Oracle® Email

Administrator's Guide

Release 9.0.3

August 2002

Part No. B10033-01

ORACLE®

Oracle Email Administrator's Guide, Release 9.0.3

Part No. B10033-01

Primary Author: Ginger Tabora

Graphic Artist: Valarie Moore

Contributors: Ranadhir Aligireddy, Alex Chan, Marcus Chan, Ashish Consul, Claus Cooper, Vikas Dhamija, Tanya Hitaisinee, Sandra Lee, Nagaraj Mandya, Howard Narvaez, Ricardo Rivera, Sekhar Varanasi, Harvinder Walia

Send Us Your Comments

Oracle Email Administrator's Guide, Release 9.0.3

Part No. B10033-01

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

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

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

- Electronic mail: infodev@us.oracle.com
- FAX: (650) 506-7228 Attn: Oracle Email Documentation Manager
- Postal service:
  Oracle Corporation
  Oracle Email Documentation Manager
  500 Oracle Parkway, 4OP11
  Redwood City, CA 94065
  USA

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

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

Table of Contents

3 Administration and Provisioning

4 Servers and Processes

5 Error Messages

6 Command Line Interface

Preface

The Oracle Email Administrator's Guide is intended for anyone planning, configuring, managing, or monitoring Oracle Email. It provides an introduction to the components and concepts of Oracle Email and describes the planning, configuring, and management tasks you will perform.

This preface contains these topics:

- Audience
- Organization
- Related Documentation
- Conventions
- Documentation Accessibility

Audience

The Oracle Email Administrator's Guide is intended for anyone planning, configuring, managing, or monitoring Oracle Email. It provides an introduction to the components and concepts of Oracle Email and describes the planning, configuring, and management tasks you will perform.

Organization

This book contains the following chapters:

Chapter 1, "Introduction"

This chapter contains an overview of the Oracle Email system and describes its major features.

Chapter 2, "Getting Started"

This chapter contains information on the administration tools and explains how to configure, start up, shut down, and refresh the Oracle Email system .

Chapter 3, "Administration and Provisioning"

This chapter contains information on administering Oracle Email.

Chapter 4, "Servers and Processes"

This chapter contains information on the different servers and processes of the Oracle Email system.

Chapter 5, "Error Messages"

This chapter contains information on Oracle Email error messages.

Chapter 6, "Command Line Interface"

This chapter contains information on the Oracle Email command line interface.

Appendix A, "Server Statistics"

This chapter contains information on the Oracle Email server statistics.

Appendix B, "Oracle Email Access Control Lists"

This chapter contains information on the Oracle Email access control lists.

Related Documentation

Oracle Email documentation is available in HTML and PDF.

- *Oracle Collaboration Suite Release Notes*
- *Oracle Collaboration Suite Administrator's Guide*
- *Oracle Collaboration Suite User's Guide*
- *Oracle Voicemail & Fax Release Notes*
- *Oracle Voicemail & Fax Administrator's Guide*
- *Oracle Email Release Notes*
- *Oracle Email Application Developer's Guide*
- Oracle Email *JAVA API Documentation*

For more information, see these Oracle resources:

- *Oracle Enterprise Manager Administrator's Guide*
- *Oracle9i Database Administrator's Guide*
- *Oracle9i Application Server Database Administrator's Guide*
- *Oracle9i SQL Reference*
- *Oracle Net Services Administrator's Guide*

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

`http://oraclestore.oracle.com/`

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

`http://www.oraclebookshop.com/`
Other customers can contact their Oracle representative to purchase printed documentation.

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

`http://otn.oracle.com/admin/account/membership.html`

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

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

To access the database documentation search engine directly, please visit

http://tahiti.oracle.com

For more information on Request for Comments (RFCs), please visit

http://www.ietf.org

Conventions

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

- Conventions in Text
- Conventions in Code Examples

Conventions in Text

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

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle9i Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.

Convention	Meaning	Example
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to open SQL*Plus. The password is specified in the `orapwd` file. Back up the datafiles and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `true`. Connect as `oe` user. The `JRepUtil` class implements these methods.
`lowercase italic monospace (fixed-width) font`	Lowercase italic monospace font represents placeholders or variables.	You can specify the *`parallel_clause`*. Run `U`*`old_release`*`.SQL` where *`old_release`* refers to the release you installed prior to upgrading.

Conventions in Code Examples

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

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

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

Convention	Meaning	Example
`[ ]`	Brackets enclose one or more optional items. Do not enter the brackets.	`DECIMAL (`*`digits`*` [ , `*`precision`*` ])`
`{ }`	Braces enclose two or more items, one of which is required. Do not enter the braces.	`{ENABLE | DISABLE}`
`|`	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	`{ENABLE | DISABLE}` `[COMPRESS | NOCOMPRESS]`

Convention	Meaning	Example
...	Horizontal ellipsis points indicate either: ■ That we have omitted parts of the code that are not directly related to the example ■ That you can repeat a portion of the code	CREATE TABLE ... AS *subquery*; SELECT *col1*, *col2*, ... , *coln* FROM employees;
. . .	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	SQL> SELECT NAME FROM V$DATAFILE; NAME ---------- /fsl/dbs/tbs_01.dbf /fs1/dbs/tbs_02.dbf . . . /fsl/dbs/tbs_09.dbf 9 rows selected.
Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;
Italics	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/*system_password* DB_NAME = *database_name*
UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;
lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;

Documentation Accessibility

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

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

1

Introduction

This chapter provides an overview of the Oracle Email system and describes its major features.

This chapter contains the following topics:

- Oracle Email Overview
- Oracle Email Features

Oracle Email Overview

Oracle Email is a reliable, scalable, and secure messaging system that reduces administration, hardware, and software costs by providing a consolidated mail store.

Oracle Email uses the Oracle9*i* database as a single message store for e-mail taking advantage of the Oracle core competencies in providing access to, storing, and managing all types of information. Using the highly scalable and reliable Oracle9*i* message store as a foundation, Oracle Email provides message delivery, standards-based client access, browser-based clients, and administration utilities.

Oracle Email Features

Oracle Email is designed to grow to almost any size while maintaining its performance and ease of administration. The Oracle Email system can be customized based on how many messages need to be stored, how many users access the system under peak loads, and how many messages are sent and received over a period of time. The Oracle Email internet computing architecture enables customers to support thousands of users on a single system, if necessary. Customers have the option of creating a two-tier system with a single host supporting a few thousand users, or a three-tier system with the protocol access servers are separated from the message database supporting many thousands of users. This architecture enables customers to add hardware at any tier, expanding the system to support a virtually unlimited number of users.

Message Store

Oracle Email stores all messages in the Oracle 9i database. This eliminates the need to synchronize message stores. Oracle Email users can access and manage all messages from the interface of their choice, including a Web browser, phone, PDA, and fax machine. The Oracle9*i* database enables Oracle Email to offer data availability, data integrity, low recovery time, and fault-tolerance capability. Oracle Email takes advantage of Oracle Email multithreading, parallel processing, high availability support, and high performance.

Open Standards-Based Messaging

Oracle Email enables users to access messages with the messaging client of their choice. Messages can be accessed using any client compliant with Internet Message Access Protocol (IMAP4) or Post Office Protocol (POP3), such as Netscape Messenger, Microsoft Outlook Express, or Eudora Pro Lite. Oracle Email provides

directory services using the light-weight directory access protocol (LDAP) standard compliant Oracle Internet Directory.

Thin Client

The Oracle Email Thin Client provides Internet access to Oracle Email through a standard Web browser. Browser-based clients provide all of the advantages of internet computing: increased reliability because no dedicated client is needed; decreased support and administration costs due to the system being maintained in a professional data center; and increased message access because there are no local message storage requirements. Users can access and manage all aspects of their Oracle Email account from the Thin Client.

Extended Server Side Filters

Oracle Email extends the server side filters that were available in the previous release. Filters now provide additional predefined actions that can be applied to a wider range of mail events, including sending, forwarding, deleting, and expunging a message from the system. These filters include a PL/SQL Application Programming Interface (API) that enables customers to write their own customized actions for filters. Server side filters also provide integration with the Oracle9*i* Advanced Queueing to enable sophisticated e-business applications to work with mail. Server Side Filters can be used to identify and delete virus messages as they come. In addition, there is a standalone virus cleanup utility that can be used to scan the whole mail system for virus infections.

Integration With Other Applications

PL/SQL and Java programmers can create custom applications to integrate Oracle Email with other applications. Oracle Email APIs enable applications to directly manipulate stored messages as well as create outgoing messages that follow the MIME standard. Large numbers of messages can be processed and managed by applications integrated with Oracle Email combined with server side filters.

Enhanced Administration Features

Oracle Email simplifies administration and management by integrating with Oracle Enterprise Manager, enabling consolidated, Web-based management of the total Oracle environment as well as integration into existing system monitoring infrastructures. Oracle Email also supports multiple domains with delegated administration on the same system, enabling hosting.

2

Getting Started

This chapter discusses the administration tools and explains how to configure, start up, shut down, and reinitialize the Oracle Email system.

This chapter contains the following topics:

- Pre-Configuration Checklist
- Configuring Oracle Email
- Using the Administration Tools
- Starting Up, Shutting Down, and Reinitializing Oracle Email

Pre-Configuration Checklist

This section describes pre-configuration procedures that must be done prior to confuguring Oracle Email.

This section contains the following topics:

- Verifying the Java and OracleText Options on the Mail Store Database
- Verifying and Starting the Oracle9iAS Infrastructure and Application Servers
- Registering the Database with Oracle Internet Directory
- Setting the Database init.ora Parameters on the Mail Store Database
- Creating Mail Store Tablespaces and Schema

Verifying the Java and OracleText Options on the Mail Store Database

To verify that the Java and Oracle Text Options were installed and configured on the mail store database, run the following sql query as `sysdba`:

```
SQL> select comp_id, version, status from dba_registry;
```

Verifying and Starting the Oracle9*i*AS Infrastructure and Application Servers

Verify that the infrastructure and application servers are running:

```
ps -ef | grep http
```

To start the infrastructure and application servers:

```
% opmnctl startall
```

Registering the Database with Oracle Internet Directory

An Oracle9*i* database is required to install the mail store. Before a database can be configured as a mail store, it must be registered with the Oracle Internet Directory infrastructure. If the database is not already registered with the Oracle Internet Directory, it can be registered using the Oracle database configuration assistant . Once the database is registered with Oracle Internet Directory, at a later point any changes to the connect identifier can be made using Oracle Net Manager.

To register the mail store database with the Oracle Internet Directory infrastructure, the following procedures must be performed using the Oracle Net Configuration Assistant (NETCA) and the Database Configuration Assistant (DBCA):

- Running Oracle Net Configuration Assistant (NETCA)
- Running Oracle Database Configuration Assistant (DBCA)

Running Oracle Net Configuration Assistant (NETCA)

Perform the following steps to run Oracle Net Configuration Assistant (NETCA):

1. Start Oracle Net Configuration Assistant (NETCA):

   ```
   $ORACLE_HOME/bin/netca
   ```

2. Select **Directory Service Usage Configuration**
3. Click **Next**.
4. Select **Select the directory server you want to use. The directory server must already be configured for Oracle usage**.
5. Click **Next**.
6. Select Oracle Internet Directory as the directory server you want to
7. use and choose Next.
8. Enter connect information about the Oracle Internet Directory infrastructure.
9. Click **Next**.
10. Select the root OracleContext (cn=OracleContext) as the default
11. Oracle Context in the directory and finish the netca configuration.

Running Oracle Database Configuration Assistant (DBCA)

Perform the following steps to run the Oracle Database Configuration Assistant (DBCA):

1. Start Oracle Database Configuration Assistant (DBCA) to register the mail store database with Oracle Internet Directory infrastructure.

   ```
   $ORACLE_HOME/bin/dbca
   ```

2. Select **Configure database**.
3. Click **Next**.
4. Select the mail store database instances.
5. Click **Next**.

6. In the Directory Services screen, Select the **Yes, register the database**.
7. Enter a user DN and password to connect to the Oracle Internet Directory infrastructure. For example, `acmeadmin`.
8. Complete the Oracle Database Configuration Assistant (DBCA) configuration.

Setting the Database init.ora Parameters on the Mail Store Database

Set the database `init.ora` parameters on the mail store database to the following values:

```
processes=150 or higher
open_cursors=300 or higher
dml_locks=200 or higher
shared_pool_size=32000000 or higher
java_pool_size=40000000 or higher
```

Creating Mail Store Tablespaces and Schema

The Oracle Email configuration wizard creates tablespaces and schema for mail store. If you want to customize tablespace storage parameters or data files, you can create them before running configuration wizard.

For the name of mail store tablespaces and their default storage parameters refer to the `$ORACLE_HOME/oes/install/sql/tblspc.sql` script.

is 30). The tablespace named ESTERSTORE that is only used for tertiary storage of old messages periodically messages. To enable tertiary storage, pre-create tablespace `ESTERSTORE` on a different disk prior to installation. After which, create an instance of the housekeeping server with the `TertiaryStore` parameter enabled, and parameter `Tertiary Store Age Threshold` to the desired age value in terms of number of days (the default

---

Note: If you pre-create tablespaces, the Oracle Email configuration wizard log shows some errors indicating tablespace creation failed. These errors can be ignored.

See Also: Chapter 4, "Servers and Processes", for more information on tertiary storage and how to configure server proceses

Configuring Oracle Email

This section describes how to configure the Oracle Email mail store, and middle tier servers.

This section contains the following topics:

- Configuring the Oracle Email Mail Store
- Configuring the Oracle Email Middle Tier

Configuring the Oracle Email Mail Store

Configuring the mail store database does the following:

- Creates tablespaces for the mail schema
- Creates mail tables and indexes
- Loads mail related PL/SQL packages
- Loads mail related stored java procedures
- Configures the mail store with Oracle Internet Directory

To configure the Oracle Email mail store, perform the following steps on the middle tier server:

9. Run the `umconfig.sh` script located on the application server:
   - For UNIX, enter the following command:

     ```
     $ORACLE_HOME/oes/bin/umconfig.sh
     ```

   - For Windows, enter the following command:

     ```
     %ORACLE_HOME%\oes\bin\umconfig.bat
     ```

   The Unified Messaging Configuration screen appears.
10. Select **Mail Store Database Configuration**.
11. Click **Next**. The Mail Store Database Configuration screen is displayed.
12. Enter the following information in the corresponding fields:

Table 2–1

Field	Description
Database Hostname	The name of the machine on which the database is located
SID	The system identifier of the mail store
Port Number	The port number on which the listener is listening
System Password	The system password for the host database
CTXSYS Password	The password for the OracleText account

> **Note:** To use OracleText with the Oracle9*i* database, the `ctxsys` account must be unlocked. Installation of OracleText requires the `ctxsys` account to be present.

13. Click **Next**. The `ES_Mail` password screen is displayed.

14. Enter the `ES_MAIL` password and confirm it. If an `ES_MAIL` password is not entered, the default is `es`.

> **Note:** The mail store schema is owned by the `ES_MAIL` database user.

15. Click **Next**. The `UMADMIN` password screen is displayed.

> **Note:** `UMADMIN` is an administrator account created in the Oracle Internet Directory server during the application server installation of Oracle Email. It account owns specific Oracle Email entries in the directory. After installation, administrators should log in to the administration tool using the `UMADMIN` account and create an initial Oracle Email user. Thereafter, they can delegate system and domain administration responsibilities to other users.

16. Enter the `UMADMIN` password and confirm it. If a password is not entered, the default value `welcome` is stored in Oracle Internet Directory and the database as the `UMADMIN` password.
17. Click **Next**. The Create Unified Messaging Domain screen is displayed.
18. Enter the domain name. This domain name is used for users' e-mail addresses.
19. Click **Next**. The Configuration Tools screen is displayed and the mail store configuration begins.

Once the mail store configuration is complete, the End of Installation screen is displayed.

Log files for `umconfig.sh` are located in the following directory:

- Unix:

```
$ORACLE_HOME/oes/log/
```

- Windows:

```
%ORACLE_HOME\oes\log\
```

> **Note:** When adding subsequent mail stores through the `umconfig.sh` script, the following error is reported in `$ORACLE_HOME/oes/log/createmailstore.log` file:
>
> javax.naming.NameNotFoundException: [LDAP: error code 32 - No Such Object]; remaining name
>
> This error message can be ignored.

Manually Configuring the Oracle Email Mail Store

> **Note:** The manual steps described below are an alternative to running the `mailstore.sh` script through the user interface.

Enter the following command to manually run the Oracle Email mail store configuration scripts with parameters:

- On Unix:

```
cd $ORACLE_HOME/oes/bin
install_mailstore.sh <connect_str> true <system_passwd>
<ctxsys_passwd> <SID> <host_name> <port_number> UM_SYSTEM <ORACLE_HOME>
<es_mail_passwd> <umadmin_passwd> <oid_flag> <domain_name>
```

- On Windows:

```
cd %ORACLE_HOME%\oes\bin
install_mailstore.sh <connect_str> true <system_passwd>
<ctxsys_passwd> <SID> <host_name> <port_number> UM_SYSTEM <ORACLE_HOME>
<es_mail_passwd> <umadmin_passwd> <oid_flag> <domain_name>
```

Where:

Table 2–2

Parameter	Description
`connect_str`	The mail store database connect string.

Table 2–2

Parameter	Description
System Password	The system password for the host database
CTXSYS Password	The password for the OracleText account
`SID`	The mail store database SID.
`host_name`	The mail store database host name.
`port_number`	The mail store database port number (default 1521).
`es_mail_passwd`	The password for the ES_MAIL database user. If an `ES_MAIL` password is not entered, the default is `es.`
`umadmin_passwd`	`UMADMIN` is an administrator account created in the Oracle Internet Directory server during the application server installation of Oracle Email. It account owns specific Oracle Email entries in the directory. After installation, administrators should log in to the administration tool using the `UMADMIN` account and create an initial Oracle Email user. Thereafter, they can delegate system and domain administration responsibilities to other users. If a password is not entered, the default value `welcome` is stored in Oracle Internet Directory and the database as the `UMADMIN` password.

Table 2–2

Parameter	Description

oid_flag	f Oracle Email entries have been created in Oracle Internet Directory, the value is 1. Otherwise, it is 0.
	
	To determine if the Oracle Internet Directory infrastructure is configured for Oracle Email, run the following commands from the middle tier ORACLE_HOME.
	
	`rm $ORACLE_HOME/oes/log/exists.txt`
	`java -classpath`
	`$ORACLE_HOME/jlib/esinstall.jar:$ORACLE_`
	`HOME/jlib/repository.jar:$ORACLE_`
	`HOME/jlib/esldap.jar:$ORACLE_`
	`HOME/jlib/jndi.jar:$ORACLE_`
	`HOME/jlib/ldap.jar:$ORACLE_`
	`HOME/jlib/providerutil.jar`
	`oracle.mail.install.ESDSInstallQuery $ORACLE_HOME`
	`um_system.`
	
	After running this query, check if the `$ORACLE_HOME/oes/log/exists.txt` file exists. If it exists, the value of `oid_flag` should be 1, otherwise, the value is 0.
	
	■ On Windows:
	
	`del %ORACLE_HOME%\oes\log\exists.txt`
	`java -classpath`
	`%ORACLE_HOME%\jlib\esinstall.jar;%ORACLE_`
	`HOME%\jlib\repository.jar;%ORACLE_`
	`HOME%\jlib\esldap.jar;%ORACLE_`
	`HOME%\jlib\jndi.jar;%ORACLE_`
	`HOME%\jlib\ldap.jar;%ORACLE_`
	`HOME%\jlib\providerutil.jar`
	`oracle.mail.install.ESDSInstallQuery %ORACLE_HOME%`
	`um_system.`
	
	After running this query, check if the `%ORACLE_HOME%\oes\log\exists.txt` file `exists`. If the file exists, the `oid_flag` value should be 1, otherwise, it is 0

domain_name	The local domain name. For example, `acme.com`.

Note: The default `INSTALLATION_NAME` is `UM_SYSTEM`.

Installing the umbackend.tar File

> **Note:** If the mail store database was not installed from Oracle Collaboration Suite, the umbackend.tar file must be downloaded from the Oracle Technology Network website.

Perform the following steps to install the umbackend.tar file:

1. Copy the umbackend.tar file from $ORACLE_HOME/oes directory on the application server to the $ORACLE_HOME on the mail store database. If the mail store is located on another system, the umbackend.tar file for the database platform must be transferred from the application server to the mail store database as the database owner.
2. Untar the umbackend.tar file:

   ```
   tar xvf umbackend.tar
   ```

3. Run the following commands to run the installer:

   ```
   cd backend/Disk1
   ./runInstaller
   ```

4. Follow the screen prompts to complete the Oracle Email backend installation.

Installing the install_infra.sql File

Administrators must run <mailstore_database_oracle_home/oes/install/sql/install_infra.sql on the mailstore database machine after installing the umbackend.tar file.

Configuring the Oracle Email Middle Tier

Configuring the middle tier does the following:

- Configures middle tier with Oracle Internet Directory
- Configures middle tier with mailstore
- Creates Oracle Email server instances

To configure the Oracle Email middle tier servers, perform the following steps:

1. Run the umconfig.sh script located on the application server:
   - For UNIX, enter the following command:

```
$ORACLE_HOME/oes/bin/umconfig.sh
```

- For Windows, enter the following command:

```
%ORACLE_HOME%\oes\bin\umconfig.bat
```

The Unified Messaging Configuration screen is displayed.

2. Select **Middle Tier Configuration**.
3. Click **Next**. The Middle Tier Configuration screen is displayed.
4. Enter the mail store global database name. This is the name of the middle tier mail store database.

> **Note:** If the UMADMIN password and the domain name were specified during the mail store configuration, the next screen that displays is the Configuration Tool screen. Otherwise, you must specify the UMADMIN password and the domain name.

5. Click **Next**. The UMADMIN password screen is displayed.

> **Note:** UMADMIN is an administrator account created in the Oracle Internet Directory server during the application server installation of Oracle Email. It account owns specific Oracle Email entries in the directory. After installation, administrators should log in to the administration tool using the UMADMIN account and create an initial Oracle Email user. Thereafter, they can delegate system and domain administration responsibilities to other users.

6. Enter the UMADMIN password and confirm it. If a password is not entered, the default value welcome is stored in Oracle Internet Directory and the database as the UMADMIN password.
7. Click **Next**. The Create Unified Messaging Domain screen is displayed
8. Enter the the domain name to be created. This domain is used for users' e-mail addresses.
9. Click **Next**. The Configuration Tools screen is displayed and the middle tier configuration begins.

Once the middle tier configuration is complete, the End of Installation screen is displayed.

Log files for `umconfig.sh` and `umconfig.bat` are located in the following directories:

- Unix:

  ```
  $ORACLE_HOME/oes/log/
  ```

- Windows:

  ```
  %ORACLE_HOME%\oes\log\
  ```

Manually Configuring the Oracle Email Middle Tier

> **Note:** The manual steps described below are an alternative to running the `middletier.sh` script through the user interface.

Use the following command to run the Oracle Email Middle Tier configuration scripts with parameters:

- On Unix:

  ```
  cd $ORACLE_HOME/oes/bin
  install_middletier.sh true true UM_SYSTEM <ORACLE_HOME>
  <umadmin_passwd> <oid_flag> <global_db_name> <domain_name>
  ```

- On Windows:

  ```
  cd $ORACLE_HOME/oes/bin
  install_middletier.sh true true UM_SYSTEM <ORACLE_HOME>
  <umadmin_passwd> <oid_flag> <global_db_name> <domain_name>
  ```

Where:

Table 2–3

Parameter	Description
`umadmin_passwd`	`UMADMIN` is an administrator account created in the Oracle Internet Directory server during the application server installation of Oracle Email. It account owns specific Oracle Email entries in the directory. After installation, administrators should log in to the administration tool using the `UMADMIN` account and create an initial Oracle Email user. Thereafter, they can delegate system and domain administration responsibilities to other users. If a password is not entered, the default value `welcome` is stored in Oracle Internet Directory and the database as the `UMADMIN` password.

Table 2–3

Parameter	Description

oid_flag	f Oracle Email entries have been created in Oracle Internet Directory, the value is 1. Otherwise, it is 0.
	
	To determine if the Oracle Internet Directory infrastructure is configured for Oracle Email, run the following commands from the middle tier ORACLE_HOME.
	
	`rm $ORACLE_HOME/oes/log/exists.txt`
	`java -classpath`
	`$ORACLE_HOME/jlib/esinstall.jar:$ORACLE_`
	`HOME/jlib/repository.jar:$ORACLE_`
	`HOME/jlib/esldap.jar:$ORACLE_`
	`HOME/jlib/jndi.jar:$ORACLE_`
	`HOME/jlib/ldap.jar:$ORACLE_`
	`HOME/jlib/providerutil.jar`
	`oracle.mail.install.ESDSInstallQuery $ORACLE_HOME`
	`um_system.`
	
	After running this query, check if the `$ORACLE_HOME/oes/log/exists.txt` file exists. If it exists, the value of `oid_flag` should be 1, otherwise, the value is 0.
	
	▪ On Windows:
	
	`del %ORACLE_HOME%\oes\log\exists.txt`
	`java -classpath`
	`%ORACLE_HOME%\jlib\esinstall.jar;%ORACLE_`
	`HOME%\jlib\repository.jar;%ORACLE_`
	`HOME%\jlib\esldap.jar;%ORACLE_`
	`HOME%\jlib\jndi.jar;%ORACLE_`
	`HOME%\jlib\ldap.jar;%ORACLE_`
	`HOME%\jlib\providerutil.jar`
	`oracle.mail.install.ESDSInstallQuery %ORACLE_HOME%`
	`um_system.`
	
	After running this query, check if the `%ORACLE_HOME%\oes\log\exists.txt` file `exists`. If the file exists, the `oid_flag` value should be 1, otherwise, it is 0

global_db_name	The database global name. For example, `acmedb.foo.acme.com`
domain_name	The local domain name. For example, `acme.com`.

Using the Administration Tools

This section describes the different administration tools used to administer the Oracle Email system.

This section contains the following topics:

- Oracle Enterprise Manager
- Thin Client

Oracle Enterprise Manager

> **See Also:** *Oracle9i Application Server Administrator's Guide* for more information about Oracle Enterprise Manager

Oracle Enterprise Manager is a Web-based tool that enables administrators to perform some of the management tasks for the Oracle9*i* database and Oracle9*i* Application Server. The Oracle Enterprise Manager can be used to administer Oracle Email service processes. Through Oracle Enterprise Manager, administrators can perform the following tasks on Oracle Email system:

- Startup
- Shutdown
- Reinitialize
- Modify default parameters

To perform administration tasks for Oracle Email through Oracle Enterprise Manager, administrators must navigate to the following URL:

```
http://<machine name>:1810
```

Thin Client

> **See Also:** Chapter 3, "Administration and Provisioning" for more information on domain and user provisioning

Using the Oracle Email Thin Client, administrators can perform domain and user provisioning tasks. Through the Oracle Email Thin Client, administrators can do the following:

- Create and modify domain settings for users and distribution lists

- Create, delete, modify, and view e-mail, fax users, voice mail users, and distribution lists
- Add and delete members to and from distribution lists
- View all the distribution lists a specific user is on
- Create, delete, modify, and view server-side filters
- Create, delete, and modify lists

To perform administration tasks for Oracle Email through the Thin Client, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Starting Up, Shutting Down, and Reinitializing Oracle Email

This section explains how to start, stop, and reinitialize the Oracle Email system.

This section contains the following topics:

- Verifying and Starting the Listener for the Mail Store
- Verifying and Starting the Listener for the Middle Tier
- Starting the Oracle Email System
- Stopping the Oracle Email System
- Reinitializing the Oracle Email System
- Creating a Public User

> **See Also:** Chapter 4, "Servers and Processes", for more information on how to start up, shut down, and reinitialize individual processes

Verifying and Starting the Listener for the Mail Store

The Oracle Net Listener must be running on the mail store database so that the system can establish database connections from the Oracle Email system and clients.

To verify that the listener is running, enter the following:

- On Unix:

```
% lsnrctl status listener_es
```

- On Windows:

  From command prompt, enter the following command:

  ```
  lsnrctl status listener_es
  ```

If the computer returns a message that contains the line `no listener`, then the listener must be started.

To start the listener, enter the following:

- On Unix:

  ```
  % lsnrctl start listener_es
  ```

- On Windows:

  From command prompt, enter the following command:

  ```
  lsnrctl start listener_es
  ```

  From the Services Console:

  1. Select **Start->Settings->Control Panel->Services**.
  2. Select **listener_es**.
  3. Click **Start**.

Verifying and Starting the Listener for the Middle Tier

To verify that the listener is running, enter the following:

- On Unix:

  ```
  % lsnrctl status listener_es
  ```

- On Windows:

  From command prompt, enter the following command:

  ```
  lsnrctl status listener_es
  ```

If the computer returns a message that contains the line `no listener`, then the listener needs to be started.

To start the listener, perform the following steps:

- On Windows:

From command prompt, enter the following command:

```
lsnrctl start listener_es
```

- On Unix:

  1. Configure the listener with protocol addresses and other configuration parameters, using Oracle Net Configuration Assistant or Oracle Net Manager.

     See Also: Chapter 12, "Configuring and Administering the Listener," of the *Oracle9i Net Services Administrator's Guide*

  2. Log in as superuser (root) and set file ownership and access permissions for the listener executable (`tnslsnr`) and its dependent shared libraries so that these files can be modified only by the superuser. The `tnslsnr` is located in the `$ORACLE_HOME/bin` directory.
  3. Verify that the permissions of the individual directories found in the path names to these files, starting with the root directory, are modified in the same way.
  4. Start the listener as root.
  5. At the operating system prompt, enter `tnslsnr` with optional command line arguments.

     ```
     tnslsnr [listener_name] [-user user] [-group group]
     ```

     where:

Table 2–4

Name	Description
listener_name	Specifies the name of the listener, as `listener_es`. If omitted, the default name LISTENER is used.
-user user	Specifies the user whose privileges the listener uses when superuser (root) privileges are not needed. After performing the privileged operations, the listener gives up root privileges irreversibly.
listener_name	Specifies the name of the listener, as `listener_es`. If omitted, the default name LISTENER is used.

Starting the Oracle Email System

Starting an Oracle Email service starts all the processes comprising that service type, such as IMAP4 and SMTP.

Using Oracle Enterprise Manager, perform the following procedure to start an Oracle Email system:

1. Navigate to the Oracle9*i* Application Server home page
2. Select Oracle Email.
3. Click **Start**.

Stopping the Oracle Email System

Stopping an Oracle Email system sends a request to the operating system to shut down all of the Oracle Email processes. One reason an administrator would want to stop the Oracle Email system is to perform maintenance on the system, such as upgrading the server hardware or software. It is not possible for the processes to be running while certain kinds of upgrades are performed.

Using Oracle Enterprise Manager, perform the following procedure to stop an Oracle Email system:

1. Navigate to the Oracle9*i* Application Server home page.
2. Select Oracle Email.
3. Click **Stop**.

Reinitializing the Oracle Email System

Reinitializing an Oracle Email process informs the operating system to reload its operational settings from the Oracle Internet Directory server. The process does not stop running, which means that users continue to receive uninterrupted service. Whenever a Oracle Email process parameter is modified, it must be reinitialized to make the changes take effect.

Using Oracle Enterprise Manager, perform the following procedure to reinitialize an Oracle Email process:

1. Navigate to the Oracle9*i* Application Server home page
2. Select Oracle Email.
3. Click **Reinitialize**.

Creating a Public User

After configuring Oracle Email, administrators must create a public user account through delegated administration service (DAS). This public user corresponds with the initial user that will be created through the Thin Client administration tool.

> **See Also:** Oracle Internet Directory Administrator's Guide for more information on using DAS

Once the public user has been created, administrators can navigate to `http://<machine name>:<port number>/um/admin/UMAdminLogin.uix`, to create the initial domain and user.

> **See Also:** Chapter 3, "Administration and Provisioning" for more information on creating the initial domain and user.

3

Administration and Provisioning

This chapter discuses how to administer Oracle Email.

This chapter contains the following topics:

- Initial Administration Tasks
- Managing Oracle Email
- Scanning and Removing Viruses

Initial Administration Tasks

To perform initial administration tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port number>/um/admin/UMAdminLogin.uix
```

Once administrators have logged in, they can perform the following tasks:

- Create the initial domain
- Create the initial user
- Change the administrator password

After performing initial administration tasks, Oracle Corporation disabling the initial administration pages to avoid inadvertent changes to the system.

Creating the Initial Domain

> **Note:** This task is to be performed only if domain creation failed during the configuration of Oracle Email.

Perform the following steps to create the initial domain:

1. Navigate to the Administration page.
2. Enter the `UMADMIN` password. This is the password designated during configuration.
3. Select **Domain** > **Domain Management** > **Create Domain**.
4. Select the parent domain from the drop down list.
5. Enter the domain name.
6. Click **Create Domain**.

Creating the Initial User

Note: Oracle Corporation that the initial user created be an Oracle Email system administrator.

Perform the following steps to create the initial user:

1. Navigate to the Administration page.
2. Select **User** > **User Management** > **Create User**.
3. Enter the user's ID.
4. Select the domain to which the user belongs.
5. Select the public user domain.
6. Select the mail store from the drop down list.
7. Enter the quota for the user
8. Select the user's role from the drop down list.
9. Click **Create**.

Changing Administrator Password

Perform the following steps to change the administrator password:

1. Navigate to the Administration Page.
2. Select **Preferences**> **Change Password**.
3. Enter the following information in the corresponding fields:
   - Old Password
   - New Password
   - Confirm Password
4. Click **Submit**.

Managing Oracle Email

Using the Thin Client, administrators can perform domain, user, list, and alias management tasks by clicking on the appropriate tab.

Under the Overview tab, you can view what components are installed on the different middle tier hosts. To administer these components, click on the host links and you will be redirected to the Oracle Enterprise Manager.

To perform administration tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Managing Domains

Using the Thin Client, administrators can perform domain management tasks, such as create domains, modify domain settings for users, and modify domain settings for lists.

To perform administration tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Creating Domains

Perform the following steps to create domains:

1. Using the Thin Client, navigate to the Administration page.
2. Select **Domain** > **Create Domain**.
3. Select the installation name.
4. Select the parent domain from the drop down list.
5. Enter the new domain name in the corresponding field.
6. Click **Submit**.

Modifying Domain Settings for Users

> **Note:** When the preferences for a domain are modified using the Thin Client administration tool, only the new entries created after the modifications contain the new set of preferences. For example, if the administrator changes the default mail quota of mail users of the `oracle.com` domain to 60MB, the new users created in the `oracle.com` domain have the new 60MB quota. The existing users in `oracle.com` retain the old mail quota.

Perform the following steps to modify domain preferences for users:

1. Using the Thin Client, navigate to the Administration page.
2. Select **Domain** > **Domain Settings for Users**.
3. Select the installation name from the drop down list.
4. Select the domain you want to modify.
5. Click **Submit**.
6. Modify the preferences you want to change.

The following is a list of user domain parameters:

> **Note:** These parameter is present at the domain level preferences for users. The values set here are inherited by all new users in the domain during user creation time.

Mail Store

This parameter stores the DN of the mail store the user belongs to. The mail store for a user is the database that contains mail user information.

E-mail Quota

This parameter stores the e-mail quota of a mail user in megabytes. This parameter is present at the domain level preferences for users.

Voice Quota

This parameter stores the quota for the voice mail user in megabytes. This parameter is present at the domain level preferences for users.

Number of Email Display (Web Mail)
This parameter displays the number of message headers displayed on a single page of the Thin Client.

Mail User Index Type
This parameter is associated with individual users and is present at the user level in Oracle Internet Directory, enabling OracleText configuration on a per user basis.

Domain Control ACI
This parameter specifies the domains a user can access.

Document Binary
This parameter is associated with individual users and is present at the user level in Oracle Internet Directory, enabling OracleText configuration on a per user basis. This parameter enables binary attachment search for users

Allow External Access (Web Mail)
This is an end user parameter that enables account access from outside a firewall. This preference is only used when the client has been configured to check for this preference before enabling users to connect. Typically, the WebMail client is configured in this manner when it is facing outside the firewall. This configuration can be used in conjunction with another WebMail client that is inside the firewall and not configured to check for this preference so that a user can log in from within the firewall, enable, or disable the preference, and then subsequently log in or prevent log in from outside the firewall. For security purposes, this preference is set to disabled by default.

View in New Window (Web Mail)
This is an end user parameter that specifies whether the message view display should occur in a new browser window. When disabled, the application displays message views in the same window.

Display All Headers (Web Mail)

This is an end user parameter that specifies whether a message view displays all the headers that a message contains. If disabled, only the common headers are shown. The commom headers are:

- From
- To
- CC
- Subject
- Date
- Priority

7. Click **Submit**.

Modifying Domain Settings for Lists

Perform the following steps to modify domain preferences for lists:

1. Using the Thin Client, navigate to the Administration page.
2. Select **Domain** > **Domain Settings for Lists**.
3. Select the installation from the drop down list.
4. Select the domain you want to modify.
5. Click **Submit**.
6. Modify the preferences you want to change.

The following is a list of parameters for list domains:

Maximum Message Size

This parameter indicates the maximum size (in bytes) of a message delivered to the list. Any message larger than this size is rejected.

Group Admin Mail ID

This parameter is the mail ID to which all commands to the list should be sent. The default is `<listname>-admin@domain`.

Group Type

This parameter specifies the type of the list. Possible values are announcement, discussion, edited, or moderated.

Group Subscription Type
This parameter denotes the type of subscription control on this list. Valid values are open, restricted, or closed.

Group Topic
This parameter is a single line phrase describing the topic of discussions on this list.

Group Information Text
This parameter is a multi-line field that owners can use to include descriptive text about the list.

Group Auto Reconfirm
This parameter is set to true if the owner wants all subscription requests to be reconfirmed with the user.

Group Invite Text
This parameter is a multi-line text sent in e-mail to users invited by a list owner to join their list.

Group Post Type
This parameter is defines the level of control a list owner wants on who can post messages to their list. Valid values are open or subscriber.

Group Editor's List
This parameter contains the list of users (mail IDs) who are the editors of the list.

Group Moderator's List
This parameter contains the list of users (mail IDs) who are the moderators of the list.

Group Merge Tag
This parameter enables a list owner to support mail merge or scheduled mail delivery, this attribute contains the tag that is used for specifying mail merge and scheduler tags.

7. Click **Submit**.

Managing Users

Using the Thin Client, administrators can perform user management tasks, such as add, remove, and modify e-mail users.

To perform administration tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Through the Oracle Email Overview page, you can view what components are installed on the different middle tier hosts. To administer these components, click on the host links and you will be redirected to the Oracle Enterprise Manager.

Adding E-mail Users

> **Note:** A base user must exist in Oracle Internet Directory before they can be added as an e-mail user. If a base user does not exist, a message displays with a link directing you the Oracle Internet Directory Delegated Administration Service page.

Perform the following steps to add e-mail users:

1. Using the Thin Client, navigate to the Administration page.
2. Select **User** > **E-mail User Management** > **Add User**.
3. Select the domain from the drop down list.
4. Enter the user ID in the corresponding field.
5. Enter the base user domain.
6. Select the mail store from the drop down list.
7. Enter the quota value in the corresponding field.
8. Select the role from the drop down list.
9. Click **Add**.

Modifying E-mail User Parameters

Perform the following steps to modify e-mail user parameters:

1. Using the Thin Client, navigate to the Administration page.
2. Select **User** > **E-mail User Management** > **Modify User**.

3. Enter the user ID in the **Search Criteria** field.
4. Select the user's domain from the drop down list.
5. Click **Go**.
6. Modify the parameters you want to change.
7. Click **Modify**.

E-mail User Parameters

The following is a list of e-mail user parameters:

User ID

This parameter specifies the user ID.

Mail Store

This parameter stores the DN of the mail store the user belongs to. The mail store for a user is the database that contains mail user information.

E-mail Quota

This parameter stores the e-mail quota of a mail user in megabytes.

Additional Voice Quota

This parameter stores the additional quota for the voice mail user

User State

This parameter defines if the user is active or inactive. If the parameter is set to active, the user will be able to receive and send e-mail. If the parameter is set to inactive, the user will not be able to receive and send e-mail. If this parameter is not set, it defaults to active.

Auto Reply Text

This parameter stores auto reply text to be used by the user for the auto reply feature.

Auto Reply Expiration

This parameter stores the auto reply expiration date in the following format:

`MMDDYY:HH24:MI:SS:TZH:TZM`

Forward E-mail Address

This parameter stores the e-mail addresses for the auto forward feature.

Document Binary

This parameter is associated with individual users and is present at the user level in Oracle Internet Directory, enabling OracleText configuration on a per user basis. This parameter is present at the domain level preferences for Users. The values set here are inherited by all new users in the domain during user creation time.

Role

This parameter defines if the user is a regular user, a system administrator, or a domain administrator.

External Web Access

This parameter enables the user to have e-mail access outside the firewall.

Removing E-mail Users

> **Note:** When a mail user is removed, any shared folders and public shared folders owned by that user are also deleted

Perform the following steps to remove e-mail users:

1. Using the Thin Client, navigate to the Administration page.
2. Select **User** > **E-mail User Management** > **Remove User**.
3. Enter the search criteria
4. Select the domain to which the user belongs to.
5. Click **Go**.
6. Select the user you want to delete.
7. Click **Remove**.

Managing Lists

Using the Thin Client, administrators can perform list management, such as create lists, modify list properties, delete lists, show lists, add and delete list members, show list members, and show all the lists a member is on.

To perform list management tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Creating Lists

Perform the following steps to create a list:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Distribution List Management** > **Create a new list**.
3. Select the domain from the drop down list.
4. Select **SMTP** or **List Server** from the **Distribution List Type** drop down list. The distribution list type defines the mailing list type.
5. Click **Go**.
6. Enter the following information in the corresponding fields and click **Create**.
   - Distribution List Name
   - Owner
   - Maximum Message Size
   - Group Topic
   - Group Invite Text
   - Group Editor's List
   - Group Moderator's List
   - Group Merge Tag
   - Group Auto Reconfirm
   - Group Type
   - Group Subscription Type
   - Group Post Type

   See Also: "List Server Parameters" for parameter definitions

Modifying Lists Properties

Perform the following steps to edit list properties:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Distribution List Management** > **Edit list properties**.
3. Enter the list name, or enter * to display all available lists.
4. Select the domain of the list from the drop down list.

5. Select the list you want to make changes to.
6. Edit the properties you want to change.
7. Click **Modify**.

List Server Parameters

The following is a list of list server properties:

Distribution List Name

This parameter specifies the name of the distribution list.

Owner

This parameter specifies the person who owns or is responsible for the list.

Maximum Message Size

This parameter specifies indicates the maximum size, in bytes, of a message delivered to a list. Any message larger than this size is rejected.

Group Topic

This parameter is a single line phrase describing the topic of discussions on this list.

Group Invite Text

This parameter is a multi-line text that is sent in a mail to users invited by a list owner to join their list.

Group Editor's List

This parameter is contains the list of users (mail IDs) who are the editors of the list.

Group Moderator's List

This parameter is contains the list of users (mail IDs) who are the moderators of the list.

Group Merge Tag

This parameter enables a list owner to support mail merge or scheduled mail delivery, this attribute contains the tag used for specifying mail merge and scheduler tags.

Group Auto Reconfirm

This parameter is set to true if the owner wants all subscription requests to be reconfirmed with the user.

Group Type

This parameter specifies the type of the list. Possible values are:

- Announcement: Messages can only be sent to the list. Replies to the list are not accepted.
- Discussion: Messages can be sent and replied to on the list.
- Edited: Only certain people known as editors can post mails to the list. Messages from others are rejected.
- Moderated: Every mail posted to the list are sent to one or more persons known as moderators. The mail is delivered to the list only if at least one of the moderators approves the mail.

Group Subscription Type

This parameter denotes the type of subscription control on this list. Valid values are:

- Open: Anybody can subscribe to a list. No controls are imposed on who can subscribe
- Restricted: All subscriptions to the list need approval by the owner of the list.
- Closed: Users can subscribe to a list only if they are invited to join the list by the owner. All other subscription requests to the lists are automatically turned down.

Group Post Type

This parameter dines the level of control a list owner wants on who can post messages to their list. Valid values are open or subscriber.

- Open: Anybody can post a mail to the list.
- Subscriber: Only a subscriber to a list can post a mail to the list. Non-subscribers that send mails to the list are rejected.

Deleting Lists

Perform the following steps to delete a list:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Distribution List Management** > **Delete list(s)**.
3. Enter the list name, or enter * to display all available lists.
4. Select the domain of the list from the drop down list.
5. Click **Go**.

6. Select the list you want to delete.
7. Click **Delete**.

Adding and Deleting List Members

Perform the following to steps to add or delete list members:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Membership Management** > **Add/Remove Members**.
3. Enter the list name, or enter * to display all available lists.
4. Select the domain of the list from the drop down list.
5. Click **Go**.
6. Select the list for which you want to add members.
7. Enter or remove information in the following fields:
   - Members (user) - Users on this system that are members of this list
   - Members (list) - Lists that are members of this sub-lists
   - Members (alias) - Aliases that are members of this list
   - Members (foreign) - Users foreign to this system who are members of this list
8. Click **Modify**.

Showing Lists

Perform the following steps to view all the lists belonging to a particular domain:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Distribution List Management** > **Show list(s)**.
3. Enter the list name, or enter * to display all available lists.
4. Select the domain of the list from the drop down list.
5. Click **Go**.
6. Select the list you want to view.

Showing Members

Perform the following steps to show list members:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Membership Management** > **Show Members**.
3. Enter the list name, or enter * to display all available lists.
4. Select the domain of the list from the drop down list.
5. Click **Go**.
6. Select the list for which you want to view members.

Showing All the List a User is On

Perform the following steps to show all the lists a user is on:

1. Using the Thin Client, navigate to the Administration page.
2. Select **List** > **Miscellaneous Functions** > **Show all memberships of a user**.
3. Enter the user's name.
4. Select the user's domain from the drop down list.
5. Click **Show Memberships**.

Managing Aliases

Using the Thin Client, administrators can perform alias management tasks, such as create, modify, and delete aliases.

To perform administration tasks for Oracle Email, administrators must navigate to the following URL:

```
http://<machine name>:<port>/um/traffic_cop
```

Creating a New Alias

Perform the following steps to create a new alias.

1. Using the Thin Client, navigate to the Administration page.
2. Select **Alias** > **Alias Management** > **Create a new alias**.
3. Select the domain from the drop down list.
4. Click **Go**.
5. Enter the alias name.
6. Enter the alias target.

7. Enter the description.
8. Click **Create**.

Editing Alias Properties

Perform the following steps to edit alias properties:

1. Using the Thin Client, navigate to the Administration page.
2. Select **Alias** > **Alias Management** > **Edit alias properties**.
3. Enter the search criteria.
4. Select the domain from the drop down list.
5. Click **Go**.
6. Select the alias you want to modify.
7. Click **Modify**.
8. Modify the properties you want to change.
9. Click **Modify**.

Deleting Aliases

Perform the following steps to delete aliases:

1. Using the Thin Client, navigate to the Administration page.
2. Select **Alias** > **Alias Management** >**Delete alias(es)**.
3. Enter the search criteria.
4. Select the domain from the drop down list.
5. Click **Go**.
6. Select the alias you want to delete.
7. Click **Delete**.

Scanning and Removing Viruses

E-mail viruses typically have the form of an executable program as an e-mail attachment. The program gets executed on the client machine when the attachment is opened by an unsuspecting user, causing various forms of damage to the computer or the network. Oracle Email server contains a PL/SQL utility package MAIL_AV that can scan and remove email viruses. To use this package, one simply writes a SQLPLUS script that uses this package or execute procedures in this package directly from SQLPLUS.

MAIL_AV package allows users to identify virus messages using a variety of search criteria, such as subject, date, and size. Once messages with an attached virus are identified, the messages are moved to a designated folder away from their recipients. An administrator can manually examine the messages in the designated folder and remove them using any mail client. In cases where messages are wrongly identified as a virus, the MAIL_AV package provides functions to restore the message back to the original recipient's folder.

Because MAIL_AV has the ability to remove messages from regular users, it is considered a privileged package. Only sessions logged on as es_mail database user can execute this package.

Usage Examples

The following are summaries and usage examples for the procedures in the MAIL_AV package:

Quarantine

The quarantine procedure has the following syntax:

```
PROCEDURE quarantine (p_endday IN DATE,
                      p_dayrange IN NUMBER,
                      p_attribute IN NUMBER,
                      p_pattern IN VARCHAR2,
                      p_folder IN VARCHAR2);
```

The quarantine procedure identifies virus messages using a given pattern and moves them to a designated folder. The caller of the procedure must have write authorization to the folder. Authentication is done by using MAIL_SESSION package.

> **See Also:** *Oracle Email Application Developer's Guide* for more information

Parameters `p_endday` and `p_dayrange` can used to narrow down the virus search to within certain time frame. Parameter `p_attribute` takes one of the following three values:

```
MAIL_AV.ATTR_SUBJECT
MAIL_AV.ATTR_ATTACHMENT
MAIL_AV.ATTR_SENDER
```

Parameter `p_pattern` is the identifying string for the virus. Parameter `p_folder` is the designated folder name to which virus-infected messages are moved.

The following example logs in as user `SYSADMIN`, and scans the whole mail server for messages with an attachment name containing `.exe` within the last seven days, and moves them to the `/infected` folder.

```
declare
  sessionid number;
begin
  mail_session.login('sysadmin', <password>, <ldaphost>, sessionid);
    mail_av.quarantine(sysdate, 7, mail_av.attr_attachment, '.exe',
'/infected');
end;
/
```

Quarantine II

The Quarantine procedure can take on the following format enabling IMAP style search criteria:

```
PROCEDURE quarantine (p_criteria IN VARCHAR2,
                      p_folder IN VARCHAR2);
```

This quarantine procedure form identifies virus messages using an IMAP style search criteria for enhanced searching. All IMAP search commands are supported. The advantage of using this procedure not only includes the expanded list of search item, but also the ability to combine search criteria using logical operations such as "and" or "or."

> **See Also:** Internet RFC 2060: Internet Message Access Protocol, version 4, rev 1, for more information on IMAP search commands

Use the new form of quarantine procedure, the following script identifies and moves messages with subject "snow white" and from acme.com, that's also sent since Jan 2002:

```
declare
    sessionid number;
begin
    mail_session.login('sysadmin', <password>, <ldaphost>, sessionid);
    mail_av.quarantine('SINCE 01-Jan-2002 SUBJECT "snow white" SENDER
"aol.com"', '/infected');
end;
/
```

Restore

There are two procedures to restore messages already quarantined back to their original folders:

```
PROCEDURE restore (p_messageid IN NUMBER);
PROCEDURE restoreall;
```

The restore procedure takes a given message ID and restore it back to its original folder. If the message ID does not exist, the procedure does nothing. The `restoreall` procedure restores all messages quarantined regardless which designated folders are used to store the messages. These procedures are useful when a message is wrongly identified as a virus message and must be restored back to its recipients.

4

Servers and Processes

This chapter discusses the different servers and processes of the Oracle Email system.

This chapter contains the following topics:

- Mail Store
- IMAP4 and POP3 Processes
- SMTP Process
- Housekeeping Process
- List Server Process
- Managing Server Processes
- Server Parameters
- Server Log Files
- SSL Setup
- Thin Client

Mail Store

The Oracle Email mail store is the location where messages and folder information are stored. If a message is destined for many accounts on that mail store, only one copy of the message is stored and links to the message are sent to all recipients. Folders can be private, shared, or public. A single mail store can store mail for one domain or several different domains. If you have an extremely large domain, it is possible to have multiple mail stores support a single domain.

Applications can make direct PL/SQL procedure calls to act upon messages similar to the voice mail client use standard protocol servers or JMA+ libraries. Calls made to the stored procedures are executed in the mail store.

The mail store loads stored procedures that enable the following features:

- High-Level Interface
- Process Close to the Data
- Extensible Store

High-Level Interface

The Oracle Email mail store has a comprehensive and high-level PL/SQL interface. Clients make high-level calls to the mail store.

Because the logic of the PL/SQL procedures take place inside the Oracle9*i* database mail store, close to the actual data, mail transactions are greatly simplified and more reliable. Oracle Email, by definition, enables multiple clients and client types to access the same mail store. By having all clients enter through the same interface and perform processing in the mail store, all clients achieve the same goals with the same behavior.

For example, a typical Oracle Email installation enables a user to listen to voice mail messages from any one of three clients:

- Telephone handset client
- Standard IMAP4 mail client
- Web Client

In a typical telephone situation, if a voice mail message has not been heard, the Message Waiting Indicator (MWI) light on the user's telephone is on. If all voice mail messages have already been listened to, the light is off. Because the MWI logic is placed inside the mail store and not in the Web client, standard client, or

telephone handset, the solution consistently and appropriately turns the light off no matter which client was used to listen to the voice mail messages.

Process Close to the Data

All of the Oracle Email scalable protocol servers and access servers perform tasks within the stored PL/SQL procedures running within the mail store. One can query an entire mail store with a simple request and the mail store efficiently carries it out. This comprehensive high-level interface enables effective support for system-wide rules, individual rules, server side filters, and spam control, for a very large and dynamic mail store without serious degradation. It also affords Oracle Email the ability to query and update the mail store efficiently.

Extensible Store

The mail store can be extended by or integrated with other Oracle product capabilities. Filters can be applied that automatically act on all mail messages. The filters can either perform common mail tasks on a mail message, or they can pass the mail message to an external program.

For example, an Oracle Email e-mail server can be set up to filter all mail messages sent to the abstract Helpdesk mail account of a company. Each message is broken down into its main body and any attachments. Each part is then passed, individually, through the Oracle Text engine. The results can then be routed to the appropriate support representative, who can choose from a list of possible responses to the specific request or create and log a new response.

> **See Also:** Chapter 2, "Getting Started" for more information on how to install a mail store

Modifying Mail Store Default Parameters

Using Oracle Enterprise Manager, perform the following steps to modify mail store default parameters:

1. Navigate to the Oracle Email Service Targets page.
2. Select a server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the mail store for which you want to make changes.
4. Modify the parameters you want to change.

5. Click **Apply**.

Modifying Mail Store Connection Parameters

Using Oracle Enterprise Manager, perform the following steps to modify mail store default parameters:

1. Navigate to the Oracle Email Service Targets page.
2. Select a server type. For example: IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the mail store for which you want to make changes.
4. Modify the parameters you want to change.
5. Click Apply.

Mail Store Parameters

The following is a list of mail store parameters.

Timeout

Connections idle for more than this time value are terminated, to maintain an optimum number of open connections

Increment

Oracle Email can increase the number of connections to be opened to the database bye this number, if the current number of connections are less than maximum. Valid values are 0 and above.

Minimum

Specifies the minimum number of connections in the connection pool. Valid values are 0 and above.

Maximum

Specifies the maximum number of connections that can be opened to the database. Once this value is reached, no more connections are opened. Valid values are 1 and above.

Database Host

The machine on which the database is located.

Mail Store Port

Port number the database listener is listening on.

Default Value: 1521

Database SID

System identifier of the mail store database.

Default Value: none

User Name

The user name used to log into the mail store database as the mail user.

Default Value: es_mail

Password

The password for the mail store user name.

Default Value: es

Oracle Text Settings

The integration of Oracle Text and Oracle Email extends the e-mail server functionalities. This enables text search in e-mails, e-mail theme generation, and e-mail formatting functions such as, highlight and markup.

Oracle Text is installed by default when Oracle Email is installed. However, if database user `ctxsys` is not present at the time of installation, the Oracle Text installation will fail.

There are two user level Oracle Internet Directory parameters associated with the configuration of Oracle Text:

- `orclMailUserIndexType`: This parameter enables text search capability for users. When this parameter is set to 1 or 2 for a user, the user can use any supported client to perform server side search on message bodies.
- `orclMailIsDocBinary`: This parameter controls whether only the text or the complete contents of e-mail messages should be used for e-mail theme generation and e-mail formatting functions

The following table describes the parameter values for `orclMailUserIndexType` and `orclMailIsDocBinary`

Oracle Internet Directory Parameter Name & Associations	Type	Possible Values
`orclMailUserIndexType/ text search`	number	0:do not index incoming e-mail (default) 1:for incoming e-mails, index text contents only 2:for incoming e-mails, index both text and binary contents
`orclMailIsDocBinary/ document service`	Boolean	false: when requesting document service, process only text contents (default) true: when requesting document service, index both text and binary contents

The parameters are independent of each other, and can be configured at the user level. The parameters are present at the domain level preferences for viewers. The values set here are inherited by all new users in the domain during user creation time. To view or modify parameter values, use Oracle Enterprise Manager.

Oracle Text provides both a Java Software Developer's Kit (SDK) and a PL/SQL SDK for application integration. Applications can interface with the SDKs to use or extend Oracle Text functionalities.

Quota

There are two quota values that can be set for a user: `user-quota` and `voice-quota`. All e-mails and voice mails are delivered to the user as long as the user is under `user-quota`. When `user-quota` is reached, all e-mails are held in the system and are not delivered to the user. However, voice mail delivery continues as long as the user's total usage is under `user-quota` plus `voice-quota` value. For example, if the `user-quota` is 50MB and the voice-quota is 20MB, e-mail delivery stops after the user's usage is 50MB but the voice mail delivery continue until the user reaches 70MB.

When the user cleans up the account and the usage is under the `user-quota` plus `voice-quota` value, voice mail delivery starts again. When the usage is under `user-quota`, e-mail delivery starts again. It is important to note that both e-mails and voice mails contribute to the user-quota calculations. When the usage reaches the `user-quota`, it means that the sum of e-mails and voice mails is equal to the

`user-quota` value. Voice-quota is an additional buffer provided to users so that voice mail delivery is not affected when users reach their quota.

In addition to stopped mail delivery, users cannot save new messages in the server folders when `user-quota` is reached. For example, saving a copy of outgoing messages to the sent folder is not allowed. The IMAP server informs the client that the user is over quota and is trying to save new mail.

Server Side Filters

Server side filters enable users to create mailbox filters on the server. Users can use the Thin Client to create rule-based actions, such as message foldering, vacation reply, spam filter, and wireless notification. Because the filters are created on the server, the actions are carried out whether the user is online or not.

The filters engine enables customized auto-actions on e-mail messages being processed, based on a variety of customizable events and conditions.

The filters engine is an automated layer between server processes and message objects. It provides scripted operations done on behalf of the server processes to the message objects. The filters engine maintains its script database in the mail store.

IMAP4 and POP3 Processes

Internet Message Access Protocol (IMAP) and Post Office Protocol (POP), are protocols for retrieving e-mail messages.

The POP3 protocol provides mail manipulation services for certain types of smaller nodes on the Internet where it is often impractical to maintain a message transport system, or in situations where it is undesirable to keep an Internet connection open for long periods of time. Messages are temporarily stored on the server until they are downloaded to a client machine.

The IMAP4 protocol provides a set of functionality to manipulate mail messages as well as mail folders, which are stored on the server. It provides primitives to allow optimization of online performance, especially when dealing with large MIME messages. The IMAP4 protocol also provides the capability for an off-line client to re-synchronize with the server.

Standards Supported

The IMAP4 and POP3 servers support the IMAP4 and POP3 protocols as described in the RFC-2060 and RFC-1939 respectively.

See Also: www.ietf.org for more information on RFCs

The IMAP4 Server also supports the following IMAP4 extensions:

- RFC-2087 IMAP4 QUOTA extension
- RFC-2088 IMAP4 non-synchronous literals
- RFC-2859 IMAP4 UIDPLUS extension
- RFC-2177 IMAP4 IDLE command
- Draft IMAP4 - SORT Extension

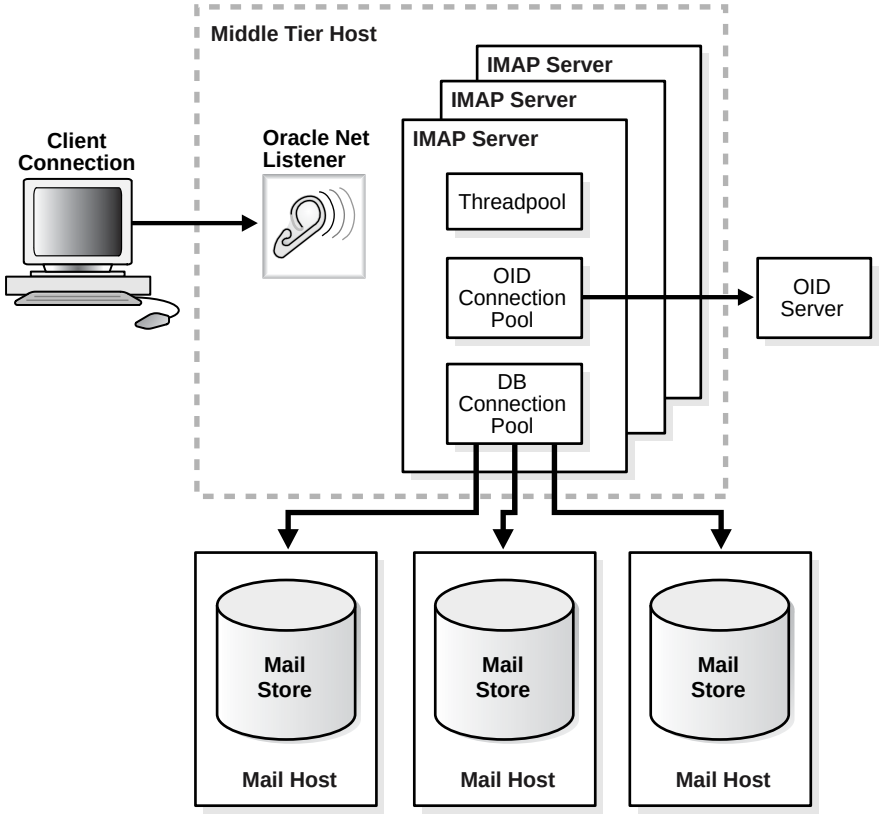
Process Architecture

By using the scalable protocol server programming framework, the IMAP4 and POP3 Servers obtain the benefits of multithreading, database connection sharing, and load balancing, which enables the servers to support thousands of concurrent user connections.

The IMAP4 and POP3 support a large number of client connections, using a very small amount of system resources for each client connection. It maintains a pool of worker threads, which handles the work for the clients and also a pool of database connections shared across client connections. When a client request comes in, a thread from the pool of worker threads is assigned to it. The worker threads read the command from the client, obtain a database connection from the database pool, and perform the operation. After the database connection is released back to the pool, the thread returns to the worker thread pool. There can be multiple mail stores in the system. The IMAP4 and POP3 servers can be set up to create database connection pools to more than one mail store. Administrators should check the IMAP4 and POP3 server parameters on how to control the size of the pools.

Many operating systems have limitations on the number of file descriptors and sockets a single process can open, requiring more than one instance of an IMAP4 or POP3 server to be running. If there is more than one instance running on the server, the listener distributes the load between them. Administrators should verify that the operating system parameter controlling the file descriptors for a process is correctly set.

Figure 4–1 Process Architecture



Process Log Writing

The IMAP4 logs are written in `$ORACLE_HOME/oes/log/<install_name>/imap/<pid>/<pid>.log`. Different log levels determine the amount of information produced by from the servers. There are five different log levels. Their names and values from least to most are:

Error Level	Numeric Value
Internal Errors	1

Errors	6
Warnings	11
Notification	16
Trace	21
Dump	26

SMTP Process

Simple Mail Transfer Protocol (SMTP), is a protocol for sending e-mail messages between servers. Most e-mail systems that send mail over the Internet use SMTP to send messages from one server to another; the messages can be retrieved with an e-mail client using either POP or IMAP. Mail clients generally use SMTP to send messages to a mail server.

The SMTP server handles all inbound and outbound mail. It implements the SMTP protocol and interacts with DNS and Oracle Internet Directory servers to get information about hosts and users.

Standards Supported

The following is a list of the SMTP server -supported RFCs.

See Also: www.ietf.org for more information on RFCs.

SMTP Base Protocol

RFC 821 - Simple Mail Transfer Protocol (SMTP)

RFC 1123 - Requirements for Internet hosts - application and support

SMTP Extensions

RFC 1869 - SMTP Service Extensions

RFC 1652 - SMTP Service Extension for 8bit-MIME transport

RFC 1870 - SMTP Service Extension for Message Size Declaration

RFC 1891 - SMTP Service Extension for Delivery Status Notifications

RFC 1894 - An Extensible Message Format for Delivery Status Notifications (DSNs)

RFC 2034 - SMTP Service Extension for Returning Enhanced Error Codes

Mail Format Standards

RFC 822 - Standard for the format of ARPA Internet text messages

RFC 2045 - MIME Part 1: Format of Internet Message Bodies

RFC 2046 - MIME Part 2: Media Types

RFC 2047 - MIME Part 3: Message Header Extensions for Non-ASCII Text

RFC 2048 - MIME Part 4: Registration Procedures

RFC 2049 - MIME Part 5: Conformance Criteria and Examples

Various Configurations

Oracle Email has a flexible architecture that enables users to set up a single, double, or multiple tier configuration that is appropriate to a site's needs. The following are examples of various configurations.

Figure 4–2 **Simple Single Node Setup**

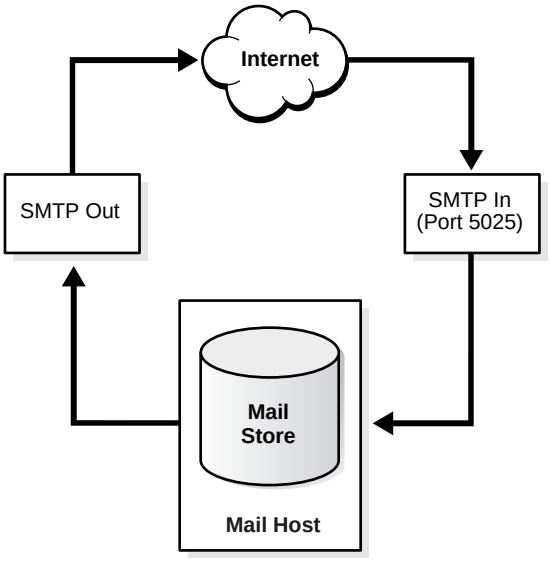


Figure 4-2 is the simplest setup where there is only one mail store and SMTP server running on the same host. This configuration can be used for supporting a small numbers of users.

Figure 4–3 Two Tier Setup

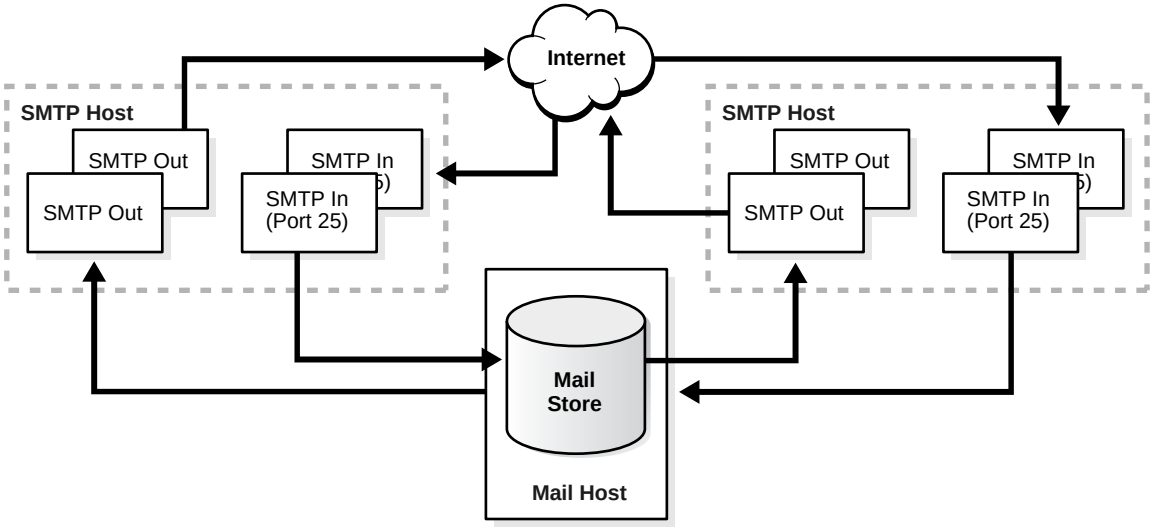


Figure 4-3 is a single mail store setup, with the processes divided into two tiers: the backend running the database and the middle tier running SMTP and other protocol servers. It provides fault tolerance and the flexibility to run multiple SMTP servers and distribute load across them. Load balancing and fault tolerance can be achieved by running the servers behind a network director or by having multiple MX records for the domain.

Figure 4–4 Multiple Mail Stores Setup

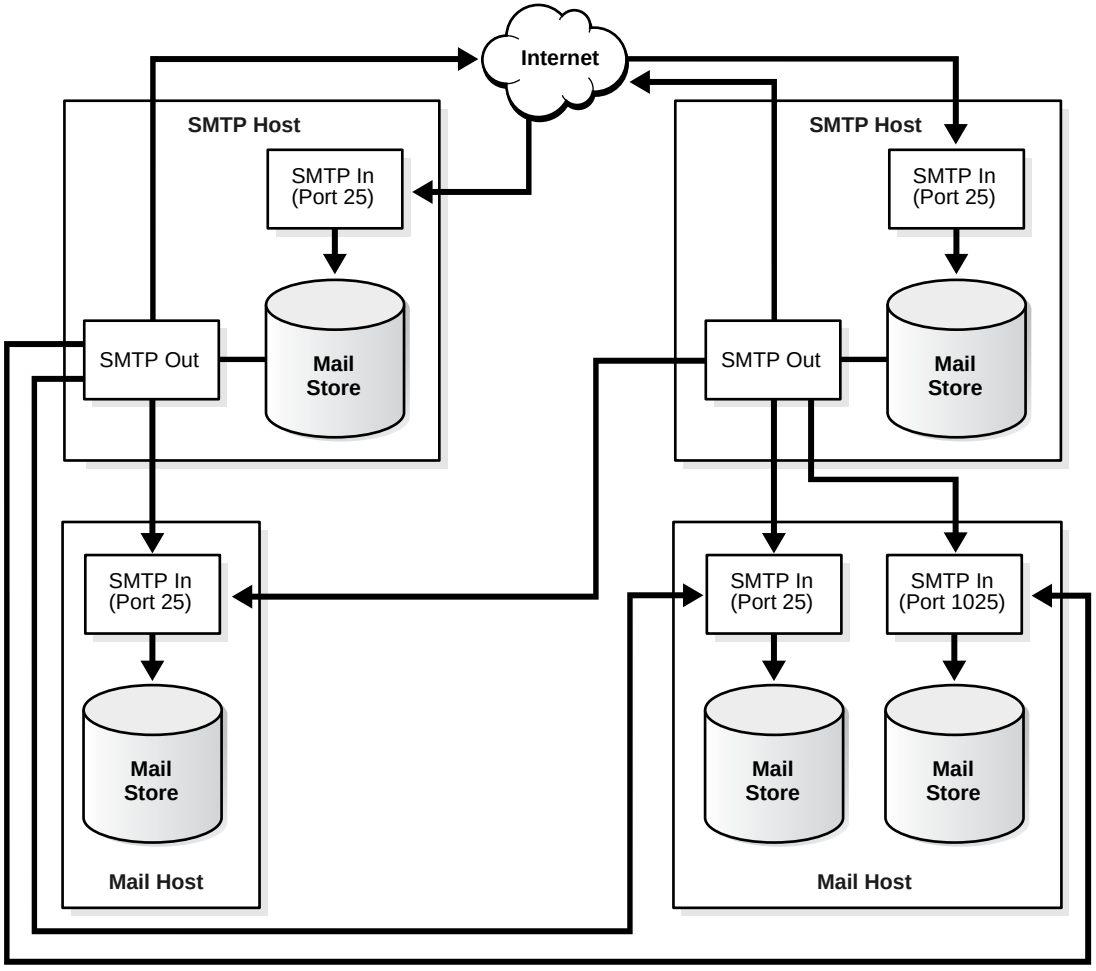


Figure 4-4 shows two kinds of configurations: multiple mail stores on different hosts and multiple mail stores on the same host. Each SMTP server serves only one mail host and each mail store must have a SMTP server. The mail stores on the SMTP hosts are used as SMTP queues and do not contain users.

Message Flow

The SMTP message transfer agent engine is responsible for the incoming SMTP connection. It receives incoming messages and performs address lookup and rewriting. The Oracle Internet Directory server is queried to find and authenticate the addresses. Addresses are rewritten based on the rewriting rules. Anti-spamming rules is applied. If all of the above are successful, the SMTP message transfer agent accepts the message and inserts it into the corresponding queue based on the destination address.

If the recipient is an outside user, the message is stored in the relay queue awaiting further processing. If the recipient is local, the message is stored in the local delivery queue. To determine if an address is local, the parameter `SMTPlocaldomains` is used. This parameter contains the list of domains that are considered local. The local delivery module picks up the message later, applies the rules, if any, and delivers it to the user's inbox.

If administrators do not want to process the messages immediately for performance reasons, messages can be stored in the submission queue and marked as submitted or unprocessed. This is controlled by the server parameters. Messages created by the SDK applications are also marked as submitted in this situation.

The messages in the submission queue are picked up by the outbound queue processor. For relay messages, the outbound processor queries the DNS server, applies the rules against them, and sends them out using SMTP. The submitted messages first go through the Address Rewriting and DNS resolution module. After which, the outbound queue processor sends them to the local delivery queue or to the Internet, depending on whether the messages' recipients are local or not.

During the address resolution phase, if the server determines that the message should be sent to a distribution list handled by the list server, it places the message in the list server queue. The list server then picks up this message, expands the distribution list and delivers the message.

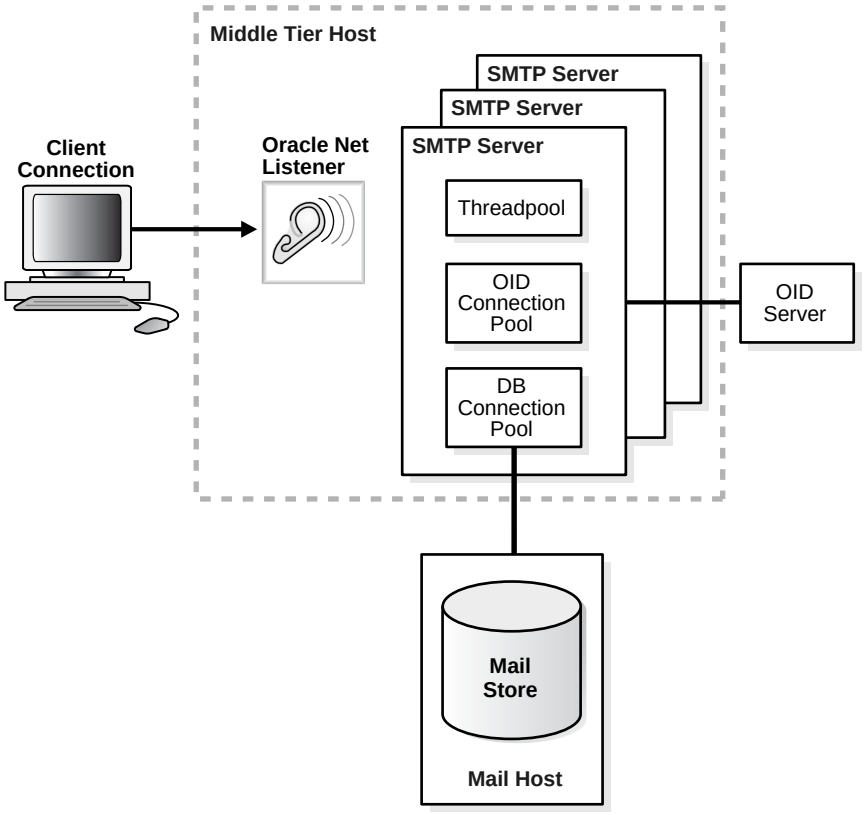
If the message is for a user on a different mail store, then the message is placed in the relay queue. The outbound server picks up the message and transfers the mail to the other mail store using the SMTP protocol.

Inbound SMTP Server Architecture

The Oracle Net listener listens on the SMTP port, default 25, and transfers connections to the SMTP server. The SMTP server maintains a pool of worker threads and two other pools: a database connection pool to the mail store and a pool of connections to the Oracle Internet Directory server. Upon receipt of a new client

connection, a thread is picked up from the pool to handle the request. It performs name resolution using a connection from the Oracle Internet Directory pool and then inserts the mail into the database using a connection from the database pool. At this point the SMTP connection to the client is terminated, but if there are local recipients, the worker thread continues to process the mail and performs local delivery.

Figure 4–5 SMTP Inbound



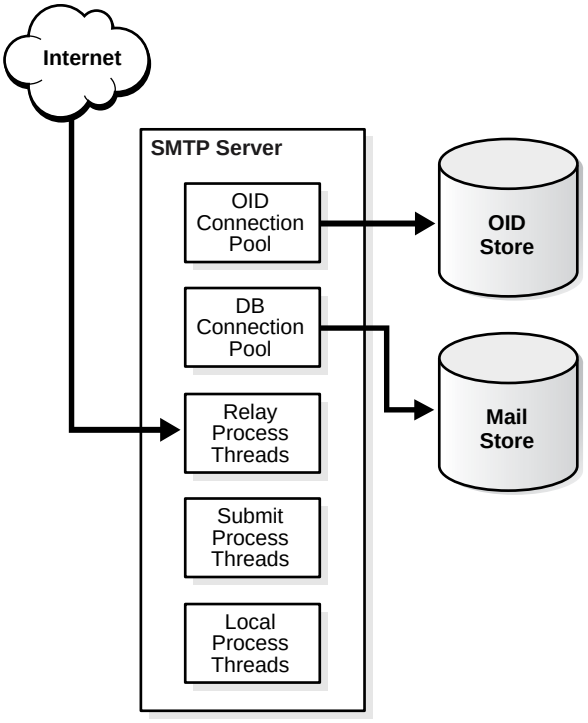
Outbound SMTP Server Architecture

The outbound STMP server has three main threads for each queue: submission, local, and relay. Each of the threads poll the database for messages in its queue. Whenever there are messages to process, a new thread is spawned to process the mail.

The SMTP server also maintains two other pools: a database connection pool to the mail store and a pool of connections to the Oracle Internet Directory server. When a thread is started to process a mail, it picks up a database connection from the pool and gets connections from the Oracle Internet Directory pool as needed. After the mail is processed, the db session is returned back to the pool.

If the delivery of an e-mail fails, the message is returned into the queue and delivery is retried after intervals defined by the `minqueueage` parameter. If the attempted re-deliveries are unsuccessful during the interval equal to the `queuetimeout` parameter, a delivery failure message is sent to the sender.

Figure 4–6 SMTP Outbound



Antispamming

The SMTP server supports a variety of anti-spam methods to prevent users and domains from spamming and to prevent the server from being used as relay for other domains. The various spam prevention methods are:

Method	Description
Relay blocking:	The SMTP server can be set up to block all relaying. Alternatively it can setup to relay only a known set of domains. For example to allow relaying of all messages received from hosts within the `foo.com` domain, the `RelayDomains` parameter can be set up to `foo.com`.
Reject messages received from certain domains:	The `RejectSender` parameter can be used to set up a list of domains from which messages are not allowed.

Method	Description
Reject messages from certain senders:	The `Rejectrecipients` parameter contains the list of recipients who are not allowed to receive messages.
Reject messages for certain recipients:	The `RejectDomains` parameter contains the list of senders who are not allowed to send messages.
Prevent service attack denials:	The SMTP `Floodmax` and `TimeInterval` parameters can be set to control the number of messages coming from a host. If more than `Floodmax` connections and messages are received from a particular host in the `Timeinterval`, all further messages from that host are rejected until the rate of incoming messages from the host falls below the `Floodmax` level.

Virus Scanning

The SMTP MTA supports virus scanning software from Symantec. Symantec's CarrierScan Server provides a native TCP/IP protocol to communicate with the client. Oracle Email provides a client program that can send the messages to the Symantec server and read the results of the scan. The MTA spawns this client program through the external filter process and pass the messages to it. The MTA delivery module verifies the returned value and decides whether to accept the message, or to reject the message and issue a DSN.

Processing

The LDAP parameter orclMailSmtpExternalFilter of the SMTP server must be set to TRUE, and orclMailSmtpExternalFilterProcess must be set to the name of the client program. When the local delivery module processes a message, it spawns the client program and passes the message to the new process. The client process communicates with the Symantec server using the native TCP/IP protocol. The client sends the protocol version, request and the message. The server reply consists of a reply code, scan results, and the modified message.

The local delivery module verifies the return value and decides whether to accept the message, or to reject the message. If there are any errors, the MTA rejects the message and send back a DSN.

Configuring the Virus Scanning Client

In the `$ORACLE_HOME/oes/admin/` directory, create a file for the Symatec server in the following format named `symantec.cfg`:

```
<host ip> <port>
```

where:

Name	Description
host ip	The IP address of the host running the Symantec server.
port	The port the Symantec server listens on.

Return Codes

```
SCSCANFILE_INF_NO_REP  (-3)
SCSCANFILE_INF_PARTIAL_REP  (-2)
SCSCANFILE_INF_REPAIRED  (-1)
SCSCANFILE_CLEAN   (0)
SCSCANFILE_FAIL_CONNECT  (1)
SCSCANFILE_FAIL_INPUTFILE  (2)
SCSCANFILE_FAIL_ABORTED  (3)
SCSCANFILE_INVALID_PARAM (4)
SCSCANFILE_FAIL_RECV_FILE (5)
SCSCANFILE_FAIL_MEMORY  (6)
SCSCANFILE_FAIL_FILE_ACCESS  (7)
```

External Process Interface

After a message is inserted into the database and before it is delivered, an external process can be used to filter the mail. An example of this is an virus scanner that can be invoked to scan the messages and reject messages that are infected. The external process returns a true or false indicating whether to accept or reject the mail. It is controlled by the following server parameters:

- `orclmailsmtpexternalfilter`: A Boolean operator to indicate if external filter processing is turned on.
- `orclmailsmtpexternalfilterprocess`: Path name of the executable for the external filter process

Address Rewriting Rules

An address rewriting rule can have the following attributes:

Attribute	Type (Length)	Description
RuleNo	NUMBER	Specifies the number of the sequence
Pattern	VARCHAR(255)	Pattern of the addresses you want to rewrite.

Result	VARCHAR(255)	Result that shows how you want to change addresses that fit the pattern.
Description	VARCHAR(255)	Description of the rule

Rule Symbols

Address rewriting rules contain symbols that determine how an address is parsed and how it is changed.

Symbol	Description
$+	Represents a non-empty string in a pattern.
$*	Represents an empty or non-empty string in a pattern.
$1, $2	Identifies parts of an address in a result.
$:	If this symbol is at the beginning of a result, it indicates that the rule should be applied only once.
$@	Stop immediately and ignore the remaining rules. This symbol must appear at the beginning of the result.

Rule Execution Guidelines

Address renaming rules are applied sequentially, starting with Rule 1. All rules are applied, unless a result starts with `$@`, that stops rule execution immediately and ignores any remaining rules. If a rule has a syntax error or cannot be executed, it is ignored.

A rule is applied to its own output in a loop until application of the rule yields no more changes in the result. The next rule in the sequence is then applied. After all the rules have been executed, an Oracle Internet Directory resolution is performed on the result. If the Oracle Internet Directory resolution returns a changed address such as an alias, the address rewriting rules are applied to the changed address, and the Oracle Internet Directory resolution is performed again. When the Oracle Internet Directory resolution yields no more changes, the rule execution process is done.

Housekeeping Process

Housekeeping is a standalone component that directly interacts with the mail store database. It is a background process performing cleanup tasks that can be set to run

according to a schedule. During Oracle Email installation, a housekeeping job is created by default and has a default configuration associated with it. Administrators can manually alter its schedule or add more instances of the job.

Job scheduling and management are taken care of by Oracle Enterprise Manager. During the time a housekeeping process is running, it responds to administrative requests such as reporting job progress, reinitializing job parameters, or shutting down.

> **See Also:** *Oracle Enterprise Manager Administrator's Guide* for more information on job scheduling

Other servers interact with housekeeping by producing garbage. There are primarily three types of garbage - producing agents:

- SMTP
- IMAP and POP3
- Housekeeping

Housekeeping performs tasks in multiple stages, with some stage of operations producing garbage for another stage. For example, when performing message expiration stage, the housekeeping process produces messages that are later consumed by the pruning stage.

SMTP server creates and processes messages that are mostly in transit. The messages stay in queues until the SMTP server finishes processing them. The processed messages are marked as processed, sent to the housekeeping process, and removed from the system. When users delete messages through clients, the messages are marked for deletion and are picked up by housekeeping process for actual deletion.

The housekeeping log files are located in two areas: the mail store and the middle tier. The log file on the mail store contains information on the progress of housekeeping tasks, the log file on the middle tier contains information on the status of the process.

Tertiary storage

Mail boxes are dynamic in nature. Mail constantly enters the store and is processed and deleted. Oracle Email housekeeping cleans up deleted messages and can be set to move old and rarely read mail messages to a tertiary tablespace.

The process of moving old messages to a tertiary tablespace is called tertiary storage. It enables administrators to move older messages to larger and less expensive disks according to a configurable age threshold. This process frees up valuable space on the primary disk to hold more recent and frequently accessed messages.

> **See Also:** for information on how to configure Oracle Email housekeeping server to perform tertiary storage

List Server Process

List servers provide a means of public list management as well as integration with other messaging services or applications.

The Oracle Email list server enables users to own and administer public mailing lists. The lists can be set up as a means of distributing information to groups of people or as a discussion forum. In addition, lists can be set up with restricted membership, where users must be approved before becoming a member. Lists can also be set up so that any messages sent out are moderated, where only certain members can send out messages. For example, the administrator of a mailing list may screen out advertisements.

When a distribution list has a large number of members, the Oracle Internet Directory `Max Search Results Entries` parameter must be configured to return a large number of entries to enable the list resolution API to return all the members. The Oracle Internet Directory `Max Search Results Entries` parameter can be configured through `oidadmin`.

APIs provided with the Oracle Email list server enable users to customize lists and messages sent out to a list. The Oracle Email list server features can be used for applications such as marketing campaigns where special non-transferable offers are sent and readable only by the intended recipients. For example, a user can use the list server APIs to query a database of sales information to create a list of all customers who have made purchases in the past three months, then send coupons by e-mail to each of the customers with discounts based on the amount of their purchases.

List Server Mail Interface

The list server mail interface command is a list of commands that administrators and users can send to the list server to perform certain tasks. The `setattribute` command is used by administrators to set values for various list parameters.

See Also: *Oracle Collaboration Suite User's Guide* for more information on list server mail interface commands for users.

Setattribute

Use this command to set values for the various parameters of the lists by an administrator.

Syntax

```
setattribute type=<list type> subscription=<subscription type> topic="<list
topic>" autoreconfirm=<true/false> post=<post type> editor=<editor mailid>
moderator=<moderator mailid> invitetext="<multi-line text>"
```

Although none of the parameters are mandatory, every `setattribute` command should have at least one parameter. The following is a list parameters and their definitions:

Parameter	Definition
type	This sets the type of the list. Valid values are announcement, discussion, edited, and moderated.
subscription	This sets the type of subscription control you want on the list. Valid values are open, restricted, and closed.
topic	This sets the list topic. The topic value needs to be enclosed within quotes.
autoreconfirm	This sets whether subscription commands should reconfirmed by the list server or not. Valid values are true or false.
post	This sets the posting type allowed by the list. Valid values are open or subscriber.
editor	For an edited list, this parameter sets the editor for the list. More than one editor parameter can be set with a single `setattribute` command.
moderator	For a moderated list, this parameter sets the moderator for the list. More than one moderator parameter can be set with a single `setattribute` command.
invitetext	This parameter specifies text that will be part of the mail sent to users an owner invites to join his/her list using the invite command. The parameter value should be enclosed within quotations. This text can be multi-lined and individual lines should be separated with a newline (\n) character.

Mail Merge and Schedule Mail Delivery

The list server supports mail merge and scheduled mail delivery. This feature can be enabled for a list by providing a value for the merge tag property of the list.

Mail Merge

Mail merge: Enables customized mail to be delivered to every list recipient. The list server supports two types of mail-merge:

- Standard mail-merge: In this type of mail-merge, the contents of a mail can be customized for each recipient with the following values:

```
Recipient's mail address(recipient_mail_address)
Recipient's first name(recipient_first_name)
Recipient's last name(recipient_last_name)
Recipient's full name(recipient_full_name)
Current date(current_date)
Current time(current_time)
```

To use this feature, use the mail merge tag in the appropriate sections of the mail. For example, If the list's mail merge property is `orcl`, and the mail is addressed with the recipient's full name, the mail looks like the following:

```
Dear <orcl>recipient_full_name</orcl>,
    ...
    ...
```

- PL/SQL mail-merge: Enables embedding of PL/SQL in messages. For every recipient, the PL/SQL function is executed and the output is embedded in the mail before delivery. The PL/SQL function must return a `varchar2` string. As a parameter to the PL/SQL function, any of the parameters defined in the standard mail-merge can be included.

For example, if you have a PL/SQL `getsalary` function that returns the salary of an individual given his mail address, you can send a mail to a list of employees letting them know their salaries as follows.

```
Dear <orcl>recipient_full_name</orcl>,
    Your salary is <orcl>getSalary(recipient_mail_address)</orcl>.
    ...
```

By default, the list server looks for the PL/SQL function in the mail store that the server is connected to. If the function is on a different database, a database link must be created to that database in `ES_MAIL` schema and use that in the mail-merge tag.

For example, the `getSalary` function is defined in a different database, create a database link called `dblink`. The mail will now look like this:

```
Dear <orcl>recipient_full_name</orcl>,
    Your salary is <orcl>getSalary(recipient_mail_address)@dblink</orcl>.
    ...
```

Scheduled Mail Delivery

This feature can be used to schedule mail delivery to a list at a particular time. This feature can be enabled by providing a value for the mail merge property of the list. In the mail, specify the delivery time of the mail by putting in the schedule mail delivery tag anywhere in the mail. The following is an example of how the tag appears if the mail merge property of the list is `orcl`.

```
<orcl>send_schedule=DD-MON-YYYY hh24:mi [TZH:TZM]</orcl>
```

Parameter	Description
`DD`	The date
`MON`	The 3 letter abbreviation for the month
`YYYY`	The year
`hh24`	The time in a twenty-four hour period
`mi`	The time in minutes
`TZH`	The optional time zone hour offset
`TZM`	The optional time zone minute offset

If `TZH` and `TZM` are not specified, the list server uses the sender's time zone to schedule delivery of the mail.

Managing Server Processes

This section discusses how to start, stop, reinitialize, and modify server processes.

Starting, Stopping, or Reinitializing All Server Processes

Starting an Oracle Email process starts all the processes comprising that service type, such as IMAP and POP.

Stopping an Oracle Email system sends a request to the operating system to shut down all of the Oracle Email processes. A reason and administrator would want to stop the Oracle Email system is to perform maintenance, such as upgrading the server hardware or software. It is not possible for the processes to be running while certain kinds of upgrades are performed.

Reinitializing an Oracle Email process informs the operating system to reload its operational settings from the Oracle Internet Directory server. The process does not stop running, which means that users continue to receive uninterrupted service. Whenever a Oracle Email process parameter is modified, it must be reinitialized to make the changes take effect.

Note: The following functions can only be executed if at least one instance has been created.

Using Oracle Enterprise Manager, perform the following steps to start, stop, or reinitialize all server processes:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Click **Start, Stop,** or **Reinitialize**.

Creating a Server Instance

Using Oracle Enterprise Manager, perform the following steps to create a server instance:

To create a new IMAP4 server instance with default parameters:

1. Navigate to the Oracle Email Service Targets page.

2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Click **Create**. This creates a new server instance with default parameters.

To create a new server instance with the same parameter values as an existing server instance:

1. Select the process with the parameters you want to replicate.
2. Click **Create Like**. This creates a new server instance with the same parameters as the selected server instance.

Deleting a Server Instance

Warning: Deleting an Oracle Email process may disable some or all e-mail processes.

Using Oracle Enterprise Manager, perform the following steps to delete a server instance:

1. Navigate to the Oracle Email Service Targets page.

Note: A process must be shut down before it can be deleted.

2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server process you want to delete.
4. Click **Delete**.

Starting a Server Instance

Using Oracle Enterprise Manager, perform the following steps to start a server instance:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server instance you want to start.

4. Click **Start**.

Stopping a Server Instance

Using Oracle Enterprise Manager, perform the following steps to stop a server instance:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server instance you want to stop.
4. Click **Stop**.

Reinitializing a Server Instance

Note: Servers must be reinitialized whenever parameters are modified.

Using Oracle Enterprise Manager, perform the following steps to reinitialize a server instance:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server process you want to reinitialize.
4. Click **Reinitialize**.

Note: reinitializing does not interrupt user actions because the service is not brought down.

Modifying Parameters for a Specific Server Instance

Note: Servers must be reinitialized whenever parameters are modified.

Using Oracle Enterprise Manager, perform the following steps to modify parameters for a specific server instance:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server instance you want to modify.
4. Modify the parameters you want to change.
5. Click **Apply**.

To make the changes take effect, you must reinitialize the server instance.

Modifying Server Default Parameters

Note: Servers must be reinitialized whenever parameters are modified.

Using Oracle Enterprise Manager, perform the following steps to modify server default parameters:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type. For example, IMAP, POP, SMTP, Housekeeping, or List server.
3. Select **Change Settings**.
4. Modify the parameters you want to change.
5. Click **Apply**.

To make the changes take effect, you must reinitialize the server.

Server Parameters

This section provides parameter definitions for the different servers.

IMAP4

The following is a list of IMAP4 server parameters and their definitions:

LDAP Connection Pool Increment

This parameter specifies the number of Oracle Internet Directory connections the pool is increased by.

Range: none

Default Value: 0

LDAP Minimum Connection Pool

This parameter specifies the minimum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Maximum Connection Pool

This parameter specifies the maximum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Connection Pool Time Lag

This parameter specifies the time lag (1/100th of a second) permitted before increasing the pool. For example, if the time lag value is 100 it means 1 second. This indicates that if the concurrent connections arrive within 1 second then the server must wait.

LDAP Connection Retry Interval

This parameter specifies the maximum time in seconds the server has to wait to get a connection after it has reached the maximum number of connections.

LDAP Reconnection Timeout

This parameter specifies the time after which the server tries to reconnect to Oracle Internet Directory.

LDAP Number of Retry Before Erroring

This parameter specifies the total number of times the server attempts to connect to Oracle Internet Directory.

Protocol Server Increment Thread

This parameter specifies the number of threads the client connection pool is increased by.

Range: 1-999

Default Value: 1

Protocol Server Thread Timeout

This parameter specifies the number of seconds after which an idle thread is cleaned up.

Range: 0-65535

Default Value: 1860 seconds

Protocol Server Minimum Threads

This parameter specifies the minimum number of threads available for client connection handling.

Range: 1-1000

Default Value: 1

Protocol Server Maximum Threads

This parameter specifies the maximum number of threads available for client connection handling.

Range: 0-1000

Default Value: 500

Presentation Name

This parameter determines which port the listener is listening for the IMAP service. Selecting **Custom** enables administrators to specify their own presentation name.

Range: string

Default Value: IMAP

Custom Name

This parameter applies on if the presentation name is set to custom.

SSL Enabled

This parameter applies if the presentation name is set to custom.

Process Log Level

This parameter specifies the log messages level.

Range: 0-30

Default Value: 6

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Range: 4294967295 (32bits, multi-value)

Default Value: 0

Default Domain

This parameter specifies the default domain for user login. If the user does not provide a domain when logging in, the value of this parameter is used.

Default Value: none

Maximum Number of Clients

This parameter specifies the maximum number of clients allowed to connect to the server instance.

Range: 0-1000

Default Value: 1000

Maximum Rule Nesting Level

This parameter specifies the maximum number of times a nesting rule can be applied to a message. In general, setting this parameter to a smaller number increases overall performance, but not for systems that use rules heavily.

Range: >=1

Default Value: 20

Cache Size

This parameter specifies the caching level. For small caching levels, no mail information is cached in the middle tier IMAP server. For medium caching levels, certain parts of mail are cached. Increasing the cache size increases the memory requirements on the middle tier.

Range: small, medium

Default Value: small

Get New Mail Interval

This parameter specifies the interval to wait before checking for new mail. The IMAP server does not check for new mail until the time interval has elapsed. If clients send a large number of check new mail requests to the server it affects performance.

Range: 0-65535

Default Value: 120 seconds

Timeout Interval

Auto-logout timeout interval. If a client does not perform any operations within this interval, it is disconnected.

Range: 0-65535

Default Value: 1800 seconds

Debug User

This parameter enables administrators to select a user name to obtain more debug information in the log files for that user.

Range: None

Default Value: None

POP3

The following is a list of POP3 server parameters and their definitions:

LDAP Connection Pool Increment

This parameter specifies the number of Oracle Internet Directory connections the pool is increased by.

Range: none

Default Value: 0

LDAP Minimum Connection Pool

This parameter specifies the minimum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Maximum Connection Pool

This parameter specifies the maximum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Connection Pool Time Lag

This parameter specifies the time lag (1/100th of a second) permitted before increasing the pool. For example, if the time lag value is 100 it means 1 second. This indicates that if the concurrent connections arrive within 1 second then the server must wait.

LDAP Connection Retry Interval

This parameter specifies the maximum time in seconds the server has to wait to get a connection after it has reached the maximum number of connections.

LDAP Reconnection Timeout

This parameter specifies the time after which the server tries to reconnect to Oracle Internet Directory.

LDAP Number of Retry Before Erroring

This parameter specifies the total number of times the server attempts to connect to Oracle Internet Directory.

Presentation Name

This parameter determines which port the listener is listening for the POP service. Selecting **Custom** enables administrators to specify their own presentation name.

Range: string

Default Value: POP

Process Log Level

This parameter specifies the log messages level.

Range: 0-30

Default Value: 6

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Range: 4294967295 (32bits, multi-value)

Default Value: 0

Default Domain

This parameter specifies the default domain for user login. If the user does not provide a domain when logging in, the value of this parameter is used.

Default Value: none

Maximum Number of Clients

This parameter specifies the maximum number of clients allowed to connect to the server instance.

Range: 0-1000

Default Value: 1000

Maximum Rule Nesting Level

This parameter specifies the maximum number of times a nesting rule can be applied to a message. In general, setting this parameter to a smaller number increases overall performance, but not for systems that use rules heavily

Range: >=1

Default Value: 20

POP3 Delete Allowed

Enables read messages to be deleted from the server.

Range: [yes, no]

Default Value: no

POP3 Retrieval

Enables all or unread messages to be retrieved from the server. If the parameter is set to all, the server retrieves all mails. If the parameter is set to unread or any other value, the server only retrieves unread messages.

Range: [unread, all]

Default Value: unread

SMTP In

The following is a list of SMTP In server parameters and their definitions:

LDAP Connection Pool Increment

This parameter specifies the number of Oracle Internet Directory connections the pool is increased by.

Range: none

Default Value: 0

LDAP Minimum Connection Pool

This parameter specifies the minimum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Maximum Connection Pool

This parameter specifies the maximum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Connection Pool Time Lag

This parameter specifies the time lag (1/100th of a second) permitted before increasing the pool. For example, if the time lag value is 100 it means 1 second. This indicates that if the concurrent connections arrive within 1 second then the server must wait.

LDAP Connection Retry Interval

This parameter specifies the maximum time in seconds the server has to wait to get a connection after it has reached the maximum number of connections.

LDAP Reconnection Timeout

This parameter specifies the time after which the server tries to reconnect to Oracle Internet Directory.

LDAP Number of Retry before Erroring

This parameter specifies the total number of times the server attempts to connect to Oracle Internet Directory.

Spam Flood Interval

This parameter specifies the time interval, in minutes, to detect spam flooding.

Range: none

Default Value: none

Spam Maximum Flood Count

SMTP server signals flooding if the number of messages and connections from a single host exceeds the Spam Max. Flood Count within the Spam Flood Interval.

Range: none

Default Value: none

Native Anti-Spamming

Turns on anti-spamming checks. If this parameter is not set, all anti-spamming checks are turned off.

Range: Boolean

Default Value: false

Relay Allowed

Enables domains determined by SMTP Relay Domains Allowed. Anti-spamming check.

Range: string

Default Value: false

Reject Domains

This parameter specifies the list of domains and sub-domains to be rejected. Anti-spamming check.

Range: multi-value

Default Value: none

Reject Senders

This parameter specifies the list of senders to be rejected. Anti-spamming check

Range: multi-value

Default Value: none

Reject Recipients

This parameter specifies the list of recipients to be rejected. Anti-spamming check.

Range: multi-value

Default Value: none

Reject IPs

This parameter specifies a list of IP addresses to be rejected.

Range: none

Default Value: none

Trusted IPs

List of IP addresses from which connections are permitted, regardles the criteria.

Range: none

Default Value: none

Trusted Domains

This parameter specifies a list of allowed domains or subdomains from which mail is received, regardles the criteria.

Range: none

Default Value: none

Relay Domains Allowed

This parameter specifies the list of domains to relay through. Anti-spamming check.

Range: multi-value

Default Value: none

Recipient Rewriting Rules

This parameter rewrites rules for recipients.

Range: multi-value

Default Value: none

Sender Rewriting Rules

This parameter rewrites rules for senders.

Range: multi-value

Default Value: none

Local Domains

This parameter specifies the list of local domains.

Range: multi-value

Default Value: none

Protocol Server Increment Thread

This parameter specifies the number of threads the client connection pool is increased by.

Range: 1-999

Default Value: 1

Protocol Server Thread Timeout

This parameter specifies the number of seconds after which an idle thread is cleaned up.

Range: 0-65535

Default Value: 1860 seconds

Protocol Server Minimum Threads

This parameter specifies the minimum number of threads available for client connection handling.

Range: 1-1000

Default Value: 1

Protocol Server Maximum Threads

This parameter specifies the maximum number of threads available for client connection handling.

Range: 0-1000

Default Value: 500

Presentation Name

This parameter specifies the presentation name used by the server to register with the Oracle9*i* listener.

Range: string

Default Value: ESSMI

Process Log Level

This parameter specifies the log messages level.

Range: 0-30

Default Value: 10

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Range: 4294967295 (32bits, multi-value)

Default Value: 0

Maximum Number of Clients

This parameter specifies the maximum number of clients permitted to connect to the server at one time.

Range: 0-1000

Default Value: 1000

Maximum Rule Nesting Level

This parameter specifies the maximum number of times a nesting rule can be applied to a message. In general, setting this parameter to a smaller number increases overall performance, but not for systems that use rules heavily

Range: >=1

Default Value: 20

Checkpoint Interval

This parameter specifies the number of recipients processed in a single relay delivery attempt.

Range: >=1

Default Value: 4

Fallback MX Host

This parameter, when set, specifies the host where relay messages are sent.

Range: string

Default Value: none

Maximum Hop Count

This parameter specifies the maximum number of hops a message can go through.

Range: >=1

Default Value: 25

Maximum Message Size

This parameter specifies the maximum allowed incoming message size in bytes.

Range: >=0

Default Value: 0

SMTP Minimum Queue Age

If the message has been in the queue less than the SMTP Min. Queue Age, the message is skipped for a delivery attempt.

Range: integer

Default Value: 30 minutes

Postmaster Copy

If the postmaster address is set, a copy of the delivery status notification is sent to it.

Range: string

Default Value: none

Message Timeout

When a SMTP server is shutdown it may be in the middle of processing certain messages. When the server is restarted, it looks for messages that are being processed. If the messages remain in the same state for this parameter value interval (in minutes), it assumes that the messages are left over from the previous run and processes them again.

Range: integer

Default Value: 30

SMTP Queue Timeout

This parameter specifies the maximum time a message can be in the queue.

Range: >=1

Default Value: 5 days

Use Errors To

This parameter determines if the Errors To header is to be used for delivery status notifications.

Range: Boolean

Default Value: False

Connection Number

This parameter specifies the number of SMTP connections the outbound SMTP server caches for future delivery to the same host.

Range: >1

Default Value: 0

Authentication

This parameter determines if SMTP authentication is enabled. If mandatory is selected, users must authenticate themselves before sending any messages. If optional is selected, users may authenticate themselves, although the SMTP server

will accept the message even if the authentication fails. If none is selected, no authentication is required.

Range: String

Default Value: Optional

Submit Only

This parameter submits inbound messages without resolving the recipient.

Range: Boolean

Default Value: false

External Filter

Enables external filter processing if set to true.

Range: Boolean

Default Value: false

External Filter Process

This parameter specifies the path for the executable of the external process.

Range: string

Default Value: none

SMTP Out

The following is a list of SMTP Out server parameters and their definitions:

LDAP Connection Pool Increment

This parameter specifies the number of Oracle Internet Directory connections the pool is increased by.

Range: none

Default Value: 0

LDAP Minimum Connection Pool

This parameter specifies the minimum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Maximum Connection Pool

This parameter specifies the maximum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Connection Pool Time Lag

This parameter specifies the time lag (1/100th of a second) permitted before increasing the pool. For example, if the time lag value is 100 it means 1 second. This indicates that if the concurrent connections arrive within 1 second then the server must wait.

LDAP Connection Retry Interval

This parameter specifies the maximum time in seconds the server has to wait to get a connection after it has reached the maximum number of connections.

LDAP Reconnection Timeout

This parameter specifies the time after which the server tries to reconnect to Oracle Internet Directory.

LDAP Number of Retry before Erroring

This parameter specifies the total number of times the server attempts to connect to Oracle Internet Directory.

Spam Flood Interval

This parameter specifies how often to detect spam flooding. Anti-spamming check.

Range: none

Default Value: none

Spam Maximum Flood Count

SMTP server signals flooding if the number of messages and connections from a single host exceeds the Spam Max. Flood Count parameter within the Spam Flood Interval parameter. Anti-spamming check.

Range: none

Default Value: none

Native Anti-Spamming

Turns on anti-spamming checks. If this parameter is not set, all anti-spamming checks are turned off.

Range: Boolean

Default Value: false

Relay Allowed

Enables domains determined by SMTP Relay Domains Allowed. Anti-spamming check.

Range: string

Default Value: false

Sender Rewriting Rules

This parameter rewrites rules for senders. Used only by the SMTP In server.

Range: multi-value

Default Value: none

Local Domains

This parameter specifies the list of local domains. Used only by the SMTP Out server.

Range: multi-value

Default Value: none

Process Log Level

This parameter specifies the log messages level.

Range: 0-30

Default Value: 10

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Range: 4294967295 (32bits, multi-value)

Default Value: 0

Maximum Rule Nesting Level

This parameter specifies the maximum number of times a nesting rule can be applied to a message. In general, setting this parameter to a smaller number increases overall performance, but not for systems that use rules heavily

Range: >=1

Default Value: 20

Checkpoint Interval

This parameter specifies the number of recipients processed in a single relay delivery attempt.

Range: >=1

Default Value: 4

Fallback MX Host

This parameter, when set, specifies the host where relay messages are sent.

Range: string

Default Value: none

Maximum Hop Count

This parameter specifies the maximum number of hops a message can go through.

Range: >=1

Default Value: 25

Maximum Message Size

This parameter specifies the maximum allowed incoming message size in bytes.

Range: >=0

Default Value: 0

SMTP Minimum Queue Age

If the message has been in the queue less than the SMTP Min. Queue Age, the message is skipped for a delivery attempt.

Range: integer

Default Value: 30 minutes

Postmaster Copy

If the postmaster address is set, a copy of the delivery status notification is sent to it.

Range: string

Default Value: none

Message Timeout

When a SMTP server is shutdown it may be in the middle of processing certain messages. When the server is restarted, it looks for messages that are being processed. If the messages remain in the same state for this parameter value interval (in minutes), it assumes that the messages are left over from the previous run and processes them again.

Range: integer

Default Value: 30

SMTP Queue Timeout

This parameter specifies the maximum time a message can be in the queue.

Range: >=1

Default Value: 5 days

Use Errors To

This parameter determines if the Errors To header is to be used for delivery status notifications.

Range: Boolean

Default Value: False

Connection Number

This parameter specifies the number of SMTP connections the outbound SMTP server caches for future delivery to the same host.

Range: >1

Default Value: 0

Authentication

This parameter determines if SMTP authentication is enabled.

Range: Boolean

Default Value: false

External Filter

Enables external filter processing if set to true.

Range: Boolean

Default Value: false

External Filter Process

This parameter specifies the path for the executable of the external process.

Range: string

Default Value: none

Housekeeping

The following is a list of housekeeping parameters and their definitions:

Collection

This parameter determines whether to run the collection task. The collection task collects or reclaims space taken up by messages that are no longer used by removing the message data. Oracle Corporation recommends scheduling this task to run continuously, to keep up with the rate of messages coming in from outside the server.

Range: Enalbed or Disabled

Default Value: Enabled

Expiration

This parameter determines whether to run the expiration task. The expiration task expires or deletes messages set to expire on or before the current time according to a timer, by moving such messages to the system trash folder. The expiration timer is a folder attribute and can be set by users. Oracle Corporation recommends running this task only once a day.

Range: Enabled or Disabled

Default Value: Disabled

Text Synchronization

This parameter specifies whether to perform Oracle Text index synchronization task. Performing synchronization is essential to content-based searching. Doing it frequently should greatly increase search performance. However when expected rate of incoming message is low, doing it too frequently increases the server load unnecessarily. If content-based searching through Oracle Text is used heavily, it is recommended that a dedicated housekeeping instance should be created for this task with a sleep time of five to ten minutes.

Range: Enabled or Disabled

Default Value: Disabled

Text Optimization

This parameter specifies whether to perform Oracle Text optimization task. Oracle Text optimization improves performance of index synchronization. Without optimization, synchronization performance degrades over time. It is recommended that this task should be done about once per week. A dedicated housekeeping instance should be created with task enabled and a sleep time of 24*7 (168) hours.

Range: Enabled or Disabled

Default Value: Disabled

Pruning

This parameter determines whether to run the pruning task. The pruning task clears up message queues and the system trash folder, and marks un-referenced messages for the collection task. Oracle Corporation recommends scheduling this task to run continuously, to keep up with user message deletion activity.

Range: Enabled or Disabled

Default Value: Enabled

Tertiary Store

This parameter determines whether to run the tertiary store task. This task archives old messages by moving them to another tablespace, presumably a cheaper and larger storage designed for archiving. Oracle Corporation recommends running this task on a monthly basis.

Range: Enabled or Disabled

Default Value: Disabled

Process Log Level

This parameter specifies the log message level.

Range: 0-30

Default Value: 6

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Range: 4294967295 (32bits, multi-value)

Default Value: 0

Tertiary Storage Age Threshold

This parameter specifies the minimum age of messages in number of days, for archiving. If the tertiary storage task is turned on, housekeeping tries to archive messages older than the number of days specified in this parameter. Oracle Corporation recommends setting this parameter to at least 30.

Range: Non negative number

Default Value: 30

Maximum Rule Nesting Level

This parameter specifies the maximum number of times a nesting can be applied to a message. In general, setting this parameter to a smaller number increases overall performance, but not for systems that use rules heavily.

Range: >=1

Default Value: 20

Process Sleep Duration

This parameter specifies the interval between two consecutive starts of the task processing in hours. If the task takes less than this amount of time to finish, the housekeeping process sleeps for the rest of the duration. If the task takes more time than the amount of sleep time specified, the process does not sleep but instead runs continuously.

Range: >=0

Default Value: 60 minutes

List Server

The following is a list of list server parameters and their definitions:

LDAP Connection Pool Increment

This parameter specifies the number of Oracle Internet Directory connections the pool is increased by.

Range: none

Default Value: 0

LDAP Current Connection Pool

This parameter specifies the current number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Minimum Connection Pool

This parameter specifies the minimum number of Oracle Internet Directory connections in the pool.

Range: none

Default Value: 10

LDAP Maximum Connection Pool

This parameter specifies the maximum number of Oracle Internet Directory connections in the pool.

Default Value: 10

Range: none

Process Log Level

This parameter specifies the message log level.

Default Value: 6

Range: 1-30

Process Debug Level

This parameter specifies the debug messages level. To enable statistics, set the value to 512.

Default Value: 0

Number of Mails Processed Concurrently

This parameter specifies the number of messages to be processed simultaneously by the list server.

Default Value: 50

Range: Any positive number greater than zero.

Number of Threads per Mail

This parameter specifies the number of threads to be spawned to deliver list server messages to subscribers.

Default Value: 10

Range: Any positive number greater than zero.

Number of Recipients per Batch

This parameter specifies the number of users that each of the user threads deliver messages to.

Default Value: 1000

Range: Any positive number greater than zero.

Local Domains

This parameter specifies the list of local domains served by the list server process.

Range : multi-value

Default : none

Server Log Files

This sections provides log file locations for the different servers.

IMAP4

The following specifies the IMAP4 log file locations:

On UNIX:

```
$ORACLE_HOME/oes/log/<install_name>/imap/<process_id>/<process_id>.log.
```

On Windows:

```
%ORACLE_HOME\oes\log\<install_name>\imap\<process_id>\<process_id>.log.
```

POP3

The following specifies the POP3 log file locations:

On UNIX:

```
$ORACLE_HOME/oes/log/<install_name>/pop/<pid>/<pid>.log
```

On Windows:

```
%ORACLE_HOME\oes\log\<install_name>\pop\<pid>\<pid>.log
```

SMTP In and SMTP Out

The following specifies the SMTP In and SMTP Out log file locations:

On UNIX:

```
$ORACLE_HOME/oes/log/<install_name>/smtp_in/<pid>/<pid>.log
$ORACLE_HOME/oes/log/<install_name>/smtp_out/<pid>/<pid>.log
```

On Windows:

```
%ORACLE_HOME\oes\log\<install_name>\smtp_in\<pid>/<pid>.log
%ORACLE_HOME\oes\log\<install_name>\smtp_out\<pid>\<pid>.log
```

Housekeeping

The following specifies the housekeeping log file locations:

Mail Store on UNIX:

```
$ORACLE_HOME/oes/log/collector/<SID>.*/text.log
```

Mail Store on Windows:

```
%ORACLE_HOME\oes\log\collector\<SID>.*\text.log
```

Middle Tier on UNIX:

```
$ORACLE_HOME/oes/log/<install_name>/gc./<pid>/<pid>.log
```

Middle Tier on Windows:

```
%ORACLE_HOME\oes\log\<install_name>\gc.\<pid>\<pid>.log
```

List Server

The following specifies the list server log file locations:

On UNIX

```
$ORACLE_HOME/oes/log/<install_name>/list/<pid>/<pid>.log
```

On Windows

```
%ORACLE_HOME%\oes\log\<install_name>\list\<pid>\<pid>.log
```

SSL Setup

Secure Sockets Layer (SSL), is a protocol used for transmitting private documents over the Internet. SSL works by using a public key to encrypt data that is transferred over the SSL connection. Many Web sites use the protocol to obtain confidential user information, such as credit card numbers. By convention, URLs requiring an SSL connection start with `https:` instead of `http:`.

Obtaining a SSL Server Certificate

In order for the server to communicate securely with clients, customers must obtain an SSL Server Certificate for their machine and configure their network listener to use that certificate.

> **See Also:** Chapter 16 of the *Oracle Advanced Security Administrator's Guide*, for how to use the Wallet Manager to create a wallet and store SSL certificates.

NOTE: You must have a separate certificate for each machine on which the protocol server processes are running, but you can use the same certificate for all protocol server processes on the same machine.

In the Oracle environment, you can use the Oracle Wallet Manager to create and store certificates and the corresponding private keys securely.

To obtain a certificate, use the Wallet Manager to perform the following steps:

1. Create a new wallet, if one does not already exist. The same wallet can be used by all the servers running on that machine.
2. Generate a certificate request. This generates the corresponding private key and store it in the wallet.
3. Enter the host name along with the domain name as the Common Name while generating a certificate request.
4. Send the certificate request to a Certificate Authority like VeriSign and get it signed.
5. Store the signed certificate in the wallet with the Auto Login option enabled on. You should see the certificate status set to Ready.

6. Remember to store the wallet with the Auto Login option enabled. The option is under the Wallet menu option in the Wallet Manager.

   This creates a `cwallet.sso` file in addition to the `ewallet.p12`, that is the actual wallet. The files can be found in the following location: `/etc/ORACLE/WALLETS/<userid>`.

Configuring the Network Listener for SSL

The `listener.ora` file is updated with the required SSL and non- SSL listening end points for both the IMAP4 and POP3 servers during installation. Users only need to set the wallet location and other optional SSL parameters in the `listener.ora` and the `sqlnet.ora` files for the listener to receive SSL connections. This can be done manually or by using the Oracle Network Manager.

Setting the Wallet Location Manually

Add the following `WALLET_LOCATION` and `SSL_CLIENT_AUTHENTICATION` entries in the beginning of the `$TNS_ADMIN/listener.ora` and `$TNS_ADMIN/sqlnet.ora` files:

```
WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = <Directory path containing the cwallet.sso file>)
    )
  )

SSL_CLIENT_AUTHENTICATION = FALSE
```

The following is what a typical directory parameter value looks like:

```
/etc/ORACLE/WALLETS/<userid>
```

The `SSL_CLIENT_AUTHENTICATION` parameter should be set to True, if the client needs to be authenticated by the server. This requires clients to present their certificates during the SSL handshake.

> **See Also:** Chapter 7 of the *Oracle Advanced Security Administrator's Guide* to set the wallet location using Oracle Network Manager.

If the SSL_CLIENT_AUTHENTICATION parameter is not set, the default setting is true and clients are required to present a certificate during the SSL handshake. If the intent is only to secure the communication, not to authenticate the client using the certificate, then this parameter should be set to false.

Configuring Protocol Servers for SSL

IMAP and POP protocol servers can be configured to use SSL for securely communicating with and authenticating clients. To use the SSL client connections, administrators can configure an existing server instance or create a new instance.

A separate server instance is necessary to use SSL and non SSL connections. One server instance cannot manage both types of connections. By default, server instances are configured to manage non SSL connections only. The default listening end points for both IMAP and POP protocol servers are created in the `listener.ora` file during installation.

Perform the following steps to configure a SSL server instance:

1. Log in to Oracle Enterprise Manager.
2. Select the Application Server instance where Oracle Email is installed.
3. Click on **Oracle Email**.
4. Click on **IMAP** or **POP**.
5. Click on the process instance.
6. Select **IMAPSSL**, **POPSSL**, or **Custom** from the drop down list.
   - If **Custom** is selected provide a specific presentation name in the corresponding field.
   - If **Custom** is selected, change the `SSL Enabled` parameter to true and verify that there is a description entry in the `listener.ora` file for the presentation name. Verify that `Protocol` is set to TCPS and that the `PORT` set to the default SSL port number of the protocol. For IMAP, the default SSL port number is 993 and for POP, it is 995.
7. Reinitialize the server instance.

Thin Client

The Thin Client gives users a simple and fast means to access messages and other self service features through a web browser. A user points their browser to a predetermined URL to log in to their e-mail account. Their inbox is rendered dynamically. The logic to render a user's folders, messages, public directory and personal address book runs at the Oracle9*i* Application Server web server. The browser acts merely as a keyboard and screen. There is no processing or data storage on the desktop.

The Thin Client provides a standard, out of the box web mail solution, along with a tool kit that can extend and modify the standard solution.

Thin Client Tool Kit Properties

The Thin Client log files are derived from values in the following `toolkit.properties` file:

- `toolkit.logdirectory`

The files are placed in `toolkit.logdirectory`; the filename is

```
$ORACLE_HOME/um/log/webmail_client/xxxx/text.log
```

The tool kit provides a framework for easy additions or modifications of simple functionality or presentation of the Thin Client. For example, a deployment can replace the Oracle logo with a different graphic. The tool kit can also enforce aspects of the application such as the availability of basic actions or functionality. This reduces the amount of development effort required to customize a solution for customer specific needs.

Properties are defined in the following directory:

```
$ORACLE_HOME/j2ee/oc4j_um/config/oc4j.properties
```

The following line ensures that the Thin Client reads this file into the environment only at startup. This line must not be changed or removed.

```
ct_env=set
```

SMTP Server Settings

Remove the comment and change this value to point to the SMTP host.

```
mail.smtp.host=%machinehost%
```

Remove the comment and change this value to point to the port address for the SMTP service listener. The SMTP server default port 25). This value is seldom changed, except to distribute the load for sending many messages.

```
mail.smtp.port=25
```

Set the fully qualified domain name to be appended to addresses that are not fully qualified.

```
mail.host.qualifiedname=oracle.com
```

If someone sends an e-mail message to recipient, the Thin Client first attempts to resolve the name from the user's personal address book and other LDAP directories. If the recipient does not exist, the program assumes the message is for someone in the fully qualified domain. This value is appended to recipient and sent. If the mail domain was set to acme.com, recipient would be re-written as `recipient@acme.com`.

Tool Kit Default Settings

State File

The state file is a XML file that defines the navigation behavior of the Thin Client. This file provides a way to easily define and manage state file transitions in the client. A state transition defines the logic that must be executed by the application when the user moves from one state file to another.

The state file for configuration can be found in the following directory:

```
toolkit.statefile=%ORACLE_HOME%/um/client/config/statefile.xmll
```

- URL Directory for Pages:

  ```
  toolkit.clientdir=/templates/
  ```

- Image Directory for Images

  ```
  toolkit.imagedir=/um/images/
  ```

- Suffix for State -> Pages

  ```
  toolkit.pagesuffix=.uix
  ```

- Supported Languages (Based on Java locale conventions)

  ```
  supportedLanguages=en,ar,cs,da,de,el,es,fi,fr,fr_
  ```

```
CA,hu,it,iw,ja,ko,nl,no,pl,pt,pt_BR,ro,ru,sk,sv,th,tr,zh_CN,zh_TW

voicemailLanguages=en-GB,da,nl,fr,de,it,es,sv,en-US
```

- URLs for the Administration and Preferences Tabs

  Components of the Thin Client application are integrated by a set of tabs visible on every page, enabling easy navigation from one component to another. The following define URLs that tie together parts of the Thin Client applications:

```
client.admintab.url=/um/DomainManagement.jsp
client.preferencestab.url=/um/PFAccountBasicSettings.jsp
client.admin.url=/um/DomainManagement.jsp
client.preferences.url=/um/PFAccountBasicSettings.jsp
client.corporate.url=/um/traffic_cop
client.product.url=http://www.oracle.com
client.portal.url=http://www.oracle.com
client.privacystatement.URL=
```

- Images

```
client.image.corporate=/um/images/corporateBrand_oracle.gif
client.image.product=/um/images/branding_collaborationsuite.gif
client.image.portal=/um/images/globalbutton_returntoportal.gif
client.image.login=/um/images/globalbutton_login.gif
client.image.logout=/um/images/globallogout.gif
client.image.preferences=/um/images/globalpreferences.gif
client.image.help=/um/images/globalhelp.gif
```

Thin Client Log Files

The Thin Client log files can be found in the following directory:

```
toolkit.logdirectory=%ORACLE_HOME%/um/log
toolkit.logfilename=Webmail_Client
toolkit.loghostclient=%machinehost%
toolkit.loglevel=error
toolkit.debugmode=false
```

5

Error Messages

This chapter includes component-specific errors, listed in numerical order. The error codes are divided into the following groups:

- Overview
- IMAP4 and POP3
- SMTP
- Housekeeping
- List Server
- Thin Client

Overview

Error messages may appear in any part of Oracle Email. Users may see them in the end-user interface, and administrators may see them in the administrative tools and process logs.

Sometimes, more than one error is displayed. A list of error messages is called an error stack. The bottommost error in the stack is typically the cause of the error.

> **Note:** The error stack may contain error messages from other Oracle products that Oracle Email uses. When these additional errors appear, refer to the documentation for the given product.

IMAP4 and POP3

The following is a list of IMAP4 and POP3 error messages:

101, 0, Login failed
Cause: Invalid user name or password used for `LOGIN` command.

Action: Check the user name, password and try again.

102, 0, No of auth/login tries exceeded. Exiting
Cause: Used all your allowed login attempts

Action: Check the user name and password, then retry in a new session.

103, 0, User logged out
Cause: IMAP/POP session ended either by `LOGOUT/QUIT` command or because of some other fatal server error like unable to read or write to client connection anymore.

Action: Session end by `LOGOUT/QUIT` command is normal. If you suspect an abnormal connection termination, check for other errors in this error chain in server log file.

104, 0, Authorization succeeded
Cause: Successful login via authenticate command

Action: None

105, 0, Authorization failed
Cause: Unsuccessful login attempt via authenticate command

Action: Check the user credentials and try again.

106, 0, Could not retrieve folder id for folder={sarg0}. Error#{narg0}
Cause: Possibly a non-existent folder name was used.

Action: Correct the folder name and try again. If folder name is correct, check and resolve any other database errors in this error chain.

107, 0, Failed to get header info for folder={sarg0} with fid={narg1}. Error#{narg0}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_FOLDER_API is loaded. Check and resolve any other database errors in this error chain.

108, 0, Failed to update folder={sarg0} with fid={narg1}. Error#{narg0}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_FOLDER_API is loaded. Check and resolve any other database errors in this error chain.

109, 0, Failed to connect to database {sarg1}. Error#{narg0}
Cause: Server unable to create OCI connection pool.

Action: Make sure database is up and configured correctly in Oracle Internet Directory.

110, 0, Conntected to database {sarg1}
Cause: Successful connections to the database

Action: None

111, 0, Failed to get statement handle {narg1} with Error#{narg0}.
Cause: Database related error

Action: Check for OCI error in this error chain

112, 0, Autologout: idle {narg0} minutes.
Cause: Your session was idle for too long

Action: Send noop or any other command before timeout.

113, 0, Out of free Memory. Requested {narg0} bytes.
Cause: No more free memory is available to server.

Action: Reduce the load on server by reducing any of following: threads, max. clients, OCI sessions or Oracle Internet Directory connections. Make sure enough free memory is available for server on your system.

114,0, Module {sarg0}: nesting level too deep, no stats
Cause: Internal error

Action: Contact customer support

117, 0, Failed to get body parts for messageID={narg0}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_FOLDER_API is loaded. Check and resolve any other database errors in this error chain.

118, 0, Failed to get database session for db={sarg0}. Error#{narg0}
Cause: There are no more free sessions available in OCI connection pool.

Action: This may be a temporary error due to a spike in load. You may need to reevaluate your system to reduce the number of clients connecting to this database, increase the number of sessions in pool or tune the system in general to get faster response.

119, 0, Failed to insert subscribed folder={sarg0}. Error #{narg0}
Cause: Database error.

Action: Check the OCI errors in this error chain

120, 0, Failed to rename folder={sarg0})to {sarg1}. Error#{narg0}
Cause: Trying to rename a non-existent folder, or the new name is already in use or not allowed.

Action: Make sure folder with old name exists and new name is not already in use or contains restricted characters. Check for any other database errors in this error chain.

121, 0, Failed to set SEEN flag for msgid={narg0} in fid={narg1}. Error#{narg2}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_FOLDER_API is loaded. Check and resolve any other database errors in this error chain.

122, 0, Failed to get shell for msgid={narg0}. Error#{narg1}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_FOLDER_API is loaded. Check and resolve any other database errors in this error chain.

123, 0, Failed to create hierarchical folders {sarg0}. Error#{narg0}
Cause:

- You cannot create INBOX in any case insensitive form.
- You may be trying to create a folder which already exists.

Action: Check the folder name you are trying to create. Also check for any OCI errors in this error chain.

124, 0, Failed to expunge {narg0} msgs from folder with fid={narg1}. Error#{narg2}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if `ES_FOLDER_API` is loaded. Check and resolve any other database errors in this error chain.

125, 0, Bad flags list
Cause: Syntax error in flag list for Store command.

Action: Correct the syntax for flag list.

126, 0, Failed to get folder Id for folder={sarg0}. Error#{narg0}
Cause: Possible causes are:

- You may be looking for a non-existent folder
- For a shared folder you may not have read permissions on folder.

Action: Make sure you are looking for the right folder and is spelled correctly. If it is a shared folder, check its configuration and permissions in Oracle Internet Directory. Check and resolve any other database errors in this error chain.

117, 0, Failed to create shared folder={sarg0}. Error#{narg0},{sarg1}
Cause: Database error.

Action: Check and resolve database errors in this chain.

128, 0, Failed to delete shared folder={sarg0}. Error#{narg0},{sarg1}
Cause: Possible causes are:

- You may be trying to delete a non-existent folder.
- Only the shared folder owner can delete the shared folder.

Action: Check the name of the folder and make sure you are the owner of the shared folder you are trying to delete. Check for database errors in this error chain.

129, 0, Failed to rename shared folder={sarg0} to {sarg1}. Error#{narg0},{sarg2}
Cause: Possible causes are:

- You may be trying to rename a non-existent folder.
- Only shared folder owner can rename it.
- New name is already in use or not allowed.

Action: Make sure you are the owner of shared folder or retry with a different name.

130, 0, Failed to change ACI on shared folder={sarg0}. Error#{narg0},{sarg1}
Cause: Database error.

Action: Check the database and Oracle Internet Directory error logs.

131, 0, Failed to determine if this folder or any child is shared.{sarg0}. Error#{narg0}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if `ES_FOLDER_API` is loaded. Check and resolve any other database errors in this error chain.

132, 0, Failed to determine Folder space usage for user={sarg0}. Error#{narg0}
Cause: This could be due to an OCI error.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if `ES_FOLDER_API` is loaded. Check and resolve any other database errors in this error chain.

133, 0, Bad message in Folder={narg0},mid={narg1},muid={narg2}. Null value for {sarg0}
Cause: One of the required message attributes is missing in database.

Action: Make sure all required packages are loaded in the database correctly.

SMTP

The following is a list of SMTP server error messages:

100, 0, Memory allocation failed
Cause: The process is consuming too much memory.

Action: Reduce the number of threads running and restart the process.

101, 0, Memory realloc failed
Cause: The process is consuming too much memory.

Action: Reduce the number of threads running and restart the process.

103, 0, failed to create thread
Cause: There are too many threads in the process.

Action: Reduce the number of threads and restart the server. If the problem persists, contact technical support.

175, 0, ESDSGetEntry failed {sarg0}
Cause: The Oracle Internet Directory server may be down.

Action: Restart the Oracle Internet Directory server. If the problem still exists contact technical support.

176, 0, ESDSGetEntry for entrytype failed {sarg0}
Cause: The Oracle Internet Directory server may be down.

Action: Restart the Oracle Internet Directory server. If the problem still exists contact technical support.

177, 0, ESDSGetAttribute failed for {sarg0}
Cause: The Oracle Internet Directory server may be down.

Action: Restart the Oracle Internet Directory server. If the problem still exists contact technical support.

200, 0, loop detected for the recipient: {sarg0}
Cause: The address resolution for the recipient resulted in a loop.

Action: Make sure the data present in the Oracle Internet Directory server does not introduce any loops for the recipient. Check if auto forward attribute for the recipient introduces a chain ending with original recipient.

201, 0, orclobjectid not populated in Oracle Internet Directory for usr: {sarg0}
Cause: Mandatory attribute `orclobjectid` is missing in Oracle Internet Directory.

Action: Populate correct value for the user in Oracle Internet Directory.

205, 0, failed to deliver to user inbox: {sarg0}
Action: Make sure all required packages are loaded in the database correctly. Check if `ES_MESSAGE_API` is loaded.

208, 0, failed to index msg for user: {sarg0} index type: {sarg1}
Action: Make sure all required packages are loaded in the database correctly. Check if `ES_OT_API` is loaded.

209, 0, message rejected by rules for usr: {sarg0}
Cause: The user rule resulted in rejection of the message.

Action: None.

210, 0, message rejected by the recipient {sarg0} using replymode: reject
Cause: Auto reject is set in Oracle Internet Directory entry for the recipient.

212, 0, failed to delete local recipients
Cause: There may be OCI errors.

Action: Make sure all required packages are loaded in the database correctly. Check if `ES_MESSAGE_API` is loaded.

213, 0, local delivery failed for user: {sarg0}
Action: Check the log for exact reason for failure prior to this message, and see any correction for the user's setup is needed.

225, 0, failed to pickup unprocessed messages
Cause: Error in recovery processing.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_QUEUE_API is loaded.

226, 0, failed to requeue messages
Cause: Error in recovery processing.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_QUEUE_API is loaded.

243, 0, path for external filter process is NULL in Oracle Internet Directory
Action: Populate `orclmailsmtpexternalfilterprocess` with path for virus scanner executable if virus scanning is enabled.

302, 0, User {sarg0} logon failed. Oracle Internet Directory returns {narg0}
Cause: Unable to authenticate user in Oracle Internet Directory.

Action: Check user name and password to see if they are correct.

401, 0, Error {narg0}: Unable to get msgid
Cause: Unable to get next message ID from database.

Action: Check if the schema is installed and if the package is valid.

402, 0, Error {narg0}: Unable to store envelope
Cause: Unable to insert envelope information into database.

Action: Check if the schema is installed and if the package is valid.

403, 0, Error {narg0}: Unable to store recipient
Cause: Unable to insert recipient information into database.

Action: Check if the schema is installed and if the package is valid.

404, 0, Error {narg0}: Unable to store {sarg0} queue
Cause: Unable to insert the message into a queue.

Action: Check if the schema is installed and if the package is valid.

405, 0, Error {narg0}: Unable to insert the message
Cause: Unable to insert message into database.

Action: Check the OCI error and ORACLE error.

406, 0, Error: Routing loop detected
Cause: Message may be in a loop by checking the Received: headers.

Possible causes: Loop in address rewriting rules; Auto-forward between addresses;.forward set up by UNIX mail senders.

Action: Check the rewriting rules and auto-forward setup and notify the sender.

407, 0, Error: Unable to read from client
Cause: Unable to read from client.

Action: Check network connections.

500, 0, spam check failed for IP address: {sarg0}
Cause: DNS server failed to verify that the IP address of the SMTP client is correct.

501, 0, spam check failed for host: {sarg0}
Cause: DNS server failed to verify that the host is a valid internet host.

502, 0, spam check failed for sender: {sarg0}
Cause: The sender is either in the list of rejected senders or rejected domains.

503, 0, spam check failed for recipient: {sarg0}
Cause: This could be due to either relay is not allowed for the non-local recipient's domain OR the non-local recipient is in the list of rejected recipients.

650, 0, failed to get submit recipients
Cause: This could be due to OCI errors.

651, 0, failed to delete submit recipients
Cause: This could be due to OCI errors.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_MESSAGE_API is loaded.

652, 0, failed to insert resolved recipients
Cause: This could be due to OCI errors.

Action: Make sure all required packages are loaded in the database correctly. In particular, check if ES_MESSAGE_API is loaded.

Housekeeping

The following is a list of housekeeping error messages:

Oracle error {sarg0} occurred during expiration
Cause: An RDBMS error prevented house keeper from successfully performing expiration.

Action: Correct the generic RDBMS error and try running house keeper again.

Oracle error {sarg0} occurred during queue pruning
Cause: An RDBMS error prevented house keeper from successfully performing pruning.

Action: Correct the generic RDBMS error and try running house keeper again.

Oracle error {sarg0} occurred during pruning
Cause: An RDBMS error prevented house keeper from successfully performing pruning.

Action: Correct the generic RDBMS error and try running house keeper again.

Oracle error {sarg0} occurred during collection
Cause: An RDBMS error prevented house keeper from successfully performing collection.

Action: Correct the generic RDBMS error and try running house keeper again.

Oracle error {sarg0} occurred during tertiary storing
Cause: An RDBMS error prevented house keeper from successfully performing tertiary storage.

Action: Correct the generic RDBMS error and try running house keeper again.

List Server

The following is a list of list server error messages:

Msg-id: 5002(An error occurred while performing a database operation. Error= {sarg0})
Cause: The cause for this error is available in the error message itself.

Action: Look at the oerr error for the error specified in the error message.

Msg-id: 5003(Error occurred while connecting to the Oracle Internet Directory server on {sarg0}port {narg0} bind dn {sarg0})
Cause: The Oracle Internet Directory server is down or has stopped responding or is listening on a different port.

Action: Restart the Oracle Internet Directory server if it is not running. Otherwise, restart the list server and specify the correct host name and port number of the Oracle Internet Directory server.

Msg -id: 5004(Error initializing process control)
Cause: Either the database or the Oracle Internet Directory server is not running or has stopped responding.

Action: Restart the database and Oracle Internet Directory server. If they are running, then restart the list server.

Msg-id: 5021(Error modifying user {sarg0} entry. Error = {narg0})
Cause: An Oracle Internet Directory error occurred while trying to process a command for the user.

Action: Check if the user entry on the Oracle Internet Directory server is still valid.

Thin Client

The following is a list of Thin Client error messages:

An error occurred while adding attachments
Cause: WebMail was unable to add the attachments.

Action: Try again.

No folder name was specified
Cause: The user did not specific a folder name.

Action: Enter a folder name.

An error occurred; unable to create the new folder
Cause: WebMail was unable to create the folder.

Action: Try again.

A folder by that name `<foldername here>` already exists
Cause: The user specified a folder name that is being used by another folder.

Action: Name the folder with a new name or put the folder in a different location.

An error occurred while creating the message
Cause: WebMail could not create a new message object.

Action: Try creating a message again.

No valid To: recipients found
Cause: User did not specify a valid e-mail address in the **To** field.

Action: Try again and specify a valid e-mail address.

Error occurred during message creation
Cause: WebMail could not create a new message object.

Action: Try creating a message again.

Invalid parameter specified for attachment removal
Cause: WebMail experienced a problem when removing the attachment.

Action: Contact your system administrator.

Invalid attachment index was received
Cause: WebMail attachment indices are misaligned.

Action: Recreate the message.

No message IDs were specified for deletion
Cause: The user did not select messages for deletion.

Action: Select the message for deletion.

An error occurred during message deletion
Cause: The message does not exist.

Action: Contact your system administrator.

An error occurred while compacting the folder
Cause: This is a problem with the voice mail messages in the folder.

Action: Contact your system administrator.

No message IDs were specified for forwarding
Cause: The user did not select a message before selecting **Forward**.

Action: Select a message before selecting **Forward**.

More than one message specified for forwarding
Cause: Multiple messages were selected for forwarding.

Action: Select one message at a time for forwarding.

Invalid message specified
Cause: The message selected could not be forwarded.

Action: Try selecting another message, if that does not work, contact your system administrator.

An error occurred while preparing the message for forwarding
Cause: The selected message could not be processed for forwarding.

Action: Try again or contact your system administrator.

The destination folder does not exist
Cause: The destination folder selected does not exist.

Action: Select another destination folder.

No message IDs were specified for move
Cause: The user did not select a message before selecting **Move**.

Action: Select a message before selecting **Move**.

An error occurred while performing message move
Cause: WebMail could not process the move request.

Action: Try again or contact your system administrator.

There are no more messages in this folder
Cause: No messages exist before of after the current message.

Action: Try another folder.

An error occurred opening the next message
Cause: WebMail could not open the next message.

Action: Try again or contact your system administrator.

There are no messages before this one in this folder
Cause: No messages exist before of after the current message.

Action: Try another folder.

An error occurred opening the previous message
Cause: WebMail could not open the previous message.

Action: Try again or contact your system administrator.

Unable to find folder
Cause: The folder is not accessible.

Action: Check the shared permission or contact your system administrator.

Folder does not exist
Cause: There is no such folder in the account.

Action: Contact your system administrator.

An error occurred while opening the folder
Cause: WebMail experienced problems opening the folder.

Action: Contact your system administrator.

Error occurred during communication with the message store
Cause: Possibly a network problem.

Action: Contact your system administrator.

No message ID specified
Cause: Internal error.

Action: Contact your system administrator.

Error retrieving message
Cause: The message may have been deleted, but the browser is looking a cached or old pages.

Action: Refresh the message list and try again.

No message IDs were specified for reply
Cause: The user did not check any messages before selecting **Reply.**

Action: Select a message before selecting **Reply.**

More than one message specified for reply
Cause: The user selected multiple messages for reply.

Action: Select only one message at a time.

Invalid message specified
Cause: WebMail could not process the message for reply.

Action: Try again or contact your system administrator.

Error occurred while preparing the message for reply
Cause: Internal error

Action: Contact your system administrator

Error while sending message
Cause: Internal error

Action: Contact your system administrator.

No folder was specified for editing
Cause: The user did not select a folder in the folder list before selecting **Edit**.

Action: Select a folder from the folder list.

The specified folder does not exist in the mail store
Cause: The folder selected is not available.

Action: Verify that the folder exists, or contact you system administrator.

An error occurred while preparing the folder for editing
Cause: Internal Error

Action: Contact your system administrator.

You cannot rename special system folders
Cause: The user tried to rename the inbox.

Action: None, the inbox cannot be renamed.

No new name was specified
Cause: The user did not specify a name for the folder.

Action: Specify a name for the folder.

A folder with that name already exists
Cause: Internal error

Action: Contact your stem administrator.

Unable to rename folder
Cause: Internal error

Action: Contact your stem administrator.

An error occurred while trying to update the folder
Cause: Internal error

Action: Contact your stem administrator.

Error while setting previous state
Cause: Internal error

Action: Contact your stem administrator.

You are no longer connected to the mail store
Cause: The session has timed out.

Action: Contact your stem administrator.

6

Command Line Interface

This chapter contains general instructions on how to use the command-line interface. It also contains an entry for each command available in the command-line interface. Each command is followed by a brief description of its purpose. In addition, the proper syntax, keywords, and command parameters are provided.

This chapter contains the following topics:

- OESCTL
- OESMON
- OESUCR
- OESDL
- OESRL
- OESCHART

OESCTL

The `oesctl` command enables an Oracle Email administrator to perform some configuration and control operations on Oracle Email services.

This command is used from within a command shell, such as `/bin/csh` on Unix systems, and provides a subset of the functionality available on Oracle Enterprise Manager pages for Oracle Email. For example, `oesctl` can be used by an administrator to start an Oracle Email IMAP4 server, but it cannot be used to modify IMAP service parameters.

Getting Usage Information

Without arguments, `oesctl` prints out the following usage information:

```
% oesctl
oesctl  [ [<command>] [<target>|<instance>] ]
```

Where commands can be any of the following:

Table 6–1

Command	Description
startup	Starts all the processes associated with the target or instance.
shutdown	Shuts down all the processes associated with the target or instance.
create instance	Creates an instance on a target.
delete instance	Creates an instance on a target.
refresh	Causes the target or instance to reload parameters from Oracle Internet Directory.
show targets	Displays a list of possible targets.
show status	Displays the status of the target.
show processes	Displays the status of the processes associated with the target.

OESCTL Syntax

The syntax of `<target>` is `<host>:<installation>:<service>`

`<host>` is the host name of the computer on which server processes run

`<installation>` is always `um_system` for the this release

<service> is one of the following: gc, list, smtp_in, smtp_out, imap, pop

The syntax of <instance> is <target>:<instance_id>

The meaning of the different service names are:

- gc: housekeeper service
- list: secure list service
- smtp_in: inbound SMTP service
- smtp_out: outbound SMTP service
- imap: IMAP service
- pop: POP service

<instance_id> is a number assigned to an instance when it is created. These numbers are selected automatically at instance creation time. Instance numbers cannot be configured by administrators.

Examples

The following examples are executed from a command shell running on a host named mail server.

OESCTL Configuration Operations

The configuration operations query or update the current configuration.

The query operations are:

```
% oesctl show targets
% oesctl show processes <target>
% oesctl create instance <target>
% oesctl delete instance <target>
```

Getting the List of Available Targets

```
% oesctl show targets
TARGET: mailserver:um_system:gc
TARGET: mailserver:um_system:imap
TARGET: mailserver:um_system:list
TARGET: mailserver:um_system:pop
TARGET: mailserver:um_system:smtp_in
TARGET: mailserver:um_system:smtp_out
```

Getting the List of Process Instances for a Target

In the following examples, there is one process instance configured for the IMAP service running on the host mail server, and there are no process instances for the POP service. A service must have at least one process instance before it can be started. From the above example we know that the POP service cannot be started on the host mail server.

```
% oesctl show processes mailserver:um_system:imap
mailserver:um_system:imap:101771055406040653

% oesctl show processes mailserver:um_system:pop
No processes for mailserver:um_system:pop
```

Creating a Process Instance

```
% oesctl show processes mailserver:um_system:gc
No processes for mailserver:um_system:gc

% oesctl create instance mailserver:um_system:gc
Succesfully created a new instance for a total of: 1

% oesctl show processes mailserver:um_system:gc
mailserver:um_system:gc:101778964029981136
```

The list of process instances for the target `mailserver:um_system:gc` was checked prior to instance creation, and it was empty. The create command was used to create a new process instance for the target, after which the process instance list was checked again and found to contain the new instance.

Deleting a Process Instance

```
% oesctl show processes mailserver:um_system:gc
mailserver:um_system:gc:101778964029981136

% oesctl delete instance mailserver:um_system:gc
Succesfully deleted an instance for a total of: 0

% oesctl show processes mailserver:um_system:gc
No processes for mailserver:um_system:gc
```

The list of process instances for the target mail server: `um_sytem`: `gc` was checked prior to instance deletion. The delete command was used to delete the process instance found, after which the process instance list was checked again and found to contain no processes.

OESCTL Control Operations

The control operations display or alter the operational state of targets and instances.

The control operations are:

```
% oesctl show status <target>
% oesctl startup <target>
% oesctl startup <instance>
% oesctl shutdown <target>
% oesctl shutdown <instance>
% oesctl refresh <target>
% oesctl refresh <instance>
```

Starting and Stopping a Target

```
% oesctl show processes mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257
mailserver:um_system:gc:101779029537864556

% oesctl show status mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 <stopped>
mailserver:um_system:gc:101779029537864556 <stopped>

% oesctl startup mailserver:um_system:gc
ok
ok

% oesctl show status mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 is alive. Message from console: null
mailserver:um_system:gc:101779029537864556 is alive. Message from console: null

% oesctl shutdown mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 Housekeeper is terminated.  Message
from console: null
mailserver:um_system:gc:101779029537864556 Housekeeper is terminated.  Message
from console: null
% oesctl shutdown mailserver:um_system:gc
No processes configured to be running for mailserver:um_system:gc
```

If `oesctl` is used to start a target, each configured process instance is started.

Starting and Stopping an Instance

```
% oesctl startup mailserver:um_system:gc:101779027179112257
ok
```

```
% oesctl show status mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 is alive. Message from console: null
mailserver:um_system:gc:101779029537864556 <stopped>

% oesctl shutdown mailserver:um_system:gc:101779027179112257
ok:Housekeeper is terminated.  Message from console: null
```

In some situations administrators may want to start or stop only a particular process instances. In this case, `oesctl startup <instance>` and `oesctl shutdown <instance>` are used.

Refreshing Targets and Instances

```
% oesctl refresh mailserver:um_system:gc:101779027179112257
ok:is refreshed. Message from console: null

% oesctl refresh mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 is refreshed.  Message from console:
null
mailserver:um_system:gc:101779029537864556 is refreshed.  Message from console:
null
```

Refreshing a process instance sends the instance a message to reload its process parameters from Oracle Internet Directory.

Refreshing a service target refreshes each started process instance.

The refresh functionality can be used to change a process parameter and have the change take effect without having to stop and restart running processes. For example, IMAP service log level can be changed in Oracle Internet Directory and refreshed without disconnecting any users that are currently connected to the IMAP service. Executing a shut down followed by a startup changes the logging behavior, and temporarily disconnect users.

OESMON

The `oesmon` command enables customers to obtain raw metric data from the Oracle Email server processes. The output of the `oesmon` command uses ASCII characters.

A metric is a string or a number. Every metric is associated with a managed object. Managed objects are associated with other managed objects in a parent-child relationship, forming a hierarchical tree structure of managed objects and metrics. The metrics are always leaves of the tree.

A numeric metric is a gauge or a counter. A gauge measures the current amount of an object and is characterized by a value going up and down. A counter measures an accumulated value and is characterized by the value remaining the same or becoming larger. If the value of a counter goes past the maximum supported number, the value wraps around to 0.

Handling of Units

The integer values tracked by numeric metrics measure quantities in some unit of measure. The unit of measure for a given metric is not maintained internally and is not printed out by the `oesmon` command. Instead, the units are defined by the server-specific product documentation.

> **See Also:** Appendix A, "Server Statistics" for more information on server-specific statistics

Metric Names

All metrics and all managed objects have names that are case-sensitive and contain only alphanumeric characters, including the underscore character. A name cannot contain spaces or dot characters. The full name of object O is formed by connecting all the names along the path from the tree root to O. In this case, O may be a metric or a managed object.

For example:

`.MTA` is the full name of a managed object named MTA. It is at located directly under the root of the tree.

`.MTA.connections` is the full name of a managed object named connections that is a child of the MTA object.

`.MTA.connections.out.current` is the full name of a numeric metric that tracks the number of currently active outbound SMTP connections.

`.MTA.connections.out.total` is the full name of a metric that tracks the number of outbound SMTP connections created since startup.

Examples

Getting the Usage Message

Running `oesmon` without any arguments, it shows the command type:

Output

```
% oesmon
Usage: oesmon targets | names <target> | get <target> <name>
```

Getting the List of Available Service Targets

Command

```
oesmon targets
```

Purpose

Shows all possible service targets that can be polled with oesmon:

Output:

```
% oesmon targets
TARGET: mailserver:um_system:gc
TARGET: mailserver:um_system:imap
TARGET: mailserver:um_system:list
TARGET: mailserver:um_system:pop
TARGET: mailserver:um_system:smtp_in
TARGET: mailserver:um_system:smtp_out
```

Showing Metric Names for a Given Service Target

Command

```
oesmon names mailserver:um_system:smtp_in
```

Purpose

This command queries each process instance for the metric names it currently has defined.

Output

```
% oesmon names mailserver:um_system:smtp_in
.DUMP.OIDStatus.Connection
.DUMP.Threads.dump
```

```
.ES_SPS.socket.currload
.ES_SPS.socket.sockmax
.ES_SPS.thread.currthreads
.ES_SPS.thread.thrmax
.MTA.uptime
.MTA.connections.in.current
.MTA.connections.in.total
.MTA.msgs.deferred.current
.MTA.msgs.deferred.total
.MTA.receive.kbytes
.MTA.receive.messages
.MTA.receive.recipients
.MTA.receive.time
.MTA.transmit.bytes
.MTA.transmit.bytes_local
.MTA.transmit.messages
.MTA.transmit.messages_local
.MTA.transmit.recipients
.MTA.transmit.recipients_local
.um.admin.os_pid
.um.admin.uptime
```

The `oesmon names mailserver:um_system:imap` command contacts each process instance that belongs to the service `mailserver:um_system:imap` and determines which metrics are currently defined for the process.

Many metrics are defined as soon as the process starts up, but some are created dynamically during operation, and are not available at all times. Therefore, the output from using the `oesmon names` command does not always give the same list of metric names. For example, an IMAP server process does not have any metrics available about particular user until that user logs in at least one time.

Querying a Metric Value from a Running Server

Command

```
oesmon get mailserver:um_system:smtp_
in.MTA.connections.in.total
```

Purpose

This command queries each SMTP in-bound process to find out how many connections it has accepted.

Output

```
% oesmon get mailserver:um_system:smtp_in .MTA.connections.in.total
```

```
.MTA.connections.in.total = 352
.MTA.connections.in.total = 0
```

The output can determine that the `smpt_in service` for `mailserver: um_system` has two process instances configured and running. The first process has received 352 connections and the second has not received any.

The metric given to the command was the complete name of a single metric. It is possible to retrieve values for all metrics associated with a managed object, as is shown in the following example.

Query Multiple Related Metrics for a Managed Object

Command

```
oesmon get mailserver:um_system:smtp_in .MTA.transmit
```

Purpose

This command queries in-bound SMTP processes to find out transmission metrics.

Output

```
% oesmon get mailserver:um_system:smtp_in .MTA.transmit
.MTA.transmit.bytes = 3282806
.MTA.transmit.bytes_local = 3282806
.MTA.transmit.messages = 330
.MTA.transmit.messages_local = 330
.MTA.transmit.recipients = 698
.MTA.transmit.recipients_local = 698
.MTA.transmit: metric not found
```

The name `.MTA.transmit` is the name of a managed object, not a metric. In this case, oesmon returns all metrics and children managed objects associated with `.MTA.transmit`.

The output can determine that the `smpt_in service` for mail server: `um_system` has two process instances configured and running. The first process has transmitted a number of e-mail messages, and the second process has not transmitted any.

Error Output

The `oesmon` command returns error results if a metric cannot be found or if a process instance cannot be contacted.

Example of output for undefined metric:

```
% oesmon get mailserver:um_system:smtp_in .nosuchmetric
```

```
.nosuchmetric: metric not found
```

Example of output for process that does not respond:

```
% oesmon get mailserver:um_system:smtp_in .MTA.connections.in.total
<no response>
```

OESUCR

The `oesucr` command does the following:

- Creates and deletes Oracle Email users
- Changes e-mail addresses
- Specifies a real domain for users
- Supports different character encoding types
- Creates users directly through command line

`OESUCR` takes a file as an input. For user creation, the file should contain a list of records, separated by an empty line. Each record contains information above a user to be created. Each line in a record is a name-value pair for an attribute for the e-mail user in the directory. Each record must have at least three mandatory attributes

- `mail`
- `orclmailquota`
- `baseuserdn`

For user deletion, the file should contain one line listing all the users to be deleted, separated by a comma.

This command only creates and deletes e-mail users, not corresponding public users. For user creation, the public users must exist prior to creating the corresponding e-mail users. For user deletion, after running the tool, the users are no longer valid e-mail users, but they are still users in the directory.

Usage

```
% oesucr <file> [-v] [-d]
```

`<file>` is the path to the file containing the user records of the users to be created or the list of users to be deleted.

The -v flag prints out debug messages. The -d flag deletes users. -v and -d can be used together.

oesucr <filename> -change

<filename> is a text file with the following format:

```
<old email address1>=<new email address1>
```

For example:

```
user1@us.oracle.com=newuser1@us.oracle.com
```

After running the command, user1@acme.com becomes newuser1@acme.com.

esucr <filename> -encoding=UTF-8

The file is read as UTF-8. The file must be saved in the corresponding encoding.

Examples

Creating Users

The example file user_file contains the following records:

```
mail=testuser1@us.oracle.com
orclmailquota=400000000
baseuserdn=cn=testuser1,cn=users,o=oracle,dc=com

mail=testuser2@us.oracle.com
orclmailquota=400000000
baseuserdn=cn=testuser2,cn=users,o=oracle,dc=com
```

Running the % oesucr user_file creates two e-mail users called testuser1 and testuser2. Each record in the file contains only the three mandatory attributes.

> **Note:** The corresponding public users must exist before running the OESUCR.

Creating Users with Optional Attributes

For a file user_file containing the following records:

```
mail=testuser1@us.oracle.com
```

```
orclmailquota=400000000
userpassword=welcome
orclMailDomainControlAci=domain

mail=testuser2@us.oracle.com
orclmailquota=400000000
baseuserdn=cn=testuser2,cn=users,o=oracle,dc=com
```

Running % `oesucr user_file` creates two-e-mail users called `testuser1` and `testuser2`. The role of the first user is set to domain administrator.

Deleting Users

The example file `user_file` contains the following line:

```
mail=testuser1@us.oracle.com,testuser2@oracle.com,testuser3@oracle.com
```

Running % `oesucr user_file -d` deletes the e-mail users: `testuser1@us.oracle.com`, `testuser2@oracle.com`, and `testuser3@oracle.com`.

> **Note:** Corresponding public users are not deleted by `OESUCR`

Creating a User Through Command Line

The following is an example of how to create a user through command line, with out creating a new file. Only one user can be created at a time:

```
oesucr -cmd mail=user1@acme.com
baseuserdn=cn=user1,cn=users,dc=us,dc=acme,dc=com orclmailquota=400000000 <other
optional attributes>
```

All parameters are separated by a space and have the same names as those used in the file. All mandatory attributes must be specified, and can take any valid optional attributes.

Specifying a Real Domain for Users

The following is an example of how to specify a real domain for users:

```
mail=user1@company1.com
realdomain=acme.com
baseuserdn=......
orclmailquota=......
```

The e-mail address of the user becomes `user1@company1.com`. However, in Oracle Internet Directory, `company1.com` may not exist, because the user is created under acme.com

OESDL

`OESDL` is the command line tool for adding users to and removing users from distribution lists.

The `oesdl` tool takes a file as an input. The file should contain a list of records, separated by an empty line. Each record contains information to manipulate one distribution list. Each record must have the name of the list and a list of users.

For adding users to a list, the user type must be indicated. A regular user , a distribution list, an alias, or a foreign user to a distribution list can be added.

When adding users to a list, you can create the list at the same time if it does not exist. To create a new list, the owner must be specified.

Usage

```
% oesdl <file> [-v]
```

`<file>` is the path to `[-v]` the file containing the list records.

The `-v` flag prints out debug messages.

Examples

Adding Users to a Lists

The example file `list_file` contain the following records:

```
listname=list1@oracle.com
action=add
newlist=n
usertype=U
users=user1@oracle.com,user2@oracle.com,user3@oracle.com

listname=list2@oracle.com
action=add
newlist=n
usertype=L
users=list1@oracle.com
```

Running % oesdl list_file adds user1, user2, and user3 to list1@oracle.com, list1 must already exist. It also adds list1@oracle.com to another list called list2@oracle.com.

The usertype can be one of the following:

- U for regular user
- F for foreign user
- L for a distribution list
- A for an alias

Adding Users to a New List

The example file list_file contains the following records:

```
listname=list1@oracle.com
action=add
newlist=y
owner=user1@oracle.com
usertype=U
users=user1@oracle.com,user3@oracle.com
```

Running % oesdl list_file, creates a new list called list1@oracle.com, set its owner to user1@oracle.com, and then adds users: user1@oracle.com, and user3@oracle.com to the new list.

Removing Users from a Distribution List

The example file user_file contains the following lines:

```
listname=list1@oracle.com
action=delete
usertype=U
users=user1@oracle.com,user2@oracle.com

listname=list2@oracle.com
action=add
newlist=y
owner=user1@oracle.com
usertype=U
users=user1@oracle.com,user2@oracle.com
```

Running % oesdl list_file removes user1 and user2 from list1@oracle.com. It then creates a new list called list2, sets the owner

user1@oracle.com, and then add user1, and user2 to the new list list2@oracle.com.

OESRL

The oesrl command enables administrators to create and manage server side rules from a command line.

oesrl can create server side rules specified in a text file or list server side rules to the standard output. When creating rules, two formats of text file are accepted, the Java properties file format and the XML format. When listing rules, only the XML format is listed.

Usage

```
% oesrl
Usage: oesrl [-c <file> | -x <file> | -p <ruleowner>]
```

-c <file>: create rules based on property <file>.

-x <file>: creates rules based on XML <file>.

-p <ruleowner>: prints <ruleowner> rules in XML.

File Formats

Property file

Property files are text files with name-value pairs. Names can be organized hierarchically and separated by periods. The following are the top-level property names used in the file:

- Ruleowner: The qualified name of the rule owner. If the rules are owned by a user, it is the user's e-mail address. If the rules are domain specific, the rule owner should be a domain name such as acme.com. If the rules are system wide, the rule owner is the system installation name stored in Oracle Internet Directory. One file can contain only rule owner.
- Ruletype: Describes the rule owner types. The rule types are:
  - User
  - Domain
  - System

- Debug: If this parameter is set to true, the `oesrl` utility prints out debugging messages. This property is optional.
- Event#: There are up to six distinct events that can be defined in this file. Each event can only appear once. The events defined should have a sequence number starting from 1, such as `event1`. The events are:
  - Relay
  - Reception
  - Deliver
  - Copy
  - Flag change
  - Expunge

Under each event, one can define unlimited number of rules using property name `<eventname>.rule#`, where `<eventname>` is one of the six events and # is a sequence number starting from 1. For example, the property `deliver.rule1` defines the name of the first rule under the deliver event. All attributes of this rule can be further defined under the prefix `deliver.rule1`.

Under each rule, one can define actions and their parameters. Often times rules come with conditions that need to be defined. The following is a list of property names corresponding to each rule attribute:

Note: # means it can be replaced by a sequence number starting from 1.

- `<eventname>.rule#.action#`: the sequence of actions that can be defined under `<eventname>.rule#`.

  The choices of values are listed in Oracle Email Java API documentation under Java class `CommandType`.
- `<eventname>.rule#.action#.param#`: the parameter sequence needed for the action `<eventname>.rule#.action#`.
- `<eventname>.rule#.active`: an optional property that can be set to true or false indicating whether <eventname>.rule# is active.
- `<eventname>.rule#.attr#, <eventname>.rule#.op#, <eventname>.rule#.operand#`: together they represent a condition

associated with the rule <eventname>.rule# as long the same sequence number is used.

The choices of values for <eventname>.rule#.attr# are listed in Oracle Email Java API documentation under Java class AttributeType. The choices of values for <eventname>.rule#.op# are listed in Oracle Email Java API documentation under Java class OperatorType.

- <eventname>.rule#.negate#: an optional property value that can be set to true or false, indicating whether the condition number should be negated
- <eventname>.rule#.param#: in case when <eventname>.rule#.attr# requires a parameter (such as xheader), use this property to specify the parameter value.
- <eventname>.rule#.case#: an optional property value that can be set to true or false, indicating whether <eventname>.rule#.op# is case sensitive .
- <eventname>.rule#.cond: if multiple conditions are needed, use this property to specify whether and or or should be used to combine the conditions.

  Oracle Corporation recommends listing rule properties in order, so that readability of the property file is maximized. When running the oesrl utility, listing rule properties in order is not required.

XML

XML is used as the storage format of server side rules. An XML rule representation can be created directly and oesrl can be used to load the rules into the system. The XML file specified needs to a valid XML file according to the rules XML schema. To obtain the XML schema for rules, extract the schema file oracle/mail/sdk/rule/mail_rule.xsd from the Java SDK library esmail_sdk.jar under $ORACLE_HOME/jlib.

Examples

Creating User Rules Using Property File Input

This example demonstrates how to use property files to specify rules for a user, and how to use the oesrl utility to save the rules.

```
% cat > rules.properties
ruleowner=user1@oracle.com
ruletype=user
event1=deliver
```

```
deliver.rule1=Moving private messages
deliver.rule1.cond=or
deliver.rule1.attr1=rfc822to
deliver.rule1.op1=contains
deliver.rule1.operand1=user1@oracle.com
deliver.rule1.attr2=rfc822cc
deliver.rule1.op2=contains
deliver.rule1.operand2=user1@oracle.com
deliver.rule1.action1=moveto
deliver.rule1.action1.param1=/user1/Private
^D

% oesrl –c rules.properties
```

Creating User Rules Using XML File Input

This example demonstrates how to use XML files to specify rules for a user, and how to use the `oesrl` utility to save the rules.

```
% cat > rules.xml
<account qualifiedName=user1@oracle.com>
 <rulelist event=deliver>
  <rule description=Moving private messages>
   <condition junction=or>
   <condition>
    <attribute tag=rfc822to/>
    <operator op=contains/>
    <operand>user1@oracle.com</operand>
   </condition>
   <condition>
    <attribute tag=rfc822cc/>
    <operator op=contains/>
    <operand>user1@oracle.com</operand>
   </condition>
   </condition>
   <action>
    <command tag=moveto/>
    <parameter>/user1/Private</parameter>
   </action>
  </rule>
 </rulelist>
</account>
^D

% oesrl –x rules.xml
```

Retrieving Rules

This example demonstrates how to use list rules for a user in XML format.

```
% oesrl -p user1@oracle.com
<account qualifiedName=user1@oracle.com>
 <rulelist event=deliver>
  <rule description=Moving private messages>
   <condition junction=or>
   <condition>
    <attribute tag=rfc822to/>
    <operator op=contains/>
    <operand>user1@oracle.com</operand>
   </condition>
   <condition>
    <attribute tag=rfc822cc/>
    <operator op=contains/>
    <operand>user1@oracle.com</operand>
   </condition>
   </condition>
   <action>
    <command tag=moveto/>
    <parameter>/user1/Private</parameter>
   </action>
  </rule>
 </rulelist>
</account>
```

OESCHART

OESCHART is a charting utility that enables administrators to create graphs illustrating the system's current load and performance. Upon invocation it reads the values necessary in the `.ini` file to connect to an Oracle database, query the mail statistics tables, and then creates an image file. Other metrics plug into one of the standard display types, such as showing the rate or current values. The images are only created when the program is run, therefore to have dynamically updated charts administrators need to schedule periodic invocations of the program, such as with `cron` on UNIX systems or `WinAT` for Windows or as a DBMS job.

Usage

```
oeschart <property file>
```

File Formats

Mechanism

When a mail process starts up and is retrieving runtime attributes from Oracle Internet Directory there is an attribute `<attribute name>` that specifies how often the process should place statistics in the repository database.

When the processes startup they initialize a number of counters to zero and update these internal counters every time an associated operation is performed. There is a separate thread which keeps track of these counters and inserts them into the statistics database at the time interval specified in the Oracle Internet Directory. All of the counters held in the e-mail server processes are started up and are not reset. Because the processes keep track only of counters, the memory over head is minimal.

To disable the insertion of statistics, the insert period should be set to zero seconds. For the processing servers, the collection of statistics is a relatively light operation in terms of CPU and memory consumption.

Schema

The data is held in a small number of tables, with one view that summerizes the data. Administrators of this component should query the tables directly after a few processes have entered their initial data. `OESChart` is only one mechanism of mining this data. For more specific information, administrators should query the schema.

There is no automated clean up functionality. The tables grow until historical folders are deleted or exported . It is important to occasionally monitor the `esperftbl` tablespace.

es_perf_process

Name	Value	Description
process_id	Number not null	An internal assigned unique number for each process.
process_dn	Varchar2 (500)	The dn of the process in Oracle Internet Directory

es_perf_metric

Name	Value	Description
metric_id	Not Null	An internal assigned unique number for each metric.
metric_name	Varchar2 (100)	The name of the metric.
metric_type	Not Null	

es_perf_timestamp

Name	Description
`timestamp_id`	An internal assigned unique number for each time period that a process inseted int statistics time stamp.
Date	The time the data was inserted.

es_perf_sample

Name	Value	Description
process_id	Number not null	S.A.
metric_id	Varchar2 (100)	S.A.
timestamp_id	Number not null	S.A.
nvalue	Number	The numberical value of the metric (if the metric is numeric)
Svalue	Varchar2 (1000)	The String value of the metric (if the metric is a string)

es_perf_data

Name	Value
es_perf_data	(view)
process_dn	Varchar2 (500)
metric_name	Varchar2 (100)
metric_type	Number not null
timestamp	Date

Name	Value
nvalue	Number not null
Svalue	Varchar2 (1000)

Graph Styles

The `oeschart` utility enables four types of graphs to bespecified , three are variations on a xy scatter graph and the fourth is a bar chart illustrating IMAP command statistics.

xy_current

This charts whatever current value was entered in the database, there is no data manipulation done on the entries. It should be used on metrics such as, .ES_ `SPS.socket.currload` which is a point in time metric.

xy_rate

This chart calculates the difference between two data points and divides the resulting value by the time differencee, which gives the rate of the transaction. All times are standardized on seconds. The graphed value becomes Y transactions per second, such as in the metric `.ESPROTO.transmit.bytes`. And by using the xy_ rate graph displays the outgoing number of transmitted bytes per second.

xy_cummulative

This chart illustrates the total amount of work performed in a given period and shows the cumulative counters for a metric, while taking into account process restarts. This only works on metrics that are cumulative counters. It does take into account additional processes that were added or deleted, and any process restarts.

Mandatory Entries

The `.ini` file specifies all the necessary information to create the charts. The following tables describe mandatatory entries and optional values:

Table 6–2

Parameter	Description
Server	The database host name containing the statistics
port	The listener port for that database

Table 6–2

Parameter	Description
sid	The sid or service name for that server.
username	Username to access the account
password	Password for that account
process_dn	The query used to gather statistics, such as `process_dn='%<value in ini file>%` used to retrieve all processes that follow this dn pattern. This enables administrators to graph a specific process, a set of processes, or the entire system by specifying the level of detail entered for this parameter.
metric_name	The metric to query.
graph_type	Possible values are xy and bar
Image_file_name	The name of the file to be generated
image_title	The title to display on the graph.
number_of_hours	Specifies the number of hours since the present to graph.

Optional Parameters

Table 6–3

Parameter	Default	Description
encode_type	gif	Possible values are gif & png
image_dir	./	
aggregate_ time_period	600	This is the time span in which multiple processes logging will be grouped together and the metrics combined to show the aggregate value. For example, suppose we had 2 IMAP servers running, IMAP1 & IMAP2. IMAP1 logged its statistics at 3:00pm and IMAP2 at 3:02 pm. (The servers log their statistics every x seconds (specified in ldap) relative to when they started, so in this case, IMAP2 must have been started 2 minutes after IMAP1) When showing the total number of sockets on the system we need to combine the values from IMAP1 and IMAP2. The aggregate_time_period metric defines what is an acceptable window for different processes statistics to be combined. This should be the same as the submit period specified in LDAP for this process type.
max_lifetime	300	The number of seconds until the program terminates.

Table 6–3

Parameter	Default	Description
show_statistics	FALSE	Displays at the bottom of the graph the number of data points, the min, max, average, meadian, stddev and 95%
debug	FALSE	Provides a detailed output of the utility.

Examples

.ini File

The following is an example of the `.ini` file that graphs the number of sockets connected to an IMAP server.

```
server=testdb.us.oracle.com
sid=test
port=1521
username=es_mail
password=es
process_dn=test1:um_system:imap:
metric_name=.ES_SPS.socket.currload
graph_type=xy_current
image_file_name=socketcount
image_title=Socket count on test1
image_dir=./
number_of_hours=24

encode_type=png
show_statistics=true
aggregate_time_period=600
debug=false
max_lifetime=120
```

Selecting Graphs to Display in Oracle Enterprise Manager

Administrators can schedule graphs illustrating information to be created and placed in their own custom HTML page or in the Oracle Enterprise Manager, such as the number of connected sockets, the response time to log in, and the number of queued outbound messages.

Displaying Graphs into Oracle Enterprise Manager

Perform the following steps to display graphics in Oracle Enterprise Manager:

1. Modify target metadata definition

Under the `$ORACLE_HOME/emdw/sysman/admin/metadata` directory is a list of target metadata definition files. For a particular target definition, the following elements must be added under the `<InstanceProperties>` section:

```
      <!--This property specifies the total number of statistic charts to be
displayed -->
      <InstanceProperty NAME="totalNumberOfStats" CREDENTIAL="FALSE"
       OPTIONAL="TRUE">
         <Display>
            <Label NLSID="totalnumberofstats">Total Number of Statistics</Label>
         </Display>
      </InstanceProperty>

     <!--This property specifies the header title for the first charting picture
-->
      <InstanceProperty NAME="Title0" CREDENTIAL="FALSE"
       OPTIONAL="TRUE">
         <Display>
            <Label NLSID="stat0">Statistic Number 0</Label>
         </Display>
      </InstanceProperty>

     <!--This property specifies the tool tips string for the first charting
picture, coded to Section 508 standards-->
      <InstanceProperty NAME="ToolTips0" CREDENTIAL="FALSE"
       OPTIONAL="TRUE">
         <Display>
            <Label NLSID="tooltips0">This is tooltips 0 for ADA</Label>
         </Display>
      </InstanceProperty>

     <!--This property specifies the relative picture path under the servlet for
the first charting pictyre.-->
      <InstanceProperty NAME="PicPath0" CREDENTIAL="FALSE"
       OPTIONAL="TRUE">
         <Display>
            <Label NLSID="picpath0">Picture Path 0</Label>
         </Display>
      </InstanceProperty>
```

```
     <!--This property specifies the physical path for the first charting
picture. The admin code will test if the file exists according to the path
below-->
      <InstanceProperty NAME="PicPhysicalPath0" CREDENTIAL="FALSE"
       OPTIONAL="TRUE">
         <Display>
            <Label NLSID="picphysicalpath0">Picture Physical Path 0</Label>
         </Display>
      </InstanceProperty>
```

To increase the number of charts displayed, the property value of `<totalNumberOfStats>` must to be changed accordingly under the `targets.xml` file and the addition picture properties must be defined under the naming standard:

`Title[N], ToolTips[N], PicPath[N], PicPhysicalPath[N]`

Where N is a non negative natural number.

2. Modify targets.xml file to specify the property instance values.
3. Add the following property to the specific target section, under the `$ORACLE_HOME/emdw/sysman/emd/targets.xml`:

```
  <Property NAME="totalNumberOfStats" VALUE="1"/>
  <Property NAME="ToolTips0" VALUE="My First Statistic Tool Tips"/>
  <Property NAME="PicPhysicalPath0"
VALUE="<…>/sysman/webapps/emd/ias/umsg/es/images/pic1.gif"/>
  <Property NAME="PicPath0" VALUE="/emd/ias/umsg/es/images/pic1.gif"/>
  <Property NAME="Title0" VALUE="My First Statistic Header"/>
```

If any of the following situations occur, the charting picture is skipped and not displayed on Oracle Enterprise Manager.

- `totalNumberOfStats` is missing, zero, or ,not a number
- `Title[N]` is missing for the particular chart
- `ToolTips[N]` is missing for the particular chart
- `PicPath[N]` is missing for the particular chart
- `PicPhysicalPath[N]` is missing for the particular chart
- The chart picture file specified under `PicPhysicalPath[N]` does not exist.

A

Server Statistics

The DBMS_STATS package generates statistics for the entire Oracle Email table or index. Statistics are transferred between statistics table and data dictionary, and can be used only when they are stored in the data dictionary. The statistics table enables users to export or import statistics from one database to another. Oracle Email includes statistics that are collected from a mature system, because there is not enough statistical data when the system is first installed. Users can choose to import these statistics into their systems prior to using their own statistics.

This chapter contains the following topics:

- POP3 Statistics
- IMAP4 Statistics
- SMTP In Statistics
- SMTP Out Statistics
- Housekeeping Statistics
- List Server Statistics

POP3 Statistics

The following is a list of POP3 server statistics and their descriptions:

Table A–1

Statistic	Description
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.ES_SPS.socket.currload	The current number of client connections
.ES_SPS.socket.sockmax	The maximum number of client connections allowed
.ES_SPS.thread.currthreads	The number of threads the server is currently using
.ES_SPS.thread.thrmax	The maximum number of threads the server creates.
.DUMP.DBconnections.dump	Internal Information
.DUMP.OIDStatus.Connection	Internal Information
.DUMP.Threads.dump	Internal Information
.ESPROTO.uptime	The amount of time the server has been up
.ESPROTO.COMMAND.total	total number of commands executed
.ESPROTO.COMMAND.<PO3_COMMAND>.success	where PO3_COMMAND is one of the commands defined by the POP3 protocol
.ESPROTO.COMMAND.<PO3_COMMAND>.totalcalls	The total number of calls for that command
.ESPROTO.COMMAND.<PO3_COMMAND>.fail	The total number of failed calls for that command
.ESPROTO.USERS.LOGIN.<userid>	Where userid is the user who has used the server since the server was started. A value of 1 indicates that user is still logged in.Otherwise, it is 0.

Table A–1

Statistic	Description
.ESPROTO.connections.lost	The total number of client connections that have disconnected
.ESPROTO.connections.timeout	The total number of client connections that have timed out.
.ESPROTO.connections.total	The total number of client connections
.ESPROTO.receive.bytes	The total number of bytes received by the server
.ESPROTO.transmit.bytes	The total number of bytes sent by the server

IMAP4 Statistics

The following is a list of IMAP4 server statistics and their descriptions:

Statistic	Description
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.ES_SPS.socket.currload	The current number of client connections
.ES_SPS.socket.sockmax	The maximum number of client connections allowed
.ES_SPS.thread.currthreads	The number of threads the server is currently using
.ES_SPS.thread.thrmax	The maximum number of threads the server can create.
.DUMP.DBconnections.dump	Internal Information
.DUMP.OIDStatus.Connection	Internal Information
.DUMP.Threads.dump	Internal Information
.ESPROTO.uptime	The amount of time the server has been up
.ESPROTO.COMMAND.total	The total number of commands executed

Statistic	Description
.ESPROTO.COMMAND.<IMAP_ COMMAND>.success	Where IMAP_COMMAND is one of the commands defined by the IMAP protocol (RFC2060). A value of 1 means that command has succeeded. A value of 0 means the command has failed
.ESPROTO.COMMAND.<IMAP_ COMMAND>.totalcalls	The total number of failed calls for that command
.ESPROTO.COMMAND.<IMAP_ COMMAND>.fail	The total number of failed calls for that IMAP command
.ESPROTO.USERS.LOGIN.<userid>	where userid is the user who has used the server since the server was started. A value of 1 indicates that user is still logged in, 0 otherwise.
.ESPROTO.connections.lost	The total number of client connections that have disconnected
.ESPROTO.connections.timeout	The total number of client connections that have timed out.
.ESPROTO.connections.lost	The total number of client connections that have disconnected
.ESPROTO.connections.timeout	The total number of client connections that have timed out.
.ESPROTO.connections.total	The total number of client connections
.ESPROTO.receive.bytes	The total number of bytes received by the server
.ESPROTO.transmit.bytes	The total number of bytes sent by the server

Table A–2

Statistic	Description
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.ES_ SPS.socket.currload	The current number of client connections

Table A–2

Statistic	Description
.ES_SPS.socket.sockmax	The maximum number of client connections allowed
.ES_SPS.thread.currthreads	The number of threads the server is currently using
.ES_SPS.thread.thrmax	The maximum number of threads the server creates
.DUMP.DBconnections.dump	Internal Information
.DUMP.OIDStatus.Connection	Internal Information
.DUMP.Threads.dump	Internal Information
.MTA.uptime	Time string describing when this MTA came up
.MTA.connections.in.current	The current number of inbound SMTP Connections
.MTA.connections.in.total	The total number of inbound SMTP connections
.MTA.msgs.deferred.current	The current number of messages deferred
.MTA.msgs.deferred.total	The total number of messages deferred
.MTA.receive.kbytes	The total number of kilobytes received
.MTA.receive.messages	The total number of messages received
.MTA.receive.recipients	The total number of recipients received
.MTA.receive.time	The total time receiving data (ms)
.MTA.transmit.bytes	The total number of bytes transmitted
.MTA.transmit.bytes_local	The total number of bytes transmitted to local entities

Table A–2

Statistic	Description
.MTA.transmit.messages	The total number of messages transmitted
.MTA.transmit.messages_loca	The total number of messages transmitted to local entities
.MTA.transmit.recipients	The total number of recipients transmitted
.MTA.transmit.recipients_local	The total number of recipients transmitted to local entities

SMTP In Statistics

The following is a list of SMTP In server statistics and their descriptions:

Table A–3

Statistic	Description
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.ES_SPS.socket.currload	The current number of client connections
.ES_SPS.socket.sockmax	The maximum number of client connections allowed
.ES_SPS.thread.currthreads	The number of threads the server is currently using
.ES_SPS.thread.thrmax	The maximum number of threads the server creates
.DUMP.DBconnections.dump	Internal Information
.DUMP.OIDStatus.Connection	Internal Information
.DUMP.Threads.dump	Internal Information
.MTA.uptime	Time string describing when this MTA came up

Table A–3

Statistic	Description
.MTA.connections.in.current	The current number of inbound SMTP Connections
.MTA.connections.in.total	The total number of inbound SMTP connections
.MTA.msgs.deferred.current	The current number of messages deferred
.MTA.msgs.deferred.total	The total number of messages deferred
.MTA.receive.kbytes	The total number of kilobytes received
.MTA.receive.messages	The total number of messages received
.MTA.receive.recipients	The total number of recipients received
.MTA.receive.time	The total time receiving data (ms)
.MTA.transmit.bytes	The total number of bytes transmitted
.MTA.transmit.bytes_local	The total number of bytes transmitted to local entities
.MTA.transmit.messages	The total number of messages transmitted
.MTA.transmit.messages_loca	The total number of messages transmitted to local entities
.MTA.transmit.recipients	The total number of recipients transmitted
.MTA.transmit.recipients_local	The total number of recipients transmitted to local entities
Statistic	**Description**
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.ES_SPS.socket.currload	The current number of client connections

Table A–3

Statistic	Description
.ES_SPS.socket.sockmax	The maximum number of client connections allowed
.ES_SPS.thread.currthreads	The number of threads the server is currently using

SMTP Out Statistics

The following is a list of SMTP Out server statistics and their descriptions:

Table A–4

Statistic	Description
.um.admin.os_pid	The operating system process id
.um.admin.uptime	The amount of time the server has been up
.DUMP.OIDStatus.Connection	Internal Information
.MTA.uptime	Time string describing when this MTA came up
.MTA.connections.broken	The number of broken connections encountered by the MTA
.MTA.connections.failed	The number of failed connections from the MTA to another MTA
.MTA.connections.rejected	The number of rejected connections
.MTA.connections.rejection_reason	Description of reason for most recent rejection
.MTA.connections.out.current	The current number of outbound SMTP connections
.MTA.connections.out.current_foreign	The current number of outbound SMTP connections to MTAs in foreign domains
.MTA.connections.out.current_native	The current number of outbound SMTP connections to MTAs in native domains
.MTA.connections.out.total	The total number of outbound SMTP connections

Table A–4

Statistic	Description
.MTA.connections.out.total_foreign	The total number of outbound SMTP connections to foreign domains
.MTA.connections.out.total_native	The total number of outbound SMTP connections to MTAs in native domains
.MTA.dl.receive.count	The number of messages sent to distribution lists
.MTA.msgs.deferred.current	The current number of messages deferred
.MTA.msgs.deferred.total	The total number of messages deferred
.MTA.msgs.delivered.totaltime	The total time inserting data into db (ms)
.MTA.ndr.inbound	The total number of non delivery reports generated by inbound mail
.MTA.ndr.loop	The total number of messages not delivered due to mail loops
.MTA.ndr.outbound	The total number of non delivery reports generated by outbound mail
.MTA.queued.out.kbytes	The kilo bytes queued awaiting to be sent out to the Internet
.MTA.queued.out.messages	The messages queued awaiting to be sent out to the Internet
.MTA.transmit.bytes	The total number of bytes transmitted
.MTA.transmit.bytes_foreign	The total number of bytes transmitted to foreign domain MTA′s
.MTA.transmit.bytes_local	The total number of bytes transmitted to local entities
.MTA.transmit.messages	The total number of messages transmitted
.MTA.transmit.messages_foreign	The total number of messages transmitted to foreign domain MTA′s
.MTA.transmit.messages_local	The total number of messages transmitted to local entities

Table A–4

Statistic	Description
.MTA.transmit.messages_native	The total number of messages transmitted to native domain MTA's
.MTA.transmit.messages_relay	The total number of messages transmitted during relay operations
.MTA.transmit.recipients	The total number of recipients transmitted
.MTA.transmit.recipients_foreign	The total number of recipients transmitted to foreign domain MTA's
.MTA.transmit.recipients_local	The total number of recipients transmitted to local entities
.MTA.transmit.recipients_native	The total number of recipients transmitted to native domain MTA's
.MTA.transmit.time	The total time transmitting data (ms)
.MTA.transmit.time_foreign	The total time transmitting data to foreign domain MTA's (ms)
.MTA.transmit.time_native	The total time transmitting data to native domain MTA's (ms)
.MTA.transmit.time.local	The total time spent transmitting data to local entities (ms)
.MTA.transmit.time.relay	The total time transmitting data during relay operations (ms)

Housekeeping Statistics

The following is a list of housekeeping server statistics and their descriptions:

Table A–5

Statistic	Description
.GC.processed.expirables	The number of message instances expired by a particular housekeeping process
.GC.processed.prunables	The number of message instances removed from the system trash folder by a particular housekeeping process

Table A–5

Statistic	Description
.GC.processed.queued_prunables	The number of message references removed from the system trash queue by a particular housekeeping process
.GC.processed.collectables	The number of unreferenced messages removed from the system by a particular housekeeping process
.GC.processed.tertiary_storables	The number of messages moved to tertiary storage by a particular housekeeping process
.GC.pending.expirables	The number of message instances awaiting expiration remaining in the system
.GC.pending.prunables	The number of message instances remaining in the system trash folder
.GC.pending.queued_prunables	The number of message references remaining in the system trash queue.
.GC.pending.collectables	The number of identified unreferenced messages remaining in the system
.GC.pending.tertiary_storables	The number of messages eligible for tertiary storage remaining in the system

List Server Statistics

The following is a list of list server statistics and their descriptions:

Table A–6

Statistic	Description
.DUMP.OIDStatus.Connection	Internal information.
.SLIST.connections.busy	The number of busy database connections
.SLIST.connections.total	The total number of database connections
.SLIST.process.current_mail_threads	The number threads running in the server processing mails
.SLIST.process.current_mails	The number of mails being processed in the server

Table A–6

Statistic	Description
.SLIST.process.current_user_threads	The number of threads running in the server that are delivering mails to users
.SLIST.process.total_mails	The total number of mails that have been processed by the server since startup
.SLIST.queue.pending	The number of mails waiting to be processed by the list server
.um.admin.os_pid	The operating system process ID
.um.admin.uptime	The amount time the server has been up

B

Oracle Email Access Control Lists

This section provides an overview of access control list policies set for Oracle Email in Oracle Internet Directory. These directory access control lists are set in Oracle Internet Directory during the infrastructure installation phase.

This appendix contains the following topics:

- Mail Server Access Control Lists
- Oracle Email Privilege Groups

Mail Server Access Control Lists

> **See Also:** *Oracle Internet Directory Administrator's Guide* for more information on access control lists

The Oracle Email LDAP schema and entries are installed during the installation of Oracle Internet Directory. In Oracle Internet Directory, the `cn=Products` container under OracleContext, contains all product specific information. The mail server container underneath this product container contains all the Oracle Internet Directory entries related to the e-mail server component of Oracle Email.

The `%s_OracleContextDN%` parameter described in the following access control lists can be the root or subscriber OracleContext.

During installation, the following privilege group is created:

```
cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products,%s_OracleContextDN%
```

The members of this group are the e-mail server component administrators. Various access control lists on `cn=EMailServerContainer,cn=Products, %s_OracleContextDN%` entry are as follows:

- Access control list for the group `cn=iASAdmins`, `cn=Groups,%s_ OracleContextDN%` giving browse, add, delete and proxy permissions. This is required for the `iasadmins` to be able to do a proxy to the `EmailServerContainer`.
- Access control list with DN equals `owner` or `targetdn` attribute giving read, search, write, selfwrite, and compare permissions to all entries. Since the mail users in the e-mail directory information tree have references to the organization level users, this ACL enables users to modify only entries they own. This prevents end users from modifying other users' entries, or entries they are not supposed to modify.
- Access control list enabling any user binding in "Simple" mode to have read and search permissions. This is required as the public users are stored outside the e-mail directory information tree. This bind mode "Simple" is added to restrict anonymous lookups using some client tools, such as Netscape Navigator.
- Access to the e-mail subtree is denied to everybody else.

This example specifies the access control list that must be set in order for the public distribution lists to be searchable through standard clients. In an e-mail domain, the distribution lists are stored under the list container. For example, if the domain is `oracle.com`, the list container `cn=List,dc=oracle,dc=com,cn=um_`

system,cn=EMailServerContainer,cn=Products,cn=OracleContext needs to have access control list "access to entry by * (browse)".

OID Group Membership for EmailAdminsGroup

The cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products, %s_OracleContextDN% also is added to the following groups in order to have permissions for e-mail related directory operations.

Table B–1

Group	Permissions
cn=ComputerAdmins, cn=Groups,%s_ OracleContextDN%	The addition of EmailAdminsGroup to this group enables the e-mail administrators to create process entries under cn=Computers.
cn=UserProxyPrivilege, cn=Groups,%s_ OracleContextDN%	The addition of EmailAdminsGroup to this group enables the e-mail administrators to proxy as the end users.
cn=AuthenticationServices,cn=Groups,%s_ OracleContextDN%	The addition of EmailAdminsGroup to this group enables the e-mail servers to compare the user's password at the time of authentication.
cn=verifierServices,cn=Groups,%s_ OracleContextDN%	The addition of EmailAdminsGroup to this group enables the e-mail servers to compare the orclpasswordverifier;email attribute. This is required for the voice mail authentication.

Oracle Email Privilege Groups

The following privilege groups are created for Oracle Email e-mail server component administration:

Group

cn=MailstoreAdminsGroup,cn=MailStores,cn=um_system,cn=EMailServerContainer, cn=Products,cn=OracleContext

Permissions

This group has read, search, compare, selfwrite, write access to the attribute orclPasswordAttribute of the mail store entry, everybody else is denied access to this attribute.

Members

```
cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products,cn=OracleContext
cn=DomainAdminsGroup,<Domain RDNs>,cn=um_system,cn=EMailServerContainer,
cn=Products,cn=OracleContext - if exists
```

Group

```
cn=DomainAdminsGroup,<Domain RDNs>,cn=um_system,cn=EMailServerContainer,
cn=Products,cn=OracleContext
```

where, `<Domain RDNs>` for the domain oracle.com is the string `dc=oracle,dc=com`

Note: This group is present in a system where domain administrators have been created from the Thin Client administration pages.

Permissions

This group has add, delete, browse, read, search, compare, and write permissions on the particular domain.

Members

```
Domain administrator user's DN
cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products,cn=OracleContext
```

Index

L

M

O

P

Q

R

S

T

U

V