Oracle® Email

Administrator's Guide

Release 2 (9.0.4)

Part No. B10720-01

June 2003

ORACLE®

Oracle Email Administrator's Guide, Release 2 (9.0.4)

Part No. B10720-01

Primary Author: Ginger Tabora

Contributors: Vicky Cao, Ashish Consul, Vikas Dhamija, Tanya Hitaisinee, Harvinder Walia, Anthony Ye

Table of Contents

3 Servers and Processes

5 Backing Up and Recovering Oracle Email

6 Charting and Monitoring

7 Command Line Interface

8 Parameters and Log Files

9 Error Messages

A Shared Folders

B Alias and Distribution List Look Up

C Oracle Email Access Control Lists

D Co-existence

E Server Statistics

F Oracle Email Supported RFCs

Index

Send Us Your Comments

Oracle Email Administrator's Guide, Release 2 (9.0.4)

Part No. B10720-01

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: ocsdocs_us@us.oracle.com
- FAX: (650) 633-3838 Attn: Oracle Email
- Postal service:
  Oracle Corporation
  Oracle Email Documentation
  500 Oracle Parkway, 2OP5
  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.

Preface

The Oracle Email Administrator's Guide is intended for anyone 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, "Provisioning"
This chapter contains information on the administration tools and explains how to provision domains and users.

Chapter 3, "Servers and Processes"
This chapter contains information on the different servers and processes of the Oracle Email system.

Chapter 4, "Security"
This chapter contains information on Oracle Email Security.

Chapter 5, "Backing Up and Recovering Oracle Email"
This chapter contains information on mail recovery for the Oracle Email system.

Chapter 6, "Charting and Monitoring"
This chapter discuses the Oracle Email system charting and monitoring tools.

Chapter 7, "Command Line Interface"
This chapter contains information on the Oracle Email command line interface.

Chapter 8, "Parameters and Log Files"
This chapter provides the Oracle Email system parameters and log file locations.

Chapter 9, "Error Messages"
This chapter contains information on Oracle Email error messages.

Appendix A, "Shared Folders"
This section contains information on Oracle Email shared folders.

Appendix B, "Alias and Distribution List Look Up"
This chapter contains information on alias and distribution list look up.

Appendix C, "Oracle Email Access Control Lists"
This chapter contains information on Oracle Email access control lists.

Appendix D, "Co-existence"
This chapter contains information on how Oracle Email can co-exist with other mail systems.

Appendix E, "Server Statistics"
This chapter contains information on Oracle Email server statistics.

Appendix F, "Oracle Email Supported RFCs"
This chapter contains information on RFCs supported by Oracle Email.

Related Documentation

Oracle Email documentation is available in HTML and PDF.

- *Oracle Email Application Developer's Guide*
- *Oracle Email JAVA API Reference*
- *Oracle Email Migration Tool Guide*
- *Oracle Collaboration Suite Setting Preferences*
- *Oracle Collaboration Suite Using WebMail*

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*

Printed documentation is available for sale in the Oracle Store at

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

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

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

To access the database documentation search engine directly, please visit

`http://tahiti.oracle.com`

Conventions

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

- Conventions in Text
- Conventions in Code Examples

Conventions 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.

Convention	Meaning	Example
UPPERCASE monospace (fixed-width) font	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a NUMBER column. You can back up the database by using the BACKUP command. Query the TABLE_NAME column in the USER_TABLES data dictionary view. Use the DBMS_STATS.GENERATE_STATS procedure.
lowercase monospace (fixed-width) font	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter sqlplus to open SQL*Plus. The password is specified in the orapwd file. Back up the datafiles and control files in the /disk1/oracle/dbs directory. The department_id, department_name, and location_id columns are in the hr.departments table. Set the QUERY_REWRITE_ENABLED initialization parameter to true. Connect as oe user. The JRepUtil class implements these methods.
lowercase italic monospace (fixed-width) font	Lowercase italic monospace font represents placeholders or variables.	You can specify the *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]
...	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;

Convention	Meaning	Example

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 contains Oracle Calendar, Oracle CorporateSync, and Oracle Outlook Connector.

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 protocol access servers separate 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 Oracle9*i* database. 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. Oracle Email takes advantage of Oracle9*i* database multithreading, parallel processing, high availability support, and high performance.

Open Standards-Based Messaging

Oracle Email enables users to access messages with the standards-based messaging client of their choice. Messages can be accessed using any client compliant with

Internet Message Access Protocol (IMAP) or Post Office Protocol (POP), 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.

WebMail

Oracle WebMail 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 WebMail.

Extended Server Side Filters

Oracle Email provides a wide range of server side filters that enable certain actions to be taken at various events. A variety of built-in actions, such as move, delete, and forward can be used to quickly assemble complex filtering logic with optimized e-mail operations. Filters can be defined to cover a wide range of events, such as delivery, copy, delete, and relay, enabling fine-grained control over a message's lifespan. In addition to built-in actions, server side filters include a PL/SQL Application Programming Interface (API), enabling customers to write their own customized actions and conditions.

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. Combined with server-side rules, large numbers of messages can be processed and managed by applications integrated with Oracle Email.

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

Provisioning

This chapter discusses how to administer Oracle Email domains and users.

This chapter contains the following topics:

- Managing Oracle Email
- Managing Domains
- Managing Users
- Managing Aliases

Managing Oracle Email

> **Note:** You must have administrator privileges to perform e-mail management tasks. If you have administrator privileges, you will see the administration tab in the WebMail client.

To perform management tasks for Oracle Email, you must navigate to the following URL, substituting your site's e-mail *machine_name* and *port* and retaining the rest:

`http://machine_name:port/um/traffic_cop`

Using WebMail, you can perform domain, user, list, alias, news, and policy 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.

Managing Domains

A domain brands your e-mail addresses as being from your company. An e-mail domain can have sub-domains, which can be administered separately even on the same system, with the following advantages:

- Convenience in accommodating sub-domains with different maintenance schedules, which is typical for sub-domains in different geographic regions
- Ease of administering sub-domains with different default attributes, which is common for sub-domains belonging to different organizations

Using WebMail, you can perform domain management tasks, such as modifying default attributes for new users or new lists, managing domain settings, and creating domains.

Domain Attributes

The following describes the different domain attributes:

Table 2–1 Domain Attributes

Attribute	Description	Acceptable Values	Default Value
Location in Public Namespace	Specifies the distinguished name of the LDAP container in Oracle Internet Directory, which contains all distribution lists in public namespace for client lookup	A valid distinguished name within the list server to which distribution lists are synchronized	None
Object Classes for Creation in Public Namespace	Contains the list of LDAP `objectclasses` used while creating the distribution lists in public namespace. The list of LDAP `objectclasses` must include the `groupofnames` or `groupofuniquenames` parameter		None
Naming Attribute for Creation in Public Namespace	Contains the naming attribute to be used while creating the distribution lists in public namespace		`CN`
Flashback Mail Recovery	Enables or disables flashback mail recovery for e-mail users. This attribute does not affect the flashback mail recovery capability for administrators using the PL/SQL interface	`Enable` or `Disable`	`Disable`

Creating Domains

Domains created through WebMail are Oracle Email domains. The base domain is created automatically during the Oracle Collaboration Suite infrastructure installation. E-mail domain names can be different than the base domain. For example, you can create e-mail sub-domains of `company.com` named `a.company.com`, `b.company.com`, and `c.company.com`.

Perform the following steps to create additional domains:

1. Navigate to the WebMail client administration page.
2. Select **Domain** > **Create Domain**.
3. Select the name of the installation for which you want to create the new domain.
4. Select the parent domain from the drop down list.

5. Enter the new domain name in the corresponding field.
6. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Modifying Domain Settings

> **Note:** Preferences modified for a domain apply only to entries created after the modifications. For example, if the default mail quota for the `oracle.com` domain is changed to 60MB, only users newly created in that domain have the new 60MB quota. Existing users in that domain retain their old mail quota.

Perform the following steps to modify domain settings:

1. Navigate to the WebMail client administration page.
2. Select **Domain** > **Domain Settings**.
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.
7. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Managing Users

Using the WebMail, you can perform user management tasks, such as adding, removing, and modifying e-mail users.

E-mail User Attributes

The following table describes the attributes for e-mail users:

Table 2–2 E-mail User Attributes

Attribute	Description
User ID	This parameter specifies the user ID. This attribute is read-only and cannot be modified.

Table 2–2 E-mail User Attributes

Attribute	Description
Mail Store	This parameter specifies the database to be used as mail store for the user. This attribute is read-only and cannot be modified.
E-mail Quota	This parameter specifies the e-mail quota of a mail user in bytes.
Additional Voice Quota	This parameter specifies the additional quota for the voice mail user in bytes
User State	This parameter defines the user as active or inactive. If User State is active, the user can receive and send e-mail; if inactive, the user cannot receive and send e-mail.
Forward E-mail Address	This parameter stores the e-mail addresses for the auto forward feature.
Document Binary Search	This parameter controls what is used for e-mail theme generation and e-mail formatting functions: only the text, or the complete contents of e-mail messages
Role	This parameter defines the user as either a regular user, a system administrator, or a domain administrator.
Text Synchronization	This parameter enables the user to have text search capability on message bodies.
Number of E-mail Display (WebMail)	This parameter specifies the number of message headers displayed at one time on the WebMail client. For example, you can specify to display 20 messages at a time.
Mail User Index Type	This parameter specifies if text indexing should be performed on only the e-mail text or both the e-mail and the attachment.
Domain Control ACI	This parameter specifies whether the user is a system administrator, domain administrator, or regular user
Display All Headers (WebMail)	This parameter specifies whether WebMail headers are displayed in detail.

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 continues until the user reaches 70MB.

When the user cleans up the account and the usage is under `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 `user-quota` calculations. When the usage reaches the `user-quota`, it means that the sum of e-mails and voice mails is equal to `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 when trying to save new outgoing mail.

Adding E-mail Users

Note: A base user must exist in Oracle Internet Directory before an e-mail account can be created. If the intended e-mail user has no entry in the directory, a message displays with a link directing you to the directory's Delegated Administration Service page. Create the user entry there, and then the e-mail account can be created.

Perform the following steps to add e-mail users:

1. Navigate to the WebMail client administration page.
2. Select **User** > **E-mail User Management** > **Add User**.
3. Select the domain from the drop down list.
4. Enter the new user's ID in the **User ID** field.
5. Enter the base user domain.
6. Select the mail store from the drop down list.
7. Enter the e-mail quota in bytes in the corresponding field.
8. Select the new user's role from the drop down list.
9. Click **Add**.

Modifying E-mail User Attributes

Perform the following steps to modify an existing user's attributes:

1. Navigate to the WebMail client 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**.

Setting Default New User Attributes

Perform the following steps to set the default attributes of new users in a particular domain. All new e-mail users have these attributes, which can be changed later.

1. Navigate to the WebMail client administration page.
2. Select **Domain** > **Default New User**.
3. Select the installation from the drop down list.
4. Select the domain you want to modify.
5. Click **Submit**.
6. Modify the attributes.
7. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

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 individual e-mail users:

1. Navigate to the WebMail client administration page.
2. Select **User** > **E-mail User Management** > **Remove User**.

3. Enter the search criteria for the user you want to delete.
4. Select the domain to which the user belongs.
5. Click **Go**.
6. Select the user you want to delete.
7. Click **Remove**.

Managing Aliases

An alias is a shorter or more descriptive name you can use when sending a message to a long user ID or list name. The message still reaches that original ID or list; the alias is like a pointer, effectively redirecting the message to the intended receiver.

For example, if Jane Doe changes her name to Jane Roe, you can create an alias so that mail sent to her original account, `jane.doe@acme.com`, is automatically redirected to her new account, `jane.roe@acme.com` . This alias prevents her losing messages sent to her old user ID while she notifies senders of her new one.

WebMail enables you to create, modify, and delete aliases.

Alias Attributes

The following describes the alias attributes:

Table 2–3

Attribute	Description	Acceptable Values	Default Value
Name	This parameter specifies the name by which the alias is referred.	`user ID, list, or alias`	None
Target	This parameter specifies the alias.	`user ID, list, or alias`	None
Description	This parameter specifies the description of the alias.	Text string	None

Creating a New Alias

Perform the following steps to create a new alias.

1. Navigate to the WebMail client 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. A target can be a user ID, list, or another alias.
7. Enter the description.
8. Click **Create**.

Editing Alias Properties

Perform the following steps to edit properties of an existing alias:

1. Navigate to the WebMail client 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 attributes you want to change.
9. Click **Modify**.

Deleting Aliases

Perform the following steps to delete an alias:

1. Navigate to the WebMail client administration page.
2. Select **Alias** > **Alias Management** >**Delete alias(es)**.
3. Enter the search criteria for the alias you want to edit.
4. Select the domain from the drop down list.
5. Click **Go**.
6. Select the alias you want to delete.
7. Click **Delete**.

3

Servers and Processes

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

This chapter contains the following topics:

- Mail Store
- Managing Services and Processes
- IMAP4 and POP3 Processes
- SMTP Process
- Housekeeping Process
- List Server Process
- NNTP Server Process
- WebMail

See Also: Chapter 4, "Security" for information on virus scrubber

Mail Store

Messages and folder data are stored in the Oracle Email mail store. A message destined for many accounts is stored only once, and links to the message are sent to all recipients. A single mail store can store mail for one domain or several different domains. Conversely, a single extremely large domain can be supported by multiple mail stores. Folders can be private, shared, or public.

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, such as IMAP, POP, SMTP, Housekeeping, or List server.
3. Click on **Mail Store Connection Parameters**.
4. Select the mail store for which you want to make changes.
5. Modify the parameters you want to change.
6. Click **Apply**.

Mail Store Parameters

Table 3–1 lists the mail store parameters.

Table 3–1 Mail Store Parameters

Parameter	Description
Timeout	Number of seconds a connection can remain idle before being terminated. Optimizes number of available connections for active users.
Maximum	Maximum number of connections that can be opened to the database. Once this value is reached, no more connections are allowed. Can be 1 or more.
Increment	Number of connections to the database that Oracle Email can add if the current number of connections is less than maximum. Can be 0 or more.
Minimum	Minimum number of connections to the database. Can be 0 or more.

Server Side Filters

Server side filters enable users to create mailbox filters on the server. Users can use the WebMail 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 that are carried out depend on if the user is online or using the client on which the rules were created on.

The SMTP inbound and outbound servers execute the rules as messages are received and sent.

Managing Services and Processes

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

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

Stopping an Oracle Email service sends a command to the service processes to shut down. System maintenance might be one reason an administrator might want to do this, such as upgrading the server hardware or software. The Oracle Email processes cannot be running while this kind of upgrade is being performed.

Whenever a Oracle Email process parameter is modified, the service must be reinitialized to make the changes take effect.

Reinitializing an Oracle Email service causes the processes to reload their operational settings from Oracle Internet Directory without stopping. Users continue to receive uninterrupted service.

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

Starting, Stopping, or Reinitializing All Server Processes

To start, stop, or reinitialize all server processes, use Oracle Enterprise Manager as follows:

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

Creating a Server Instance

To create a server instance, use Oracle Enterprise Manager as follows:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type, such as IMAP, POP, SMTP, Housekeeping, or List server.
3. Click **Create**. This creates a new server instance with default parameters.

Creating a Server Instance with the Same Parameter Values as an Existing Server Instance

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

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

To delete a server instance, use Oracle Enterprise Manager as follows:

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

1. Shut down the process you intend to delete.
2. Navigate to the Oracle Email Service Targets page.
3. Select the server type, such as IMAP, POP, SMTP, Housekeeping, or List server.
4. Select the server process you want to delete.
5. Click **Delete**.

Starting a Server Instance

To start a server instance, use Oracle Enterprise Manager as follows:

1. Navigate to the Oracle Email Service Targets page.

2. Select the server type, such as IMAP, POP, SMTP, Housekeeping, or List server.
3. Select the server instance you want to start.
4. Click **Start**.

Stopping a Server Instance

To stop a server instance, use Oracle Enterprise Manager as follows:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type, such as 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. However, reinitializing does not interrupt user actions because the service is not brought down.

Use Oracle Enterprise Manager to reinitialize a server instance as follows:

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

Modifying Server Default Parameters

All new server instances are created with default parameters for that server type that can later be modified for specific server instances.

> **Note:** Servers must be reinitialized whenever parameters are modified.

To create a server instance with the same parameters as an existing server instance, use the **Create Like** option.

> **See Also:** "Creating a Server Instance with the Same Parameter Values as an Existing Server Instance" for instructions on using the Create Like option.

To modify server default parameters, use Oracle Enterprise Manager as follows:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type, such as IMAP, POP, SMTP, Housekeeping, or List server.
3. Select **Change Settings**.
4. Modify the parameters you want to change.
5. Click **Apply**.
6. Reinitialize the server to make the changes take effect.

Modifying Parameters for a Specific Service Process

> **Note:** Servers must be reinitialized whenever parameters are modified.

To modify parameters for a specific service process, use Oracle Enterprise Manager as follows:

1. Navigate to the Oracle Email Service Targets page.
2. Select the server type, such as 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**.
6. Reinitialize the server instance to make the changes take effect.

IMAP4 and POP3 Processes

Table 3–2 describes the features of these two protocols for retrieving e-mail messages.

Table 3–2 Features of the POP3 and IMAP4 Protocols

Protocol	Description of Features
POP3, the Post Office Protocol	Provides mail manipulation services for smaller Internet nodes where it can be impractical to maintain a message transport system or 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.
IMAP4, the Internet Message Access Protocol	Provides functionality to manipulate mail messages and mail folders stored on the server and to enable an off-line client to re-synchronize with the server. Also has primitives enabling optimization of online performance, especially for large MIME messages.

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. These benefits enable the servers to support thousands of concurrent user connections, each using very few system resources.

This framework maintains a pool of worker threads handling the work for the clients and a pool of database connections shared across client connections. An incoming client request has a worker thread assigned to it, which reads the client command, obtains a database connection, and performs the operation. After the database connection is released back to the pool, the thread returns to the worker thread pool.

A system can contain multiple mail stores, and the IMAP4 and POP3 servers can be configured to create database connection pools to more than one mail store. Administrators use the IMAP4 and POP3 server parameters to control the size of the pools.

Many operating systems limit the number of file descriptors and sockets a single process can open. Such limits can make it necessary to run more than one instance of an IMAP4 or POP3 server, in which case the listener distributes the load between them. Administrators must verify the correctness of the operating system parameter controlling the file descriptors for such processes.

IMAP and POP Server Parameters

See Also: Chapter 8, "Parameters and Log Files" for detailed information on IMAP and POP server parameters

Managing IMAP and Pop Servers

See Also: "Managing Services and Processes" for instructions on creating, deleting, or setting parameters for IMAP and POP servers

SMTP Process

Simple Mail Transfer Protocol (SMTP) enables sending e-mail messages between servers, and is used by most Internet e-mail systems. Mail clients generally use SMTP to send messages to a mail server, and use either POP or IMAP to retrieve messages.

The SMTP server handles all inbound and outbound mail, implementing the SMTP protocol and interacting with DNS and Oracle Internet Directory servers for information about hosts and users.

Various Configurations

The flexible architecture of Oracle Email enables users to set up a single or multi-tier configuration appropriate to a site's needs, as in the following configurations.

Single Node Setup

A single node setup has one mail store and SMTP server running on the same host, supporting a small numbers of users.

Single Mail Store Setup

A single mail store setup divides two processes into two tiers. The backend runs the database; the middle tier runs SMTP and other protocol servers. This configuration provides fault tolerance and the flexibility to run multiple SMTP servers with distributed loads by running the servers behind a network director. Alternatively, it could have multiple mail exchanger (MX) records for the domain.

Multiple Mail Store Setup

A multiple mail store setup can have two configurations for multiple mail stores: on different hosts and on the same host. Each SMTP server serves only one mail store and each mail store must have an SMTP server. The mail stores on the SMTP hosts are used as SMTP queues and do not contain users.

Message Flow

The SMTP inbound service is responsible for handling the incoming SMTP connection. It receives incoming messages, queries the Oracle Internet Directory server to find and authenticate the addresses, and rewrites those addresses based on the rewriting rules. Anti-Spam rules are 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, for performance reasons, administrators do not want to process the messages immediately, messages can be stored in the submission queue and marked as submitted or unprocessed. Messages created by the SDK applications are also placed in the submission queue and marked as submitted. Server parameters control this use of the submission queue.

The messages in the submission queue are picked up by the SMTP outbound server. For relay messages, the SMTP outbound server queries the DNS server, applies the rules against them, and sends them out using SMTP. For submitted messages, processing by the address rewriting and DNS resolution module happens first. After that, the SMTP outbound server sends them to the local delivery queue or to the Internet, depending on whether the messages' recipients are local or not.

During address resolution, the server can determine that the message is for a distribution list handled by the list server. If so, the server places the message in the queue for the list server, which then picks up the message, expands the distribution list, and delivers the message.

Messages for users on a different mail store are placed in the relay queue. The outbound server picks up and delivers the messages to the SMTP inbound service for the other mail store.

SMTP Inbound Server Architecture

The SMTP inbound server listens for client requests, processes incoming messages and either delivers them locally or places them into queues for further processing.

The Oracle Net listener listens for incoming clients requests on the SMTP port, default 25, and transfers connections to the SMTP server. The SMTP server maintains three thread pools to perform its tasks:

- Worker thread pool, through which it handles client requests
- Oracle Internet Directory server thread pool, through which it performs user authentication and address resolution
- Database connection thread pool to the mail store, through which it delivers local messages.

Note: The number of threads in each pool should depend on the number of users on the system and the e-mail load.

Upon receipt of a connection request, a worker thread is picked up from the pool to handle the request. It performs name resolution on the incoming message by using a connection from the Oracle Internet Directory pool. Any anti-virus and anti-spam rules are applied, and then the recipient rewriting rules are applied to the e-mail addresses to determine whether the message is to be delivered locally or sent to another mail store or out to the Internet.

> **See Also:** "Rewriting Rules" for more information on SMTP address rewriting rules

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. At this point the SMTP connection to the client is terminated, but if the message has local recipients, the worker thread continues to process the mail and perform local delivery.

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

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.

> **See Also:** "List Server Process" for more information about the Oracle Email list server

If the recipient is determined to be local, the message is stored in the local delivery queue. To determine if an address is local, the `SMTPlocaldomains` parameter is used. This parameter contains the list of domains that are considered local.

If the recipient is an outside user, either on another mail store or on the Internet, the message is stored in the relay queue to await further processing by the SMTP outbound server.

SMTP Outbound Server Architecture

The STMP outbound queue processor processes messages in the submission, local, and relay queues. It has a main thread for each queue that periodically polls the database for messages in its queue. Whenever there are messages to process, a new thread is spawned to process the mail.

The outbound queue processor also maintains two other thread pools:

- Oracle Internet Directory server thread pool, through which it performs address authentication. If `SMTP_auth` is turned on, this thread pool is also used for user authentication prior to sending out a message.
- Database connection pool to the mail store, through which it delivers local messages.

The size of all of these thread pools can be set through Enterprise Manager.

When a thread is started to process a message from one of the queues, 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 database and Oracle Internet Directory threads are returned to the pool.

Messages in the submission queue are treated as not yet processed, and so must go through the anti-virus and anti-spam rules and rewriting rules to determine their destination. After processing, the messages are either placed in the local delivery queue or in the relay queue.

Messages in the local delivery queue are destined for a local mailbox, so the queue processor applies local rewriting and other rules, if any, and inserts the mail into the user's inbox in the database using a connection from the database pool.

Relay messages require further processing because their recipients are either on a different local message store or outside the email system. Relay messages first go through the sender rewriting rules, then the system rules are invoked for `event relay` and external filter processing. If the system rule and external filter processing are successful, the DNS resolution takes place. The SMTP outbound queue processor then sends them to another mail store in the system or to the Internet using SMTP, depending on whether the messages' recipients are local or not.

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.

Rewriting Rules

The SMTP address rewriting rules enables you to check and correct an e-mail message's addresses before sending it to its final recipient destinations. Rules resolve a focused or internal format address into a mailer, host, user triplet that can be delivered.

Table 3–3 Mailer, Host, User

Parameter	Description
Mailer	Specifies the Oracle Email SMTP process daemons used for delivery. Note: This is the only mailer available. Creating alternative mailers is not a feature of this application.
Host	Specifies either a fully qualified host name, such as `hostname.acme.com`, or a domain name, such as `foo.com`.
User	Specifies the recipient user name.

To execute and complete name resolution, the inbound SMTP process parses each rule for every address in the message envelope during mail delivery.

Components of Rewriting Rules

Headers for the message and the envelope are distinctly different. The envelope headers are generated by that receiving e-mail application, rather than by the

sender. `Received:` headers are the envelope headers, and relate only to the envelope `From` and envelope `To` fields.

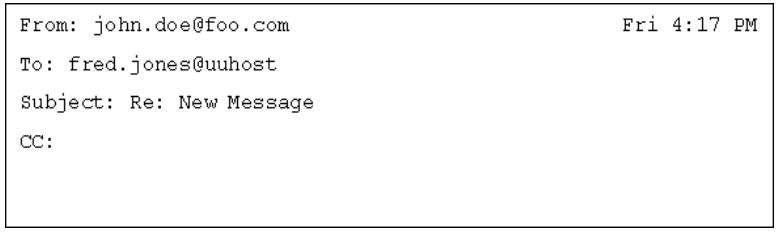
The envelope From header is created from the `MAIL FROM` entry in the received message. For example, when a sending machine puts `MAIL FROM: jsmith@acme.com` in the message, the envelope `From` is `jsmith@acme.com`.

Similarly, the envelope `To` is derived from an incoming message line, such as `RCPT TO: john.smith@acme.com`. The information for envelope `To` and envelope `From` is stored in a different location from the header.

Mail is routed based solely on the envelope `To` data rather than on the message `To:` or `From:` headers supplied by the sender. These headers contain no significant envelope information and can misrepresent who sent the mail to whom, as is illustrated in the example below.

Figure 3–1 shows e-mail addresses within a header, where an envelope is created by the `From:` and `To:` fields from the header.

Figure 3–1 Message Header



A handshake between two SMTP systems executing transactions through port 25 involves a series of action dialogs for each message being delivered. These messages can be seen on the receiving or sending systems only by running in debug mode, as illustrated in the following example. The example illustrates why the routing of mail ignores the `Message From` and `Message To` headers, which can be faked.

Example of Original Headers

```
HELO acme.org
250 mail.rico.net Hello ernie.com [104.65.21.123], pleased to meet you
MAIL FROM: forged-address@acme.org
250 forged-address@acme.org... Sender ok
RCPT TO: john.smith@acme.org
250 john.smith@acme.org... Recipient OK
```

```
DATA
354 Enter mail, end with "." on a line by itself
From: another-forged-address@moreover.com
To: (your address suppressed for stealth mailing and annoyance)
.
250 OAA08757 Message accepted for delivery
```

Resulting Headers as Seen by the Recipient

```
Received: from acme.org ([104.128.23.115]) by mail.rico.net (8.8.5)
From: another-forged-address@moreover.org
To: (your address suppressed for stealth mailing and annoyance)
```

Notice that the only true data seen by this recipient is in the `Received` line, which was taken from the `RCPT TO` entry actually sent. The apparent sending addresses need not have any relationship to the physical facts. They are taken from the data of the envelope `From`, `message From:`, and `message To:` lines exactly as entered by the sender, with no necessary relationship to what is factual.

This example illustrates why the `From`, `From:`, and `To:` headers are not reliable in mail, because they can easily be forged.

Rule Execution Guidelines

Address renaming rules are applied sequentially, beginning with rule 1. All rules are applied, unless a result starts with `$@`, which immediately stops rule execution 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 the application of the rule does not yield anymore 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 due to an alias, for example 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 rule does not yield any more changes, the rule execution process is done.

Oracle Email Rewriting Rules

To understand rewriting rules, you must understand their components: a left hand side (LHS) and a right hand side (RHS) as explained in Table 3–4, used in this format:

```
Pattern (LHS),Result (RHS) [,Description ]
```

where:

Table 3–4 Rewriting Rule Format

Format	Description
Pattern (LHS)	Specifies the pattern to be changed
Result (RHS)	Specifies that whenever the pattern is seen, it is changed with this result.
Description	Specifies the administrator's rule notation, and is not used in address name resolution. Anything after the last comma does not require quotes.
Comma (,)	Separates the LHS, RHS, and Description. No spaces are allowed between the commas, nor before the first comma.

When the Pattern (LHS) is compared against the address and finds a match, the Result (RHS) replaces that match in the address. The comparison is not case-sensitive. If no match is found, then this rule is skipped and the next rule is applied. A rule can be applied to an address resulting from applying a previous rule.

Tokens and Matching

When processing an address for rewriting by a rule, the SMTP daemon first separates the address into parts called "tokens" and stores them into a buffer called the "workspace."

A rule's Pattern (LHS) is also divided into tokens, which are then compared to the tokens in the workspace. If the two sets of tokens are identical, it's a match, and the result of the left hand side comparison is true.

Operators for Rewriting Rules

If rules always had to match addresses exactly, too many rules would be required: it would render their usage unproductive. Instead, operators such as wild cards can also be used to match arbitrary text in the workspace. To make the entire Pattern (LHS) match, wildcard operators match as little as possible.

The following operators are used as wildcards or token identifiers.

- `$*` = zero or more tokens, and prefers zero, or the fewest possible, to satisfy a `Pattern (LHS)` match.

Figure 3–2 Changing First.Last

```
LHS        RHS           Description
        |             |
$*.$*,   $1.$2@uuhost,  Changing the first.last to first.last@uuhost
```

For example:

```
fred.jones resolved by the Pattern (LHS) rule
Result (RHS) rule = fred.jones@uuhost
```

- $+ = one or more tokens, and prefers one, or the fewest possible, to satisfy an Pattern (LHS) match.

  To illustrate, consider passing the address john.jones@home. ORG to this Pattern (LHS):

Figure 3–3 Changing Uppercase to Lowercase

```
LHS             RHS          Description
              |            |
$* @ $+ .ORG,  $1@$2.org,   Changing the uppercase to lower case
```

For example:

john.jones@home.ORG resolved by the Pattern (LHS)rule ,

```
Result (RHS)= john.jones@home.org
$* (matches zero or more) = john.jones
@ matches exactly
$+ (matches one or more) = home
```

- $- = exactly one token

  john rewritten by rhs $- = ->john

  If the result was john@uuhost, then the rule would not match.

RHS Operator Descriptions

$1,$2 identifies Pattern (LHS) tokens to be passed over into the Result (RHS). These are copied by position from the Result (RHS) location.

`rhs $*.$*` where `lhs $1.$2`

$: Indicates that the rule should be applied only once.

`rhs john` rewritten by `$:$1` = `john`

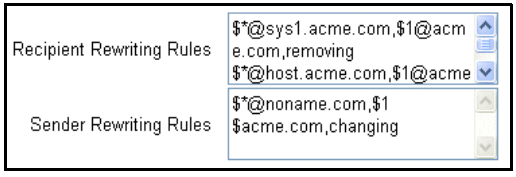
$@ Exactly none. Rules are not applied beyond this point if the $@ operator is reached during the rewriting rule processing, .

Designing an Oracle Email Address Rewriting Rule

Oracle Email uses two types of address rewriting rules:

- Sender rewriting rules: Apply to all sender addresses.
- Recipient rewriting rules: Apply to all incoming and outgoing recipient addresses.

Figure 3–4 Recipient and Sender Rewriting Rules



Rules can be written through Oracle Enterprise manager. Rules are executed in the order they are entered.

Examples of Rewriting Rules

The following example takes the **From:** (sender) and the **To:** (recipient) addresses and rewrites them using the rewriting rules.

- Sender Rewriting Rules

```
$*@$+.com,$1@uuhost.com, "This changes john.doe@foo.com to
john.doe@uuhost.com"
```

Rule:

1. Match anything before the @ sign and take the one token after the @ sign with the `.com` at the end and change it.

2. Keep the user name and pass it to the RHS through `$1`, which is in direct order or the 1st token from the LHS, `john.doe` and pass the `@` sign as is, but change the `$2` token (second token) and change it to `uuhost.com`.

   The receiving SMTP daemon accepts this message, and accepts `john.doe@uuhost.com` as the sender of the message. It is important to remember that the header information is never changed from its original entries.

- Recipient Rewriting Rules

```
$*.$*@uuhost,$1.$2@foo.com, "This changes fred.jones@uuhost to
fred.jones@foo.com"
```

Rule:

1. Capture both the first name and the last name of any address that has `uuhost` after the `@` sign.
2. Bring those tokens over as `$1` and `$2` respectively, and keep a period (.) between them.
3. After the `@` sign, replace `uuhost` with `foo.com`.

- Rewriting an E-mail Address

The following example shows how to rewrite an e-mail address using `fred.jones@uuhost`. The address points to `uuhost` which is a Unix to Unix copy (UUCP) system name. The message is sent using the UUCP software which requires the address form of `uuhost!username`, and that the current address be rewritten for UUCP. Consider the following example:

```
$*@uuhost,uuhost!$1,"Changing from to UUCP address"
```

Using the following rewriting rule, you can change this address to a more compatible Internet address such as `fred.jones@foo.com`.

Figure 3–5 Message Flow through Rewriting Rules

```
LHS       (S)     RHS     (S)Description
           |               |
uuhost!$*,   $1@foo.com,    "Changing Fred's address"
```

where:

`$*` token in `Pattern (LHS)` resolves as anything after the exclamation point (!).

$* = `fred.jones`.

The comma (,) is the separator between the LHS, RHS, and Description.

The $1 or first token in LHS string ($*) moves to the RHS is as.

Message headers are not rewritten during SMTP address name resolution. The address is parsed and rewritten by the delivery daemon rewriting rules, and passed as a logical address to the receiving daemon, which then parses and resolves it.

SMTP Server Parameters

See Also: Chapter 8, "Parameters and Log Files" for detailed information on SMTP parameters

Managing SMTP Servers

See Also: "Managing Services and Processes" for instructions on creating, deleting, or setting parameters for SMTP servers

Housekeeping Process

Housekeeping is a standalone component, operating in the background, that directly interacts with the mail store database to do cleanup tasks. During Oracle Email installation, a housekeeping job is created by default with a default configuration, but administrators can manually alter its schedule or add more instances of the job.

Job scheduling and management are handled by Oracle Enterprise Manager. While a housekeeping process is running, it responds to administrative requests to report job progress, reinitialize job parameters, or shut down.

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

Other servers mark messages for deletion by the housekeeper after their intermediate output is no longer needed, and housekeeping removes it, which is called "garbage collection." Three types of agents produce such intermediate output that is later destroyed by housekeeping:

- SMTP
- IMAP and POP3
- Housekeeping itself

Housekeeping performs tasks in multiple stages, some of which produce messages for another stage. For example, during message expiration processing, the housekeeping process produces messages later consumed by the pruning stage.

The SMTP server creates and processes messages that are mostly in transit, which stay in queues until the SMTP server finishes processing them and marks them as processed. Messages so marked are sent to the housekeeping process, which removes them from the system. Messages that users delete through clients are marked for deletion and picked up by housekeeping for deletion from the mail store.

The housekeeping log files, located in the mail store and the middle tier, respectively contain information on the progress of housekeeping tasks and the status of the process. See the section on SMTP Process.

Oracle Text

Integrating Oracle Text and Oracle Email extends the e-mail server functionality, enabling 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.

Two user-level Oracle Internet Directory parameters are associated with the configuration of Oracle Text:

- `User Index Type`: Enables text search capability for users. Set to 1 or 2 for a user enables that user to do server side search on message bodies, using any supported client.
- `Doc Binary`: Controls what is used for e-mail theme generation and e-mail formatting functions: only the text, or the complete contents of e-mail messages.

The following table describes the parameter values for `User Index Type` and `Doc Binary`.

Table 3–5 Oracle Internet Directory Parameter Name & Associations

Oracle Internet Directory Parameter Name & Associations	Type	Possible Values
`User Index Type/ 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
`Doc Binary/ document service`	Boolean	false: when requesting document service, process only text contents (default) true: when requesting document service, index both text and binary contents

These user-level configuration parameters are independent of each other and are viewable as domain level preferences. They are inherited by all new users created in the domain. 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.

Except for zipped attachments, Oracle Email message bodies and attachments can be indexed and later searched for text strings, themes, gists, or formatting, such as highlight and markup. To be searchable, the contents of a mail message body must be indexed by the Oracle Text server. If indexing is enabled, Oracle Email puts candidate messages into a queue for Oracle Text to index. The created index is later usable for performing a message body search.

Oracle Email user accounts can be enabled for text searching only, or for binary search as well. Text indexing enables searching message bodies for content, using IMAP clients that support message body searching or using Oracle Collaboration Suite's Ultra Search component. This feature is available only to users whose accounts are text-enabled.

A user enabled for text searching can search for strings in text and HTML files only.

A user enabled for text and binary searching can also search for strings in binary files, such as PDF files.

Applications that integrate with Oracle Email can use Oracle Text indexing through the PL/SQL and Java APIs.

> **See Also:** *Oracle Email Application Developer's Guide* for more information on using Oracle Email APIs to find themes and gists in e-mail messages

Verifying Oracle Text Installation

Before text indexing can be used, Oracle Text must be installed and configured. Oracle Text is installed by default when Oracle Email is installed. The Oracle Text installation fails if the database user `ctxsys` is not present at the time of installation.

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;
```

If Oracle Text was installed correctly, an output similar to following displays:

```
COMP_ID                        VERSION                        STATUS
------------------------------ ------------------------------ -----------
…
CONTEXT                        9.2.0.2.0                      VALID
```

If Oracle Text was not installed and configured on the mail store database, it must be configured manually.

> **See Also:** *Oracle Collaboration Suite Installation Guide* for further instructions on installing and configuring Oracle Text.

Creating a Housekeeper Process to Index Text

Oracle Text periodically processes a message queue filled by a Housekeeper instance.

To create a housekeeping instance to queue messages for text indexing, perform the following steps:

1. Using Oracle Enterprise Manager, navigate to the housekeeping page.
2. Create a new Oracle Email housekeeping instance by clicking Create or Create Like.
3. Click on that housekeeping instance to go to its parameter page.
4. Enable Text Synchronization.
5. Disable Pruning and Collection options.

6. In the Process Sleep Duration field, enter how often the housekeeper should queue messages for indexing, in minutes.

   For example, if the housekeeper should queue messages for indexing every three minutes, enter 3 in the field.

   Note: Set the housekeeper to index messages about as frequently as clients check for new mail. For example, five minutes is a good starting point.

7. Set **Execution Mode to Daemon**.
8. Click **Apply**.
9. Return to the housekeeping page.
10. Click **Start**.

    As an ongoing process, the housekeeper wakes up the message store periodically to submit messages that are at least three days old to Oracle Text for indexing.

Enabling Text Indexing for a User

To enable content indexing for a user, perform the following steps:

1. Navigate to the WebMail client administration page.
1. Select **Policy** > **Anti-Virus**.
2. Select **Modify User**.
3. Enter the user ID.
4. Select domain from pulldown list, if necessary.
5. Click Find.
6. Click on the user's name to bring up the policy page for that user.
7. Select the `User Index Type` parameter.
8. Set the `Text Synchronization` parameter of the user by selecting **Text Only**.
9. Click **OK**.

System administrators can enable text indexing for all users on the system; domain administrators can do so only for users within the domain they manage. Once text

indexing is enabled for a user, message body searching becomes available as soon as the housekeeper has indexed that user's messages.

Performance Considerations

All messages for an Oracle Text-enabled Oracle Email user are queued by the housekeeping process for indexing. However, users typically delete most messages almost immediately after they are received, wasting both the space in the queue and the CPU time for indexing.

Improving Performance through Optimized Text Index

Text search performance can be improved by periodically optimizing the existing Oracle Text index. Since many indexed messages are deleted or moved, the Oracle Text index bits are no longer consecutive, slowing down searching. Search time can be reduced by periodic clean-up of the Oracle Text index, removing entries that refer to deleted or moved messages.

Optimization can be done by a housekeeper process. A new housekeeper instance should be assigned to do this unless performance requires optimization to be done at the same frequency as indexing.

To create a housekeeper process to optimize the text index, perform the following steps:

1. Using Oracle Enterprise Manager, navigate to the housekeeping page.

Create a new Oracle Email housekeeping instance by clicking **Create** or **Create Like**.

2. Click on that housekeeping instance to go to its parameter page.
3. Enable **Text Synchronization**.
4. Ensure all other GC Operations are disabled.
5. In the **Process Sleep Duration** field, enter the frequency the housekeeper should index messages in minutes.

   For example, if the housekeeper should queue messages for indexing every three minutes, enter 3 in the field.

Note: Set the housekeeper to index messages about as frequently as clients check for new mail. For example, five minutes is a good starting point.

---

6. Set Execution Mode to Daemon.
7. Click **Apply**.
8. Return to the housekeeping page.
9. Click **Start**.

   The housekeeper wakes up periodically the message store for messages that are at least three days old to submit to Oracle Text for indexing, and is an ongoing process

This ongoing housekeeper process wakes up periodically to clean up the Oracle Text index.

Tertiary Storage

Administrators can configure Oracle Email to move messages to tertiary storage based on the age of the message. Although this process frees up valuable space on the primary disk for newer, more frequently accessed messages, users can still access messages in tertiary storage as before.

Message stores tend to grow constantly. Mail continually enters the store, and while many messages are deleted, more are saved. Generally, older messages are accessed less, so storing them on less expensive, slower disks while keeping them accessible to users may be an acceptable way to reduce costs. Depending on the storage mechanisms used for tertiary storage, users should not be aware that their older messages have been moved to a different physical disk.

Tertiary storage in Oracle Email is enabled through the housekeeper. The Oracle Email housekeeper can be set to move older messages to a tablespace named `ESTERSTORE`, which is reserved for tertiary storage of old messages. The age of messages to be moved to tertiary storage is set through the `Tertiary Storage Age Threshold` parameter.

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

Tertiary storage can be initially planned as part of an Oracle Email system or it can be implemented later. By default, the `ESTERSTORE` tablespace is created on the same disk as all other tablespaces when initially installing Oracle Email.

Table 3–6 gives the four considerations determining how tertiary storage tablespace should be handled:

Table 3–6 Tertiary Storage Usage and The ESTERSTORE Tablespace

If Tertiary Storage is:	Then The ESTERSTORE Tablespace Should:
Never enabled for the system	Remain empty.
Enabled for Oracle Email	Be set up on a disk different from the primary mail store, either before or after installing Oracle Email.
To be implemented for a new Oracle Email system	Be created on a disk different from the primary mail store, before installation of Oracle Email.
To be implemented for an existing Oracle Email system	Be moved, with the `es_tbody` table, from its default location on the same disk as the primary mail store onto a separate disk.

> **See Also:** Chapter 11, "Managing Tablespaces" of the *Oracle9i Database Administrator's Guide Release 2 (9.2)* for more information on creating and moving tablespaces

Moving the ESTERSTORE Tablespace

Perform the following steps to move the `ESTERSTORE` tablespace after Oracle Email has already been installed:

1. Back up the database.
2. Identify the datafiles for `ESTERSTORE` tablespace

   For example: The following query of the data dictionary view `DBA_DATA_FILES` lists the datafile names of the `ESTERSTORE` tablespace:

   ```
   select file_name from dba_data_files
   where tablespace_name='ESTERSTORE';
   FILE_NAME
   ----------------------------------------
   /usr/app/oracle/product/mailstore/dbf/erstore.dbf
   ```

3. Take the `ESTERSTORE` tablespace offline.

   ```
   alter tablespace esterstore offline normal;
   ```

4. Copy the datafiles for `ESTERSTORE` tablespace using the operating system to a different disk

5. Use the ALTER TABLESPACE statement with the RENAME DATAFILE clause to change the file names for the ESTERSTORE tablespace to a new location.

```
alter database esterstore rename datafile
/usr/app/oracle/product/mailstore/dbf/erstore.dbf to
file_name_in_new_location;
```

6. Bring the ESTERSTORE tablespace online.

```
alter tablespace esterstore online;
```

See Also: *Oracle9i Database Administrator's Guide Release 2 (9.2)*

Enabling Tertiary Storage

After the ESTERSTORE tablespace has been created, enable tertiary storage for an instance of Oracle Email housekeeper using the following steps:

1. Using Oracle Enterprise Manager, navigate to the housekeeping page.
2. Create a new Oracle Email housekeeping instance by clicking **Create** or **Create Like**.
3. Navigate to the parameter page of newly created housekeeping instance.
4. Set the `Tertiary Store` parameter to **Enable**.
5. Disable **Pruning** and **Collection**.
6. In the **Process Sleep Duration** field, enter how often the housekeeper should perform tertiary storage, in minutes.

   For example, if the housekeeping process performs tertiary storage every week, enter 10080 (60*24*7).
7. In the **Tertiary Storage Age Threshold** field, enter the age, in days, of messages you want to move to tertiary storage. The default is 30 days.

   For example, if you enter 60 in this field, messages that are 60 days old are moved to tertiary storage.
8. Click **Apply**.
9. Return to the housekeeping page.
10. Click **Start**.

    The housekeeping server periodically moves messages of the appropriate age into tertiary storage.

Housekeeping Parameters

See Also: Chapter 8, "Parameters and Log Files" for detailed information on housekeeping parameters

Managing Housekeeping

See Also: "Managing Services and Processes" for instructions on creating, deleting, or setting parameters for housekeeping

List Server Process

List servers enable public list management as well as integration with other messaging services or applications.

Users can own and administer public mailing lists as a way to distribute information to groups of people or as a discussion forum. If desired, restrictions can be placed on membership, requiring prior approval, and on outgoing messages, requiring screening by one or more moderators who control what messages are sent out. For example, a mailing list administrator may screen out advertisements.

The list server is installed with Oracle Email, with default values set for all list server parameters. Administrators can modify these values to meet performance or feature requirements. For example, a distribution list with a large number of members requires changing the Oracle Internet Directory `Max Search Results Entries` parameter. It must be configured to return a large number of entries to enable the list resolution API to return all the members. This parameter can be configured through `oidadmin`.

> **See Also:** *Oracle Internet Directory Administrator's Guide*, for more information on setting the `MaxSearchResults` parameter

APIs provided with the Oracle Email list server enable users to customize lists and messages sent out to a list. For example, marketing campaigns can send special non-transferable offers readable only by the intended recipients. As another example, a user can query a sales information database to create a list of all customers who have made purchases in the past three months. Customers on that list can then receive e-mail coupons with discounts based on the amount of their purchases.

List Attributes

List attributes include

- group type, described in Table 3–7,
- subscription type, described in Table 3–8, and
- posting type, described in Table 3–9.

Administrators can set or modify these types using the setattribute command described in the List Server Mail Interface section below.

A list's group type, set by the list owner during list creation, controls list attributes. Examples include what headers go on mail delivered to a list, or whether the list is

moderated. The list owner can change the group type after the list is created. Table 3–7 describes the different group types:

Table 3–7 Group Type

Type	Description
announcement	Restricts replies: No auto-replies or DSNs to the mail sent on the list are delivered to the sender Announcements have no `Reply-To` header. Replies to announcement mail are delivered only to the originator of the announcement.
discussion	Restricts replies to the specific subtopic discussion group on a list: Used when there are multiple discussions occurring on a list. Each e-mail posted has a `Reply-To` header containing the name of the relevant list. Replies to e-mails sent to this list are sent to the original sender and list. Auto-replies and delivery status notifications (DSNs) are not blocked and are delivered to the sender
edited	No `Reply-To` header is set in the mail delivered to the list. Auto-replies or DSNs to the mail sent on the list are not delivered to the sender.
moderated	Restricts postings: only e-mail from those e-mail addresses (moderators) stored in the `Group Moderators List` attribute is posted. Mail sent to the list is first delivered to all the moderators. Before mail can be posted to the list, at least one moderator must approve the mail within 3 days of the delivery.

"Subscription type," described in Table 3–8, controls who can subscribe to a list.

Table 3–8 Subscription Type

Type	Description
open	No approval required: any user can subscribe.
restricted	Approval required: subscription requests are sent to the list owner; users are added only if approved.
closed	No approval possible: subscription requests are not received; users are added only if the lister owner invites them.

Table 3–9 describes the list posting type, which can restrict non-members' postings.

Table 3–9 Posting Type

Type	Description
subscriber	Only list subscribers can post a mail to the list. Mails from non-subscribers are rejected.
open	Both subscribers and non-subscribers can post mail to the list.

List Parameters

Table 3–10 describes the list parameters.

Table 3–10 List Server Parameters

Parameter	Description	Acceptable Values	Default Value
Archive Name	Name of the newsgroup archive for this list. Validation is performed in the code to ensure that only valid existing newsgroups are given	Any valid newsgroup name	None
Archiving	If TRUE, messages to this list are archived as a newsgroup in the NNTP server; otherwise not.	TRUE or FALSE	
Autoreconfirm	If set to true the subscribe, unsubscribe, suspend, and resume requests must be reconfirmed with the user.	TRUE or FALSE	
Create New Archive on News Store	If selected, this option creates the newsgroup in the news store specified.		
Editor	List of users (mail IDs) for the editors of the list. Multiple editors can be set with one `setattribute` command.		
External Procedure	Name of the external procedure used to resolve the list, in the following format: `schema-name.procedure-name@database-link`		
Group Approvers	E-mail addresses for the list approvers, who approve subscriptions to a list with `Group Subscription Type` set to "restricted." If an approver is set, all subscriptions to the list must be approved by the approver and not the list owner. A list owner can also be an approver. Any approver can approve a subscription request.		
Group Auto Reconfirm	If TRUE, requests to subscribe, unsubscribe, suspend, resume or invite must be reconfirmed with the user.		

Table 3–10 List Server Parameters

Parameter	Description	Acceptable Values	Default Value
Group Has Archive	If TRUE, the list is archived, and all mails addressed to the list are archived.	TRUE or FALSE	FALSE
Group Information Text	Multi-line owner-provided descriptive text about the list		
Group is External	If TRUE, the list is resolved externally; otherwise locally.	TRUE or FALSE	FALSE
Group Merge Tag	A tag used for specifying mail merge and scheduler tags, enabling a list owner to support mail merge or scheduled mail delivery		
Invitetext	Multi-line text sent in e-mail to users invited by a list owner to join the list. When setting this parameter through the `setattribute` command, the parameter value should be enclosed within quotations		
List State	State of the list, active or inactive:active - posting is permittedinactive - the list is not recognized as a recipient, no posting is permittedmigrating - the list is being migrated into this Oracle Collaboration Suite installation	`active`, `inactive`, `migrating`	active
Mail Store	If specified, the mail store on which the messages addressed to this list are queued until the list server processes them. If not, then messages addressed to this list is queued wherever they are received.		
Moderator	List of users (mail IDs) who are the moderators of the list Owner List owner Post, Open or Subscriber. If SUBSCRIBER, then only subscribers to the list can post messages to the list; otherwise anyone can		
Subscription	Type of subscription control placed on the list, one of the three shownOpen, Restricted, or Closed		
Topic	Single-line phrase describing the topic of discussions on this list, enclosed within quotes		

Table 3–10 List Server Parameters

Parameter	Description	Acceptable Values	Default Value
Type	Type of the list, one of the following four: Announcement, Discussion, Edited, or Moderated		
Unsubscribe Not Allowed	If TRUE, allows only the list owner to unsubscribe a member from a list. If FALSE, anyone can unsubscribe.	TRUE or FALSE	FALSE
Use Existing Archive	Names an existing newsgroup as the list archive		

List Server Mail Interface

The email list server performs certain tasks when it receives commands by e-mail from administrators or users. The `setattribute` command is used by administrators to set values for various list parameters. For example, if John, the list owner of `test@acme.com` wants to set the list type to `moderated`, he would have to send an e-mail to the list administrator with the e-mail body containing the following line:

```
setattribute type=moderated
```

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"
```

Archiving Lists

Note: To enable list archiving, the NNTP server must be configured and running.

A list owner can have all mails sent to a list stored as messages in an archive in the NNTP server archives. Such an archive operates as a newsgroup and can be browsed using a standard news client.

An administrator must specify archiving as a property of the list in order to archive messages.

When a list is archived, a newsgroup is created with a name reflecting the original list. For example, the name of the NNTP archive newsgroup for the list `abc@foo.com` becomes the following:

```
ListArchive.abc
```

Once a list has been created, the domain administrator can begin archiving, which affects only mails sent after a list's archive property is set. No messages prior to that time are archived.

List archives must have the post parameter disabled. A mail is added to the archive only when a mail is delivered to the list. Mails cannot be added to an archive by any other mechanism.

List archives must be local to the domain of the list. Global newsgroups cannot be associated with a list as an archive.

Administrators can set expiration periods for list archives, such as one month, meaning that messages are only stored in the archive for one month, and then deleted. The expiration policy for a list's archive is the corresponding newsgroup's expiry attributes.

External Lists

External lists provide a way for the membership of a list to be stored outside of Oracle Collaboration Suite, while using the list server to deliver mails to such a list. A list owner or domain administrator can configure a list to be external by checking the external list option in the list properties page. A PL/SQL procedure for resolving the addresses of the list members must be created on the mail store the list server is connected to. If the PL/SQL procedure is on a different database, then a database link must be created from the mail store to the other database.

The PL/SQL procedure must have the following syntax:

```
procname(listid IN VARCHAR2,
    return_count IN NUMBER,
    count OUT NUMBER,
    recipients OUT TABLE OF VARCHAR2(2000))
```

The list server calls the procedure two times while resolving an external list:

The first value passed for the `return_count` parameter is 1. It receives in return the count out parameter, containing just the number of recipients in the list, and not the list of recipient addresses.

The second value passed for the `return_count` parameter is 0, causing the procedure to return a table of `varchar2(2000)` in the recipient out parameter. Each row of that table contains a recipient's full e-mail address.

Example

The following is an example of how to create the `get_cust_list` PL/SQL procedure.

1. Using the Oracle Collaboration Suite administration pages, login as domain administrator.
2. Navigate to the list `cust_list@acme.com`.
3. Edit the properties of the list.
4. Select true in the list box for the **Group Is External** field.
5. Set the `External procedure` parameter to `get_cust_list`.
6. Connect as `es_mail` to the mail store the list server is connected to create the following PL/SQL procedure:

```
CREATE OR REPLACE PROCEDURE get_cust_list(listid IN VARCHAR2,
return_count_flag IN NUMBER,
cnt OUT NUMBER,
recipients OUT dbms_sql.varchar2_table)
as
begin
if (return_count_flag = 1) then
-- when this procedure is called with the value of the second parameter as
-- 1, it is expected to return the total number of users in the list in the
third
-- parameter
select count(*) into cnt from customer_list;
else
-- when this procedure is called with the value of the second parameter as
-- anything other than 1, it is expected to return the recipients in the
-- external list in the fourth parameter with each recipient in one row
-- of the output varchar2 table
select customer_mail
bulk collect into recipients
from customer_mail;
end if;
end;
```

The previous example assumes that `cust_list@acme.com` is a list of customers

maintained in a database table by a different application. This procedure uses a table `customer_list` described below.

Table 3–11 cust_list

Column Name	Data Type	Explanation
customer_mail	varchar2 (1000)	Mail ID of a customer

Assuming that the table `customer_list` is populated with email addresses, sending a mail to the list `cust_list@acme.com` using an Oracle Email SMTP server delivers the mail to all the recipients in the `customer_list` table.

Mail Merge

Mail merge enables customized mail to be delivered to every list recipient. List owners or domain administrators must decide on a mail merge tag for a list and set it in the list properties page. The mail merge tag can be a single word or a group of words. This feature can be enabled for a list by providing a value for the merge tag property of the list. The list server supports two types of mail merge:

Table 3–12 Types of Mail Merge and Customizable Features

Type of Mail Merge	Description	Customizable Features
Standard mail merge	Message contents can be customized for each recipient with the values in the Customizable Features column.	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`)
PL/SQL mail merge	Similarly customizable, but also enables embedding of PL/SQL in messages. (The PL/SQL function must return a `varchar2` string.)	For each recipient, the PL/SQL function is executed and the output is embedded in the mail before delivery. Any parameter defined for standard mail merge (as above) can be included as a parameter to the PL/SQL function.

For standard mail merge, use the mail merge tag appropriate to a corresponding section 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>,
...
...
```

For PL/SQL mail merge, if you have a PL/SQL getsalary function that returns an individual's salary, given his mail address, you can use it in the mail. For example, you can embed the function call in the mail you send 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 the `ES_MAIL` schema, and that link must be referenced in the mail merge tag. For example, if the getSalary function is defined in a different database and you've created a database link called dblink, your mail merge message would need to look like the following:

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

Example

> **Note:** The following example assumes that lists and users have been setup correctly with the list server process configured and running.

The following example shows how to create the `get_sal` PL/SQL procedure:

1. Using the Oracle Collaboration Suite administration pages, log in as domain administrator.
2. Navigate to the list `all_emp@acme.com`.
3. Edit the properties of the list and set the `Group merge tag` parameter to mail merge.
4. Connect as `es_mail` to the mail store supported by the list server and create the following PL/SQL procedure:

```
CREATE OR REPLACE FUNCTION get_sal(email IN VARCHAR2) RETURN VARCHAR2
```

```
mon varchar2(10);
tmp number;
ret varchar2(4000);
begin
-- get the month and salary value for the user
select month, salary into mon, tmp from emp_payroll where employee=email;

-- concatenate to form a string
ret := mon || ‘ is $’ || tmp;

return ret;
end;
```

The procedure assumes that some application puts employee payroll information into a database table. The table named `emp_payroll` in the previous example contains the following columns:

Table 3–13 emp_payroll

Column Name	Data Type	Explanation
employee	varchar2 (1000)	Mail ID of an employee
month	varchar2 (10)	Month for which the salary is stored
salary	number	Salary of the employee

Send a mail with mail merge tags embedded in it as given below. This sends a mail to each recipient in the list `all_emp@acme.com` with his/her salary details.

> Dear <mailmerge>recipient_full_name</mailmerge>, your salary for the month of <mailmerge>get_sal(recipient_mail_address)</mailmerge>. The salary has been credited into your account.
>
> Thanks
>
> Payroll

Scheduled Mail Delivery

Scheduled mail delivery enables administrators to schedule mail delivery to occur at a particular time, such as during low traffic hours, possibly minimizing server loads during peak usage hours. Otherwise, delivery of very large messages or of mailings to lists with large numbers of subscribers can degrade performance.

This feature can be enabled by providing a value for the mail merge property of the list. Specify the delivery time for a message by putting the schedule mail delivery tag anywhere in the mail. The following example shows how, using `orcl` as the tag for the mail merge property of the list:

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

<orcl>send_schedule=23-JUN-2003 21:45 -08:00</orcl>
```

Note: `+/-` before `TZH:TZM` is mandatory.

Table 3–14 send_schedule

Parameter	Description
`DD`	The date
`MON`	The three 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 Lists

Using WebMail, you can

- create, delete, or show lists or list members,
- modify list properties, and
- show all the lists a member is on.

Creating Lists

To create a list, perform the following steps:

1. Navigate to the WebMail client 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.
   - 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
   - Group Approvers list
7. Click **Create**.

Modifying List Properties

To edit list properties, perform the following steps:

1. Navigate to the WebMail client 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**.

Modifying Default New List Attributes

To modify domain preferences for lists, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **Domain** > **Default New List**.
3. Select the installation from the drop down list.
4. Select the domain you want to modify.
5. Click **Submit**.
6. Modify the attributes you want to change.
7. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Deleting Lists

Perform the following steps to delete a list:

To delete a list, perform the following steps:

1. Navigate to the WebMail client 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**.

Showing Lists

To view all lists in a particular domain, perform the following steps:

1. Navigate to the WebMail client 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

To show list members, perform the following steps:

1. Navigate to the WebMail client 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.

Adding and Deleting List Members

To add or delete list members, perform the following steps:

1. Navigate to the WebMail client 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-list
   - 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 All the List a User is On

To show all the lists subscribed to by a user, perform the following steps:

1. Navigate to the WebMail client 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**.

List Server Parameters

See Also: Chapter 8, "Parameters and Log Files" for detailed information on list server parameters

Managing List Servers

See Also: "Managing Services and Processes" for instructions on creating, deleting, or setting parameters for list servers

NNTP Server Process

Network News Transport Protocol (NNTP) is used to distribute, query, post, and retrieve news articles from the Internet using a reliable stream-based mechanism. NNTP enables news-reading clients to select news articles from a central database, enabling subscribers to retrieve only the articles they want to read. The net news model provides indexing, cross-referencing, and message expiration. For server-to-server interaction, NNTP is designed to efficiently transmit net news articles over a reliable communication channel. Receiving and sending news articles is an interactive mechanism so that articles already present are not re-transmitted.

The NNTP server installed with Oracle Email uses the standard mail store for article repository and the Oracle Collaboration Suite infrastructure directory service to store operational parameters. All protocol exchanges are performed over a stream-based connection.

During installation, default values are set for all Oracle Email NNTP server parameters, which administrators can modify to meet performance or feature requirements of their site.

About News Servers

One or more news servers used by the same community of users is called a news site. Such sites can exchange news articles, transmitting locally posted articles to other sites to provide (and serve) a wider audience. News servers that exchange news articles are called peers.

News articles collected into similar-topic groupings are called newsgroups, such as articles about sailing or articles about the Oracle database. A peer can be configured to download articles only for particular newsgroups.

Users of news services exchange information by posting and reading news messages. Posting means a news user composes a message in a standard newsreader and sends it to the news server for storage, after which other users can read it.

The NNTP service maintains a list of peer servers, the newsgroups each is configured to receive, and a list of newsgroups that the NNTP service delivers. The administrator for each newsgroup specifies the peers to be fed articles from that newsgroup. Once peers and newsgroups are configured and the feed rules are set, the service is ready for posting, reading, and feeding news.

Storage Requirements

News articles are stored in a standard Oracle Collaboration Suite mail store. Inbound and outbound servers connected to a mail store only handle newsgroups created in that mail store. News articles are automatically expired strictly by the housekeeping process.

The NNTP service also creates a history record to track articles already received, for which new entries are rejected by the server. Having a long history expiry avoids repeated entry of the same article into the mail store. History records are also subject to expiry and collection by the housekeeping service.

The storage needed for NNTP service depends on the volume of incoming traffic and the expiry policy of the server instances, such as how long articles are retained. Each article's expiration date and history record are established when it is stored, and cannot later be changed.

Article Caching for Performance

You can configure the NNTP inbound service to cache articles in memory, improving performance for articles that are requested repeatedly, since no new mail store access is needed. Caching can only be done for articles less than 4 kilobytes in

size. You can adjust the total cache size to accommodate the number of articles you want to cache, to provide newsreaders with quicker response times for popular articles.

NNTP Processes

There are two types of newsgroup exchanges, which are known as feeds:

- Inbound: The NNTP server receives the feed
- Outbound: The NNTP server transmits the feed

The processes required for NNTP service are shared between the inbound and outbound servers.

NNTP Inbound

The NNTP inbound server has two functions:

- Incoming feed: accepts news articles and waits for connections from remote hosts
- Read or Post: accepts and waits for connections from news-reading clients and enables them to post or read news articles

The NNTP inbound server identifies the connecting host for each connection it receives. For a peer, the server prepares to receive the news feed. If the connecting host is not a peer, it is a newsreader and only has permission to read and post articles.

When a newsgroup is configured, administrators can specify the number of peers that must be sent articles for that newsgroup. Based on the number of peers specified, the NNTP inbound server creates queues of incoming messages that must be passed on to other peers .

NNTP Outbound

The NNTP outbound server periodically connects to peer news servers configured to receive news feeds and offers a list of new articles queued for sending. Peer server parameters determine what is offered and what is acceptable or rejected by the peer. See Peer Server Parameters.

The NNTP outbound server maintains a list specifying the newsgroups for each peer server. When the outbound server contacts a peer and provides a list of new articles, the remote host's response determines which articles are sent.

Peer Server Parameters

Table 3–15 describes the peer server parameters in alphabetical order.

Table 3–15 Peer Server Parameters

Parameter	Description	Acceptable Values	Default Value
Host Name	Internet host name of the peer, used by NNTP Inbound to recognize an incoming connection as a peer connection	Canonical host name of the peer as returned by DNS	None
Inbound Feed Accepted Newsgroups(s)	Names of newsgroups for which feed is accepted from this peer. If any groups are specified, then messages are accepted only if addressed to one or more of them.	Multi-value newsgroup names. Wildcards are allowed	None
Inbound Feed Rejected Newsgroups(s)	Names of newsgroups for which feed is rejected from this peer. If any groups are specified, then messages are rejected if addressed to one or more of them. This is checked after Inbound Feed Accepted Newsgroups(s). If a newsgroup appears in both `accepted newsgroups` and `rejected newsgroups`, it is rejected.	Multi-value newsgroup names. Wildcards are allowed	None
Installation	The Oracle Collaboration Suite installation for which this peer is to be configured	An installation in which newsgroups have been created, and NNTP server has been configured	um_ system
Outbound Feed Newgroups	Names of newsgroups for which feed is offered to this peer. If any groups are specified, then only messages posted to any of these groups are offered.	Multi-value newsgroup names. Wildcards are not allowed.	None
Port	NNTP port on which the peer listens, used by NNTP Outbound to connect to the peer	None	119

Managing Peer Servers

You can use WebMail to add, edit, or delete peer servers, as the following sections describe.

Adding Peer Servers

To add a peer server, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Peer Server Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Click **Add**.
6. Enter information in the appropriate fields.
7. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Editing Peer Server Properties

To edit a peer server, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Peer Server Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the peer server for which you want to edit properties.
6. Click **Edit**.
7. Edit the properties you want to change.
8. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Deleting Peer Servers

To delete a peer server, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Peer Server Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the peer server you want to delete.
6. Click D**elete**.

NNTP Server Parameters

See Also: Chapter 8, "Parameters and Log Files" for detailed information on NNTP server parameters

Managing NNTP Servers

See Also: "Managing Services and Processes" for instructions on creating, deleting, or setting parameters for NNTP servers

About Newsgroups

A newsgroup is a collection of messages discussing a particular subject, posted to an internet site and redistributed through Usenet, a worldwide network of news discussion groups. There are two types of newsgroups, public and private:

- Public newsgroups have worldwide distribution, are maintained at many news sites, and are intended to contain non-sensitive information. The NNTP service exchanges newsgroup articles among news sites.
- If your Oracle Collaboration Suite installation hosts more than one domain, public newsgroups are shared, which greatly reduces the storage space required for news articles. An inbound server always services all public newsgroups configured for an Oracle Collaboration Suite installation.
- Private newsgroups belong to and are visible in only one domain, where they are intended as internal discussion groups. Access to private newsgroups is permitted only if the inbound server has the local domain parameter set to that domain. An inbound server instance cannot host private newsgroups for more than one domain. For example, an inbound server instance with the local domain set to `acme.com` serves all public newsgroups in addition to only those private newsgroups that belong to the `acme.com` domain.

Newsgroups are organized into subject hierarchy. The first few letters of the newsgroup name indicates the major subject category; sub-categories are represented by a subtopic name. Users can post to existing newsgroups, respond to previous posts, and create new newsgroups. Some newsgroups have a moderator, a designated person who decides which postings to allow or to remove.

Three attributes are associated with newsgroups:

- Name
- Description

- Posting allowed

Newsgroup Parameters

Table 3–16 describes the newsgroup parameters in alphabetical order.

Table 3–16 Newsgroup Parameters

Parameter	Description	Acceptable Values	Default Value
Article Retention Day(s)	Number of days articles are stored	Number of days	None
Description	Brief description of the newsgroup	A single line of text	None
Mail Store	Mail store of the newsgroup	The mail store where the newsgroup was created	NULL
Moderated Newsgroup	If YES, this newsgroup is moderated; otherwise not.	YES or NO	NO
Moderator(s)	E-mail addresses of the moderators	Valid e-mail addresses of the moderators	None
Newsgroup Name	Name of the newsgroup	Only 7-bit English ASCII characters are allowed	None
Owner	Name of the newsgroup owner	A valid e-mail address of the group owner	None

Managing Newsgroups

You can use the WebMail client to add, edit, or delete private and public newsgroups.

Adding Private Newsgroups

To add a private newsgroup, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Private Newsgroup Management**.

3. Select the installation from the drop down list.
4. Click **Go**.
5. Click **Add**.
6. Enter information in the appropriate fields.
7. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Editing Private Newsgroup Properties

To edit a private newsgroup, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Private Newsgroup Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the private newsgroup.
6. Click **Edit**.
7. Edit the properties you want to change.
8. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Deleting Private Newsgroups

To delete a private newsgroup, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News**> **Public Newsgroup Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the private newsgroup.
6. Click D**elete**.

Adding Public Newsgroups

To add a public newsgroup, perform the following steps:

1. Select **News** > **Public Newsgroup Management**.
2. Select the installation from the drop down list.

3. Click **Go**.
4. Click **Add**.
5. Enter information in the appropriate fields.
6. Click **Submit** to commit the changes or **Cancel** to return to the previous page.

Editing Public Newsgroup Properties

To edit a public newsgroup, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Public Newsgroup Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the public newsgroup.
6. Click **Edit**.
7. Edit the properties you want to change.
8. Click **Submit** to commit the changes or **Revert** to return to the original settings.

Deleting Public Newsgroups

To delete a public newsgroup, perform the following steps:

1. Navigate to the WebMail client administration page.
2. Select **News** > **Public Newsgroup Management**.
3. Select the installation from the drop down list.
4. Click **Go**.
5. Select the public newsgroup.
6. Click **Delete**.

WebMail

The WebMail client gives users a simple and fast means to access messages and other self-service features through a web browser. A user points his browser to a predetermined URL to log in to his e-mail account and sees the inbox rendered dynamically.

All programs execute in the Oracle9*i* Application Server, including the logic to render a user's folders, messages, public directory and personal address book. There is no processing or data storage on the desktop. The browser acts merely as a keyboard and screen.

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

Tool Kit Default Settings

State File

The state file contains XML defining the navigation behavior of the WebMail client. This file facilitates defining state transitions in the client, when the user moves from one state to another, and managing the end state to present to the user.

The location of the state file for the WebMail client is determined by the `toolkit.statefile` property:

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

> **Note:** Percent symbols (%) in the `oc4j.properties` file are substituted to reflect the real directory path during installation.

The `toolkit.statefile` property is set in the `$ORACLE_HOME/j2ee/OC4J_UM/config/oc4j.properties` file.

WebMail Properties

> **See Also:** Chapter 8, "Parameters and Log Files" for detailed information on WebMail properties

4

Security

This chapter discusses Oracle Email system security.

This chapter contains the following topics:

- Overview
- SSL
- Anti-Spam
- Anti-Virus
- Virus Scrubber
- Virus Scanning and Removal through PL/SQL Scripts

Overview

E-mail system security has many aspects and implications. Each component of the system has potential vulnerabilities in addition to possible breaches through user error or violation of documented security policies. Examples include careless password management or cooperation with deceptive phone calls purporting to be from IT workers.

Security issues include:

- Components of the e-mail system to identify the areas to consider when designing a secure architecture
- Elements of security to consider in the design, such as password policies.

Email System Component Security

Each core component of an email system has unique security issues and vulnerabilities that you must address in designing your system and security policies. Security decisions must often balance the goals of maximum protection and unlimited access. Most decisions that increase security inevitably reduce the level of access for ordinary users.

The core components are the message store, the middle tier, the identity management infrastructure, and the mail clients.

Message Store

Table 4–1 describes the elements providing security in the message store.

Table 4–1 Security Components of the Message Store

Message Store Element	Security Effect
Oracle9*i* database	Traditional database security prevents unauthorized access.
Data access management	Normal database authentication mechanisms protect e-mail, too, and can be restricted to specific accounts or trusted users.
Signed e-mail and S/MIME support	Mail clients can provide e-mail security with digital signatures and S/MIME, part of an overall security strategy supported by Oracle preserving message integrity.

Middle Tier

In the middle tier, more vulnerabilities arise, because this is the access point for most users. Security concerns and ease of use for normal end-users must be balanced based to build a workable implementation.

Since protocol servers, such as IMAP, POP, or SMTP, are potential targets for attack by hackers looking for security weaknesses, you should enable only the protocols you require. Where appropriate, enforce authentication for all client connections, and consider using SSL for the underlying network connection. For SMTP, authentication can prevent inbound mail traffic. In this case, ensure that the anti-spam, anti-relay, and anti-virus controls are setup appropriately to minimize the risks posed by incoming mail traffic.

For HTTP servers, only minimum information should be available through any web servers giving access through web clients. Lock down access to any URL except for the thin client. To protect the security of e-mail date and password, enable SSL access only through the thin clients.

Providing adequate security of the middle tier, particularly the SMTP server, can be problematic because by design, SMTP accepts and routes inbound traffic to its destination. While this design makes mail exchange possible, it also provides possible avenues of attack. Restrictions in the openness of the SMTP server should be weighed against the loss of usability.

SMTP mail is inherently insecure because it enables users to negotiate directly with receiving and relaying SMTP servers. Sophisticated users can create messages that will trick a naive recipient into believing that they came from somewhere else. Constructing such a message so the trickery cannot be detected by an expert is more difficult, but not so much as to deter a determined and knowledgeable hacker.

Consequently, as knowledge of Internet mail increases, so does the knowledge that, at the transport level, SMTP mail inherently cannot be authenticated, nor can integrity checks be provided. Real mail security lies only in end-to-end methods that secure message bodies by using digital signatures, such as PGP or S/MIME.

Identity Management Infrastructure

This infrastructure controls and manages all aspects of directory, authentication, and single sign-on (SSO) operations.

Database security rules protect the underlying Oracle*9i* instance. Middle tier and storage servers, as well as web clients, must access this information to operate. Access is required though LDAP and possibly HTTP/HTTPS, and access to these protocols should be limited to only those servers truly requiring such access.

If web clients are deployed on the public Internet, the SSO components should be implemented on servers separate from the rest of the infrastructure. This separation makes it possible to provide protection for these components behind firewalls.

Because end user passwords are managed by the infrastructure, password policies should be maintained, such as enforcing acceptable password sizes, randomness, and frequency of change.

Unused or inactive mailboxes should be routinely cleaned or locked to minimize the risk of unauthorized use.

Mail Clients

Most mail clients have configuration options enabling support for increased levels of security when connecting to the server. For example, support for connections over SSL and protocol authentication can require special configuration. Ensure that users are aware of risky behaviors, such as storing passwords in ordinary files on PCs, and the configuration options or changes required to minimize those risks.

Network Security

Security features of the product enable separate components to be configured securely. The more restrictive access to an organization's network is the messaging system's security. Ensuring that the rest of your intranet is secure reduces the chances of unauthorized attempts to access components of the messaging system.

Firewalls

Firewalls play a large role in protecting the security of your implementation. Firewalls must be configured appropriately, with more than firewall in place, and regularly monitored for intrusion. It is important not to assume that everything is safe because you have deployed a firewall.

You have to determine what protocols to enable at the various points in your network. This decision often requires evaluating the trade-off between providing wide access to legitimate users and yet still protecting vulnerabilities from unauthorized use.

At a minimum, sending and receiving e-mail messages from the Internet requires that you enable inbound and outbound connections through port 25, the default SMTP connection socket. For other protocols, such as IMAP and POP, determine whether enabling public Internet access is worth the risks and cost. The risks include unauthorized access, and the costs include extra configuration and administration to maintain this infrastructure. In a typical enterprise, such access is not required, and all protocol tiers can be well protected within an intranet environment. Access for non-office based workers can be managed through a separate VPN or remote access infrastructure.

If you enable access through any of these protocols from the Internet, then security can be improved by using authentication and SSL. Authentication provides some protection to the protocols, and SSL aids data encryption of network traffic.

The storage component of Oracle Email should be located behind any firewall implementation, with minimal access through SQLnet from the middle tier processes in other parts of the DMZ.

You should close down all firewall access other than explicitly required port and host connections. Closely managing and minimizing potential security vulnerabilities is a key part of any secure configuration.

Never assume that implementing a secure configuration means no new vulnerabilities and risks can arise. Watch for security updates from Oracle and security updates affecting Internet protocols to ensure that you maintain a secure environment.

Non-Technical Considerations

Any security implementation is only as good as its users' awareness of security issues. Many security breaches are the result of simple human factors allowing intruders to gain access to user accounts through simple human deception. Administrators must keep the following factors in mind:

- Understand who has access to sensitive information.
- Understand that database administrators can generally access trusted level information.
- Implement password policies: minimum lengths, frequent changes.
- Remove unused accounts.
- Do not start unused services: run only what you need.

Establish security policies for each level of security that applies to different parts of your system, including who has access to them and how to respond to security breaches.

SSL

Secure Sockets Layer (SSL) is a protocol 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

For servers to communicate securely with clients, customers installing Oracle Email must obtain an SSL Server Certificate for each machine and configure its 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 run. A single certificate can support all protocol server processes on one machine.

In the Oracle environment, you can use the Oracle Wallet Manager for secure creation and storage of certificates and the corresponding private keys.

To obtain a certificate, use the Wallet Manager as described in the *Oracle Advanced Security Administrator's Guide*. The general steps are as follows:

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

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

Manually Setting Wallet Location and Client Authentication

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
```

A typical directory parameter value looks like the following line:

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

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`.

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

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.

Two separate server instances are necessary to use both SSL and non-SSL connections. One server instance cannot manage both types of connections. By

default, server instances are configured to manage Internet connections only. The default listening end points for both IMAP and POP protocol servers are created in the `listener.ora` file during installation.

To configure a SSL server instance, do the following steps:

1. Log in to Oracle Enterprise Manager.
2. Select the application server instance where Oracle Email is installed.
3. Click **Oracle Email**.
4. Click **IMAP** or **POP**.
5. Click the process instance.
6. Select **IMAPSSL**, **POPSSL**, or **Custom** from the drop-down list.

   If you select Custom:

   - Provide a specific presentation name in the corresponding field.
   - Change the `SSL Enabled` parameter to TRUE and verify that there is a description entry in the listener.ora file for the presentation name you specified. Verify that protocol is set to TCPS and that the PORT set to the default SSL port number of the protocol.

   The default SSL port number for IMAP is 993. The default SSL port number for POP is 995.
7. Reinitialize the server instance.

Configuring SSL from Protocol Servers and Oracle Internet Directory

All server process instances have a description parameter. The following parameter shows a sample set of configuration options:

```
-sslenable=yes -sslport=3061 -sslwalletloc=file:/directory_
name/oracle/work/genwallt/client
-sslwalletpassword=welcome12 -sslmode=3
```

Where:

- `sslenable` has two possible values: yes or no
- `sslport` is the port on which Oracle Internet Directory is listening. Oracle Internet Directory should be configured to listen on both SSL and non-SSL mode
- `sslwalletloc` is the location of the client wallet in the file system.

- `sslwalletpassword` is the password used to access the wallet.
- `sslmode`: valid values are
  - `1`: SSL without authentication is performed (no PKI certificate exchanges), only channel encryption and data integrity
  - `2`: One-way authentication is performed. For example: Oracle Internet Directory server authentication
  - `3`: Both the client protocol servers and Oracle Internet Directory server authentication are performed

If any values are not stored correctly, a `non!-ssl mode of auth` occurs.

> **See Also:** Oracle Internet Directory Administrator's Guide for more information regarding Oracle Internet Directory SSL.

Configuring SSL for WebMail

Perform the following steps to configure SSL for WebMail:

1. In the `toolkit.properties` file, set the `oracle.mail.ldap.connectssl=true/false` property to `true.`
2. Place the `oracle.mail.ldap.connectssl=true/false` property in the `$ORACLE_HOME/oes/admin/oesadmin.properties` file

Anti-Spam

The SMTP server supports a variety of anti-spam methods to prevent users and domains from misusing the e-mail system. Examples include flooding the e-mail system with undesired, unsolicited messages, and using the server as a spam relay for other domains.

Third Party Anti-Spam Filters

A third party anti-spam filter agent can be run in front of the SMTP server to check whether incoming messages are spam. After completing the spam check it passes the mail to the SMTP server. Anti-spam filters are configured to either reject spam mails or to change or add headers to indicate that the mail could be spam.

This release of Oracle Email does not process any specific spam headers. However, because the format and values of the new headers are known, a user can set up

server-side or client-side rules to move spam mail out of the `INBOX` or delete them based on certain criteria.

Setting Up Third Party Anti-Spam Filters

If the third party spam filter and SMTP server are running on the same machine, the filter should listen to the default SMTP port 25. The SMTP server listens to a different port. When messages come into the system, the spam filter filters the mail and takes action on it. If the mail passes the spam check, it is sent to the SMTP server. The communication between the third party spam filter and SMTP server must be done over SMTP protocol.

Native Anti-Spam

The SMTP server supports native anti-spam checks, which are more efficient than third party anti-spam filters because they eliminate the costs of external agent execution and passing mail. Checks are performed on each input from the sender to identify spam mails at an earlier stage. Native anti-spam checks do not analyze message contents, but all mail messages are checked for the following:

- Sender's address
- Sender's domain
- Recipient's address
- Recipient's domain
- IP address of the sending computer
- Domain of the sending computer

The server supports two types of settings: an acceptance list and a rejection list. The values in these lists are domains, IP addresses, and senders. The appropriate accept and reject list is checked against the SMTP command being processed.

Anti-spam checks are performed during the following operations:

- When the SMTP connection is opened
- When the client connects
- After each SMTP protocol command, in the following order:

  `HELO/EHLO, MAIL FROM, RCPT TO`

Typically, SMTP servers inside a firewall do not need to have anti-spam checking enabled. Outside the firewall, however, servers receiving inbound mail messages need anti-spam protection.

Oracle Email servers have a global `native anti-spamming` parameter that is checked by each instance. If this value is set to `FALSE`, anti-spam checking does not take place and all other parameters for anti-spam are disabled.

If native anti-spam is set to `TRUE`, the following steps occur:

1. The Internet Protocol (IP) address and domain name (based upon a DNS lookup of the IP address) of the requester are checked during the connection request to the server. The following parameters are checked:
   - `Accept Connections from IP Addresses`: If the IP address is trusted, then the process continues
   - `Reject Connections from IP Addresses`: If the IP address is here, the message is rejected and the connection is closed
   - `Accept Connections from Host Domains`: If the domain name of the computer is trusted, then the process continues. The domain name is obtained through a reverse DNS lookup of the IP address
   - `Reject Connections from Host Domains`: If the domain name is in this list, then reject the message and close the connection
   - `Prevent Service Denial Attack`: The number of messages plus the number of connection requests from this host within a time interval that is considered to be flooding
   - `Spam Flood Interval`: The time interval, in minutes, used in conjunction with `Spam Flood Count` parameter to determine whether a host spamming
2. The `HELO` or `EHLO` command is performed. This is the initial command performed before any work can begin on the SMTP server. When this command is entered, a domain name is passed as part of the command. The following parameters are set:
   - `Accept Connections from IP Addresses`: If the IP address is trusted, the process is continued
   - `Accept Connections from Host Domains`: If the domain name of the computer is obtained through reverse DNS lookup of the IP address are trusted, then the process continues

- Enable HELO DNS Check: If this parameter is set then the domain name in the helo/ehlo command is checked for existence in the DNS server. If it does not exist, the connection is rejected

3. The information in the MAIL FROM command is verified. This command contains the e-mail address of the sender. This address can be checked for spam. The following parameters are set:
   - Accept Connections from IP Addresses: If the IP address is trusted, the process is continued
   - Accept Connections from Host Domains: If the domain name of the computer is obtained through reverse DNS lookup of the IP address are trusted, then the process continues
   - Reject Connections from Sender Domains: If the domain name part of the sender's e-mail address is in this list, the message is rejected and the connection is closed
   - Reject Connections from Sender: If the sender's e-mail address is in this list, the message is rejected and the connection is closed
   - Accept Connections from Sender Domains: If the domain part of the sender's e-mail address is in the list, the process is continued
   - Accept Connections from Sender: If the sender address is a trusted address, the process is continued
   - Enable Sender DNS Check: If this is enabled, then the domain in the sender's address is checked to determine if it exists in the DNS server
4. An additional check for flooding takes place. This is required because a single connection to a message transfer agent (MTA) can send multiple messages. A flood check is performed after each message is accepted.
   - Prevent Service Denial Attack: The number of messages plus the number of connection requests from this host within a time interval that is considered to be flooding.
   - Spam Flood Interval: The time interval, in minutes, used in conjunction with Spam Flood Count to determine whether a host is spamming.
5. The RCPT TO command is verified. This command contains the e-mail address(es) of the recipient(s). This check is dependent on several parameters and differs depending on the mail destination. Again if the sending computer is trusted then the mail continues. Then each recipient is either a local user or the mail message needs to be relayed to another SMTP server. If the mail is going to

be delivered to a local user, then a check for rejected recipients is made. If the mail is to be relayed, then we check to make sure the server is allowed to relay, or allowed to relay conditional upon the domain the mail is going to as well as if the connection was initially authenticated.

- `Accept Connections from IP Addresses`: If the IP address is trusted, the process is continued.
- `Accept Connections from Host Domains`: If the domain name of the computer or the domain name sent as part of the connect request are trusted, the process is continued.
- `Accept Connections from Sender Domains`: If the domain part of the sender's e-mail address is in the list, the process is continued.
- `Accept Connections from Sender`: If the sender address is a trusted address, the process is continued.
- `Reject Recipient`: This parameter list is only used for local delivery mail messages. If the recipient name is in this list, then message is rejected and the connection is closed. This is useful for temporarily suspended accounts or restricted distribution lists.
- `Relay Allowed`: This parameter is only used for relay delivery of mail messages. The possible values are:
  - `True`: The recipient domain is checked to see if it is in the list of domains allowed to relay
  - `False`: Relay messages are not allowed
  - `Auth`: If the sender is authenticated when it first connected to the SMTP server, then the mail is allowed to continue
- `Relay Domains Allowed`: This value only reads relay mail messages when the `Allow Relay` parameter is set to `True`. This parameter provides a list of domains that the SMTP server allows to be relayed. If relaying for all domains is allowed, the parameter must be set to an asterisk (*).

Native Anti-spam Parameters

The following table describes the Oracle Email native anti-spam parameters:

Table 4–2 Anti-spam Parameters

Parameter	Description
Active	Turns on anti-spamming checks. If this parameter is not set, all anti-spamming checks are turned off.
Relay Blocking	Enables domains determined by SMTP Relay Domains Allowed.
Prevent Denial of Service Attacks	
Reject connections from sender(s)	This parameter specifies the list of senders to be rejected.
Accept connections from sender(s)	This parameter specifies the list of senders to be accepted.
Reject connections from sender domain(s)	This parameter specifies the list of domains and sub-domains to be rejected.
Accept connections from sender domain(s)	This parameter specifies the list of domains and sub-domains to be accepted.
Reject connections from Host domain(s)	This parameter specifies a list of rejected domains or sub-domains from which mail is received.
Accept connections from Host domain(s)	This parameter specifies a list of allowed domains or sub-domains from which mail is received, regardless the criteria.
Reject connections from IP address(es)	List of IP addresses from which connections are not allowed.
Accept connections from IP address(es)	List of IP addresses from which connections are permitted.
Reject recipient(s)	This parameter specifies the list of recipients to be rejected.
Allowed Relay Domain(s)	This parameter specifies the list of domains to relay through.
DNS check on helo/ehlo domains	This parameter specifies if the domain in the `helo` command should be checked in the DNS server.
DNS check on sender domains	This parameter specifies if the domain in the `mail from` command should be checked in the DNS server

Wildcard Support

Wildcard support exists on prefixes for domains and suffixes for IP addresses, to allow sub-domains and sub networks with a single entry. Entries containing only an asterisk (*) on a line by itself means all domains or IP addresses.

For example:

Valid entries `*.foo.com` (all sub-domains of `foo.com`), `*.*.foo.com`, `99.99.99.*` (any host with IP address having a prefix `99.99.99`);

Invalid entries: `*.foo.*.com` (domain), `99.*.99.*` (IP).

Adding Anti-spamming to Servers

1. Navigate to the WebMail client administration page.
2. Select **Policy** >**Anti-Spam**.
3. Select the server on which you want to add anti-spamming.
4. Select SMTP inbound or NNTP inbound from the drop down menu.
5. Enter information in the appropriate fields.
6. Click **Apply to Servers**.
7. Select the server from the list.
8. Click **OK**.
9. Restart the SMTP or NNTP server to make the changes take effect.

> **Note:** These parameters can also be modified through the Oracle Enterprise Manager administration pages.

Anti-Virus

The Oracle Email SMTP server and message store support plug-ins for third party anti-virus solutions to detect and cleanse, or delete potentially destructive messages. The message store has the capability to re-scan messages already in the store for potential virus. There are two ways to integrate with the virus scanner:

- External Filter Process: write a process that can be called from the server for virus scanning

- C Callouts: Implement the virus scanner as a C language procedure and link it with the server

External Filter Process

The SMTP server provides a way to integrate external processes to the server. The SMTP server for each mail calls this process for each mail process. Both the message body and envelope are passed through the process. The external process can then apply its own filtering mechanism and pass back a success or a failure message to the SMTP server. If a failure message is received from the external filter, the message is rejected. The external filter can also pass modified mail back to the server and that is delivered to the recipients. If the external process fails, the messages are queued and retried after the queue retry interval.

Configure the following parameters to set up external filter processing:

- `External Filter`: A Boolean that indicates if external filter processing is turned on.
- `External Filter Process`: This parameter must be in the following format:

  `name:path_name when_to_call flags`

where:

Table 4–3 External Filter Process Parameters

Parameter	Description
`name`	The name of the external filter.
`path_name`	The complete path of the process to be called.
`when_to_call`	The point at which the callout should be called. The possible values are: ■ `ENV` – After receiving the message envelope ■ `DATA` – After receiving the complete message and before local delivery ■ `NEVER` – Disables the callout ■ `RELAY` - Before relaying a message
`System Flag(s)`	Can be set to `repairmsg=1` or `repairmsg=0`. If set to `1`, then the callout can send the repaired message back to the server. If it set to `0`, the server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure. **Note:** In the WebMail client administration page, this parameter is a radio button called `Repair`.

The call to the external process has the following syntax:

```
Filter_process host=host mailfrom=mailfrom rcptto=[#_of_recipients]
msgsize=[msg_size]
```

where:

Table 4–4 External Filter Process Parameters

Parameter	Description
`filter_process`	The process set in the `External Filter Process` parameter
`host`	The name of host of the client connection
`mailfrom`	The address sent by the client during the SMTP protocol exchange
`rcptto`	The list of recipients from the SMTP protocol exchange. This is a list separated by a comma and is enclosed in brackets.
`msg-size`	The size of the message. It is optionally passed depending on `when_to_call` parameter. If the `when_to_call` parameter is set to `ENV` then this is not passed.

After the filter process has finished processing the mail, it should return the status and the changed message back to the server. This is done by writing the following information to its standard output (`stdout`)

```
status_code [version_definition]
repaired message
```

where:

Table 4–5 External Filter Process Parameters

Parameter	Description
status_code	The possible values are: ■ 0 (success) The message is clean and is to be sent to the recipient ■ 1 (failure) The message is not clean and is to be rejected ■ 2 (repaired) The message was not clean but was changed is to be sent to the recipients

Table 4–5 External Filter Process Parameters

Parameter	Description
version_definition	This parameter that can be returned by the filter process is stored with the message. An example would be to keep the virus definition database identifier in it. Because it stored with the message, it can be used later in the virus scrubber process to selectively re-scan messages.
repaired message	The modified message that should be sent to the recipients

Adding an External Filter Process

1. Navigate to the WebMail client administration page.
2. Select **Policy** > **Anti-Virus**.
3. Select the server on which you want to add anti-virus.
4. Select SMTP inbound, SMTP outbound, or list server from the drop down menu
5. Click **Go**.
6. Click **Add**.
7. Enter the filter name.
8. Select **External Filter**.
9. Enter information in the appropriate fields.
10. Click **Submit** to commit the changes or **Cancel** to return to the previous page.
11. Restart the SMTP server to make the changes take effect.

Table 4–6 External Filter Process Parameters

Parameter	Description
`Name`	The name of the external filter.
`Location`	The complete path of the process to be called.

Table 4–6 External Filter Process Parameters

Parameter	Description
When to Call	The point at which the callout should be called. The possible values are: ■ ENV – After receiving the message envelope ■ DATA – After receiving the complete message and before local delivery ■ NEVER – Disables the callout ■ RELAY - Before relaying a message
System Flag(s)	Can be set to repairmsg=1 or repairmsg=0. If set to 1, then the callout can send the repaired message back to the server. If it set to 0, the server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure. **Note:** In the WebMail client administration page, this parameter is a radio button called Repair.
Scanner Flags	Specifies any flags that must be set for the scanner. **Note:** In the WebMail client administration page, this parameter is a called Additional Flags.

External C Callouts

In addition to external process callouts, the SMTP server can call C language procedures. These procedures can be used as an inexpensive and efficient way of filtering mails. The C callouts that are implemented can obtain various message parts, such as envelope, size, and message text, through the API provided by the server. The procedure then filters and returns a reject or success with an optional modified message to the server.

Since these are linked with the server there is no overhead in the calls. But at the same time care must be taken while implementing these since they share the same process and memory space as the server. Any corruption affects the server. Each callout needs to implement a pre-defined set of functions. These functions are called by the server to initialize, send message and receive status and message back from the callout. The callouts can be specified by setting values in the Scanner Interface parameter. Multiple callouts can be specified and are called in sequence.

Each callout is specified as:

```
name:shared_library_path, when_to_call, [internal | host:port], (init, register_
callback, scan_msg, send_msg, receive_msg, close),scanner_flag,system_flags
```

where:

Table 4–7 External C Callout Parameters

Parameter	Description
`name`	The name of the external filter.
`shared_library_path`	The full path of the C shared library. This is loaded by the server at startup.
`when_to_call`	The point at which the callout should be called. The possible values are: ■ `ENV` – After receiving the message envelope ■ `DATA` – After receiving the complete message and before local delivery ■ `RELAY` – Just before relaying a message ■ `NEVER` – Essentially disables the callout
`internal or host:port`	If a host and port are needed by the scanner, enter the host name and port number of the scanner. These are passed to the scanner initialization function. If the scanner does not need a host and port, select internal.
`System Flag(s)`	Can be set to `repairmsg=1` or `repairmsg=0`. If set to `1`, then the callout can send the repaired message back to the server. If it set to `0`, the server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure. **Note:** In the WebMail client administration page, this parameter is a radio button called `Repair`.
`Scanner Flags`	Specifies any flags that must be set for the scanner. **Note:** In the WebMail client administration page, this parameter is a called `Additional Flags`.

Adding an External C Callout Procedure

1. Navigate to the WebMail administration page.
2. Select **Policy** > **Anti-Virus**.
3. Select the server on which you want to add anti-virus.
4. Select SMTP inbound, SMTP outbound, or list server from the drop down menu.

5. Click **Go**.
6. Click **Add**.
7. Enter the filter name.
8. Select **Callout**.
9. Enter information in the appropriate fields.
10. Click **Submit** to commit the changes or **Cancel** to return to the previous page.
11. Restart the SMTP server to make the changes take effect.

Table 4–8 External C Callout Process Parameters

Parameter	Description
`Name`	The name of the external filter.
`Location`	If a host and port are needed by the scanner, enter the host name and port number of the scanner. These are passed to the scanner initialization function. If the scanner does not need a host and port, select internal.
`Library Path`	The path of the C Callout library.
`When to Call`	The point at which the callout should be called. The possible values are: ■ `ENV` – After receiving the message envelope ■ `DATA` – After receiving the complete message and before local delivery ■ `NEVER` – Disables the callout ■ `RELAY` - Before relaying a message
`Initial Function`	The initialization of the scanner.
`Register Callback Function`	The callback registration function.
`Scan Function`	The function to scan the message.
`Send Function`	The function used to get the message body
`Receive Function`	The function used to send the repaired message.
`Close Function`	The scanner exit function.

Table 4–8 External C Callout Process Parameters

Parameter	Description
`System Flag(s)`	Can be set to `repairmsg=1` or `repairmsg=0`. If set to `1`, then the callout can send the repaired message back to the server. If it set to `0`, the server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure. **Note:** In the WebMail client administration page, this parameter is a radio button called `Repair`.
`Scanner Flags`	Specifies any flags that must be set for the scanner. **Note:** In the WebMail client administration page, this parameter is a called `Additional Flags`.

Applying an Existing Anti-virus Policy to a Service Process

1. Navigate to the WebMail client administration page.
2. Select **Policy** > **Anti-Virus**.
3. Select the server on which you want to add anti-virus.
4. Select SMTP inbound, SMTP outbound, or list server from the drop down menu
5. Click **Go**.
6. Select the service name from the drop down list.
7. Click **Apply to Servers**.
8. Select the Server name(s) on which the service is running.
9. In the **Override Default Process settings?** field, select **Yes**, if you want to override the process instance values set in Oracle Enterprise Manager. Otherwise, select **No**.
10. Click **OK** to commit the changes or **Cancel** to return to the previous page.

Note: Applying an existing anti-virus policy to a service, overwrites the policy currently applied to that service.

Anti-Virus with Symantec

The Oracle Email SMTP server and mail store can integrate with Symantec's Anti Virus Scan Engine (SAVSE). This enables Oracle Email to use Symantec's virus

knowledge base to detect and cleanse infected messages at both the SMTP level as well as in the mail store.

Configuring Symantec with Anti-Virus

1. Navigate to the WebMail client administration page.
2. Select **Policy** > **Anti-Virus**.
3. Select the server on which you want to add anti-virus.
4. Select SMTP inbound or outbound from the drop down menu.
5. Click **Add**.
6. Enter the filter name.
7. Select **C Callouts.**
8. Enter information in the appropriate fields. See table 4-7 for parameter descriptions.
9. Click **Submit**.
10. Select the server from the list.
11. Click **OK**.
12. Restart the SMTP server to make the changes take effect.

Table 4–9 Symantec C Callout Values

Parameter	Value
Name	The name of the external filter.
Location	The host and port number where the SAVSE is running
Library Path	*$ORACLE_HOME*/oes/lib/libessymantec.so
When to Call	DATA
Initial Function	essmasymantec_init
Callback Function	essmasymantec_register_cb
Scan Function	essmasymantec_scanmsg
Send Function	essmasymantec_send
Receive Function	essmasymantec_recv
Close Function	essmasymantec_close
System Flag(s)	Can be set to repairmsg=1 or repairmsg=0. If set to 1, then the callout can send the repaired message back to the server. If it set to 0, the server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure. **Note:** In the WebMail client administration page, this parameter is a radio button called Repair.
Scanner Flags	Specifies any flags that must be set for the scanner. **Note:** In the WebMail client administration page, this parameter is a called Additional Flags.

Virus Scrubber

The Oracle Email virus scrubber is a server process that scans for and cleans up virus-infected e-mail messages already in the message store. When rapid measures are required to immediately cleanse a system of virus-infected messages, the virus scrubber pre-scans a message store to isolate suspect messages already there based on headers such as subjects or attachment names. Pre-scanning isolates suspect messages so that users are not able to access them and possibly cause damage. Pre-scanning never deletes a message. After pre-scanning, the virus scrubber uses the external scanner to individually scan the isolated messages. A message that is deemed clean or repaired by the virus detection software is restored to its original folder.

Note: Although pre-scanning is a much faster way to isolate suspect messages in a message store than scanning all individual messages, it can quarantine clean messages.

If a message is identified as infected and not repairable, the administrator can either delete the message immediately or quarantine it to a special folder for later processing. For example, an infected message can be quarantined to wait for a future release of virus definitions that may be able to repair the message. Oracle Email can be configured to send a message to either the mail recipient or sender notifying them that it was identified as infected. Such notifications are useful to explain to users why their messages disappeared.

The virus scrubber is different from the SMTP based virus scanner that filters out virus-infected messages before they enter the system. The Oracle Email virus scrubber is a necessary complement to the SMTP virus scanner because new types of viruses continue to pop up before virus detection software can be updated to detect and repair them. There's always a possibility that by the time virus software is updated, some infected messages have already entered the system. The virus scanner can be used to retroactively rid the system of such viruses. This message store-based scanner can also be used to scan viruses coming in through a non-SMTP route such as IMAP append.

The Oracle Email virus scrubber and SMTP-based virus scanner rely on external virus detection and cleanup software such as the Symantec AntiVirus Scan Engine. Oracle Collaboration Suite provides interface libraries for third party virus tools to integrate with Oracle Email. The third party virus software must be installed properly for the virus scrubber server to be fully functional.

To run the virus scrubber only once, the process LDAP parameter `orclMailProcExecutionMode` parameter must be set to the number 1 to designate "run-once." To run the virus scrubber until the process is killed, set `orclMailProcExecutionMode` parameter to number 2.

The virus scrubber can be run as either a daemon process that runs forever or a standalone process that runs once and exit. A typical process configuration may contain one process configured as a daemon that wakes up and scan the message store monthly and another instance configured as a standalone server that can be run on demand.

The virus scrubber log files are located in the following directories:

On UNIX:

`$ORACLE_HOME/oes/log/install_name/vs/pid/pid.log`

On Windows:

`%ORACLE_HOME\oes\log\install_name\vs\pid\pid.log`

Configuring the Virus Scrubber Through WebMail

1. Navigate to the WebMail client administration page.
2. Select **Policy** > **AntiVirus** > **Mail Store**
3. Select the host name from the drop-down list.
4. Click **Go**.
5. Select **Add**.
6. Enter Information in the appropriate fields. Please refer table 4-10 for field descriptions.
7. Click **Apply to Servers**.
8. Select the server(s) for which you want to apply virus scanning.
9. Click **OK** to commit the changes or **Cancel** to return to the previous page.

Note: Selecting more than one server to apply the policy to overwrites the old configuration and sets the common configuration for all.

Configuring Virus Scrubber Through Oracle Enterprise Manager

Note: The Symantec AntiVirus Scan Engine can be purchased by contacting Symantec Corporation.

Perform the following steps to configure the virus scrubber:

1. Install and configure Oracle Collaboration Suite, Release 2 (9.0.4).
2. Install the Symantec Anti Virus Scan Engine (SAVSE) according to the product installation instructions.
3. Using Oracle Enterprise Manager, set the `scanner interfaces` parameter to a value in the following format:

```
(sym_api:midtier$OHpath/oes/lib/libessymantec.so,DATA,symantechost:symantecp
ort,(essmasymantec_init,essmasymantec_register_cb,essmasymantec_scanmsg,essm
asymantec_send,essmasymantec_recv,essmasymantec_close),true,repairmsg=1)
```

where:

Table 4–10 Virus Scrubber Parameters

Parameter	Description
symantechost	The fully qualified host name of the machine where SAVSE is installed.
symantecport	The port number that SAVSE is listening on.
midtier$OHpath	The expanded path of your middle tier `$ORACLE_HOME`. For example: `/disk1/app/oracle/product/9.0.4`

Note: Apart from three variable values shown above, the string should be exactly one single line of text.

The following virus scrubber process parameters can be accessed through Oracle Enterprise Manager:

Table 4–11 Virus Scrubber Parameters

Parameter Name	Valid Values	Default Value	Remarks
Execution Mode	Daemon	Run Once Daemon	Determines whether the server should run once and exit or stay active in the background forever (also known as the daemon mode). If a process is set to run as a daemon, it sleeps after one round of execution before starting the next round. In Run Once mode, it simply exits after the current task is finished.

Table 4–11 Virus Scrubber Parameters

Parameter Name	Valid Values	Default Value	Remarks
Concurrency Level	A positive number	10	Specifies the number of messages that should be scanned concurrently. This parameter greatly depends on the third party virus scanner software setup. It is recommended that this parameter be set within the level of concurrency supported by the third party software. For example, if the Oracle Email system is configured to use a Symantec SAVSE server that can handle 1000 concurrent virus scanning requests, the entire Oracle Email system, including the SMTP-based virus scanner, should be configured to submit roughly the same or lower number of concurrent requests to SAVSE, such as 500.
Pre-Scan	Enabled or Disabled	Enabled	Turns pre-scanning on and off. Turn on Pre-Scan to quickly isolate potential virus messages by header searches. At least one instance of virus scrubber should have pre-scan enabled since message-based scanning does not occur unless messages are first isolated by a pre-scan operation.
Pre-Scan filter	String	ALL	String containing the search criteria to be used to pre-scan messages. Search filters are specified using the IMAP search command syntax. Please refer to RFC 2060 for details of valid search clauses. For example, a search on all messages from `acme.com` with a subject "Snow White" uses the filter string 'SUBJECT "Snow White" FROM `acme.com`'.
Repair Mode	Direct or Quarantine	Direct	Specifies how to deal with messages identified by the scanner as infected or not repairable. ■ `Direct` causes the server to remove the message directly. ■ `Quarantine` causes the server to move the message to a special folder defined by the Quarantine Folder parameter and owned by Quarantine User. If repair mode is not specified, the default is direct. If `Quarantine User` or `Quarantine Folder` is not specified, but `Repair Mode` is set to "quarantine," then it is an error. The server stops processing messages until the error is corrected.

Table 4–11 Virus Scrubber Parameters

Parameter Name	Valid Values	Default Value	Remarks
Quarantine User	String	None	Identifies the owner of the Quarantine Folder into which quarantined messages should be moved. Typically this is a user account belonging to an administrator or a dedicated e-mail account for holding infected messages. If `Quarantine User` or `Quarantine Folder` is not specified, but `Repair Mode` is set to "quarantine," then it is an error. The server stops processing messages until the error is corrected.
Quarantine Folder	String	None	Specifies the folder into which quarantined messages should be moved. This folder must already exist under the Quarantine User account. If `Quarantine User` or `Quarantine Folder` is not specified, but `Repair Mode` is set to "quarantine," then it is an error. The server stops processing messages until the error is corrected.
Sender Notification	Text String	None	Contains the notification message body that the server should send to the sender of any virus-infected message. If this parameter is not set, no notification message is sent.
Recipient Notification	Text String	None	Contains the notification message body that the server should send to the recipients of any virus-infected message. If this parameter is not set, no notification message is sent.

Command-line

To run the virus scrubber, use `oesctl` startup with a service type of `vs`.

```
oesctl startup target | instance
```

For the virus scrubber, the syntax for *`target`* is

```
hostname:um_system:service_type
```

where *`hostname`* is the name of the server on which the process should run
`service_type` `vs`

The syntax of *`instance`* is *`target`*:*`instance_ID`*

where *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.

Example 1

To start the virus scrubber for all registered processes for the server named Acme, enter the following command:

```
oesctl startup acme:um_system:vs
```

Example 2

To start the virus scrubber for the process with ID `104750025197124824` on the server named Acme, enter the following command:

```
oesctrl startup acme:um_system:vs:104750025197124824
```

Virus Scanning and Removal through PL/SQL Scripts

E-mail viruses typically have the form of an executable program, such as an e-mail attachment. The program is 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 provides several different tools of virus protection, each of them suited for a different type of administration requirement.

The Oracle Email server SMTP inbound process provides integration with third party virus scanning software to scan each message that passes through the SMTP server. The server rejects the message upon arrival, preventing the virus e-mail from entering the email system.

If third party virus scanning software is not available, Oracle Email server can still reject virus messages using server side rules. Server side rules reject incoming messages based on suspicious subject lines, attachment names or sender information.

See Also

Chapter 7, "Command Line Interface" for more information on how to create system wide rules using `OESRL`

If there is a virus outbreak before the SMTP server has a chance to upgrade itself to use the latest third party software, then some virus e-mail messages are already present in user's mailbox. The Oracle Email virus scrubber process can be used to

scan the entire message store, repair or remove virus e-mail messages once the third party software is upgraded to date.

Oracle Email has a simple PL/SQL utility package MAIL_AV that scans the message store based on simple message attributes such as subject line and attachment names. To use this package, one simply writes a SQLPLUS script that uses this package or execute procedures in this package directly from SQLPLUS.

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

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

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

The p_pattern parameter is the identifying string for the virus. The p_folder parameter 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',
'/sysadmin/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 January 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"', '/sysadmin/infected');
end;
/
```

Restore

There are two procedures to restore quarantined messages 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.

5

Backing Up and Recovering Oracle Email

Disaster planning is a critically important aspect of administering an e-mail system. This chapter describes how to back up and recover Oracle Email.

This chapter contains the following topics:

- Overview of Oracle Email Backup and Recovery
- Backing Up and Recovering the Database
- Backing Up and Restoring User Data with oesbkp
- Recovering Messages with LogMiner
- Recovering Messages with Flashback Query

Overview of Oracle Email Backup and Recovery

System files and the mail store itself must be backed up regularly using the standard Oracle database tools in order to be able to recover all or part of the e-mail system and data if a disaster occurs.

In addition to the standard database backup and recovery tools, the following processes and tools allow you to backup and recover data on a more granular level:

- The `oesbkp` utility backs up and recovers folders, messages, address book entries, and server side rules for individual users
- LogMiner based mail recovery recovers deleted messages using database redo logs
- Flashback query based mail recovery recovers messages using Oracle Flashback Query

Backing Up and Recovering the Database

Oracle Email uses the Oracle database as its mail store, so standard database backup and recovery methods can be used to maintain copies of the most current contents of the e-mail system. It is important to perform a full system backup at regularly scheduled times so that the entire Oracle Email system can be restored to that snapshot if needed. Partial backups of the system can be performed between full backups so that the system can be recovered to a more recent point in time.

Oracle database backup methods include:

- Export backup, which is appropriate for small Oracle Email systems
- Hot backup, which provides online backups and restores without shutting down the system
- Cold backup, which requires shutting down the system

Note: Individual user accounts cannot be restored from database backups. To back up and restore individual user accounts, use the `oesbkp` utility, described in "Backing Up and Restoring User Data with oesbkp".

See Also:

- *Oracle9i Backup and Recovery Concepts Release 2 (9.2)*
- *Oracle9i User-Managed Backup and Recovery Guide*
- *Oracle9i Recovery Manager User's Guide*

Backing Up and Restoring User Data with oesbkp

Individual e-mail users are categorized into two states:

- Active: Can access their mailboxes
- Inactive: Cannot access their mailboxes

Individual e-mail user accounts can be backed up and restored using the `oesbkp` command line utility. `oesbkp` restores backed-up items at different levels of granularity. For example, you can restore either an entire user account or a single folder, which is particularly useful for backing up and restoring critical information.

Note: The backup and restore functionality can be applied to inactive users.

`oesbkp` backs up the following user data into flat files:

- Folders
- Messages
- Private address book entries
- Server side rules

When restoring user accounts:

- All messages in the account are restored in a new folder to avoid overwriting existing messages
- Private address entries are restored in the user's private address book, but entries that already exist are not restored.
- A user's restored server side rules overwrite existing server side rules

The `oesbkp` syntax is as follows:

```
oesbkp task={backup | restore} user=userid@domain password=admin_password
```

```
[type={all | mail | rules | addrbook}] [admindn=database_account_with_admin_
privileges] [ldaphost=host_name] [ldapport=port_number] [backupdir=directory]
[folder=folder_name]
```

Table 5–1 oesbkp Parameters

Parameter Name	Valid Values	Default Value	Description
TASK	`{backup | restore}`	None	Indicates whether this is a backup or restore request
USER	Fully-qualified username	None	Fully-qualified name, including the domain, of the user being backed up or restored. For example a valid value is `user@domain.com`, whereas `user@domain` and `user` are not valid values.
PASSWORD	Any string	None	Password for the distinguished name (DN)
TYPE	`{all | mail | rules | addrbook}`	`all`	Indicates the objects to be backed up or restored: all (folders, messages, server side rules, and address book entries); messages only; server side rules only; or address book entries only.
ADMINDN	Any valid DN for the LDAP server	`cn=orcladmin`	DN that the tool uses to bind to the LDAP server. The DN should have admin privileges, such as `orcladmin` and `umadmin.`
LDAPHOST	Any host name	`localhost`	Name of the host where Oracle Internet Directory is installed
LDAPPORT	Any integer	`389`	Port on which Oracle Internet Directory is listening
BACKUPDIR	Any valid directory	`user.dir`	Location where the backup is being created or restored from
FOLDER	Any folder name	None	Name of the folder to be backed up or restored. If no value is specified, then all available folders are backed up or restored.

Note: When maintaining multiple backups for a particular user, ensure that each backup is named uniquely to avoid overwriting files.

Note the following information regarding `oesbkp` parameters:

- Parameter and value pairs must be specified with a blank space separating them on the `oesbkp` command line . Parameters can appear in any order.
- The following parameters are mandatory:
  - TASK
  - USER
  - PASSWORD

Logging information for `oesbkp` is stored in:

`$ORACLE_HOME/oes/log/um_system/backup/`*`number`*`/text.log`

where *`number`* is generated by the system and does not represent a process ID or other such number.

`oesbkp` creates the following backup files:

- *`user`*`_rules.xml`
- *`user`*`_addrbook.ldif`
- *`user`*`_foldermap`
- *`user_n`*

  where:

  - *`user`*`_rules.xml` contains the specified user's server side rules
  - *`user`*`_addrbook.ldif` contains the specified user's address book entries
  - *`user`*`_foldermap` file contains mapping between the specified user's files and folders, which enables the backup of folders that have names containing characters that are not supported by the operating system.
  - *`user_n`* (*`user`*`_1`, *`user`*`_2`, and so forth) represents each of the user's folders

Folders are restored in subfolders in the following user account folder:

`restore_`*`dd-Mon-yyyy hh24:mi/subfoldername`*

where *`subfoldername`* is the same name as the original folder being restored, and *`dd-Mon-yyyy hh24:mi`* shows when the restore occurred (not when the backup occurred).

If a user is over quota when the backup is performed, the over-quota messages are also backed up. If a user's folder is being restored and the messages in the folder cause it to go over quota, the restore does not occur. Check the `oesbkp` log files to view the user's current quota and usage, and if necessary, increase the user's quota before restoring messages.

The following example shows how to create a full backup of all folders, messages, and private address book entries for `john@acme.com` in the `/bkp/allbkps` directory:

```
oesbkp task=backup type = all user=john@acme.com admindn=cn=orcladmin
password=abcd ldaphost=ldap.acme.com ldapport=4032 backupdir=/bkp/allbkps
```

The folders created in the `/bkp/allbkps` directory are:

```
john@acme.com_rules.xml john@acme.com_addrbook.ldif john@acme.com_foldermap
john@acme.com_1 john@acme.com_2
```

The following example shows how to restore the messages to the inbox of `john@acme.com` from the backup stored in the `/bkp/allbkps` directory:

```
oesbkp task=restore type=mail user=john@acme.com password=abcd
backupdir=/bkp/allbkps folder=inbox
```

In this example, if the restore is performed at 2:00 AM on March 17, 2003, the messages are restored in a new folder in John's account named `inbox`, which is a subfolder of `restore_17-Mar-2003 02:00`.

Recovering Messages with LogMiner

Database redo logs record all changes made to data. If a failure prevents modified data from being permanently written to the data files, the changes can be obtained from the redo logs using the Oracle9*i* Database LogMiner feature. With LogMiner, you can use SQL to read, analyze, and interpret log files.

Whenever an Oracle Email message is deleted, the change in data is recorded in a database redo log. LogMiner lets you to retrieve deleted messages from the redo logs.

To fully translate the contents of redo logs, LogMiner requires access to a database dictionary. Without the database dictionary, LogMiner returns internal object identifiers and presents data in hexadecimal. A LogMiner dictionary file contains information that identifies the database from which it was created and the time it was created. The data dictionary must be extracted prior to using LogMiner to recover Oracle Email messages.

See Also: The following for information about LogMiner:

- "LogMiner Overview" in *Oracle9i Database Concepts Release 2 (9.2)*
- "Using LogMiner to Analyze Redo Logs" in *Oracle9i Database Administrator's Guide Release 2 (9.2)*

This section contains the following topics:

- Setting Up LogMiner to Recover Mail
- Creating the Redo List File
- Creating the Redo List File
- lmmr_setup Package

Setting Up LogMiner to Recover Mail

In order to set up mail recovery, you must enable supplemental logging for the mail store database and configure the housekeeping server to record the messages being deleted into the redo logs.

Setting up LogMiner involves the following tasks:

- Enabling Supplemental Logging
- Configuring the Housekeeping Server
- Creating the LogMiner Data Dictionary File
- Creating the Redo List File

Enabling Supplemental Logging

To enable supplemental logging for the mail store database:

1. Start SQL*Plus and log in as `sys` as follows:

   ```
   $ sqlplus sys/sys_password
   ```

2. Enter the following SQL command:

   ```
   SQL > alter database add supplemental log data (primary key,unique index)
   columns;
   ```

Configuring the Housekeeping Server

Perform the following steps to configure the housekeeping server to record the messages being deleted into the redo logs.

1. Using Oracle Enterprise Manager, navigate to the housekeeping page.
2. Click on the housekeeping instance where the `collection` parameter is configured.
3. Enable the `Record Messages Being Deleted Into Redo Logs` parameter .
4. Click **Apply**.
5. Return to the housekeeping page.
6. Click **Start**.

Creating the LogMiner Data Dictionary File

To create the LogMiner data dictionary file:

1. Start SQL*Plus and log in as `sys` as follows:

   ```
   $ sqlplus sys/sys_password
   ```

2. Execute the `dbms_logmnr_d.build` procedure with the dictionary filename and location, as in the following example:

   ```
   SQL > execute dbms_logmnr_d.build('dictionary.ora', '/oracle/database/');
   ```

3. Ensure that the initialization parameter `UTL_FILE_DIR` has access to the dictionary file.

> **See Also:** The `DBMS_LOGMNR_D` package in the *Oracle9i Supplied PL/SQL Packages Reference* for information about logminer

Creating the Redo List File

The `lmmr_setup` package uses the redo list file to obtain the list of redo logs provided by an administrator. The redo list file contains redo log file names with their full path and must be listed in separate lines.

Redo logs can be stored online, archived, or both.

Note: Comments and extra spaces are not supported in redo list files.

To create a redo list file:

1. Create the file using a text editor or a directory output list command.
2. Ensure that the initialization parameter `UTL_FILE_DIR` can access the redo list file.

The following is an example of a redo list file:

```
/oracle/database/redo01.log
/oracle/database/redo02.log
/oracle/database/redo03.log
```

Using LogMiner to Recover Mail

Perform the following steps to use LogMiner to recover messages:

1. Create a file containing the list of the redo logs to be analyzed.

   See Also: "Creating the Redo List File"

2. Connect to the database as `es_mail`.
3. To recover messages:
   - Execute `lmmr_setup.setup_logmnr`, specifying the dictionary file name, redo list file location, and redo list file name.
   - Execute `mail_recovery.recover_messages` with the user name and folder name in which the recovered messages are placed.
   - Execute `mail_recovery.cleanup` after recovering the messages to drop the LogMiner shadow table, `es_lmmr_tbl`, and its indexes.

     See Also: "lmmr_setup Package"

The following example demonstrates how to execute mail recovery.

1. Run SQL*Plus.
2. Log on as the `es_mail` user.
3. Run the following PL/SQL block to recover the messages for a specified user into a newly created folder:

```
SQL> set serveroutput on;
SQL > declare
fname VARCHAR2(300);
begin
lmmr_setup.setup_logmnr(data_dictionary_file_with_full_path,
path_name_for_file_containing_redo_list, filename_for_file_containing_redo_
list);
mail_recovery.recover_messages(email_domain_of_user,emailid_of_user_without_
domain, fname);
mail_recovery.cleanup;
dbms_output.put_line(messages_recovered_into_folder||fname);
end;
/
```

lmmr_setup Package

The `lmmr_setup` package consists of the `setup_logmnr` procedure. This procedure initializes the LogMiner for mining redo logs for the specified duration. It also builds the LogMiner shadow table, `es_lmmr_tbl`, that is subsequently used for recovering messages.

The syntax is as follows:

```
PROCEDURE setup_logmnr(
p_dictionary_filename IN VARCHAR2,
p_redolist_location   IN VARCHAR2,
p_redolist_filename   IN VARCHAR2
p_starttime           IN DATE DEFAULT '01-jan-1988',
p_endtime             IN DATE DEFAULT '01-jan-2099');
```

Table 5–2 setup_logmnr Parameters

Parameter	Description
`P_DICTIONARY_FILENAME`	Full path name of the LogMiner data dictionary file
`P_REDOLIST_LOCATION`	Directory location of the redo list file

Table 5–2 setup_logmnr Parameters

Parameter	Description
P_REDOLIST_FILENAME	File name of the redo list file
P_STARTTIME	Only consider redo records with a time stamp greater than or equal to the specified start time
P_ENDTIME	Only consider redo records with time stamp less than or equal to the specified end time

mail_recovery Package

The mail_recovery package consists of the recover_messages and cleanup procedures.

The recover_messages procedure recovers messages and restores them in a specified folder.

The syntax is as follows:

```
PROCEDURE recover_messages(
p_domainname IN  VARCHAR2,
p_username   IN  VARCHAR2,
p_foldername IN OUT VARCHAR2,
p_autocommit IN BOOLEAN DEFAULT TRUE);
```

Table 5–3 recover_messages Parameters

Parameter	Description
P_DOMAINNAME	Domain name of the user
P_USERNAME	Oracle Email name of the user for whom the recovery is being performed
P_FOLDERNAME	Name of the folder in which recovered messages are restored. If the value is NULL, creates a new folder named RECMSG_*current_date_time*.
P_AUTOCOMMIT	If true, performs frequent commits If false, no commits are performed

The cleanup procedure drops the es_lmmr_tbl shadow table and its indexes. This procedure does not have any parameters.

cleanup should be called after a successful call to the recover_messages procedure.

The syntax is as follows:

```
PROCEDURE cleanup;
```

Recovering Messages with Flashback Query

This section contains the following topics:

- Using Flashback Query to Recover Messages
- MAIL_RECOVERY_FQ Package

Using Flashback Query to Recover Messages

Oracle Email end users and administrators can recover messages deleted as of a certain time using the database flashback query feature. The flashback query based e-mail recovery applies to e-mail messages transferred to another folder. Flashback query creates a snapshot of the database at a certain point in time, from which an Oracle Email user can recover all messages that are in a particular folder at a specific point in time.

Flashback query uses the retention control functionality provided by the Automatic Undo Management feature of the Oracle database. The database maintains information that is used to roll back, or undo, changes to the database. Undo information consists of records of the actions of transactions, primarily before they are committed. Retention control allows you to specify the minimum period of time for which database undo information is saved before the space is overwritten by newer transactions.

When an Oracle Email message is deleted, a record is created in the database undo logs. When flashback query retrieves the deleted message, the message is restored from the undo logs. The longer the undo information is retained, the older the deleted messages Oracle Email users can retrieve using flashback query. A message can be recovered only if retention control is enabled and the message was deleted within the specified retention period.

The length of time for which database undo information is retained depends upon the amount of available disk space, the amount of e-mail traffic going through the Oracle Email system, and the user activity on the system. The longer the undo information is retained or the heavier the activity on the e-mail system, the more disk space is required. An Oracle Email system that receives a large number of messages per day requires more disk space to retain undo information than a system that receives just a few messages per day.

> **See Also:** "Chapter 13: Managing Undo Space" in the *Oracle9i Database Administrator's Guide* Release 2 (9.2) for information about managing undo spaces, choosing the retention period for flashback queries, and calculating undo retention space requirements

Deleted messages can be retrieved with flashback query using the Microsoft Outlook client connecting to Oracle Email through the Outlook Connector. Recovered messages are recovered to the folder of the user's choice.

> **See Also:** Oracle Connector for Outlook Online Help for more information on recovering messages using Microsoft Outlook through the Outlook Connector

Flashback recovery by Oracle Email through the Outlook Connector can be enabled or disabled using the WebMail client administration pages.

To recover messages for a user using flashback query, administrators can also use the PL/SQL package `MAIL_RECOVERY_FQ`.

> **See Also:** "MAIL_RECOVERY_FQ Package"

To use flashback query, you must first create an undo tablespace. This can be done when the database containing the mail store is first created or you can add an undo tablespace to an existing database. After creating the tablespace, set the database parameter `UNDO_RETENTION` to the amount of time in seconds for which the undo information is to be retained.

To set the retention time, enter the following:

```
ALTER SYSTEM SET UNDO_RETENTION=time_in_seconds
```

For example, to retain undo information for at least 3 hours, set the `UNDO_RETENTION` parameter as follows:

```
ALTER SYSTEM SET UNDO_RETENTION = 1800
```

Recovered messages are included in quota calculations. If a user's quota is exceeded during flashback recovery, no additional messages are recovered.

A message can be recovered even if it exists in a different folder. For example, if a message was moved from a user's Inbox to FolderA and the user decided to recover the moved message into RecoverInbox, a pointer to the message would be created in RecoverInbox.

If a user tries to recover a message that already exists in the destination folder, the message retrieval fails. For example, if the message already exists in RecoverInbox and the user tries to recover that message into RecoverInbox, the recovery is not be performed.

Flashback recovery using the PL/SQL package `mail_recovery_fq` is not affected by the administrator's policy and is always available.

The `undo_retention` parameter can be set in the initialization parameter file.

The following example demonstrates how to execute flashback recovery:

1. Run SQL*Plus.
2. Log on as `es_mail` user.
3. Run the following PL/SQL block to recover all deleted messages in last 30 minutes for a specified user into a newly created folder without performing any quota check:

```
SQL> set serveroutput on;
SQL > declare
fname VARCHAR2(300);
begin
mail_recovery_fq.get_recover_messages('emailid_of_user_without_domain',
'email_domain_of_user', 30, 0, NULL, 0,
fname);
dbms_output.put_line('Messages recovered into folder'||fname);
end;
/
```

MAIL_RECOVERY_FQ Package

The `MAIL_RECOVERY_FQ` package retrieves deleted messages from one or all of a user's folders as of a specified point in time. The syntax is as follow:

```
mail_recovery_fq.get_recover_messages(
p_usernameVARCHAR2,
p_domainnameVARCHAR2,
p_int_in_minsNUMBER,
p_quotaNUMBER,
p_fromfolderVARCHAR2,
p_checksubfldrsNUMBER,
p_tofolderVARCHAR2)
```

Table 5–4 get_recover_messages Parameters

Parameter	Description
P_USERNAME	User ID of the account from which to recover e-mail
P_DOMAINNAME	Domain name of the user
P_INT_IN_MIN	Time, in minutes, to go back in the past to search for deleted e-mail.
P_QUOTA	If the value is 0, no quota check is performed. If the value is1, a quota check is performed.
P_FROMFOLDER	If a value is specified, then only P_FROMFOLDER is checked for deleted e-mail messages. If the value is null, then all of the user's folders are checked for deleted e-mail messages.
P_CHECKSUBFLDRS	If the value is 0, only P_FROMFOLDER is checked for deleted e-mail messages. If the value is 1, all subfolders for P_FROMFOLDER are checked for deleted e-mail messages.
P_TOFOLDER	Destination of the retrieved messages. If the specified folder does not exist, the folder is created by the specified name for the deleted e-mail messages. If a folder is not specified, the system creates a folder named RECMSG_*dd-Mon-yyyy hh24:mi* in which to store the deleted e-mail messages. The time stamp shows when the recovery occurred, not when the deletion occurred.

To use the mail_recovery_fq package to recover messages:

1. Connect to the database as es_mail.
2. Execute the mail_recovery_fq.get_recover_messages procedure.

6

Charting and Monitoring

This chapter describes how to collect and view Oracle Email system statistics using the `oesmon` monitoring utility and the `oeschart` charting utility.

This chapter contains the following topics:

- Using OESMON
- Using OESCHART

Using OESMON

The `oesmon` utility obtains raw metric data directly from Oracle Email server processes and provides a summary of the mail system's statistics. Each statistic is represented by either an ASCII string or a number, rendered in keyword-value pairs.

`oesmon` provides the syntax options listed in Table 6–1.

Table 6–1 oesmon Syntax Options

Command	Description
`oesmon`	Returns the usage message
`oesmon targets`	Lists all possible targets for Oracle Email servers
`oesmon names` *`target`*	Lists all of the metric names for a particular target
`oesmon get` *`target metric_name`*	Returns the statistics

A target is a concatenation of:

`host:um_system:service_type`

where:

- *`host`* is the name of the host where the service is running
- *`service`* type of the following:
  - `gc`: housekeeping process
  - `imap`: IMAP4 server
  - `list`: list server
  - `pop`: POP3 server
  - `smtp_in`: SMTP process for inbound mail routing
  - `smtp_out`: SMTP process for outbound mail routing
  - `nntp_in`: NNTP inbound news server
  - nntp_out: NNTP outbound news server
  - `vs`: virus scrubber

All statistics and managed objects have names. Names are case sensitive and contain only alphanumeric characters and the underscore character. Statistic names

are hierarchical and separated by a period (.). A request for a managed object returns all of the managed objects and statistics beneath it.

> **See Also:** Appendix E, "Server Statistics" for a complete list of the available statistics

Consider the following example:

```
oesmon  get  mycomputer:um_system:pop .um.admin
```

In this case, two values are returned for the two metric objects in the hierarchy:

- process identifier
- date and time the service was started

```
.um.admin.os_pid = 8239
.um.admin.uptime = Wed Jan 29 14:17:36 2003
.um.admin.log.discard
.um.admin.log.total
```

where:

- `.um.admin.log.discard` is the number of log messages discarded when high log levels cause log messages to be generated faster than they can be written to disk.
- `.um.admin.log.total` is the total number of log messages logged by the processes, including the ones that are discarded and written to disk.

```
oesmon  get  mycomputer:um_system:pop  .um.admin.os_pid
```

Only the process identifier is returned.

```
.um.admin.os_pid = 8239
```

Using OESCHART

Oracle Email servers track a range of metrics that are periodically stored in a set of mail statistics tables. The `oeschart` utility generates charts and images that can be used to publish reports and Web pages; providing a company with a graphic picture of the status of the mail system implementation.

This section contains the following topics:

- Setting the Statistics Collection Interval

- Cleaning Up Mail Statistics
- Mail Statistics Schema
- Creating Graphs

Setting the Statistics Collection Interval

You can set the frequency by which metrics are sampled and recorded using the `oidadmin` administration tool provided by Oracle Internet Directory.

> **Note:** The collection interval parameters are not configurable in Oracle Enterprise Manager.

The `orclMailAdminCollectionInterval` parameter specifies the number of seconds that elapse between statistics collecting. A setting of zero (0) seconds stops the service from logging statistics altogether.

You should set the `orclMailAdminCollectionInterval` parameter at the process level, rather than the instance level. If you set the value at the target instance level, statistics collection occurs at different intervals for each instance of the same type of mail service.

Collecting statistics at the same interval for all server types is not recommended. The collection process utilizes different amounts of resources and collects statistics of varying degrees, depending on which mail service is engaged in collecting the statistics. For example, the housekeeping collection process is more resource intensive and collects fewer dynamic statistics than the SMTP server collection process.

The recommended collection intervals for the different types of servers are:

- Housekeeper: 3600 (one hour)
- IMAP: 600 (ten minutes)
- POP: 600 (ten minutes)
- SMTP_IN: 600 (ten minutes)
- SMTP_OUT: 600 (ten minutes)
- List: 600 (ten minutes)
- NNTP_IN: 600 (ten minutes)

- NNTP_OUT: 600 (ten minutes)

Cleaning Up Mail Statistics

Mail statistics can be cleaned up through the housekeeping process, by enabling the `Statistics Cleanup` parameter and setting the `Tertiary Storage Age Threshold` parameter to the number of days you want to retain sample data.

Mail Statistics Schema

Each mail process has a default store database. Processes supporting multiple stores, such as the IMAP server, have a default store that is set in the `orclMailAdminStoreDN` parameter during the installation of each middle tier.

Statistical information is stored in tables in the `esperftbl` tablespace in the default mail store of the process. Because these tables can expand indefinitely, you should monitor the `esperftbl` tablespace and delete or export data as needed.

The schema consists of the following:

- es_perf_process Table
- es_perf_metric Table
- es_perf_timestamp Table
- es_perf_sample Table
- es_perf_data View

es_perf_process Table

The `es_perf_process` table is a list of the process instance records. The column `process_dn` is the complete distinguished name (DN) of the process, found in Oracle Internet Directory. The processes that produce metric data insert records into this table as needed.

Table 6–2 es_perf_process Table

Column Name	Valid Values	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 Table

Table 6–3 es_perf_metric Table

Column Name	Value	Description
metric_id	Number (not null)	Internally assigned unique number for each metric
metric_name	Varchar2 (100)	Name of the metric
metric_type	Number (not null)	If 1, the metric is numeric If 2, the metric is a string, and defines which column is important in the es_perf_sample table

es_perf_timestamp Table

The es_perf_timestamp table records each time metric data is stored into the tables. This table, along with the es_perf_data table grows without bound over time. You must clean out historic data that is no longer needed for charting or analysis at their installation. This can be done by configuring the housekeeping process.

Table 6–4 es_perf_timestamp Table

Column Name	Value	Description
timestamp_id	Number	Internally assigned unique number for each time period that a process enters a statistics time stamp
date	Date	Time the data was inserted

es_perf_sample Table

The es_perf_sample table records each metric at each timestamp. This table increases over time, so you must delete data that is no longer required in order to keep the table size manageable.

There are two possible columns that store the sample data, depending upon whether the data is numerical (nvalue) or a string (svalue).

Table 6–5 es_perf_sample Table

Column Name	Value	Description
process_id	Number (not null)	Corresponds to the process_id row in es_perf_process
metric_id	Varchar2 (100)	Corresponds to the metric_id row in es_perf_metric
timestamp_id	Number (not null)	Corresponds to the timestamp_id row in es_perf_timestamp
nvalue	Number	Numeric value of the metric (if the metric is numeric)
svalue	Varchar2 (1000)	String value of the metric (if the metric is a string)

es_perf_data View

es_perf_data is a view of the tables described in this section: es_perf_process, es_perf_metric, es_perf_timestamp, and es_perf_sample.

Table 6–6 es_perf_data View

Column Name	Value	Description
process_dn	Varchar2 (500)	The DN of the process in Oracle Internet Directory
metric_name	Varchar2 2 (100)	Name of the metric
metric_type	Number (not null)	If 1, the metric is numeric If 2, the metric is a string, and defines which column is important in the es_perf_sample table
timestamp	Date	Time the value was sampled
nvalue	Number (not null)	Numeric value of the metric (if the metric is numeric)
Svalue	Varchar2 (1000)	String value of the metric (if the metric is a string)

The SQL script that defines this view is:

```
CREATE VIEW es_perf_data AS
SELECT process_dn, metric_name, metric_type, timestamp, nvalue, svalue
FROM es_perf_process p, es_perf_metric m, es_perf_timestamp t, es_perf_sample s
```

where:

```
s.process_id = p.process_id
s.metric_id = m.metric_id ]
s.timestamp_id = t.timestamp_id;
```

Creating Graphs

`oeschart` creates graphs take a single parameter which points to a property file. The property file is a text file with keyword value pairs defining the information the utility needs to generate the graph. A valid property file would have the following mandatory and optional parameters.

The following tables describe mandatory entries and optional values:

Table 6–7 oeschart Mandatory Properties

Parameter	Description
`server`	Host name of the statistics database
`port`	Database listener port
`sid`	SID or service name for the server
`username`	Account user
`password`	Account password
`process_dn`	Query used to gather statistics, such as *`process_dn=%value_in_ini_file%`*, which retrieves all processes that follow this DN pattern. This lets you graph a specific process, a set of processes, or the entire system by specifying the level of detail
`metric_name`	Metric to query
`graph_type`	Type of graph, such as `xy` and `bar`
`image_file_name`	Name of the file being generated
`image_title`	Title to display on the graph
`number_of_hours`	Number of hours, starting from the present, to graph

Table 6–8 oeschart Optional Properties

Parameter	Default	Description
`encode_type`	`gif`	Possible values are `gif` and `png`

Table 6–8 oeschart Optional Properties

Parameter	Default	Description
image_dir	./	Directory where graphs are stored
aggregate_time_period	600	Time span in which multiple logging processes are grouped together and the metrics combined to show an aggregate value. As an example, consider two running IMAP servers, IMAP1and IMAP2. IMAP1 logged its statistics at 3:00pm and IMAP2 logged its statistics at 3:02 pm. The servers log statistics at intervals specified in seconds using oidadmin relative to when they started; in this case, IMAP2 must have been started 2 minutes after IMAP1. To show the total number of sockets on the system, combine the values from IMAP1 and IMAP2. aggregate_time_period defines what is an acceptable window for different process statistics to be combined. This should be the same as the submit period specified in oidadmin for this process type.
max_lifetime	300	Number of seconds until the program terminates
show_statistics	FALSE	Number of data points, minimum, maximum, average, and median, at the bottom of the graph
debug	FALSE	Provides a detailed output of the utility

The following is an example of an IMAP property file:

```
server=testdb.us.oracle.com
sid=test
port=1521
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
```

`oeschart` obtains information from the `es_perf` schema and generates one of four possible types of charts. Three of these are variations of scatter graphs. The fourth is a bar chart.

By executing `oeschart` in regular intervals, you are provided with a current view that can be published on a company Web site or within Oracle Enterprise Manager.

For example, you can schedule the creation of graphs that show information such as the number of connected sockets, the log in response time, and the number of queued outbound messages, and publish the results in a custom HTML page or in Oracle Enterprise Manager.

Displaying Graphs in Oracle Enterprise Manager

To display graphs in Oracle Enterprise Manager:

1. Modify the target metadata definition

   The `$ORACLE_HOME/emdw/sysman/admin/metadata` directory contains a list of target metadata definition files.

   For a particular target definition, add the following elements to 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, change the value of the `totalNumberOfStats` parameter in the `targets.xml` file accordingly, and the additional picture properties must be defined using the following naming standard:

`Title[`*N*`], ToolTips[`*N*`], PicPath[`*N*`], PicPhysicalPath[`*N*`]`

where *N* is a non-negative natural number.

2. Modify the `targets.xml` file to specify the property instance values.
3. Add the following properties to the specific target section in `$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 in 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 picture file specified under `PicPhysicalPath[N]` does not exist

7

Command Line Interface

This chapter contains general instructions on how to use the command-line interface, as well as an entry for each command available in the command-line interface. Each command entry includes a brief description of its purpose as well as the proper syntax, keywords, and command parameters.

This chapter contains the following topics:

- OESCTL
- OESUCR
- OESDL
- OESRL
- OESUTIL
- OESNG
- OESPR

OESCTL

The `oesctl` command enables an Oracle Email administrator to perform configuration and control operations on Oracle Email services.

This command is used from a command shell, such as `/bin/csh` on a Unix system. It 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 `command` can be any of the following:

Table 7–1 oesctl commands

Command	Description
startup	Starts individual processes associated with the target or instance.
shutdown	Shuts down individual 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 this release of Oracle Email.

service must be chosen from this list: `gc`, `list`, `smtp_in`, `smtp_out`, `imap`, `pop`

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

The syntax of instance is `target:`*`instance_id`*

`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 mailserver.

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 mailserver, and there are no process instances for the POP service. A service must have at least one process instance before it can be started. Since the "show targets" example shows no POP instances, 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

In the following example, the list of process instances for the target `mailserver:um_system:gc` is checked prior to instance creation, and found to be empty. The create command is used to create a new process instance for the target, after which the process instance list is checked again and found to contain the new 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
```

Deleting a Process Instance

In the following example, the list of process instances for the target mail server: `um_sytem`: `gc` is checked prior to instance deletion. The delete command is used to delete the process instance found, after which the process instance list is checked again and found to contain no processes.

```
% 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
```

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

In the following example, the show processes command reveals two instances. The status command shows they are stopped. Then the command starts them, and the status command shows them "alive." Finally, a shutdown command terminates them.

```
% 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
mailserver:um_system:gc:101779027179112257 ok
mailserver:um_system:gc:101779029537864556 ok

% oesctl show status mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 ----Heartbeat----
mailserver:um_system:gc:101779029537864556 ----Heartbeat----

% oesctl shutdown mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 ----Shutdown----
mailserver:um_system:gc:101779029537864556 ----Shutdown----
% 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

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.

```
% oesctl startup mailserver:um_system:gc:101779027179112257
ok

% oesctl show status mailserver:um_system:gc
mailserver:um_system:gc:101779027179112257 ----Heartbeat----
mailserver:um_system:gc:101779029537864556 <stopped>

% oesctl shutdown mailserver:um_system:gc:101779027179112257
mailserver:um_system:gc:101779027179112257 ----Shutdown----
```

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 all started process instance of that service.

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, the IMAP service log level can be changed in Oracle Internet Directory and refreshed without disconnecting any users currently connected to the IMAP service. Conversely, executing a shutdown followed by a startup changes the logging behavior, and temporarily disconnect users.

OESUCR

The `oesucr` command takes input from a file of user names to achieve the following tasks:

- Create and delete Oracle Email users

- Change e-mail addresses
- Specify a real domain for users
- Support different character encoding types
- Create e-mail users directly from the command line

OESUCR takes a file name as an input parameter.

For user creation, the file should contain a list of records, each followed by an empty line. Each record contains information about an e-mail user to be created. Each record in the file is a name-value pair for an attribute of the e-mail user entry in the directory. Each record must have at least three mandatory attributes:

- mail
- quota
- baseuserdn

For user deletion, the file should contain one line listing all the users to be deleted, each separated from the next by a comma.

OESUCR 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.

To change an e-mail address:

```
oesucr old_email_address1=new_email_address1  -change
```

For example:

```
%oesucr user1@us.oracle.com=newuser1@us.oracle.com
```

After this command is run, `user1@acme.com` becomes `newuser1@acme.com`.

```
%oesucr filename -encoding=UTF-8
```

where *`filename`* is the name of a file to be read as UTF-8.

The file is read as UTF-8, and must be saved in the corresponding encoding.

Examples

Creating Users

Assume 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: the email address, the quota, and the base user DN.

> **Note:** The corresponding public users must exist before running the `OESUCR`.

Creating Users with Optional Attributes

Assume a file `user_file` contains the following records:

```
mail=testuser1@us.oracle.com
orclmailquota=400000000
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*:* `testuser1` and `testuser2`. The role of the first user is set to domain administrator.

Deleting Users

Assume 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:** The corresponding entries in the directory for these public users are not deleted by `OESUCR.`

Creating a User Through Command Line

The following example shows how to create a user through the command line, without creating a new file. Only one user can be created at a time in this fashion:

```
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 example shows 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`, although the user's entry in Oracle Internet Directory is under `acme.com`. The name `company1.com` may or may not exist.

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, each followed by an empty line. Each record must have the name of the distribution list and a list of its users.

For adding users to a distribution list, the user type must be indicated, as a regular user, a distribution list, an alias, or a foreign user, as follows:

- U for regular user
- F for foreign user
- L for a distribution list
- A for an alias

You can also add users to a list that does not yet exist, by creating the list in the same command that specifies its users. See the later section Adding Users to a New List.

To create a new list, the owner must be specified.

Usage

```
% oesdl file
```

file is the path to the file containing the list records.

Examples

Adding Users to a List

Assume the example file list_file contains 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` (`usertype=U`) to `list1@oracle.com` (`usertype=L`) , which must already exist, since `newlist=n` (no). It also adds `list1@oracle.com` to another list called `list2@oracle.com`.

Adding Users to a New List

Assume 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`, sets 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

Assume 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 as `user1@oracle.com`, and adds `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 either 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.

The file parameter is defined in the next section, File Formats. Examples of using the `-c` parameter appear in Creating User Rules Using Property File Input; for the -x parameter, in Creating User Rules Using XML File Input; and for `-p`, in Retrieving Rules.

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. For rules owned by a user, it is the user's e-mail address; for rules that are domain specific, it is 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. Each property 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#: Up to six distinct events can be defined in this file; each 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 an 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. Rules often have conditions that need to be defined, using the following list of property names corresponding to each rule attribute:

> **Note:** Replace the symbol # 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 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>
```

OESUTIL

The `oesutil` command-line tool enables administrators to change passwords and delete domains.

Changing Passwords

Use the following command to change passwords:

`oesutil -umadmin_passwd` *`old new`* `-v -ocsv1`

- -v is used for debugging (optional)
- -ocsv1 is used if the mail store is ocs v1 (optional)

Deleting Domains

There are four delete options that an administrator can use:

- `user`: Deletes all users in a domain
- `alias`: Deletes all aliases in a domain
- `list`: Deletes all lists in a domain
- `all`: Deletes everything in a domain
- `news`: Deletes all newsgroups in a domain

Syntax

The following syntax is used for `oesutil`:

- `-v` used for debugging
- `- domain=domain1` specifies the domain
- `- installation=install1` specifies the installation, and is used only for in `type=all` cases

```
oesutil -delete_domain type=user domain=edu
oesutil -delete_domain type=list domain=com -v
oesutil -delete_domain type=all domain=edu installation=um_system

oesutil -delete_domain type=news domain=idc.oracle.com installation=um_system
```

OESNG

The `OESNG` command-line tool enables administrators to create and delete NNTP newsgroups in the Oracle Collaboration Suite system. This utility accepts a file as an input and creates or deletes newsgroups according to the information specified in the file.

File Format

The file passed as an input to the `OESNG` command-line tool must contain a list of records, each followed by an empty line. Each record consists of a set of *`name=value`* pairs containing information about the newsgroup to be created or deleted. Names are not case-sensitive. Unless indicated otherwise, all attributes can be specified only once for a group. Lines that begin with the "#" character are treated comments and are not processed by the tool.

For groups that are being deleted, it is sufficient to specify the name of the group and the action as delete.

Parameters

Table 7–2 OESNG Parameters

Parameter	Description	Acceptable Values	Default
Name	Name of the newsgroup to be created or deleted	Any valid newsgroup name	Mandatory (no default)
News Store	Name of the news store on which this newsgroup is to be created	Any valid news store in the Oracle Collaboration Suite system	Mandatory for newsgroup creation
Action	Action: creating or deleting the newsgroup	Create or Delete	Create
Owner	Owner of the newsgroup	Any valid e-mail address in the Oracle Collaboration Suite system	None
Description	Description for the newsgroup	Single line of text	None
Moderated Group	Boolean telling whether the group is moderated	True or False	False
Moderator	Names the moderator for the newsgroup. Can be specified more than once for a newsgroup.	Any valid e-mail address	None
Posting Allowed	Boolean telling whether posting is allowed to the group	True or False	False
Retention Days	Number of days to retain an article in a newsgroup before being expired	Any positive integer	None
Domain	Domain the group belongs to, if the newsgroup being created or deleted is a private newsgroup	Any valid domain in the Oracle Collaboration Suite system	None
Installation	Name of the installation where the newsgroup is to be created	Any valid installation name in the Oracle Collaboration Suite system	um_system

To delete groups, specify the name of the group and the action as delete.

Usage

```
oesng file
```

where *file* is the path to the file containing the newsgroups to be processed.

Examples

The following examples show how to use the OESNG command-line tool.

Creating a Simple Public Newsgroup

The following example demonstrates how to create a simple public newsgroup.

```
oesng newsfile
```

where newsfile contains the following

```
name=newsgroup1
newsstore=db1.acme.com
```

Creating a Public Moderated Newsgroup

The following example demonstrates how to create a public moderated newsgroup that permits posting and retains articles for 15 days.

```
oesng newsfile
```

where newsfile contains the following:

```
name=newsgroup2
newsstore=db1.acme.com
action=create
description=A new newsgroup
moderatedgroup=true
moderator=user1@acme.com
moderator=user2@acme.com
postingallowed=true
retentiondays=15
```

Deleting a Public Newsgroup

The following example demonstrates how to delete a public newsgroup.

```
oesng newsfile
```

where newsfile contains the following

```
name=newsgroup5
action=delete
```

Creating a Public Newsgroup

The following example demonstrates how to

- create a public newsgroup that does not allow posting,
- delete an existing private newsgroup, and
- create a private moderated newsgroup that allows posting.

```
oesng newsfile
```

where `newsfile` contains the following:

```
name=newsgroup3
newsstore=db1.acme.com
postingallowed=false

name=private.newsgroup1
domain=acme.com
action=delete

name=private.newsgroup2
domain=acme.com
newsstore=db1.acme.com
postingallowed=true
moderatedgroup=true
moderator=mod1@acme.com
moderator=mod2@acme.com
```

OESPR

The `OESPR` command-line tool enables administrators to create and delete news peers and to associate newsgroups with news peers in an OCS system. It accepts a file as an input and creates or deletes peers according to the information specified in the file.

File Format

The file that is passed as an input to the oespr command line tool must contain a list of records, each followed by an empty. Each record consists of a set of *name=value* pairs which contain information about the newsgroup to be created or deleted. Names are not case-sensitive. Unless indicated otherwise, all attributes can be specified only once for a peer.

Lines beginning with the "#" character are treated as comments and are not processed by the tool.

Parameters

The following describes the newsgroup parameters that can be specified in the file.

Table 7–3 OESPR Parameters

Parameter	Description	Acceptable Values	Default
Host Name	Fully qualified host name of the news peer being created or deleted	Any valid peer host name	Mandatory (no default)
Port	Port on which the NNTP server is running on the peer	Any valid port	119
Action	Action of creating or deleting the peer	Create or Delete	Create
Accept Group	List of groups accepted from this peer; can be specified more than once for a peer.	Any valid group name. Wildcard patterns can also be specified.	None
Reject Group	List of groups to be rejected if offered by this peer; can be specified more than once for a peer.	Any valid group name. Wildcard patterns can also be specified.	None
Feed Group	List of groups for which this peer should be fed; can be specified more than once for a peer.	Any valid public newsgroup name.	None
Installation	Name of the installation where the peer is to be created.	Any valid installation name in the Oracle Collaboration Suite system.	um_system

To delete peers, specify the host name of the peer and the action as delete.

Usage

```
oespr file
```

where *file* is the path to the file containing the peers to be processed.

Examples

The following examples show how to use the OESPR command-line tool.

Creating a Simple Peer

The following example demonstrates how to create a simple peer.

```
oespr peerfile
```

where `peerfile` contains the following:

```
hostname=host1.acme.com
```

Creating a Peer that is Fed Articles from Specific Newgroups

The following example demonstrates how to create `host1.acme.com` as a peer that is fed articles from the `comp.lang.c` and `comp.lang.java` groups on port 2119

```
oespr peerfile
```

where `peerfile` contains the following:

```
hostname=host1.acme.com
port=2119
feedgroup=comp.lang.c
feedgroup=comp.lang.java
```

Deleting a Peer

The following example demonstrates how to delete a peer.

```
oespr peerfile
```

where `peerfile` contains the following:

```
hostname=host1.acme.com
action=delete
```

8

Parameters and Log Files

This chapter provides the Oracle Email system parameters and log file locations.

This chapter contains the following topics:

- Server Parameters
- WebMail Properties
- Log Files

Server Parameters

This section provides server parameter definitions for the following categories:

- IMAP
- POP
- SMTP
- Housekeeping
- List Server
- NNTP

> **See Also:** Chapter 4, "Security" for information on virus scrubber parameters

IMAP

Table 8–1 describes the IMAP server parameters in alphabetical order.

Table 8–1 IMAP Server Parameters

Parameter	Description	Acceptable Values	Default Value
Cache Size	Caching level. When "small," no mail information is cached in the middle tier IMAP server. When "medium", certain parts of mail are cached. Increasing the cache size increases the memory requirements on the middle tier.	small, medium	small
Custom Name	Applies only if the presentation name is set to custom		
Debug User	Name of a user about whom more debug information will be sought in the log files		
Default Domain	Default domain used as user login if the user logging in does not provide one		
Get New Mail Interval	Number of seconds the IMAP server waits before checking for new mail. Large numbers of "check new mail" requests from clients affect performance.	0-65535	120
LDAP Connection Pool Increment	Number of Oracle Internet Directory connections added to the pool		1

Table 8–1 IMAP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
LDAP Connection Pool Time Lag	Time lag (in 1/100ths of a second) permitted before increasing the pool. For example, 100 means a 1 second delay after the point where a new pool connection must be added to the pool.		500 centi seconds
LDAP Connection Retry Interval	Maximum time in microseconds the server has to wait to get a connection after it has reached the maximum number of connections		100000 micro seconds
LDAP Maximum Connection Pool	Maximum number of Oracle Internet Directory connections in the pool		20
LDAP Minimum Connection Pool	Minimum number of Oracle Internet Directory connections in the pool		2
LDAP Number of Retry Before Erroring	Total number of times the server attempts to connect to Oracle Internet Directory		100
LDAP Reconnection Timeout	Number of seconds before the server tries to reconnect to Oracle Internet Directory		300 seconds
Maximum Number of Clients	Maximum number of clients allowed to connect to the server instance	0-1000	1000
Presentation Name	Port on which the listener listens for the IMAP service Selecting **Custom** enables you to specify the presentation name. When this parameter's default value has been changed, you must change the listener configuration to the same value.	string	IMAP
Process Debug Level	Debug messages level. For statistics, set 512.	4294967295 (32 bits, Multi-value)	0
Process Log Level	Log messages level	0-30	6
Protocol Server Increment Thread	Number of threads added to the client connection pool	1-999	1
Protocol Server Maximum Threads	Maximum number of threads available for client connection handling	1-1000	500
Protocol Server Minimum Threads	Minimum number of threads available for client connection handling	1-1000	1

Table 8–1 IMAP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Protocol Server Thread Timeout	Number of seconds before an idle thread is cleaned up	0-65535	1860 seconds
SSL Enabled	Applies only if the presentation name is set to custom		
Timeout Interval	Number of seconds for the auto-logout timeout interval. If no client operations occur in this time, it is disconnected.	0-65535	1800

POP

Table 8–2 describes the POP server parameters in alphabetical order.

Table 8–2 POP Server Parameters

Parameter	Description	Acceptable Values	Default Value
Default Domain	Default domain for users who do not provide a domain when logging in		None
LDAP Connection Pool Increment	Number of Oracle Internet Directory connections to be added to the pool	None	1
LDAP Connection Pool Time Lag	Number of 1/100ths of a second before increasing the pool: 100 would mean that if more than one connection arrives within 1 second, then the server must wait.		500 centi seconds
LDAP Connection Retry Interval	Maximum time in microseconds the server waits to get a connection after reaching the maximum number of connections		100000 micro seconds
LDAP Maximum Connection Pool	Maximum number of Oracle Internet Directory connections in the pool	None	20
LDAP Minimum Connection Pool	Minimum number of Oracle Internet Directory connections in the pool	None	2
LDAP Number of Retry Before Erroring	Total number of times the server attempts to connect to Oracle Internet Directory		100
LDAP Reconnection Timeout	Number of seconds before the server tries to reconnect to Oracle Internet Directory		300 seconds
Maximum Number of Clients	Maximum number of clients allowed to connect to the server instance	0-1000	1000
POP Delete Allowed	If YES, enables server to delete read messages	YES or NO	NO

Table 8–2 POP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
POP Retrieval	ALL means all mails are to be retrieved from the server; for UNREAD or any other value, only unread messages are retrieved.	UNREAD or All	UNREAD
Presentation Name	Port on which the listener listens for the POP service Selecting **Custom** enables you to specify the presentation name. When this parameter's default value has been changed, you must change the listener configuration to the same value.	string	POP
Process Debug Level	Debug messages level. For statistics, set to 512.	4294967295 (32bits, Multi-value)	0
Process Log Level	Log messages level	0-30	6

SMTP

Table 8–3 describes the SMTP server parameters in alphabetical order.

Table 8–3 SMTP Server Parameters

Parameter	Description	Acceptable Values	Default Value
Address Rewriting Rules Separator List	Defines the list of characters that should be treated as separators in address rewriting rules	String	None
Authentication	Determines if SMTP authentication is enabled. No authentication is required: If Mandatory, users must authenticate themselves before sending any messages. If Optional, users may authenticate themselves, but the SMTP server accepts the message even if authentication fails.	Mandatory Optional None	Optional
Checkpoint Interval	Number of recipients processed in a single relay delivery attempt	>=1	20
Connection Number	Number of SMTP connections the outbound SMTP server caches for future delivery to the same host	>1	20
DNS Check on HELO/EHLO Domain	If TRUE, checks whether the domain name in the helo/ehlo command exists in the DNS server. If not, the connection is rejected.	TRUE or FALSE	FALSE

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
DNS Check on Sender Domain	If TRUE, checks whether the domain in the sender's address exists in the DNS server	TRUE or FALSE	FALSE
DSN Interval	Frequency of the temporary delivery status notifications (DSNs)	Time in hours	24 hours
External Filter	Enables or disables external filter processing	True or False	False
External Filter Process	If External Filter is TRUE, then External Filter Process specifies the path for the executable of the external process in three parts:		
	name:path_name when_to_call flags		
	where:		
	name: The name of the external filter	Any name of an existing external filter	None
	path_name: The complete path of the process to be called	Any well-formed path name	None
	when_to_call: The time to call the external filter: ENV, DATA, RELAY, or NEVER, as explained below:		None
	ENV – After receiving the message envelope		None
	DATA – After receiving the complete message and before local delivery		None
	RELAY – Just before relaying a message		None
	NEVER – Essentially disables the callout		None
		0	
	flags - Should be set to 0		0
Fallback MX Host	Host where relay messages are sent when none of the MX hosts of the target domain are accessible	string	None
LDAP Connection Pool Increment	Number of Oracle Internet Directory connections added to the pool	None	1

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
LDAP Connection Pool Time Lag	Delay in 1/100ths of a second before increasing the pool. For example, 100 means 1 second, indicating that if connections arrive concurrently within 1 second, then the server must wait.		500 centi seconds
LDAP Connection Retry Interval	Maximum time in microseconds the server has to wait to get a connection after it has reached the maximum number of connections		100000 micro seconds
LDAP Maximum Connection Pool	Maximum number of Oracle Internet Directory connections in the pool	None	20
LDAP Minimum Connection Pool	Minimum number of Oracle Internet Directory connections in the pool	None	2
LDAP Number of Retry before Erroring	Total number of times the server attempts to connect to Oracle Internet Directory		100
LDAP Reconnection Timeout	Number of seconds before the server tries to reconnect to Oracle Internet Directory		300 seconds
Local Domains	Controls what domains are acceptable in e-mail addresses	Multi-value	None
Maximum Hop Count	Maximum number of hops a message can go through	>=1	25
Maximum Message Size	Maximum allowed incoming message size in bytes	>=0	0
Maximum Number of Clients	Maximum number of clients permitted to connect to the server at one time	0-1000	1000
Maximum Rule Nesting Level	Maximum number of times a nesting rule can be applied to a message. Smaller numbers increase overall performance, except for systems that use rules heavily.	>=1	20
Message Timeout	Number of minutes after an SMTP server restart, after which messages that remain in the "being processed" state will be processed again. (If an SMTP server is restarted after a shutdown, it looks for messages being processed. If they stay in the same state for this long, it reprocesses them.)	integer	30

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Native Anti-Spamming	If TRUE, turns on anti-spamming checks. If FALSE, all anti-spamming checks are turned off, including Reject and Trusted lists (domains, senders, recipients, and IPs). If Oracle Internet Directory does not have this value set to default, SMTP turns anti-spamming on.	TRUE or FALSE	TRUE
Postmaster Copy	If the postmaster address is set, a copy of the delivery status notification is sent to it.	String	None
Postmaster DSNs	Type of delivery status notifications (DSNs) postmaster wants to receive, one of the following four: All, None, Failures, Undeliverables	All, None, Failures, Undeliverables	Failures
Presentation Name	Port on which the listener listens for the SMTP service Selecting **Custom** enables you to specify the presentation name. When this parameter's default value has been changed, you must change the listener configuration to the same value.	string	ESSMI
Process Debug Level	Debug messages level. For statistics, set 512.	0-4294967295 (32bits)	0
Process Flags	Sets the local mailer flag, check point value, and enables RAC:		
	-l (local mailer flag): Messages to unknown local users are relayed to the next node, instead of being rejected.	l	No local mailer flag
	-cp=*value* (check point value): Messages are delivered to at most *value* local recipients in one transaction. 0 means delivering to all recipients in one transaction.	-cp=*value*	0
	-rac (enable optimization for RAC): Message insertion and local delivery uses the same OCI service handle.	-rac	Not enabled
Process Log Level	Log messages level	0-30	0
Protocol Server Increment Thread	Number of threads to be added to client connection pool	1-999	1
Protocol Server Maximum Threads	Maximum number of threads available for client connection handling	In 0-1000 Out 0-100	In 500 Out 50

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Protocol Server Minimum Threads	Minimum number of threads available for client connection handling	`1-1000`	`1`
Protocol Server Thread Timeout	Number of seconds before an idle thread is cleaned up	`0-65535`	`180`
Queue Poll Interval	Time in seconds the outbound server waits before checking the queue for new messages	`2 -30` seconds	`120` seconds
Recipient Rewriting Rules	Rewrite rules for recipients	Multi-value	None
Reject Domains	List of domains and sub-domains to reject, and close connection, but only if `Native Anti Spamming` is TRUE	Multi-value	None
Reject IPs	List of IP addresses to reject, and close connection, but only if `Native Anti Spamming` is TRUE	None	None
Reject Recipients	List of local recipients to reject, but only if `Native Anti Spamming` is TRUE	Multi-value	None
Reject Senders	List of senders to be rejected, but only if `Native Anti Spamming` is TRUE	Multi-value	None
Relay Allowed	If TRUE, enables relay from any domain; If FALSE, enables relay from only those domains set in Relay Domains Allowed; If AUTH, the server is to process only those messages whose senders have been authenticated by the SMTP inbound server.	`TRUE` or `FALSE` or `AUTH`	`FALSE`
Relay Domains Allowed	List of domains to allow to relay even if parameter `Allow Relay` is FALSE.	Multi-value	None

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Scanner Interfaces	Specifies C callouts for virus scanning. The form is: `name:shared_library_path, when_to_call,host_&_port,(function_set), scanner_flags, system_flags` Each element of this form is explained below:	Name of the external filter	None
	`name`	Full path of the C shared library loaded by the server at startup	None
	`shared_library_path`		
	`when_to_call`, i.e., The time to call the external filter: ENV, DATA, RELAY, or NEVER, as explained below:		
	ENV – After receiving the message envelope		None
	DATA – After receiving the complete message and before local delivery		None
	RELAY – Just before relaying a message		None
	NEVER – Essentially disables the callout		None
	`host_&_port`, as follows: - If the scanner needs a host and port, enter them as host:port_number, such as SMTP machine:3602. - If host and port are not needed, use INTERNAL.		
	`function_set` (The functions each filter callout should implement, which are called by the server to pass data to the scanner and to receive back the status and repaired messages)	`init, register_callback, scan_msg, send_msg, receive_msg, close`	None
	`repairmsg` If set to 1, callout can send the repaired message back to the server. If set to 0, server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure.	`1` or `0`	None
	`flags` Should be set to 0		None
Sender Rewriting Rules	Rewrite rules for senders; used only by the SMTP inbound server	Multi-value	None
SMTP Minimum Queue Age	Number of minutes a deferred message stays in the queue before being delivered	Integer	`30 minutes`

Table 8–3 SMTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
SMTP Process IMIP	Determines if calendar messages should be placed into the IMIP queue. For STMP inbound only.	Number	`300` seconds
SMTP Queue Timeout	Maximum number of days a message can be in the queue	`>=1`	`5 days`
SMTP Relay	Specifies the name of the relay host	Parameter type string	None
SMTP Timeout	Determines how long a connection stays open if there is no client activity. For SMTP inbound only.	`TRUE` or `FALSE`	`FALSE`
Spam Flood Interval	Number of minutes used to detect spam flooding	None	None
Spam Maximum Flood Count	SMTP server signals flooding if the number of messages and connections from a single host exceeds the value of this parameter within the `Spam Flood Interval`.	None	None
Submit Only	If TRUE, submits inbound messages without resolving recipient	`TRUE` or `FALSE`	`FALSE`
Trusted Domains	List of allowed domains or sub-domains from which mail is received, if `Native Anti Spamming` is TRUE, regardless of any other criteria	None	None
Trusted IPs	List of IP addresses from which connections are permitted, if `Native Anti Spamming` is TRUE, regardless of any other criteria	None	None
Trusted Sender	List of sender addresses against which the sender address is checked, if `Native Anti Spamming` is TRUE	Multi-string value	None
Trusted Sender Domains	List of domains against which the domain part of the sender's e-mail address is checked, if `Native Anti Spamming` is TRUE	Multi-string value	None
Use Errors To	If TRUE, uses the `Errors To` header in delivery status notifications	`TRUE` or `FALSE`	`FALSE`

Housekeeping

Table 8–4 describes the housekeeping parameters in alphabetical order.

Table 8–4 Housekeeping Parameters

Parameter	Description	Acceptable Values	Default Value
Collection	If ENABLED, runs the collection task, which collects or reclaims space taken up by messages no longer in use 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.	ENABLED or DISABLED	ENABLED
Execution Mode	Determines whether the server runs once and exits or stays active in the background forever (daemon mode). If a process is set to run as a daemon, it sleeps after one round of execution before starting the next round. Otherwise, it exits after the current task is finished.	DAEMON or RUN ONCE	DAEMON
Expiration	If ENABLED, runs the expiration task, which expires or deletes messages set to expire on or before the current time according to a timer. It moves such messages to the system trash folder. The expiration timer is a folder attribute that users can set. Oracle Corporation recommends running this task only once a day.	ENABLED or DISABLED	DISABLED
Text Synchronization	If ENABLED, performs the Oracle Text index synchronization task, which is essential to content-based searching. Doing it frequently greatly improves search performance unless the rate of incoming messages is low: then it would unnecessarily increase the server load. If content-based searching through Oracle Text is used heavily, Oracle Corporation recommends creating a dedicated housekeeping instance for this task with a sleep time of five to ten minutes.	ENABLED or DISABLED	DISABLED
Text Optimization	If ENABLED, performs the Oracle Text optimization task, which improves index synchronization performance. Otherwise, performance degrades over time. Oracle Corporation recommends running this task weekly, with a dedicated housekeeping instance with this task enabled and a sleep time of 24*7 (168) hours.	ENABLED or DISABLED	DISABLED
Pruning	Controls running the pruning task, which clears up message queues and the system trash folder, and marks unreferenced messages for collection. Oracle Corporation recommends scheduling this task to run continuously, to keep up with user message deletion activity.	ENABLED or DISABLED	ENABLED

Table 8–4 Housekeeping Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Tertiary Store	If ENABLED, runs the tertiary store task, which archives old messages by moving them to another tablespace, presumably cheaper and larger. Oracle Corporation recommends running this task monthly.	ENABLED or DISABLED	DISABLED
Process Log Level	Log message level	0-30	6
Process Debug Level	Debug messages level. For statistics, set 512.	4294967295 (32bits, Multi-value)	0
Tertiary Storage Age Threshold	Number of days, minimum, before messages are archived. If the tertiary storage task is turned on, housekeeping tries to archive messages older than this parameter. Oracle Corporation recommends setting at least 30.	Non negative number	30
Maximum Rule Nesting Level	Maximum number of times nesting can be applied to a message. Smaller numbers increase overall performance, except for systems that use rules heavily.	>=1	20
Process Sleep Duration	Number of hours between two consecutive starts of the task processing. If the task finishes before this amount of time, the housekeeping process sleeps for the rest of the duration. If the task takes more time than this parameter, the process does not sleep but instead runs continuously.	>=0	60 minutes
Statistics Collection	If ENABLED, housekeeper can delete process statistics data. The number of days set in Time Interval establishes the frequency of such deletion.	ENABLED or DISABLED	DISABLED
Record Messages Being Deleted Into Redo Logs	If ENABLED, keeps deleted message in redo logs, so that logminer-based recovery feature can be enabled.	ENABLED or DISABLED	DISABLED

List Server

Table 8–5 describes the list server parameters in alphabetical order.

Table 8–5 List Server Parameters

Parameter	Description	Acceptable Values	Default Value
Archive Mail Store	Mailstore the server should use to store all list archives. For proper access to archives, an NNTP inbound server must be configured and running against this mailstore	Any available mail store	None
Authenticated Sender	If ONLY, the list server is to process only those messages whose senders have been authenticated by the SMTP inbound server. If NONE, authentication is not required.	`only`	only
Command Mail Store	List server stores the reconfirmation on the mail store it is connected to, but the reconfirmation response can be processed by a list server connected to a different mail store in the system. In such cases, the reconfirmation is rejected. To avoid the problem of reconfirmations not being handled properly in an installation which has multiple mail stores, the list server uses the mail store configured in this parameter as the repository for command reconfirmations. All list server instances running in the system should open a connection to that mail store to store and verify all commands against that mail store only.	DN of a mail store	None
External Filter	Enables or disables external filter processing	`True` or `False`	`False`

Table 8–5 List Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
External Filter Process	If `External Filter` is TRUE, then `External Filter Process` specifies the path for the executable of the external process in three parts: *`name:path_name when_to_call flags`* where: `name`: The name of the external filter	Any name of an existing external filter	None
	`path_name`: The complete path of the process to be called	Any well-formed path name	None
	`when_to_call`: The time to call the external filter: ENV, DATA, RELAY, or NEVER, as explained below:		None
	`ENV` – After receiving the message envelope		None
	`DATA` – After receiving the complete message and before local delivery		None
	`RELAY` – Just before relaying a message		None
	`NEVER` – Essentially disables the callout		None
	flags - Should be set to 0	0	0
LDAP Connection Pool Increment	Number of Oracle Internet Directory connections added to the pool	None	`1`
LDAP Current Connection Pool	Number of Oracle Internet Directory connections currently in the pool	None	`10`
LDAP Maximum Connection Pool	Maximum number of Oracle Internet Directory connections in the pool		`20`
LDAP Minimum Connection Pool	Minimum number of Oracle Internet Directory connections in the pool	None	`2`
Local Domains	List of local domains served by the list server process	Multi-value	None

Table 8–5 List Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Maximum Message Size (bytes)	Maximum deliverable message size (in bytes) for the list. Messages larger than this are rejected.		
Number of Mails Processed Concurrently	Number of messages to be processed simultaneously by the list server	Any positive number greater than zero	`50`
Number of Recipients per Batch	Number of users each user thread delivers messages to	Any positive number greater than zero	`1000`
Number of Threads per Mail	Specifies the maximum number of threads that can be spawned at a time to distribute a mail to the members of a distribution list. Such threads are termed as "User Threads." Because each thread uses database and Oracle Internet Directory connections, this value should be increased with caution.	Any positive number greater than zero	`10`
PL/SQL Timeout	Number of minutes before a list server's PL/SQL call will be cancelled if the call has not returned. Used during PL/SQL mail-merge and external list processing.	Time in minutes	`10 seconds`
Post Master Copy	If the postmaster address is set, a copy of the delivery status notification is sent to it.	String	None
Postmaster DSNs	Type of delivery status notifications (DSNs) the postmaster wants to receive, one of the following three: All, None, Failure	`All`, `None`, `Failures`	`Failures`
Process Flags	Sets the local mailer flag and check point value. The values are: -l (local mailer flag): Messages to unknown local users are relayed to the next node instead of being rejected. -cp=value (check point value): Messages are delivered to at most value local recipients in one transaction. 0 delivers to all recipients in one transaction.	-l, `-cp=value`	No local mailer flag Check point value is 0
Process Debug Level	Debug messages level. For statistics, set `512`.		`0`
Process Log Level	Message log level	`1-30`	`6`
Recovery Interval	Number of minutes before messages marked as "being processed" are picked up for retrial by the server	Time in minutes	`90 minutes`

Table 8–5 List Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Relay Hosts	Contains the name(s) of the MTAs to which all relay messages sent out of the list server should be routed. Messages addressed to local users are not affected. If this parameter is not filled, then an MX record lookup is performed while relaying messages.		
Scanner Interfaces	Specifies C callouts for virus scanning. The form is:		
	`name:shared_library_path, when_to_call, host_&_port,function_set, repairmsg, flags`		
	Each element of this form is explained below:		
	`name`	Name of the external filter	None
	`shared_library_path`	Full path of the C shared library loaded by the server at startup	None
	`when_to_call`, i.e., The time to call the external filter: ENV, DATA, RELAY, or NEVER, as explained below:		
	ENV – After receiving the message envelope		None
	DATA – After receiving the complete message and before local delivery		None
	RELAY – Just before relaying a message		None
	NEVER – Essentially disables the callout		None
	`host_&_port`, as follows:		None
	- If the scanner needs a host and port, enter them as host:port_number, such as SMTPmachine:3602.		
	- If host and port are not needed, use INTERNAL.		
	`function_set`		
	(The functions each filter callout should implement, which are called by the server to pass data to the scanner and to receive back the status and repaired messages)	init, `register_callback`, `scan_msg`, `send_msg`, receive_msg, close	None
	`repairmsg`		
	If TRUE, callout can send the repaired message back to the server.		None
	If FALSE, server does not read any repaired message back from the callout and rejects the mail if the scanner returns failure.	TRUE or FALSE	
	`flags`		
	Should be set to 0		None

Table 8–5 List Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Temporary DSN interval	Number of hours between temporary delivery status notifications (DSNs)	Time in hours	`24 hours`

NNTP

Table 8–6 describes the NNTP server parameters in alphabetical order.

Table 8–6 NNTP Server Parameters

Parameter	Description	Acceptable Values	Default Value
Allow Peer Feed	If TRUE, this instance permits incoming feed from peers.	TRUE or FALSE	TRUE
Allow Streaming Feed	If TRUE, streaming is permitted, i.e., MODE STREAM in NNTP is enabled.	TRUE or FALSE	TRUE
Allowed Domains	List of allowed domains or sub-domains from which mail is received, if Native Anti Spamming is TRUE, regardless of any other criteria	None	None
Allowed IP Addresses	List of IP addresses from which connections are permitted, if Native Anti Spamming is TRUE, regardless of any other criteria	None	None
Allowed Senders	List of sender addresses against which the sender address is checked, if Native Anti Spamming is TRUE	Multi-string value	None
Article Cache Size (MB)	Number of megabytes for the article cache size	0-30 MB	0 MB
Authentication	Authentication scheme to be enforced, as defined in RFC 2980, to validate username (the user's full e-mail ID) and password (the Single Sign On (SSO) password)	None, Original, Simple	None
Connection Cache Size Maximum Feed Retrials	Number of connections for each peer that the outbound server contacts	0-1024	50
Default Newsgroup Subscriptions	List of newsgroups that a user subscribes to by default	Multi-value string	NULL
Disallowed Domains	If the Native Anti Spam parameter is TRUE, this parameter rejects connections from specified domains.	Multi-value string of disallowed domains, wildcards allowed	NULL, allows all domains
Disallowed IP Addresses	Identifies IP addresses to disallow connections from, if the Native Anti Spam parameter is TRUE	Multi-value string of disallowed IP addresses, wildcards allowed	NULL allows all IP addresses

Table 8–6 NNTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
Disallowed Senders	Identifies senders to reject, if anti-spam is required	Multi-value string of disallowed senders	NULL
Feed Recovery Interval (minute)	Number of minutes before a queued message marked as "in-process" is moved back to "pending", which aids in crash recovery for the NNTP outbound server	30-180	90
Feed Retry Interval (minute)	Number of minutes before retrying a message feed again	0-60	60
Inbound Peers	List of peers that send articles to this server, if the Allow Peer Feed parameter is set to TRUE. Peers are set-up after installation.	Multi-value peer names	NULL
LDAP Connection Pool Increment	Number of Oracle Internet Directory connections added to the pool	None	1
LDAP Maximum Connection Pool	Maximum number of Oracle Internet Directory connections in the pool	A number	20
LDAP Minimum Connection Pool	Minimum number of Oracle Internet Directory connections in the pool	A number	2
Local Domain	A domain created in the Oracle Collaboration Suite installation	A single value domain name	NULL
Maximum Feed Retrials	Number of times a message feed is re-tried. Combining this parameter with Feed Retry Interval tells you when a message feed is considered a permanent failure, i.e., after how many minutes.		3
Maximum News Message Size	Maximum size in bytes of an article accepted by posts or feeds. Zero indicates an unlimited size.	A number	0
Maximum Number of Clients	Maximum number of clients allowed to connect to the server instance	0-1000	1000
Native Anti-Spam	If TRUE, anti-spam checks are performed, otherwise not.	TRUE or FALSE	FALSE
News Administrator	E-mail address of the news administrator, inserted into the X-Complaints-To header of all messages posted to this server	A valid e-mail address	NULL
News Article Retention	Number of days before an article expires. This global setting applies to all articles across newsgroups.	0-180	7 days

Table 8–6 NNTP Server Parameters (Cont.)

Parameter	Description	Acceptable Values	Default Value
News History Retention	Number of days before the history entry for a news article is cleared	0-365	30 days
News Store	The Oracle Collaboration Suite mail store that NNTP inbound server connects to	Mail store in the installed Oracle Collaboration Suite where newsgroups have been created	NULL
Port	Port for the protocol service, which must match the port number in the NS listener configuration		5121
Posting Allowed	Specifies if messages can be posted to a group	Yes or No	No
Presentation Name	Port on which the listener listens for the NNTP service Selecting **Custom** enables you to specify the presentation name. When this parameter's default value has been changed, you must change the listener configuration to the same value.	string	ESNNI
Process Log Level	Log messages level	0-30	11
Protocol Server Increment Thread	Number of threads added to the client connection pool	1-999	5
Protocol Server Maximum Threads	Maximum number of threads available for client connection handling	0-1000	500
Protocol Server Minimum Threads	Minimum number of threads available for client connection handling	1-1000	1
Protocol Server Thread Timeout	Number of seconds before an idle thread is cleaned up	0-65535	300 - 399
Recommended News Distributions	List of distributions recommended on this server	Multi-value string	NULL
Socket Timeout (minute)	Number of minutes before a cached connection times out	0-30	30
Standard News Distributions	List of standard newgroup distributions	Multi-value string	NULL

Debug Level Parameters

This section provides debug level parameter definitions for the following categories:

- IMAP
- POP
- SMTP
- Housekeeping
- List Server
- NNTP
- Virus Scrubber

IMAP

Table 8–7 IMAP Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Client Logins	Enables internal debug log writing for client logins	`Enable` or `Disable`	`Disable`
Database Connections	Enables internal debug log writing for database connections	`Enable` or `Disable`	`Disable`
Folder Open	Enables internal debug log writing for the folder open function	`Enable` or `Disable`	`Disable`
Folder Synchronization	Enables internal debug log writing for folder synchronization	`Enable` or `Disable`	`Disable`
I/O Between IMAP Server and Clients	Enables internal debug log writing for I/O between the IMAP server and clients	`Enable` or `Disable`	`Disable`
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`
Memory Management	Enables internal debug log writing for memory management	`Enable` or `Disable`	`Disable`
Start/End of Client Requests	Enables internal debug log writing for client requests	`Enable` or `Disable`	`Disable`

POP

Table 8–8 POP Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Client Logins	Enables internal debug log writing for client logins	`Enable` or `Disable`	`Disable`
Database Connections	Enables internal debug log writing for database connections	`Enable` or `Disable`	`Disable`
Folder Open	Enables internal debug log writing for the folder open function	`Enable` or `Disable`	`Disable`
I/O Between POP Server and Clients	Enables internal debug log writing for I/O between the POP server and clients	`Enable` or `Disable`	`Disable`
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`
Memory Management	Enables internal debug log writing for memory management	`Enable` or `Disable`	`Disable`
Start/End of Client Requests	Enables internal debug log writing for client requests	`Enable` or `Disable`	`Disable`

SMTP

Table 8–9 SMTP Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Address Rewriting Rules	Enables internal debug log writing for address rewriting rule processing	`Enable` or `Disable`	`Disable`
Anti Spamming	Enables internal debug log writing for for anti-spamming	`Enable` or `Disable`	`Disable`
DSN Module	Enables internal debug log writing for the DSN module	`Enable` or `Disable`	`Disable`
Entire Inbound Module	Enables internal debug log writing for the entire SMTP inbound module	`Enable` or `Disable`	`Disable`
External Filter	Enables internal debug log writing for the external filter process	`Enable` or `Disable`	`Disable`
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`

Table 8–9 SMTP Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
LDAP Resolution	Enables internal debug log writing for LDAP resolution	`Enable` or `Disable`	`Disable`
List Server Interface	Enables internal debug log writing for the list server interface	`Enable` or `Disable`	`Disable`
Local Delivery	Enables internal debug log writing for the local delivery module	`Enable` or `Disable`	`Disable`
Log Message Body	Enables internal debug log writing for the message body	`Enable` or `Disable`	`Disable`
OCI Calls	Enables internal debug log writing for OCI calls	`Enable` or `Disable`	`Disable`
Outbound Main Module	Enables internal debug log writing for the outbound main module	`Enable` or `Disable`	`Disable`
Queue Processor	Enables internal debug log writing for the queue processor	`Enable` or `Disable`	`Disable`
Recovery Module	Enables recovery module debugging	`Enable` or `Disable`	`Disable`
Relay Module	Not supported in this release	`Enable` or `Disable`	`Disable`
Server Response	Enables debugging for the SMTP server responses	`Enable` or `Disable`	`Disable`
Statistics	Not supported in this release	`Enable` or `Disable`	`Disable`
Submit Module	Enables internal debug log writing for the submit module	`Enable` or `Disable`	`Disable`

Housekeeping

Table 8–10 Housekeeping Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Advanced Queue Cleanup	Informs the housekeeping process to clean up process control related data from the system	`Enable` or `Disable`	`Disable`
Statistic Cleanup	Informs the housekeeping process to clean up statistics data from the system	`Enable` or `Disable`	`Disable`

Table 8–10 Housekeeping Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Statistics Logging	Logs performance data (latency of each SQL execution) to the log file	`Enable` or `Disable`	`Disable`
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`
Log Miner Recovery	Enables log miner based mail recovery by using garbage collection throughput	`Enable` or `Disable`	`Disable`

List Server

Table 8–11 List Server Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
Address Rewriting Rules	Enables debugging for address rewriting rule processing	`Enable` or `Disable`	`Disable`
DNS Module	Enables DSN module debugging	`Enable` or `Disable`	`Disable`
External Filter	Enables debugging for the external filter process	`Enable` or `Disable`	`Disable`
List Server Interface	Enables debugging for the list server interface	`Enable` or `Disable`	`Disable`
LDAP Resolution	Enables debugging for LDAP resolution	`Enable` or `Disable`	`Disable`
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`
Local Delivery	Enables local delivery module debugging	`Enable` or `Disable`	`Disable`
OCI Calls	Enables debugging for OCI calls	`Enable` or `Disable`	`Disable`
Relay Module	Not supported in this release	`Enable` or `Disable`	`Disable`

NNTP

Table 8–12 NNTP Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`

Virus Scrubber

Table 8–13 Virus Scrubber Debug Level Parameters

Parameter	Description	Acceptable Values	Default Value
LDAP (DS) Call Tracing/Logging	Enables directory service layer tracing and logging of the server	`Enable` or `Disable`	`Disable`
Statistics Logging	Logs performance data (latency of each SQL execution) to the log file	`Enable` or `Disable`	`Disable`

WebMail Properties

Table 8–14 describes the WebMail properties in alphabetical order. Values given for the properties indicated for database and LDAP connection pooling while using OJMA, and for database connection pool parameters, are just examples and should actually be determined based on the number of users, and system load.

To edit these properties, change their values in the `oc4j.properties` file.

Table 8–14 WebMail Toolkit Properties

Property	Description	Acceptable Values	Default Value
`client.image.corporate`	URLs to use to customize corporate brand with Oracle Collaboration Suite	Any URL that points to an image	`/um/images/branding_collaborationsuite.gif`
`client.image.product`	URLs to use to customize corporate brand with Oracle	Any URL that points to an image	`/um/images/corporateBrand_oracle.gif`
`client.corporate.url`	Destination for the corporate logo link:	Any URL that points to a corporate url	`/um/traffic_cop`
`client.esdsconnpoolparam.incrementsize`	Number of connections to add to the esds client connection pool	Any integer	`1`
`client.esdsconnpoolparam.initialsize`	Initial number of connections in the esds client connection pool	Any integer	`5`
`client.esdsconnpoolparam.maxsize`	Maximum number of connections in the esds client connection pool	Any integer	`10`
`client.esdsconnpoolparam.minsize`	Minimum number of connections in the esds client connection pool	Any integer	`5`
`client.esdsconnpoolparam.shrinkingtimeoutinterval`	Time delay before esds client connection pool can be shrunk	Any integer	`1800`

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
client.esdsconnpoolparam.timeoutinterval	Maximum number of seconds the esds client waits for a free connection in the pool. If no connections are released back to the pool within that time, the directory server code throws an exception.	Any integer	30
client.ldapsearch.maxresult	Maximum number of results returned from an LDAP search. End users whose searches return more matches than this are notified that additional results exist, but that only this maximum number are shown.	Any integer	500
client.mail.defaultsort	If TRUE, the WebMail client automatically sorts by the default sort field and order, when user first logs in. The default sort field and order are set in the MailAppConstants parameter.	TRUE or FALSE	TRUE
client.mail.enforcenestedHTMLfonts	If TRUE, original fonts are to be used for HTML.	TRUE or FALSE	FALSE
client.mail.messagetransport	If SMTP, messages are sent through the SMTP server; DATABASE uses direct database interaction.	SMTP or database	SMTP
client.mail.translate.INBOX=true	If TRUE, the INBOX folder is auto-translated to the user's locale. If FALSE, the INBOX is displayed in English.	TRUE or FALSE	TRUE
client.message.charset.default	Default character set to use for outgoing messages	Any valid character set	
client.portal.url	Destination for the default portal icon link: Default is http://www.oracle.com	Any valid URL	http://www.oracle.com
client.privacystatement.url	Destination for the privacy statement link: Default is http://www.oracle.com	Any valid URL	http://www.oracle.com
client.product.url	Destination for the product logo link: Default is /um/traffic_cop		/um/traffic_cop
jdbc.connection.debug	If TRUE, enables debugging JDBC connections.	TRUE or FALSE	FALSE
mail.debug	If TRUE, enables debugging OJMA API for Oracle Email.	TRUE or FALSE	FALSE

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
mail.host.qualifiedname	Domain to add to non-qualified address recipients. If this is not set, then the user's domain is used.	Any valid domain	None
mail.imap.host	Form for installation time substitution of variables		%machinehost%
mail.imap.port	Port used when toolkit.mailstore=IMAP; the IMAP port number	Any valid IMAP port number	143
mail.smtp.host	Host used for sending messages through SMTP when client.mail.messagetransport is not set to database. This host machine is running the SMTP server.	Any valid SMTP host machine	%machinehost%
mail.smtp.port	Port used for sending messages through SMTP when client.mail.messagetransport is not set to database. This port is on the machine running the SMTP server.	Any valid SMTP port number	25
MaxTelephonePinDigits	Maximum number of digits in voice mail PINs	Any integer	12
MinTelephonePinDigits	Minimum number of digits in voice mail PINs	Any integer	7
oracle.mail.admin.ldapDebug	Enables debugging for the administration ESDS API	TRUE or FALSE	FALSE
oracle.mail.admin.ui.ojmaDebug	Enables debugging for the administration OJMA API	TRUE or FALSE	FALSE
oracle.mail.portlet.httpsToWebmail	Indicates if the WebMail title link is http or https.	TRUE or FALSE	FALSE
oracle.mail.client.prefs.autoreply	Enables access to auto reply features and options. If TRUE, enables autoreply functions for end users	TRUE or FALSE	TRUE
oracle.mail.client.prefs.autoreply.echo	Every incoming message receives an auto reply with the original message copied. Requires oracle.mail.client.prefs.autoreply to be enabled. If TRUE, enables autoreply functions for end users	TRUE or FALSE	TRUE

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
oracle.mail.client.prefs.autoreply.reject	Enables the user to select the reject option in the UI. With the reject option enabled, the server rejects all incoming messages Requires oracle.mail.client.prefs.autoreply to be enabled. If TRUE, enables autoreply functions for end users.	TRUE or FALSE	TRUE
oracle.mail.client.prefs.autoreply.reply	Enables the reply option in the UI. With the reply option enabled, every sender receives one auto reply regardless the amount of messages sent by that sender. Requires oracle.mail.client.prefs.autoreply to be enabled. If TRUE, enables autoreply functions for end users	TRUE or FALSE	TRUE
oracle.mail.client.prefs.autoreply.vacation	Enables the vacation option in the UI. With the vacation option enabled, every incoming message receives an auto reply with the original message copied. Requires oracle.mail.client.prefs.autoreply to be enabled. If TRUE, enables autoreply functions for end users	TRUE or FALSE	TRUE
oracle.mail.ldap.reconnectime	The amount of time in seconds the server waits to reconnect to the LDAP server if it is unavailable. If set to 0, the server connects immediately. If the value is not set, the reconnect time is the same as the timeout value		
oracle.mail.ldap.connectssl	If set to TRUE, enables clients to use SSL connections to Oracle Internet Directory		
oracle.mail.sdk.esmail.cache_inactivity_timeout	Number of seconds to wait for a connection before the esds client connection pool times out	Any integer values OJMA connection pool settings	300

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
`oracle.mail.sdk.esmail .cache_scheme`	Determines the cache scheme of the database connection pool DYNAMIC dynamically increases or shrinks the pool size based on the system load. FIXED WAIT waits for a specified period if a connection is not available. FIXED-NO-WAIT returns null if a connection is not available in the pool. For more details, please refer to the Oracle JDBC Developer Guide.	1 = DYNAMIC 2 = FIXED WAIT 3 = FIXED NO WAIT Use `1` or `3` for best results	1
`oracle.mail.sdk.esmail .connpool_max_limit`	Maximum number of connections in the Oracle mail sdk esmail connection pool	Any integer values OJMA connection pool settings	`10`
`oracle.mail.sdk.esmail .connpool_min_limit`	Determines the initial or minimum number of connections created in the connection pool Oracle recommends keeping this limit as low as possible to avoid holding on to unused database connections.	Depends on the number of users, system load, etc.	1
`oracle.mail.sdk.esmail .driver_type`	Determines the type of jdbc driver to be used for the database connection pool	oci8, THIN (recommended in non-RAC)	oci8
`oracle.mail.sdk.esmail .encryption`	Disables password encryption, which is mandatory in OCS V2. Can be disabled for more performance.	`FALSE or TRUE`	`TRUE`
`oracle.mail.sdk.esmail .ldap_debug`	If TRUE, enables debugging OJMA API for LDAP.	`TRUE` or `FALSE`	`TRUE`
`oracle.mail.sdk.esmail .ojma_debug`	Controls the debug output from the ojma layer	`FALSE` or `TRUE`	`FALSE`

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
toolkit.clientdir	Directory under $ORACLE_HOME/j2ee/OC4J_UM/applications/UMClientApp/um_client where the UIX pages reside: Default is /templates/	Any valid path under $OH/j2ee/OC4J_UM/applications/UMClientApp/um_client where the UIX pages reside	/templates/
toolkit.controller.url	URL for accessing the client framework controller: Default is /um/traffic_cop	A valid URL that accesses the client framework controller	/um/traffic_cop
toolkit.debugmode	Debugging mode for the WebMail client	TRUE or FALSE	FALSE
toolkit.helpdir	Relative URL path for the image, javascript, and online help files: Default is /um/help/	Any valid path to directories containing the associated files	/um/help/
toolkit.imagedir	Relative URL path for the image, javascript, and online help files: Default is /um/images/	Any valid path to directories containing the associated files	/um/images/
toolkit.jslibdir	Relative URL path for the image, javascript, and online help files: Default is /um/scripts/	Any valid path to directories containing the associated files	/um/scripts/
toolkit.logdirectory	File path location of the WebMail client log files: Default is %ORACLE_HOME%/um/log	A valid file path containing the log files	%ORACLE_HOME%/um/log
toolkit.logfilename	Name for the WebMail client log file		Webmail_Client
toolkit.loghostclient	Name for the WebMail Client host		%machinehost%

Table 8–14 WebMail Toolkit Properties (Cont.)

Property	Description	Acceptable Values	Default Value
toolkit.loglevel	Logging level of the WebMail client, from the five choices shown	internalerror warning notification trace error	error
toolkit.mail.listsubscribedfoldersonly	If TRUE, only subscribed folders are displayed.	TRUE or FALSE	FALSE
toolkit.pagesuffix	Type of suffix to append when going to targets in statefile.xml		.uix
toolkit.servlet.version	Differentiates how the servlet is forwarded to the UIX pages: should be set to 2.1 or higher. If set to <= 2.0 it reverts back to servlet.	Any servlet version	2.2
toolkit.statefile	Location and name of the statefile: %ORACLE_HOME%/um/client/config/statefile.xml is the default. %ORACLE_HOME% is translated to the real path in the oc4j.properties file. All values containing the percent symbol (%) are substituted to prevent the oc4j.properties file from containing variables	Any valid file path to a file that contains the statefile definitions	%ORACLE_HOME%/um/client/config/statefile.xml

WebMail LDAP Properties

The LDAP properties in Table 8–15 apply to all LDAP servers for search:

Table 8–15 LDAP properties for all LDAP servers

Property	Description
toolkit.ldap.pool.timeout=1000	The number of milliseconds that an idle connection can remain in the pool without being closed and removed from the pool
toolkit.ldap.pool.initsize=5	Initial size for a directory service connection pool.
toolkit.ldap.pool.prefsize=5	Preferred size for a directory service connection pool.
toolkit.ldap.pool.maxsize=10	Maximum size for a directory service connection pool.

Table 8–15 LDAP properties for all LDAP servers

Property	Description
`toolkit.ldap.search.timeout=2000`	Length of time in milliseconds a directory service search times out.
`toolkit.ldap.results.maxsize=200`	Limits the maximum size for Directory Service search results.
`toolkit.ldap.dir.total=1`	Number of LDAP directory services. Use 0 if none is available.

The LDAP properties in Table 8–16 are contained in each LDAP server:

Table 8–16 LDAP properties contained in each LDAP server

Property	Description
`toolkit.ldap.dir.1.label=Corporate Directory`	Text string to display for the directory service. The text string is "Corporate Directory," it is translated to the user's locale.
`toolkit.ldap.dir.1.url=ldap:`*`enter_your_host_here`*`:389`	Directory service host name and port number. The format is ldap://*host:port*. By default, LDAP servers listen on port 389.
`toolkit.ldap.dir.1.searchbase=dc=`*`your_subdomain`*`,dc=`*`your_domain`*	Search base for the directory service. Change this value to reflect the search base for the directory service.
`toolkit.ldap.dir.1.username=`*`enter_username_here`*	Sets the user name and password. Uncomment this to set the user name and password for a directory service that requires access control. All users using the WebMail must access the directory service using this name and password.
`toolkit.ldap.dir.1.password=`*`enter_password_here`*	

Searching Multiple Directories

To list and search multiple directories in the client, create records for each. Each subsequent directory service must have a unique ordinal. For example, the second directory service should be `toolkit.ldap.dir.2`.

Example

```
toolkit.ldap.dir.2.label=Acme Directory
toolkit.ldap.dir.2.url=ldap://ldap.acme.com:389
```

```
toolkit.ldap.dir.2.searchbase=c=US
```

Set the following two values:

```
toolkit.ldap.dir.2.username=enter_username_here
toolkit.ldap.dir.2.password=enter_password_here
```

Log Files

The process logs are written in `$ORACLE_HOME/oes/log/`*`install_name/process_name/pid/pid`*`.log`. Five different log categories determine the amount of information the servers produce. Table 8–17 lists them by name and value from least to largest quantity of output, with corresponding recommended responses.

Table 8–17 Log Levels, Meanings, and Responses

Error Level	Numeric Value and Meaning	Recommended Response
Internal Errors	0 is not a valid priority; this is the null priority	None: the message content is not properly categorized.
Internal Errors	1-5 are serious internal errors revealing that internal state is corrupted: report a bug	Administrator should file a bug with Oracle support.
Errors	6-10 are normal errors; can be corrected or addressed.	Error condition needs to be corrected.
Warnings	11-15 are warnings of possible problems.	Condition exists: may require attention
Notification	16-20 are informational notifications.	None: informational only, e.g., "initialization complete"
Trace	21-25 are used for tracing by field support engineers.	None: administrator is not the intended audience.
Dump	26 is used for debugging by the product development group	None: administrator is not the intended audience.

Log File Locations

Table 8–18 provides administration log file locations.

Table 8–18 Administration Log Files

Name	UNIX	Windows
Administration	`$ORACLE_HOME/opmn/logs/OC4J_UM.default_island`	`None`
Preferences	`$ORACLE_HOME/opmn/logs/OC4J_UM.default_island`	`None`
Portlet	`$ORACLE_HOME/opmn/logs/OC4J_UM.default_island`	`None`

Table 8–19 provides server log file locations.

Table 8–19 Server Log Files

Server	UNIX	Windows
IMAP	`$ORACLE_HOME/oes/log/`*`install_name`*`/imap/`*`pid`*`/`*`pid`*`.log`	`%ORACLE_HOME\oes\log\`*`install_name`*`\imap\`*`pid`*`\`*`pid`*`.log`
POP	`$ORACLE_HOME/oes/log/`*`install_name`*`/pop/`*`pid`*`/`*`pid`*`.log`	`%ORACLE_HOME\oes\log\`*`install_name`*`\pop\`*`pid`*`\`*`pid`*`.log`
SMTP In	`$ORACLE_HOME/oes/log/`*`install_name`*`/smtp_in/`*`pid`*`/`*`pid`*`.log`	`%ORACLE_HOME\oes\log\`*`install_name`*`\smtp_in\`*`pid/pid`*`.log`
SMTP Out	`$ORACLE_HOME/oes/log/`*`install_name`*`/smtp_out/`*`pid`*`/`*`pid`*`.log`	`%ORACLE_HOME\oes\log\`*`install_name`*`\smtp_out\`*`pid`*`\`*`pid`*`.log`
Housekeeping Mail Store	`$ORACLE_HOME/oes/log/gc/`*`SID`*`.*/text.log`	`%ORACLE_HOME\oes\log\gc\`*`SID`*`.*\text.log`
Houskeeping Middle Tier	`$ORACLE_HOME/oes/log/`*`install_name`*`/gc./`*`pid`*`/`*`pid`*`.log`	`%ORACLE_HOME\oes\log\`*`install_name`*`\gc.\`*`pid`*`\`*`pid`*`.log`
List Server	$ORACLE_HOME/oes/log/*install_name*/list/*pid*/*pid*.log	`%ORACLE_HOME%\oes\log\`*`install_name`*`\list\`*`pid`*`\`*`pid`*.log
NNTP In	$ORACLE_HOME/oes/log/*install_name*/nntp_in/*pid*/*pid*.log	`%ORACLE_HOME%\oes\log\`*`install_name`*`\nntp_in\`*`pid`*`\`*`pid`*.log

Table 8–19 Server Log Files (Cont.)

Server	UNIX	Windows
NNTP Out	$ORACLE_HOME/oes/log/*install_name*/nntp_out/*pid*/*pid*.log	%ORACLE_HOME%\oes\log*install_name*\nntp_out*pid**pid*.log
WebMail	Default: $ORACLE_HOME/um/log/Webmail_Client WebMail logging is configured by properties in oc4j.properties for the OC4J_UM application. Where: *toolkit.loghostclient* maps to a field in the log files to indicate what machine generated the log file. *toolkit.logleve*l indicates the amount of logging to do, and can be set to one of the following values; internalerror, error, warning, notification, trace. *toolkit.debugmode* controls whether or not debug info is logged for use of the ESDS API by the WebMail client code.	n/a

9

Error Messages

This chapter explains how to interpret error messages and correct errors. It lists the error codes in numerical order, divided into the following groups:

- Overview
- IMAP and POP
- SMTP
- Housekeeping
- List Server
- NNTP
- WebMail
- Virus Scrubber

Overview

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

When a list of error messages, called an error stack, is displayed, 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.

IMAP and POP

This section describes the IMAP and POP error messages. The notations 1 and 2, which appear in many cells in the "Action to Handle the Error," mean the following:

1. "Make sure all required packages have been loaded into the database correctly."
2. "In particular, check whether `ES_FOLDER_API` has been loaded. "

Table 9–1 IMAP and POP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
101, 0, Login failed	Invalid user name or password used for `LOGIN` command.	Check the user name and password. Try again.
102, 0, No of auth/login tries exceeded. Exiting	Used all your allowed login attempts	Check the user name and password; then retry in a new session.
103, 0, User logged out	IMAP/POP session ended either by `LOGOUT/QUIT` command or because of some other fatal server error, such as "unable to read or write to client connection anymore."	Session end by `LOGOUT/QUIT` command is normal. If you suspect an abnormal connection termination, check the server log file for other errors in this error chain.
104, 0, Authorization succeeded	Successful login using authenticate command	None
105, 0, Authorization failed	Unsuccessful login attempt using authenticate command	Check the user credentials and try again.
106, 0, Could not retrieve folder id for folder={sarg0}. Error#{narg0}	Possibly a nonexistent folder name was used.	Correct the folder name and try again. If the folder name is correct, check and resolve any other database errors in this error chain.

Table 9–1 IMAP and POP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
107, 0, Failed to get header info for folder={sarg0} with fid={narg1}. Error#{narg0}	Could be due to an OCI error	[1] (1). (2). Check and resolve any other database errors in this error chain.
108, 0, Failed to update folder={sarg0} with fid={narg1}. Error#{narg0}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
109, 0, Failed to connect to database {sarg1}. Error#{narg0}	Server unable to create OCI connection pool.	Make sure the database is up and configured correctly in Oracle Internet Directory.
110, 0, Connected to database {sarg1}	Successful connections to the database	None
111, 0, Failed to get statement handle {narg1} with Error#{narg0}.	Database related error	Check for an OCI error in this error chain.
112, 0, Autologout: idle {narg0} minutes.	Your session was idle for too long	Send noop or any other command before timeout.
113, 0, Out of free Memory. Requested {narg0} bytes.	No more free memory is available to the server.	Reduce the load on the server by reducing any of following: threads, max. clients, OCI sessions, or Oracle Internet Directory connections. Make sure enough free memory is available for the server on your system.
114,0, Module {sarg0}: nesting level too deep, no stats	Internal error	Contact customer support.
117, 0, Failed to get body parts for messageID={narg0}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
118, 0, Failed to get database session for db={sarg0}. Error#{narg0}	No more free sessions are available in the OCI connection pool.	This error may be temporary, 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}	Database error.	Check the OCI errors in this error chain.

Table 9–1 IMAP and POP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
120, 0, Failed to rename folder={sarg0})to {sarg1}. Error#{narg0}	▪ Trying to rename a nonexistent folder; or ▪ The new name is already in use; or ▪ Rename is not allowed.	Make sure that a folder with the old name exists and that the 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}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
122, 0, Failed to get shell for msgid={narg0}. Error#{narg1}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
123, 0, Failed to create hierarchical folders {sarg0}. Error#{narg0}	▪ You cannot create INBOX in any case insensitive form. ▪ You may be trying to create a folder that already exists.	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}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
125, 0, Bad flags list	Syntax error in the flag list for the Store command.	Correct the syntax for the flag list.
126, 0, Failed to get folder Id for folder={sarg0}. Error#{narg0}	▪ You may be looking for a nonexistent folder. ▪ You may not have read permissions for a shared folder.	Make sure you are looking for the right folder and its name 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}	Database error	Check and resolve database errors in this chain.
128, 0, Failed to delete shared folder={sarg0}. Error#{narg0},{sarg1}	▪ You may be trying to delete a nonexistent folder. ▪ Only the shared folder owner can delete the shared folder.	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.

Table 9–1 IMAP and POP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
129, 0, Failed to rename shared folder={sarg0} to {sarg1}. Error#{narg0},{sarg2}	■ You may be trying to rename a nonexistent folder. ■ Only a shared folder owner can rename it. ■ The new name is already in use or is not allowed.	Make sure you are the owner of the shared folder, or retry with a different name.
130, 0, Failed to change ACI on shared folder={sarg0}. Error#{narg0},{sarg1}	Database error	Check the error logs for the database and Oracle Internet Directory.
131, 0, Failed to determine if this folder or any child is shared.{sarg0}.Error#{narg0}	Could be due to an OCI error	(1). (2). Check and resolve any other database errors in this error chain.
132, 0, Failed to determine Folder space usage for user={sarg0}. Error#{narg0}	Could be due to an OCI error	(1). (2). 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}	One of the required message attributes is missing in the database.	(1).

[1] (1) means "Make sure all required packages have been loaded into the database correctly."
(2) means "In particular, check whether ES_FOLDER_API has been loaded."

SMTP

This section describes the SMTP error messages.

Table 9–2 SMTP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
100, 0, Memory allocation failed	The process is consuming too much memory.	Reduce the number of threads running and restart the process.
101, 0, Memory realloc failed	The process is consuming too much memory.	Reduce the number of threads running and restart the process.
103, 0, failed to create thread	There are too many threads in the process.	Reduce the number of threads and restart the server. If the problem persists, contact technical support.
175, 0, ESDSGetEntry failed {sarg0}	The Oracle Internet Directory server may be down.	Restart the Oracle Internet Directory server. If the problem still exists, contact technical support.
176, 0, ESDSGetEntry for entrytype failed {sarg0}	The Oracle Internet Directory server may be down.	Restart the Oracle Internet Directory server. If the problem still exists, contact technical support.
177, 0, ESDSGetAttribute failed for {sarg0}	The Oracle Internet Directory server may be down.	Restart the Oracle Internet Directory server. If the problem still exists, contact technical support.
200, 0, loop detected for the recipient: {sarg0}	The address resolution for the recipient resulted in a loop.	Make sure the data present in the Oracle Internet Directory server does not introduce any loops for the recipient. Check whether the auto forward attribute for the recipient introduces a chain ending with the original recipient.
201, 0, orclobjectid not populated in Oracle Internet Directory for usr: {sarg0}	Mandatory attribute `orclobjectid` is missing in Oracle Internet Directory.	Populate correct value for the user in Oracle Internet Directory.
205, 0, failed to deliver to user inbox: {sarg0}		(1). Check whether `ES_MESSAGE_API` has been loaded.
208, 0, failed to index msg for user: {sarg0} index type: {sarg1}		(1). Check whether `ES_OT_API` has been loaded.
209, 0, message rejected by rules for usr: {sarg0}	The user rule resulted in rejection of the message.	None.

Table 9–2 SMTP Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
210, 0, message rejected by the recipient {sarg0} using replymode: reject	Auto reject is set in the Oracle Internet Directory entry for the recipient.	
212, 0, failed to delete local recipients	There may be OCI errors.	(1). Check whether `ES_MESSAGE_API` has been loaded.
213, 0, local delivery failed for user: {sarg0}		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	Error in recovery processing.	(1). In particular, check whether `ES_QUEUE_API` has been loaded.
226, 0, failed to requeue messages	Error in recovery processing.	(1). In particular, check whether `ES_QUEUE_API` has been loaded.
243, 0, path for external filter process is NULL in Oracle Internet Directory		Populate `external filter process` with the path for the virus scanner executable if virus scanning is enabled.
302, 0, User {sarg0} logon failed. Oracle Internet Directory returns {narg0}	Unable to authenticate user in Oracle Internet Directory.	Check user name and password to see if they are correct.
401, 0, Error {narg0}: Unable to get msgid	Unable to get next message ID from database.	Check whether the schema has been installed and whether the package is valid.
402, 0, Error {narg0}: Unable to store envelope	Unable to insert envelope information into database.	Check whether has been installed and whether the package is valid.
403, 0, Error {narg0}: Unable to store recipient	Unable to insert recipient information into database.	Check whether has been installed and whether the package is valid.
404, 0, Error {narg0}: Unable to store {sarg0} queue	Unable to insert the message into a queue.	Check whether has been installed and whether the package is valid.
405, 0, Error {narg0}: Unable to insert the message	Unable to insert message into database.	Check the OCI error and the ORACLE error.

Table 9–2 SMTP Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
406, 0, Error: Routing loop detected	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.	Check the rewriting rules and auto-forward setup, and notify the sender.
407, 0, Error: Unable to read from client	Unable to read from client.	Check the network connections.
500, 0, spam check failed for IP address: {sarg0}	DNS server failed to verify that the IP address of the SMTP client is correct.	
501, 0, spam check failed for host: {sarg0}	DNS server failed to verify that the host is a valid internet host.	
502, 0, spam check failed for sender: {sarg0}	The sender is either in in the list of rejected senders or in the list of rejected domains.	
503, 0, spam check failed for recipient: {sarg0}	■ Relay is not allowed for the non local recipient's domain, or ■ The local recipient is in the list of rejected recipients.	
650, 0, failed to get submit recipients	Could be due to OCI errors.	
651, 0, failed to delete submit recipients	Could be due to OCI errors.	(1). In particular, check whether `ES_MESSAGE_API` has been loaded.
652, 0, failed to insert resolved recipients	Could be due to OCI errors.	(1). In particular, check whether `ES_MESSAGE_API` has been loaded.

Housekeeping

This section describes the housekeeping error messages.

Table 9–3 Housekeeping Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
Oracle error {sarg0} occurred during expiration	An RDBMS error prevented Housekeeper from successfully performing expiration.	Correct the generic RDBMS error and try running Housekeeper again.
Oracle error {sarg0} occurred during queue pruning	An RDBMS error prevented Housekeeper from successfully performing pruning.	Correct the generic RDBMS error and try running Housekeeper again.
Oracle error {sarg0} occurred during pruning	An RDBMS error prevented Housekeeper from successfully performing pruning.	Correct the generic RDBMS error and try running Housekeeper again.
Oracle error {sarg0} occurred during collection	An RDBMS error prevented Housekeeper from successfully performing collection.	Correct the generic RDBMS error and try running Housekeeper again.
Oracle error {sarg0} occurred during tertiary storing	An RDBMS error prevented Housekeeper from successfully performing tertiary storage.	Correct the generic RDBMS error and try running Housekeeper again.

List Server

This section describes the list server error messages.

Table 9–4 List Server Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
Msg-id: 5002 (An error occurred while performing a database operation. Error= {sarg0})	The cause for this error is available in the error message.	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})	The Oracle Internet Directory server ■ is down, or ■ has stopped responding, or ■ is listening on a different port.	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)	Either the database or the Oracle Internet Directory server ■ is not running, or ■ has stopped responding.	Restart the database and Oracle Internet Directory server. If they are running, then restart the list server.

Table 9–4 List Server Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
Msg-id: 5021 (Error modifying user {sarg0} entry. Error = {narg0})	An Oracle Internet Directory error occurred while trying to process a command for the user.	Check whether if the user entry on the Oracle Internet Directory server is still valid.
Msg-id: 5031 (Failed to resolve message {narg0} for external list {sarg0}. Error : {sarg1})	The cause for this error is available in the error message itself.	
Msg-id: 5025 (Error occurred while parsing command in message {narg0} : {sarg0})	The cause for the error is available in the message itself.	Correct the mail and resend it.
Msg-id: 5026 (Message {narg0} will not be processed because auth info is not available for this message)	The `orclmaillistserverauthenticatedsender` attribute is set as only for the List Server process and this mail does not have an authenticated sender.	Check whether authentication is turned on in the SMTP inbound server and the mail has been sent with authentication.
Msg-id: 5029 (Failed to recover messages)	A internal error occurred.	
Msg-id: 5030 (Failed to store message {narg0} in archive for the list {sarg0}. Error : {sarg1})	The cause for this error is available in the error message itself.	

NNTP

This section presents the NNTP error messages.

Table 9–5 NNTP Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
6000, 0, An error occurred while initializing the NNTP process.	▪ An error occurred while querying the directory server, or ▪ The server parameters had incorrect values.	Ensure that the directory server is running and all the server parameters have been set correctly.
6001, 0, Unable to initialize directory services. Server DN {sarg0}	The directory server was not running or there was an error in the command line parameters.	Check that the directory server is running and all command line parameters have been specified correctly.
6002, 0, Unable to initialize database services. Mail store {sarg0}	▪ The mail store database was down ▪ The listener was down	Ensure that the database and the listener for the mail store to which the process is connected are running.
6003, 0, Unable to allocate {narg0} bytes	The server could not obtain memory from the operating system.	Restart the server. If the problem persists, shut down other processes and also increase memory resources on the host computer
6004, 0, Database error {narg0}: {sarg0}	▪ The mail store database was down ▪ The listener was down Additional information is available in the error message.	Ensure that the database and the listener for the mail store to which the process is connected are running.
6005, 0, Directory service error {narg0}: {sarg0}	The directory server was not running. Additional information is available in the error message.	Ensure that the directory server is running.
7000, 0, Unable to initialize Oracle Net Services. Presentation name {sarg0}. Listener port {narg0}	▪ The Oracle Net Listener is not running ▪ It is not configured correctly ▪ The server parameters do not match the listener configuration.	Ensure that the server parameters match the listener configuration and that the Oracle Net Listener is running.
7001, 0 , Unable to obtain connection pool to mail store {sarg0}	The server could not initialize connection to the mail store.	Ensure that the mail store database instance is running and accepting connections.

Table 9–5 NNTP Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
7100, 0, Incompatible parameters specified: {sarg0} and {sarg1}	The parameters that have been specified are not compatible.	Consult the server documentation for more information on how to specify compatible parameters
7101, 0, Authentication failed for {sarg0}	The server received an authentication request with invalid credentials.	Verify if this is an authentication attempt by a genuine user
7102, 0, Too many authentication failures	The server detected three successive authentication failures from the same host and the connection was terminated.	Verify that these are authentication attempts by genuine users
7103, 0, Connection rejected. Disallowed domain {sarg0}	The server received a connection from a domain that is not allowed.	If connections from this domain must be allowed, the anti-spam configuration must be edited to allow this domain.
7104, 0, Connection rejected. Disallowed IP address {sarg0}	The server received a connection from a host that is not allowed.	If connections from this host must be allowed, the anti-spam configuration must be edited.
7105, 0, A database operation resulted in an error. OCI Error {narg0}: {sarg0}	A mail store operation failed.	None
7106, 0, Unable to obtain database handle to mail store {sarg0}. Error {narg0}	The server is unable to open new connections to the mail store.	Check whether the mail store database instance is running and accepting connections
7108, 0, Message rejected. Virus scan failed. (Subject: {sarg0}) (Message-ID: {sarg1})	The virus scanner detected a virus in an incoming message. The message was not delivered.	None
7110, 0, Operation {sarg0} not allowed for reader {sarg1}	The server received a feed-related request from a news reader client.	Edit the server configuration if you want to make this host a peer.
7111, 0, Parameter {sarg0} not specified. Using default {sarg1}	A required parameter was not specified. The default value was used instead.	The server configuration must be edited to specify a value for the parameter.
7112, 0, Parameter {sarg0} not set	A parameter value was not specified.	Edit the server configuration to set the parameter value.
7113, 0, No peers configured or unable to initialize all peers	The server instance is configured to allow feed. But not peer servers have been specified.	Edit the configuration to specify feed servers.

Table 9–5 NNTP Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
7114, 0, Instance identity initialization failed. Unable to determine host name	The server could not determine the name of the host on which it is running.	The name lookup service must be configured to return the host name.
7115, 0, Unable to locate peer entry: {sarg0}	A peer specified in the server configuration is invalid.	The peer configuration must be edited to specify valid peers.
7116, 0, Unable to initialize metrics collection	The metrics subsystem could not be initialized.	None
7117, 0, Unable to initialize new client connection	A new client request could not be accepted.	None
7118, 0, Invalid local group name: {sarg0}	The server detected an invalid group in the directory.	Remove the group and re-configure.
7119, 0, Unable to initialize process control subsystem	Unknown	Check whether the administration store specified in the server configuration is the same as the mail store.
8000, 0, An error occurred when starting a new thread.	The operating system limit for the maximum number of threads within a process was reached.	Increase the maximum limit on the number of threads, or reduce the value of the maximum number of threads parameter for the process.
8001, 0, An error occurred while establishing an NNTP connection with peer {sarg0}.	No route could be established to the specified peer.	Ensure that the host name and port specified for the peer are valid, and the NNTP server on the peer is running.
8002, 0, An operating system error occurred in the system call {sarg0}. Error {narg0}	An operating system error occurred in a system call.	Check the operating system error and fix accordingly.
8003, 0, Authentication to peer {sarg0} failed. Error {narg0}	The authentication information available in the peer entry was not accepted by the peer.	Ensure that the values specified for the user name and password in the peer entry are valid.
8100, 0, Failed to return connection to the peer connection cache: {sarg0}	Unknown	None
8101, 0, Failed to send an IHAVE command to peer {sarg0}	Article transmission to peer host failed.	Check that the peer is running and accepting articles.
8102, 0, Failed to read a response to the IHAVE command from peer {sarg0}	Article transmission to peer host failed.	Check that the peer host is alive and accepting articles.

Table 9–5 NNTP Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
8103, 0, Failed to transmit article with {sarg0} to peer {sarg1}	Article transmission to peer failed.	Check that the peer is alive and accepting articles.
8104, 0, An error occurred while establishing an NNTP connection with peer {sarg0}	Connection to peer host failed.	Check that the peer is alive and accepting articles.
8105, 0, An operating system error occurred in the system call {sarg0}. Error {narg0}	An error occurred in the operating system.	Check the server configuration and verify that the operating system has enough resources to support the server.

WebMail

This section describes the WebMail error messages.

Table 9–6 Webmail Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
An error occurred while adding attachments	WebMail was unable to add the attachments.	Try again.
No folder name was specified	The user did not specify a folder name.	Enter a folder name.
An error occurred; unable to create the new folder	WebMail was unable to create the folder.	Try again.
A folder by that name **<foldername here>** already exists	The user specified a folder name that is being used by another folder.	Name the folder with a new name or put the folder in a different location.
An error occurred while creating the message	WebMail could not create a new message object.	Try creating a message again.
No valid To: recipients found	User did not specify a valid e-mail address in the **To** field.	Specify a valid e-mail address.
Error occurred during message creation	WebMail could not create a new message object.	Try creating a message again.
Invalid parameter specified for attachment removal	WebMail experienced a problem when removing the attachment.	Contact your system administrator.
Invalid attachment index was received	WebMail attachment indices are misaligned.	Recreate the message.
No message IDs were specified for deletion	The user did not select messages for deletion.	Select the message for deletion.
An error occurred during message deletion	The message does not exist.	Contact your system administrator.
An error occurred while compacting the folder	This problem lies with the voice mail messages in the folder.	Contact your system administrator.
No message IDs were specified for forwarding	The user did not select a message before selecting **Forward**.	Select a message before selecting **Forward**.
More than one message specified for forwarding	Multiple messages were selected for forwarding.	Select one message at a time for forwarding.

Table 9–6 Webmail Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
Invalid message specified	The message selected could not be forwarded.	Try selecting another message. If that does not work, contact your system administrator.
An error occurred while preparing the message for forwarding	The selected message could not be processed for forwarding.	Try again or contact your system administrator.
The destination folder does not exist	The destination folder selected does not exist.	Select another destination folder.
No message IDs were specified for move	The user did not select a message before selecting **Move**.	Select a message before selecting **Move**.
An error occurred while performing message move	WebMail could not process the move request.	Try again or contact your system administrator.
There are no more messages in this folder	No messages exist before or after the current message.	Try another folder.
An error occurred opening the next message	WebMail could not open the next message.	Try again or contact your system administrator.
There are no messages before this one in this folder	No messages exist before or after the current message.	Try another folder.
An error occurred opening the previous message	WebMail could not open the previous message.	Try again or contact your system administrator.
Unable to find folder	The folder is not accessible.	Check the shared permissions or contact your system administrator.
Folder does not exist	There is no such folder in the account.	Contact your system administrator.
An error occurred while opening the folder	WebMail experienced problems opening the folder.	Contact your system administrator.
Error occurred during communication with the message store	Possibly a network problem.	Contact your system administrator.
No message ID specified	Internal error.	Contact your system administrator.
Error retrieving message	The message may have been deleted, but the browser is looking at cached or old pages.	Refresh the message list and try again.
No message IDs were specified for reply	The user did not check any messages before selecting **Reply.**	Select a message before selecting **Reply.**

Table 9–6 Webmail Error Messages (Cont.)

Error Number and Message	Cause of the Error	Action to Handle the Error
More than one message specified for reply	The user selected multiple messages for reply.	Select only one message at a time.
Invalid message specified	WebMail could not process the message for reply.	Try again or contact your system administrator.
Error occurred while preparing the message for reply	Internal error	Contact your system administrator
Error while sending message	Internal error	Contact your system administrator.
No folder was specified for editing	The user did not select a folder in the folder list before selecting **Edit**.	Select a folder from the folder list.
The specified folder does not exist in the mail store	The folder selected is not available.	Verify that the folder exists, or contact your system administrator.
An error occurred while preparing the folder for editing	Internal Error	Contact your system administrator.
You cannot rename special system folders	The user tried to rename the inbox.	None. The inbox cannot be renamed.
No new name was specified	The user did not specify a name for the folder.	Specify a name for the folder.
A folder with that name already exists	Internal error	Contact your system administrator.
Unable to rename folder	Internal error	Contact your system administrator.
An error occurred while trying to update the folder	Internal error	Contact your system administrator.
Error while setting previous state	Internal error	Contact your system administrator.
You are no longer connected to the mail store	The session has timed out.	Contact your system administrator.

Virus Scrubber

This section describes the virus scrubber error messages.

Table 9–7 Virus Scrubber Error Messages

Error Number and Message	Cause of the Error	Action to Handle the Error
10001, 0, Failed to create database connections, error={narg0}.	The server is unable to establish a database connection.	Check the generic Oracle error before this message for the exact cause.
10002, 0, Fatal database occurred.	A fatal Oracle error prevented the process from functioning.	Check the generic Oracle error before this message and correct the database problem if needed. The process restarts itself.
10007, 0, Failed to logon to the directory server, error={narg0}.	LDAP authentication failed for the server.	This error is a rare . Contact Oracle Support for more information.
10008, 0, Warning: External virus software not configured, scanning disabled.	Server parameter `orclMailScannerInterfaces` parameter is not set correctly.	Not an error. If the external virus scanner is not configured, the server does not perform virus scanning. If scanning is intended, set the parameter correctly and refresh the process.

A

Shared Folders

This section discusses the Oracle Email shared folder.

This appendix contains the following topics:

- Overview of Shared Folders
- Understanding Access Control Lists for Shared Folders
- Managing Public Folders

Overview of Shared Folders

User folders can be shared with other users, distribution lists, or with everyone in a user's domain using access control lists (ACLs). The Oracle Email system supports ACLs defined in RFC 2086.

RFC 2086 defines the following namespaces:

- Other user's namespace: A namespace that consists of mailboxes from the personal namespaces of other users.
- Shared namespace: A namespace that consists of mailboxes that are intended to be shared amongst users.

In Oracle Email system, folders within other user's namespaces are referred to as shared folders, and folders within shared namespaces are referred to as public folders.

If a folder is shared with everyone in a domain, it is called a public folder. Otherwise, if the folder is shared with one or more users or distribution lists, it is called a shared folder.

For IMAP, Oracle Email has the following prefixes for shared and public folders:

- `#Shared/`: All shared folders accessible to the user appear under this name space in the folder listing
- `#Public/`:All the public folders appear under this name space in the folder listing

> **See Also:** *Oracle Email Application Developer's Guide* for shared folder information using OJMA

Understanding Access Control Lists for Shared Folders

ACLs are used to share folders with other identifiers in the Oracle Email system. Oracle Email identifiers are listed in Table A–1.

Table A–1 Oracle Email Identifiers

Identifier	Description
Users	Users are explicitly granted permissions to a folder
Distribution lists	A distribution list is granted permission, which implies that all the members of the distribution list have complete rights. If a member is added or removed from a distribution list, the rights are automatically updated.

Table A–1 Oracle Email Identifiers

Identifier	Description
Domain	All the users in the domain have the specified permissions.

Folders cannot be shared across domains. To share folders within a domain, you must have either domain or system administrator privileges.

The following domain rights can be granted:

Table A–2 Domain Rights

Domain Right	Description
l – lookup	Allows folders to be listed
r – read	Allows messages to be read from a folder
s – seen/unseen flag	Allows `seen` and `unseen` flag changes to be kept across sessions
w – write	Enables flags other than seen and delete to be stored
i – insert	Enables messages to be appended or copied into a folder
d – delete	Enables deleted flags for messages to be stored in a folder or expunge the folder
a – administer	Enables ACLs to be set and deleted on folders that are owned by other users

Oracle Email always grants `lookup` rights with other rights. All rights are grouped with `lookup` privileges. None of the other rights are tied together.

The following rules apply to folders:

- Folders cannot be shared across domains
- Folder owners have all rights on their folders
- Only owners can rename or delete shared folders
- Sub-folders under a shared folder do not inherit any rights
- It is possible for multiple identifiers in an ACL to apply to a given user. For example, an ACL can include rights that are granted to a domain and mailing list that the user is a member of. In such cases, a union of rights are granted to the user. If a user is given specific rights, then only those rights at the user level are applicable

For example, consider a user who is a member of group G1 and list L1, which been granted the following rights:

```
identifier     rights
==========     =======
G1             li
L1             lrs
```

In this case the user's rights are a union of li and lrs, or lrsi, as derived from membership in group G1 and list L1, respectively.

User level rights take precedence over other rights. For example, if a user has lr rights at the user level, then the applicable rights are lr.

Managing Public Folders

You must have system or domain privileges to create public folders. Public folders are first created in an administrator's private namespace. To make the folders public, you must give rights to the domain identifier.

Once a public folder is created, it can be administered by other administrators without specifying any rights. Public folders must be created with a unique name because they do not have a user name prefix. For example, if administrator A1 creates a public folder called public1, then administrator A2 cannot create a public folder with the same name.

Public folders count towards the owner's e-mail quota.

Administrators can grant more rights to other identifiers. For example, a user can be granted insert (i) rights to add messages to a public folder. This folder appears twice in the user's folder listing: as a public folder and as a shared folder.

B

Alias and Distribution List Look Up

This section discusses alias and distribution list look up.

This appendix contains the following topics:

- Enabling Alias Lookup From Standard Clients
- Enabling Distribution List Lookup From Standard Clients

Enabling Alias Lookup From Standard Clients

> **Note:** Lookup is available in WebMail without having to perform end user configuration. The lookup facility is available when addressing an e-mail that the end user is composing, and when adding members to a distribution list in the address book, in addition to the basic Address Book search that is available on every page through the Search bar under the sub tab region.

To enable e-mail alias lookup from a standard client, such as Netscape Communicator, perform the following steps:

1. Run the `$ORACLE_HOME/oes/bin/oesSearchUtil.sh` script to enable or disable alias lookups from standard clients:

   ```
   $ORACLE_HOME/oes/bin/oesSearchUtil.sh -type alias -option <enable or
   disable> -domain email_domain
   ```

   where:

   `email_domain` is the e-mail domain name for which this option must be enabled or disabled.

2. Look up e-mail aliases from standard clients with a search base as `root`, or the e-mail alias container, such as:

   ```
   cn=Alias,domain_dn,cn=um_
   system,cn=EmailServerContainer,cn=Products,cn=OracleContext
   ```

   where:

   `domain_dn` is the domain DN.

   For example, if the e-mail domain is `acme.com`, then the value is the string `dc=acme,dc=com`

Another suggested configuration is to create a referral at the public namespace. For example, the following sample `ldif` file creates a container `cn=emailsearchbase,subscriber_dn` containing two referrals to the public users container and e-mail alias container:

```
dn: cn=emailsearchbase, subscriber_dn
cn: emailsearchbase
objectclass: top
objectclass: referral
```

```
objectclass: extensibleObject
ref:
ldap://oid_host:oid_port/cn=Alias,domain_dn,cn=um_
system,cn=EmailServerContainer,cn=Products,cn=OracleContext
ldap://oid_host:oid_port/cn=Users,subsriber_dn
```

where:

subscriber_dn is the distinguished name of the subscriber in Oracle Internet Directory.

oid_host is the Oracle Internet Directory host name

oid_port is the Oracle Internet Directory port

domain_dn is the domain DN.

For example, if the e-mail domain is `acme.com`, then the value is the string `dc=acme,dc=com`

Enabling Distribution List Lookup From Standard Clients

Note: Lookup is available in WebMail without having to perform end user configuration. The lookup facility is available when addressing an e-mail that the end user is composing, and when adding members to a distribution list in the address book, in addition to the basic Address Book search that is available on every page through the Search bar under the sub tab region.

E-mail distribution lists and membership information are synchronized between the private e-mail namespace and the public namespace to enable distribution list lookup from standard clients.

This synchronization option can be enabled or disabled using the WebMail administration pages.

For `dlsync` to work, the public DL container must be created and should have all permissions for the `EmailAdminsGroup`.

Sample Public Distribution List Container LDIF File

```
dn: cn=dlContainer, subsciber_dn
changetype: add
objectclass: top
```

```
objectclass: orclContainer
cn: dlContainer
orclaci: access to entry by
group="cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products,cn=OracleContext"
(add,delete,browse)
orclaci: access to attr=(*) by
group="cn=EmailAdminsGroup,cn=EMailServerContainer,cn=Products,cn=OracleContext"
(read,write,search,compare)
```

where:

subscriber_dn is the distinguished name of the subscriber in Oracle Internet Directory

Sample Object Class Definition LDIF File

```
dn: cn=subschemasubentry
changetype: modify
add: objectclasses
objectclasses: ( objectclass_oid NAME 'mailgroup' SUP groupofuniquenames
AUXILIARY MAY ( mail ) )
```

where:

objectclass_oid is the unique identifier for the `mailgroup` object class

Distribution List Synchronization Utility

The `esdssyncdl` utility synchronizes distribution lists from the e-mail private namespace under the `cn=EMailserverContainer` to a public namespace. This allows standard clients, such as Netscape Communicator, to see the distribution lists through anonymous searches. You can run `esdssyncdl` occasionally to dump or redump all distribution lists from a private namespace to a public namespace.

Whenever any update and delete occurs on the members of the private distribution list, the changes are reflected in the public distribution list. When you add or delete a distribution list using the WebMail administration pages, it occurs in the public namespace.

Synchronizing One or Multiple Distribution Lists

Running `esdlsync` with an input file containing a list of distribution lists, with one distribution list name in each line synchronizes the private e-mail namespace under `cn=EMailServerContainer,cn=Products,cn=OracleContex`, to a public

name space. Use this option when you only have a few distribution lists to synchronize.

The syntax is as follows:

```
esdssyncdl ldaphost=ldap_host      (mandatory)
 port=ldap_port                    (mandatory)
 username=superuser_DN             (mandatory)
 password=superuser_pass           (mandatory)
 preferencelocation=DN_of_the_Dl_preferences   (mandatory)
Detail: DN where Dl preferences is located. ( See Note 1 & 3. )
 inputfile=filepath
Detail: full path of the file with dls to sync,
one dl's mailid per line.  ( see Note 2 )
 flags=More_options_with_which_sync_can_be_modified
Detail: flags=all
```

The following is a usage example:

```
esdssyncdl ldaphost=gmldap01 port=389 password=welcome
username=cn=orcladmin inputfile=/tmp/dlfile
preferencelocation=dc=us,dc=oracle,dc=com,cn=um_system,
cn=EMailServerContainer,cn=Products,cn=OracleContext
```

Synchronizing All Distribution Lists from a Private E-mail Namespace

Synchronizing all private distribution lists under the `cn=EMailServerContainer`, to a public namespace can be done as a one-time task. This is the default option.

Use this option when your deployment has distribution lists populated under the `cn=EmailServerContainer` (the e-mail private namespace), you want to add the lists to a public namespace so that a standard client can see them.

The syntax is as follows:

```
esdssyncdl ldaphost=ldap_host      (mandatory)
port=ldap_port                     (mandatory)
username=superuser_DN              (mandatory)
password=superuser_pass            (mandatory)
preferencelocation=DN_of_the_Dl_preferences   (mandatory)
        Detail: DN where Dl preferences is located.
flags=<More options with which sync can be modified>
        Detail: flags=all
```

The following is a usage example:

```
esdssyncdl ldaphost=gmldap01 port=389 password=welcome
username=cn=orcladmin flags=all
preferencelocation=dc=us,dc=oracle,dc=com,cn=um_system,
cn=EMailServerContainer,cn=Products,cn=OracleContext
```

The following preferences are set in the domain where the distribution lists are present:

- `orclmaildlsynccontainerdn` is the distinguished name of the container where all public distribution lists are created.
- `orclmaildlsyncnamingattr` is the naming attribute used for public distribution lists
- `orclmaildlsyncattrstosync` is the list of attributes to be synchronized from private to public distribution lists
- `orclmaildlsyncobjectclass` is the list of `objectclasses` to be synchronized from private to public distribution lists

For the `inputfile` content, the mail ID of the distribution list should be added to each line that is to be synchronized to a public namespace.

- `dlcorp_us@acme.com`
- `dleng_app@acme.com`
- `dlsupport_us@acme.com`

The distinguished name of the LDAP location where distribution list preferences is located. This helps in running multiple `esdssyncdl` commands simultaneously for a different domain.

The DN of the domain should be the nearest domain under which all distribution lists are present.

For example, if the private distribution lists are present under:

```
cn=List,dc=us,dc=oracle,dc=com,cn=um_
system,cn=EMailServerContainer,cn=Products,cn=OracleContext
```

then the distinguished name of the preference should be as follows:

```
dc=us,dc=oracle,dc=com,cn=um_
system,cn=EMailServerContainer,cn=Products,cn=OracleContext
```

If the `inputfile` and `flags=all` options are both specified, synchronization occurs based on the `inputfile` data.

When the `sync` utility is run for a distribution list that exists in a public namespace, all existing members of the public distribution list are replaced with the members of the private distribution list.

C

Oracle Email Access Control Lists

This section discusses the access control list policies set for Oracle Email in Oracle Internet Directory. 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

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.

The installation process creates the following privilege group:

```
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 proxy to the `EmailServerContainer`.
- Access control list with `DN = 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. The bind mode "Simple" is added to restrict anonymous lookups using certain client tools, such as Netscape Navigator.
- Access to the e-mail subtree is denied to everybody else.

> **See Also:** *Oracle Internet Directory Administrator's Guide* for more information on access control lists

Oracle Internet Directory 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 C–1 Oracle Internet Directory Group Membership Permissions

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=Grou ps,%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 e-mail servers to compare the `orclpasswordverifier;email` attribute. This is required for 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

Has read, search, compare, selfwrite, and 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 `acme.com` domain is the string `dc=acme,dc=com`

> **Note:** This group is present in a system where domain administrators have been created from the WebMail 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
```

D

Co-existence

This section discusses how Oracle Email can co-exist with other mail systems.

This appendix contains the following topics:

- Overview
- MX Records
- Oracle Email Co-existence Features

Overview

Email systems are, by design, intended to co-exist with other systems. Industry standards, such as SMTP and MIME enable exchange of information between end-users. In some organizations, a deeper level of co-existence is required. For example, during the process of migrating from a legacy e-mail product to Oracle, both the old and new systems are operational. Some special considerations are required for the continued delivery of messages in these environments.

The following are examples of co-existence:

- You have an Oracle Email system, and you need to exchange messages with the outside world.
- You are migrating, and need to co-exist within your organization for a short period
- Other units of your organization use other mail systems, and are not migrating to Oracle Email

MX Records

The main issue co-existing environments must address is the role played by the exchanger (MX) record. The MX record value defines the physical network address messages for a given e-mail domain. Remote message transfer agents (MTAs) can only resolve this address by domain name.

Routing messages to the correct system of the destination mailbox can be solved in the following ways:

- Each co-existing mail system maps to a unique domain name. This is the simplest method to implement, but can be less convenient for end users. A single enterprise e-mail domain for all users is not possible, and a user's address is dependent on their e-mail system.
- Manage an alias database, possibly at an additional inbound MTA, that maps domain addresses to their destination inbox. This method works well, but creates some extra complexity for the mail administrators.
- Configure the mail system to intelligently route non-local messages to the other system. This method satisfies most requirements, although support for this type of behavior varies between vendors.

Oracle Email Co-existence Features

When messages arrive at the Oracle Email MTA, the address is examined to determine if the addressee is local. This is checked by matching the address against the `Local Domains` server setting. If the domains match and directory lookup fails to resolve the address, then the address is unknown. If the domain does not match any of the local domains, the message is kept in the relay queue.

In special circumstances, the MTA does not reject the message, but it relays it to another MTA. Server settings enable the MTA to be configured to pass on any message that is addressed to a valid local domain, but where the mailbox address is unknown. This method enables messages to be addressed using a single global domain, without the need for managing an alias database.

Perform the following steps to enable this feature:

1. Add the flag `-l` to the `Mail Process Flags` parameter of the MTA processes.
2. Set the `Relay Host` parameter to the name of the MTA to route messages.

Aliases and Rewriting Rules

A successful co-existence strategy requires that different systems can exchange messages with each other, and that users can easily address messages to reach the intended recipient.

Features of the Oracle Email, such as aliases and address rewriting rules, can assist in co-existence. Aliases provide a simple way for users to look up recipients in a directory, without needing to understand details of how messages are routed inside the system. Address rewriting rules to assist co-existence by recognizing addresses and automatically altering the intended routing path.

For example, in some cases it may be necessary to route messages to their target by including the actual location of the mail server, such as `user@smtpin.acme.com` instead of `user@acme.com`. Such an address enables the successful transport of messages from one system to another, but may result in delivery failure since the full e-mail address is not recognized at its destination. Using address rewriting, the pattern `*@smtpin.acme.com` can be rewritten to `*@acme.com` to ensure that the recipient mailbox is found. Using aliases, end users can easily select address messages from the directory without needing to know the details of the actual e-mail address required for delivery.

Troubleshooting

When configuring Oracle Email in a co-existence environment, it is sometimes difficult to trace exactly why problems occur. In order to debug any problems, you need to be familiar with the underlying protocols and routing of e-mail. Problems typically occur due to incorrect message addressing, or unexpected address re-writing by intermediate MTAs, particularly through Sendmail.

Troubleshooting tips include:

- Ensure that you fully understand the exact route that messages take between the different MTAs in your organization, and how each of these process messages in their delivery.
- Ensure that messages are deliverable from this MTA, if you are routing messages to unknown mailboxes in the local domain to a relay machine.
- When routing messages to Oracle Email from another MTA, ensure that the exact address matches against the Local Domains and any appropriate re-writing rules of the SMTP Inbound processes.
- Check that messages are actually delivered to Oracle Email with the correct addresses by using debugging tools such as command line mail (on Unix), or by using Telnet to the SMTP Inbound port directly.
- Increase the log level of the SMTP Inbound processes and examine the process log files. The log records attempted mail delivery to Oracle Email, including sender and recipient address information.

E

Server Statistics

The DBMS_STATS package generates statistics for the entire Oracle Email table or index. Statistics are transferred between the 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 appendix contains the following topics:

- POP Statistics
- IMAP Statistics
- SMTP In Statistics
- SMTP Out Statistics
- Housekeeping Statistics
- List Server Statistics
- NNTP In Statistics
- NNTP Out Statistics
- Virus Scrubber

POP Statistics

Table E–1 describes the POP server statistics:

Table E–1 POP Server Statistics

Statistic	Description
.um.admin.os_pid	Operating system process ID
.um.admin.uptime	Amount of time the server has been up
.ES_SPS.socket.currload	Current number of client connections
.ES_SPS.socket.sockmax	Maximum number of client connections allowed
.ES_SPS.thread.currthreads	Number of threads the server is currently using
.ES_SPS.thread.thrmax	Maximum number of threads the server creates.
.ESPROTO.uptime	Amount of time the server has been up
.ESPROTO.COMMAND.total	Total number of executed commands
.ESPROTO.COMMAND.<PO3_COMMAND>.totalcalls	Total number of calls for that command
.ESPROTO.COMMAND.<PO3_COMMAND>.success	Total number of successfull calls for that command
.ESPROTO.COMMAND.<PO3_COMMAND>.fail	Total number of failed calls for that command
.ESPROTO.USERS.LOGIN.<user id>	A value of 1 or more indicates that the user is still logged in. Otherwise, it is 0
.ESPROTO.connections.lost	Total number of client connections that have disconnected
.ESPROTO.connections.timeout	Total number of client connections that have timed out
.ESPROTO.connections.total	Total number of client connections
.ESPROTO.receive.bytes	Total number of bytes received by the server
.ESPROTO.transmit.bytes	Total number of bytes sent by the server

IMAP Statistics

Table E–2 describes the IMAP server statistics:

Table E–2 IMAP Server Statistics

Statistic	Description
.um.admin.os_pid	Operating system process ID
.um.admin.uptime	Amount of time the server has been up
.ES_SPS.socket.currload	Current number of client connections
.ES_SPS.socket.sockmax	Maximum number of client connections allowed
.ES_SPS.thread.currthreads	Number of threads the server is currently using
.ES_SPS.thread.thrmax	Maximum number of threads the server creates
.MTA.uptime	Time string describing when this MTA came up
.MTA.connections.in.current	Current number of inbound SMTP Connections
.MTA.connections.in.total	Total number of inbound SMTP connections
.MTA.msgs.deferred.current	Current number of messages deferred
.MTA.msgs.deferred.total	Total number of messages deferred
.MTA.receive.kbytes	Total number of kilobytes received
.MTA.receive.messages	Total number of messages received
.MTA.receive.recipients	Total number of recipients received
.MTA.receive.time	Total time receiving data
.MTA.transmit.bytes	Total number of bytes transmitted
.MTA.transmit.bytes_local	Total number of bytes transmitted to local entities
.MTA.transmit.messages	Total number of messages transmitted
.MTA.transmit.messages_loca	Total number of messages transmitted to local entities
.MTA.transmit.recipients	Total number of recipients transmitted
.MTA.transmit.recipients_local	Total number of recipients transmitted to local entities

SMTP In Statistics

Table E–3 describes the SMTP In server statistics:

Table E–3 SMTP In Server Statistics

Statistic	Description
.um.admin.os_pid	Operating system process ID
.um.admin.uptime	Amount of time the server has been up
.ES_SPS.socket.currload	Current number of client connections
.ES_SPS.socket.sockmax	Maximum number of client connections allowed
.ES_SPS.thread.currthreads	Number of threads the server is currently using
.ES_SPS.thread.thrmax	Maximum number of threads the server creates
.MTA.uptime	Time string describing when this MTA came up
.MTA.connections.in.current	Current number of inbound SMTP Connections
.MTA.connections.in.total	Total number of inbound SMTP connections
.MTA.msgs.deferred.current	Current number of messages deferred
.MTA.msgs.deferred.total	Total number of messages deferred
.MTA.receive.kbytes	Total number of kilobytes received
.MTA.receive.messages	Total number of messages received
.MTA.receive.recipients	Total number of recipients received
.MTA.receive.time	Total time receiving data
.MTA.transmit.bytes	Total number of bytes transmitted
.MTA.transmit.bytes_local	Total number of bytes transmitted to local entities
.MTA.transmit.messages	Total number of messages transmitted
.MTA.transmit.messages_loca	Total number of messages transmitted to local entities
.MTA.transmit.recipients	Total number of recipients transmitted
.MTA.transmit.recipients_local	Total number of recipients transmitted to local entities

SMTP Out Statistics

Table E–4 describes the SMTP Out server statistics:

Table E–4 SMTP Out Server Statistics

Statistic	Description
.um.admin.os_pid	Operating system process ID
.um.admin.uptime	Amount of time the server has been up
.MTA.uptime	Time string describing when this MTA came up
.MTA.connections.broken	Number of broken connections encountered by the MTA
.MTA.connections.failed	Number of failed connections from the MTA to another MTA
.MTA.connections.rejected	Number of rejected connections
.MTA.connections.rejection_reason	Description of the reason for the most recent rejection
.MTA.connections.out.current	Current number of outbound SMTP connections
.MTA.connections.out.current_foreign	Current number of outbound SMTP connections to MTAs in foreign domains
.MTA.connections.out.current_native	Current number of outbound SMTP connections to MTAs in native domains
.MTA.connections.out.total	Total number of outbound SMTP connections
.MTA.connections.out.total_foreign	Total number of outbound SMTP connections to foreign domains
.MTA.connections.out.total_native	Total number of outbound SMTP connections to MTAs in native domains
.MTA.dl.receive.count	Number of messages sent to distribution lists
.MTA.msgs.deferred.current	Current number of messages deferred
.MTA.msgs.deferred.total	Total number of messages deferred
.MTA.msgs.delivered.total time	Total time inserting data into the database
.MTA.ndr.inbound	Total number of non delivery reports generated by inbound mail
.MTA.ndr.loop	Total number of messages not delivered due to mail loops

Table E–4 SMTP Out Server Statistics

Statistic	Description
.MTA.ndr.outbound	Total number of non delivery reports generated by outbound mail
.MTA.queued.out.kbytes	Kilobytes queued awaiting to be sent out to the Internet
.MTA.queued.out.messages	Messages queued awaiting to be sent out to the Internet
.MTA.transmit.bytes	Total number of bytes transmitted
.MTA.transmit.bytes_foreign	Total number of bytes transmitted to foreign domain MTA's
.MTA.transmit.bytes_local	Total number of bytes transmitted to local entities
.MTA.transmit.messages	Total number of messages transmitted
.MTA.transmit.messages_foreign	Total number of messages transmitted to foreign domain MTA's
.MTA.transmit.messages_local	Total number of messages transmitted to local entities
.MTA.transmit.messages_native	Total number of messages transmitted to native domain MTA's
.MTA.transmit.messages_relay	Total number of messages transmitted during relay operations
.MTA.transmit.recipients	Total number of recipients transmitted
.MTA.transmit.recipients_foreign	Total number of recipients transmitted to foreign domain MTA's
.MTA.transmit.recipients_local	Total number of recipients transmitted to local entities
.MTA.transmit.recipients_native	Total number of recipients transmitted to native domain MTA's
.MTA.transmit.time	Total time transmitting data
.MTA.transmit.time_foreign	Total time transmitting data to foreign domain MTA's
.MTA.transmit.time_native	Total time transmitting data to native domain MTA's
.MTA.transmit.time.local	Total time spent transmitting data to local entities
.MTA.transmit.time.relay	Total time transmitting data during relay operations

Housekeeping Statistics

Table E–5 describes the housekeeping server statistics:

Table E–5 Housekeeping Server Statistics

Statistic	Description
.GC.processed.expirables	Number of message instances expired by a particular housekeeping process
.GC.processed.prunables	Number of message instances removed from the system trash folder by a particular housekeeping process
.GC.processed.queued_ prunables	Number of message references removed from the system trash queue by a particular housekeeping process
.GC.processed.collectables	Number of unreferenced messages removed from the system by a particular housekeeping process
.GC.processed.tertiary_ storables	Number of messages moved to tertiary storage by a particular housekeeping process
.GC.pending.expirables	Number of message instances awaiting expiration remaining in the system
.GC.pending.prunables	Number of message instances remaining in the system trash folder
.GC.pending.queued_ prunables	Number of message references remaining in the system trash queue.
.GC.pending.collectables	Number of identified unreferenced messages remaining in the system
.GC.pending.tertiary_ storables	Number of messages remaining in the system that are eligible for tertiary storage

List Server Statistics

Table E–6 describes the list server statistics:

Table E–6 List Server Statistics

Statistic	Description
.SLIST.connections.busy	Number of busy database connections
.SLIST.connections.total	Total number of database connections
.SLIST.process.current_ mail_threads	Number threads running in the server processing mails
.SLIST.process.current_ mails	Number of mails being processed in the server
.SLIST.process.current_ user_threads	Number of threads running in the server that are delivering mails to users
.SLIST.process.total_mails	Total number of mails that have been processed by the server since startup
.SLIST.queue.pending	Number of e-mails waiting to be processed by the list server
.um.admin.os_pid	Operating system process ID
.um.admin.uptime	Amount time the server has been up

NNTP In Statistics

Table E–7 describes the NNTP In server statistics:

Table E–7 NNTP In Server Statistics

Statistic	Description
.es.nntp.in.art.cache.hit	Article cache hits
.es.nntp.in.art.cache.miss	Article cache misses
.es.nntp.in.clients.article	Number of ARTICLE commands from clients
.es.nntp.in.clients.current	Number of connected clients
.es.nntp.in.clients.group	Number of GROUP commands from clients
.es.nntp.in.clients.list	Number of LIST commands from clients
.es.nntp.in.clients.post	Number of POST commands from clients
.es.nntp.in.clients.total	Number of clients serviced so far

Table E–7 NNTP In Server Statistics

Statistic	Description
.ES_SPS.socket.currload	Current client load
.ES_SPS.socket.sockmax	Max. clients allowed
.ES_SPS.thread.currthreads	Number of service threads
.ES_SPS.thread.thrmax	Max. service threads
.um.admin.os_pid	Process PID
.um.admin.uptime	Process uptime
.um.admin.log.discard	Number of discarded log messages
.um.admin.log.total	Number of total log messages

NNTP Out Statistics

Table E–8 describes the NNTP Out server statistics:

Table E–8 NNTP Out Server Statistics

Statistics	Description
.es.nntp.out.threads	Number of live feed threads, may be zero
.es.nntp.out.conn.cache.hit	Connection cache hits
.es.nntp.out.conn.cache.miss	Connection cache misses
.es.nntp.out.traffic.errors	Feed errors
.es.nntp.out.traffic.messages	Feed messages
.es.nntp.out.traffic.rejects	Feed rejects
.um.admin.os_pid	Process PID
.um.admin.uptime	Process uptime
.um.admin.log.discard	Number of discarded log messages
.um.admin.log.total	Number of total log messages

Virus Scrubber

Table E–9 describes the virus scrubber statistics:

Table E–9 Virus Scrubber Statistics

Statistics	Description
.VSCRUB.processed.prescan	Number of messages pre-scanned by the process
.VSCRUB.processed.scan	Number of messages scanned by external virus scanning software via this process
.VSCRUB.processed.infected	Number of messages deemed infected by the virus scanning software
.VSCRUB.processed.repaired	Number of messages deemed infected by the virus scanning software but able to repair and restore
.VSCRUB.pending.scan	Number of messages isolated by pre-scanning but yet to be scanned by external virus scanning software
.VSCRUB.threads.count	Number of active virus scanner threads at the moment

F

Oracle Email Supported RFCs

This appendix provides a list of the request for comments (RFCs) that are supported by Oracle Email.

Table F–1 Supported RFCs

RFC Number	Title
RFC 821	Simple Mail Transfer Protocol (SMTP)
RFC 822	Standard for the format of ARPA Internet text messages
RFC 850	Standard for Interchange of USENET Messages
RFC 0977	Network News Transfer Protocol
RFC 1034	Domain Names - Concepts and Facilities
RFC 1035	Domain Names - Implementation and Specification
RFC 1036	Standard for Interchange of USENET Messages
RFC 1123	Requirements for Internet hosts - application and support
RFC 1652	SMTP Service Extension for 8bit-MIME transport
RFC 1869	SMTP Service Extensions
RFC 1870	SMTP Service Extension for Message Size Declaration
RFC 1891	SMTP Service Extension for Delivery Status Notifications
RFC 1893	Enhanced Mail System Status Codes
RFC 1894	An Extensible Message Format for Delivery Status Notifications (DSNs)
RFC 1939	Post Office Protocol - Version 3
RFC 2034	SMTP Service Extension for Returning Enhanced Error Codes

Table F–1 Supported RFCs

RFC Number	Title
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
RFC 2060	Internet Message Access Protocol - Version 4rev1
RFC 2086	IMAP ACL extension
RFC 2087	IMAP QUOTA extension
RFC 2088	IMAP non-synchronous literals
RFC 2177	IMAP IDLE command
RFC 2342	IMAP Namespace
RFC 2821	Simple Mail Transfer Protocol
RFC 2859	IMAP UIDPLUS extension
RFC 2980	Common NNTP Extensions
RFC 3463	Enhanced Mail System Status Codes (obsoletes 1893)

Index

M

O

P

S

T

V