, 7-6
service name
character set keyword values, 3-3
SERVICE_NAME networking parameter, 6-23
SERVICE_RATE networking parameter, 8-17
SERVICE_RATE parameter, 8-17
SERVICE_RATE_listener_name networking parameter, 7-18
SERVICE_RATE_listener_name parameter, 7-18
SERVICES command, 1-9
SESSION_TIMEOUT networking parameter, 8-18
SET ASO_AUTHENTICATION_FILTER command, 2-14
SET command
of Listener Control utility, 1-10
of Oracle Connection Manager Control utility, 2-13
SET CONNECT_TIMEOUT command, 1-11
SET CONNECTION_STATISTICS command, 2-15
SET CURRENT_LISTENER command, 1-11
SET DISPLAYMODE command
of Listener Control utility, 1-12
SET EVENT command, 2-16
SET IDLE_TIMEOUT command, 2-17
SET INBOUND_CONNECT_TIMEOUT command
of Listener Control utility, 1-12
SET INBOUND_CONNECT_TIMEOUT command, of Oracle Connection Manager Control utility, 2-17
SET LOG_DIRECTORY command
of Listener Control utility, 1-13
of Oracle Connection Manager Control Utility, 2-18
SET LOG_FILE command, 1-14
SET LOG_LEVEL command, 2-19, 2-23
SET LOG_STATUS command, 1-15
SET OUTBOUND_CONNECT_TIMEOUT command, 2-20
SET PASSWORD command
of Oracle Connection Manager Control utility, 2-21
SET SAVE_CONFIG_ON_STOP command, 1-16
of Listener Control utility, 1-16
SET SESSION_TIMEOUT command, 2-22
SET TRACE_DIRECTORY command, 2-23
SET TRACE_LEVEL command, 2-23
SET TRC_DIRECTORY command, 1-16
SET TRC_FILE command, 1-17
SET TRC_LEVEL command, 1-18
SET USE_PLUGANDPLAY command, 1-19
SHARDING_KEY networking parameter, 6-22
SHOW ALL command, 2-25
SHOW command
of Listener Control utility, 1-19
of Oracle Connection Manager Control utility, 2-24
SHOW CONNECTIONS command, 2-26
SHOW CURRENT_LISTENER command, 1-19
SHOW DEFAULTS command, 2-30
SHOW DISPLAYMODE command
of Listener Control utility, 1-19
SHOW EVENTS command, 2-31
SHOW GATEWAYS command, 2-31
SHOW INBOUND_CONNECT_TIMEOUT command, 1-19
SHOW LOG_DIRECTORY command, 1-19
SHOW LOG_FILE command, 1-19
SHOW LOG_STATUS command, 1-19
SHOW PARAMETERS command, 2-32
SHOW RAWMODE command, 1-19
SHOW RULES command, 2-33
SHOW SAVE_CONFIG_ON_STOP command, 1-19
SHOW SERVICES command, 2-35
SHOW STATUS command, 2-36
SHOW TRC DIRECTORY command, 1-19
SHOW TRC_FILE command, 1-19
SHOW TRC_LEVEL command, 1-19
SHOW VERSION command, 2-36
SHOW TRC_DIRECTORY command, 1-19
SHOW TRC_FILE command, 1-19
SHOW TRC_LEVEL command, 1-19
SHOW VERSION command, 2-36
SHUTDOWN command, 2-37
SOURCE ROUTE networking parameter, 6-15
SQLNET.ALLOWED_LOGON_VERSION_CLIENT networking parameter, 5-19
SQLNET.ALLOWED_LOGON_VERSION_SERVER networking parameter, 5-20
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE networking parameter, 5-16
SQLNET.AUTHENTICATION_SERVICES networking parameter, 5-25
SQLNET.CLIENT_REGISTRATION networking parameter, 5-27
SQLNET.CLOUD_USER networking parameter, 5-27
SQLNET.COMPRESSION networking parameter, 5-29
SQLNET.COMPRESSION ACCELERATION networking parameter, 5-29
SQLNET.COMPRESSION_LEVELS networking parameter, 5-30
SQLNET.COMPRESSION_THRESHOLD networking parameter, 5-30
SQLNET.CRYPTO_CHECKSUM CLIENT networking parameter, 5-31
SQLNET.CRYPTO_CHECKSUM SERVER networking parameter, 5-31
SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT networking parameter, 5-31
SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER networking parameter, 5-33
SQLNET.DBFW_PUBLIC_KEY networking parameter, 5-33
SQLNET.DOWN_HOSTS_TIMEOUT networking parameter, 5-34
SQLNET.ENCRYPTION_CLIENT networking parameter, 5-34
SQLNET.ENCRYPTION_SERVER networking parameter, 5-35
SQLNET.ENCRYPTION_TYPES_CLIENT networking parameter, 5-36
SQLNET.ENCRYPTION_TYPES_SERVER networking parameter, 5-37
SQLNET.EXPIRE_TIME networking parameter, 5-38
SQLNET.FALLBACK_AUTHENTICATION networking parameter, 5-40
SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCP networking parameter, 5-39
SQLNET.KERBEROS5_CC_NAME networking parameter, 5-40
SQLNET.KERBEROS5_CLOCKSKEW networking parameter, 5-41
SQLNET.KERBEROS5_CONF networking parameter, 5-42
SQLNET.KERBEROS5_CONFLOCATION networking parameter, 5-42
SQLNET.KERBEROS5_REPLAY_CACHE networking parameter, 5-44
sqlnet.ora file
ADR diagnostic parameters, 5-67
collection parameter reference, 5-2
diagnostic parameters
ADR_BASE, 5-68
ADIAG_ADR_ENABLED, 5-68
LOG_DIRECTORY_CLIENT, 5-72
LOG_DIRECTORY_SERVER, 5-73
LOG_FILE_CLIENT, 5-73
LOG_FILE_SERVER, 5-73
TRACE_DIRECTORY_CLIENT, 5-74
TRACE_DIRECTORY_SERVER, 5-74
TRACE_FILE_CLIENT, 5-75
TRACE_FILE_SERVER, 5-75
TRACE_FILEAGE_CLIENT, 5-76
TRACE_FILEAGE_SERVER, 5-76
TRACE_FILELEN_CLIENT, 5-77
TRACE_FILELEN_SERVER, 5-77
TRACE_FILENO_CLIENT, 5-77
TRACE_FILENO_SERVER, 5-78
TRACE_LEVEL_CLIENT, 5-69
TRACE_LEVEL_SERVER, 5-69
TRACE_TIMESTAMP_CLIENT, 5-70
TRACE_TIMESTAMP_SERVER, 5-70
TRACE_UNIQUE_CLIENT, 5-79
parameters
ACCEPT_MD5_CERTS, 5-8
ACCEPT_SHA1_CERTS, 5-9
ADD_SSLV3_TO_DEFAULT, 5-9
BEQUEATH_DETACH, 5-67
DEFAULT_SDU_SIZE, 5-10
DISABLE_OOB, 5-11
DISABLE_OOB_AUTO, 5-11
EXADIRECT_FLOW_CONTROL, 5-9
EXADIRECT_RECVPOOLL, 5-10
sqlnet.ora file (continued)

parameters (continued)
HTTPS_SSL_VERSION, 5-12
IPC.KEYPATH, 5-12
NAMES.DEFAULT_DOMAIN, 5-13
NAMES.DIRECTORY_PATH, 5-13
NAMES.LDAP_AUTHC_IDENTIFY_BIND, 5-14
NAMES.LDAP_CONN_TIMEOUT, 5-14
NAMES.LDAP_PERSISTENT_SESSION, 5-15
NAMES.NIS_META_MAP, 5-15
NAMESCTL.TRACE_UNIQUE, 5-16
RECV_BUF_SIZE, 5-16
SDP.PF_INET_SDP, 5-17
SEC_USER_AUDIT_ACTION_BANNER, 5-17
SEC_USER_UNAUTHORIZED_ACCESS_BANNER, 5-18
SEND_BUF_SIZE, 5-18
SQLNET.ALLOWED_LOGON_VERSION_CLIENT, 5-19
SQLNET.ALLOWED_LOGON_VERSION_SERVER, 5-20
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE, 5-16
SQLNET.AUTHENTICATION_SERVICES, 5-25
SQLNET.CLIENT_REGISTRATION, 5-27
SQLNET.CLOUD_USER, 5-27
SQLNET.COMPRESSION, 5-29
SQLNET.COMPRESSION_ACCELERATION, 5-29
SQLNET.COMPRESSION_LEVELS, 5-30
SQLNET.COMPRESSION_THRESHOLD, 5-30
SQLNET.CRYPTO_CHECKSUM_CLIENT, 5-31
SQLNET.CRYPTO_CHECKSUM_SERVER, 5-31
SQLNET.CRYPTO_CHECKSUM_TYPE_CLIENT, 5-32
SQLNET.CRYPTO_CHECKSUM_TYPE_SERVER, 5-33
SQLNET.DBF_PUBLIC_KEY, 5-33
SQLNET.DOWN_HOSTS_TIMEOUT, 5-34
SQLNET.ENCRYPTION_SERVER, 5-35
SQLNET.ENCRYPTION_TYPES_CLIENT, 5-36
SQLNET.ENCRYPTION_TYPES_SERVER, 5-37
SQLNET.ENCRYPTION_CLIENT, 5-34
SQLNET.EXPIRE_TIME, 5-38
SQLNET.FALLBACK_AUTHENTICATION, 5-40
SQLNET.IGNORE_ANO_ENCRYPTION_FOR_TCP, 5-39
SQLNET.KERBEROS5_CC_NAME, 5-40
SQLNET.KERBEROS5_CLOCKSKEW, 5-41
SQLNET.KERBEROS5_CONF, 5-42
SQLNET.KERBEROS5_CONF_LOCATION, 5-42
SQLNET.KERBEROS5_KEYTAB, 5-43
SQLNET.KERBEROS5_REALMS, 5-43
SQLNET.KERBEROS5_REPLAY_CACHE, 5-44
SQLNET.RADIUS_ALTERNATE, 5-45
SQLNET.RADIUS_ALTERNATE_PORT, 5-46

sqlnet.ora file (continued)

parameters (continued)
SQLNET.RADIUS_ALTERNATE_RETRYS, 5-45
SQLNET.RADIUS_AUTHENTICATION, 5-47
SQLNET.RADIUS_AUTHENTICATION_INTERFACE, 5-47
SQLNET.RADIUS_AUTHENTICATION_PORT, 5-48
SQLNET.RADIUS_AUTHENTICATION_RETRIES, 5-48
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT, 5-49
SQLNET.RADIUS_CHALLENGE_RESPONSE, 5-49
SQLNET.RADIUS_SECRET, 5-50
SQLNET.RADIUS_SEND_ACCOUNTING, 5-50
SQLNET.RECV_TIMEOUT, 5-51
SQLNET.SEND_TIMEOUT, 5-52
SQLNET.URI, 5-53
SQLNET.USE_HTTPS_PROXY, 5-53
SSL_CERT_REVOCATION, 5-54
SSL_CIPHER_SUITES, 5-56
SSL_CRL_FILE, 5-55
SSL_CRL_PATH, 5-56
SSL_EXTENDED_KEY_USAGE, 5-57
SSL_SERVER_DN_MATCH, 5-58
SSL_VERSION, 5-59
TCP.CONNECT_TIMEOUT, 5-60
TCP.EXCLUDED_NODES, 5-60
TCP.INVITED_NODES, 5-61
TCP.NODELAY, 5-61
TCP.QUEUESIZE, 5-61
TCP.VALIDNODE_CHECKING, 5-62
TNSPING.TRACE_DIRECTORY, 5-63
TNSPING.TRACE_LEVEL, 5-63
USE_DEDICATED_SERVER, 5-64
WALLET_LOCATION, 5-65

sqlnet.ora file

parameters
SQLNET.OUTBOUND_CONNECT_TIME networking parameter, 5-44
SQLNET.OUTBOUND_CONNECT_TIMEOUT networking parameter
SQLNET.OUTBOUND_CONNECT_TIMEOUT parameters
SQLNET.OUTBOUND_CONNECT_TIMEOUT networking parameter, 5-44

Index-10
SQLNET.RADIUS_AUTHENTICATION_PORT networking parameter, 5-48
SQLNET.RADIUS_AUTHENTICATION_RETRIEVES networking parameter, 5-48
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT networking parameter, 5-49
SQLNET.RADIUS_CHALLENGE_RESPONSE networking parameter, 5-49
SQLNET.RADIUS_SECRET networking parameter, 5-50
SQLNET.RADIUS_SEND_ACCOUNTING networking parameter, 5-50
SQLNET.RECV_TIMEOUT networking parameter, 5-51
SQLNET.URI networking parameter, 5-53
SQLNET.USE_HTTPS_PROXY networking parameter, 5-53
SQLNET.WALLET_OVERRIDE, 5-53
SRC networking parameter, 8-15
SRV networking parameter, 8-15
SSL_CERT_REVOCATION networking parameter, 5-54
SSL_CIPHER_SUITES networking parameter, 5-56
SSL_CLIENT.AUTHENTICATION control parameter, 7-18
SSL_CRL_FILE networking parameter, 5-55
SSL_CRL_PATH networking parameter, 5-56
SSL_EXTENDED_KEY_USAGE networking parameter, 5-57
SSL_SERVER_DN_MATCH networking parameter, 5-58
SSL_VERSION networking parameter, 5-59, 7-19
START command of Listener Control utility, 1-21
STARTUP command, 2-38
STATUS command of Listener Control utility, 1-22
STOP command of Listener Control utility, 1-24
Structural Object Classes, C-1
SUBSCRIBE_FOR_NODE_DOWN_EVENT_listener_name control parameter, 7-19
SUPER_SHARDING_KEY networking parameter, 6-22
SUSPEND GATEWAY command, 2-40
syntax rules for network configuration files, 3-1
tdn file parameters, 8-24
TDM networking parameter, 8-25
TDM_BIND_THREAD networking parameter, 8-25
TDM_DATATYPE_CHECK networking parameter, 8-26
TDM_PRCP_MAX_CALL_WAIT_TIME networking parameter, 8-26
TDM_PRCP_MAX_TXN_CALL_WAIT_TIME networking parameter, 8-27
TDM_SHARED_THREADS_MAX networking parameter, 8-27
TDM_SHARED_THREADS_MIN networking parameter, 8-27
TDM_THREADING_MODE networking parameter, 8-28
terminated connection detection
EXPIRE_TIME parameter, 8-9
limitations, 5-38, 8-9
SQLNET.EXPIRE_TIME parameter, 5-38
tnms.ora file parameters
COMPRESSION, 6-31
COMPRESSION_LEVELS, 6-31
tnsnames.ora file parameters, 6-30
ADDRESS, 6-7
ADDRESS_LIST, 6-9
CONNECT_DATA, 6-18
CONNECT_TIMEOUT, 6-28
DESCRIPTION, 6-6
DESCRIPTION_LIST, 6-6
ENABLE, 6-10
FAILOVER, 6-11, 6-12
GLOBAL_NAME, 6-20
HTTPS_PROXY, 6-7
HTTPS_PROXY_PORT, 6-8
IGNORE_ANO_ENCRYPTION_FOR_TCPS, 6-26
INSTANCE_NAME, 6-21
tnsnames.ora file (continued)
parameters (continued)
LOAD_BALANCE, 6-12
RDB_DATABASE, 6-21
RECV_BUF_SIZE, 6-12
RETRY_COUNT, 6-29
RETRY_DELAY, 6-29
SDU, 6-13
SECURITY, 6-25
SEND_BUF_SIZE, 6-14
SERVICE_NAME, 6-23
SHARDING_KEY, 6-22
SOURCE_ROUTE, 6-15
SUPER_SHARDING_KEY, 6-22
TRANSPORT_CONNECT_TIMEOUT, 6-30
TYPE_OF_SERVICE, 6-16
TNSPING.TRACE_DIRECTORY networking parameter, 5-63
TNSPING.TRACE_LEVEL networking parameter, 5-63
TRACE command, 1-25
TRACE_DIRECTORY diagnostic parameter, 8-32
TRACE_DIRECTORY_CLIENT diagnostic parameter, 5-74
TRACE_DIRECTORY_listener_name diagnostic parameter, 7-28
TRACE_DIRECTORY_SERVER diagnostic parameter, 5-74
TRACE_FILE networking parameter, 8-18
TRACE_FILE_CLIENT diagnostic parameter, 5-75
TRACE_FILE_listener_name diagnostic parameter, 7-28
TRACE_FILE_SERVER diagnostic parameter, 5-75
TRACE_FILEAGE_CLIENT diagnostic parameter, 5-76
TRACE_FILEAGE_listener_name diagnostic parameter, 7-29
TRACE_FILEAGE_SERVER diagnostic parameter, 5-76
TRACE_FILELEN diagnostic parameter, 8-32
TRACE_FILELEN networking parameter, 8-18
TRACE_FILELEN_CLIENT diagnostic parameter, 5-77
TRACE_FILELEN_listener_name diagnostic parameter, 7-29
TRACE_FILELEN_SERVER diagnostic parameter, 5-79
TRACE_FILENO diagnostic parameter, 8-33
TRACE_FILENO networking parameter, 8-18
TRACE_FILENO_CLIENT diagnostic parameter, 5-77
TRACE_FILENO_SERVER diagnostic parameter, 5-78
TRACE_LEVEL diagnostic parameter, 8-30
TRACE_LEVEL networking parameter, 8-19
TRACE_LEVEL_CLIENT diagnostic parameter, 5-69
TRACE_LEVEL_listener_name, 7-26
TRACE_LEVEL_SERVER diagnostic parameter, 5-69
TRACE_TIMESTAMP diagnostic parameter, 5-69
TRACE_TIMESTAMP networking parameter, 8-31
TRACE_TIMESTAMP_CLIENT diagnostic parameter, 8-19
TRACE_TIMESTAMP_LISTER_NAME diagnostic parameter, 7-27
TRACE_TIMESTAMP_SERVER diagnostic parameter, 5-70
TRACE_UNIQUE_CLIENT diagnostic parameter, 5-79
TRANSPORT_CONNECT_TIMEOUT networking parameter, 6-30
TYPE_OF_SERVICE networking parameter, 6-16

U

Unsupported Control Utility Commands, A-2
Unsupported Protocols, A-3
USEDEDICATED_SERVER networking parameter, 5-64
USE_SID_AS_SERVICE_listener_name control parameter, 7-20, 8-19

V

VALID_NODE_CHECKING_REGISTRATION_listener_name, 7-20
VALID_NODE_CHECKING_REGISTRATION_listener_name control parameter, 7-20, 8-20
VERSION command
of Listener Control utility, 1-25

W

WALLETLOCATION control parameter, 7-21
WALLETLocations networking parameter, 5-65, 8-20