Oracle® Collaboration Suite

Anti-Spam Configuration

Release 2 (9.0.4)

Part No. B15614-01

November 2004

This document describes the problems caused by spam in an enterprise and Internet environment and discusses the different ways that spam is intercepted, handled, and blocked.

This document contains the following sections:

Introduction

The term **spam** refers to unsolicited, commercial e-mail sent indiscriminately to multiple mailing lists, individuals, or news groups. Spam is also commonly known as junk e-mail. Spam can also be of a noncommercial nature, for example, a joke or a chain letter.

Overview of Spam Problems

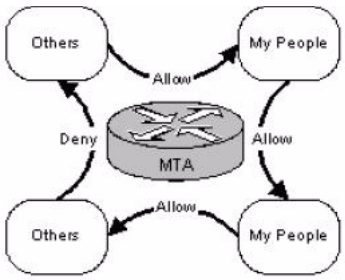
Every organization works to protect itself against two primary spam problems: open relays and unwanted e-mail.

ORACLE

Open Relays

An open relay is created when an internal e-mail resource of an organization can be used by third parties to route e-mail traffic from outside the organization (for example, from the Internet) to other outside recipients.

Figure 1 Open Relays



Unwanted E-mail

Each organization usually enables its users to receive e-mails from anyone or anywhere. But the organization's resources might be used to receive and store unsolicited e-mails that are not related to the business. This is the unwanted e-mail problem, which is generally addressed by evaluating the contents of each incoming e-mail message to determine whether or not to mark it as spam.

Preventing Spam

To protect e-mail systems against spam, organizations can use three approaches.

Disabling Open Relays

To prevent open relay spamming, organizations can:

- Disable the open relay parameter of their e-mail application
- Allow only known sources to use their **Message Transfer Agent** (MTA) as an e-mail relay.

Configuring Native E-mail Anti-Spam Features

Most e-mail systems have native anti-spam features that can be set to block e-mails from certain senders based on certain characteristics such as:

- IP addresses
- System names
- Certain domains with respect to the organization’s rules and regulations

Implementing Content-Filtering Software

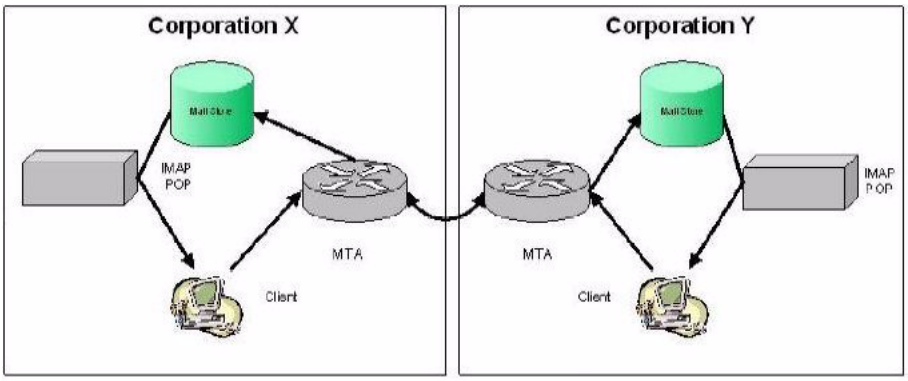
Organizations can block spam by examining each e-mail and judging whether the message is a spam or not. Anti-Spam software from third party vendors can scan e-mail messages and identify spam by applying program logic and knowledge bases.

About Message Transfer Agents

A simple view of a standard e-mail system consists of three different services: an e-mail client, an e-mail store, and an MTA.

The MTA delivers an e-mail message to the addresses specified by the originator (or sender) of the message.

Figure 2 View of Standard E-mail System Delivery



An outgoing e-mail message from a user in Corporation X sent to a user in Corporation Y travels through one or more MTAs in Corporation X, and eventually it makes it to the MTA for incoming mail in Corporation Y. Note that the MTAs within an organization route e-mail messages for e-mail clients, as well as other MTAs. In fact, some e-mail systems are configured so that there is no difference between client-to-MTA or MTA-to-MTA e-mail traffic.

The most common MTA for the UNIX environment is **Sendmail**, which communicates using the Simple Mail Transfer Protocol (SMTP).

Built-In Anti-Spam Methods in Oracle Email

Oracle provides its own MTA. The Oracle Email SMTP server supports a variety of built-in anti-spam methods to prevent Oracle users and domains from being spammed and servers from being used as a relay by external domains.

The various spam prevention methods are described in the following table:

Method	Description
Blocking relays	The SMTP server can be set up to block all relaying. Alternatively, it can be set up to relay to only a known set of domains. For example, to allow relaying of all messages received to recipients within the `foo.com` domain, the Relay Domains Allowed parameter can be set up to `foo.com`.
Rejecting messages received from certain domains	The Reject Domains parameter contains a list of domains from which messages are not allowed.
Rejecting messages from certain senders	The Reject Senders parameter contains the list of senders from whom messages will not be accepted.
Rejecting messages for certain recipients	The Reject Recipients parameter contains the list of recipients who are not allowed to receive messages.

Method	Description
Preventing **Denial of Service** (DoS) attacks	The SMTP Spam Maximum Flood Count and Spam Flood Interval parameters can be set to control the number of messages coming from a host. If connections and messages received from a particular host exceed the limits set in the Spam Maximum Flood Count and Spam Flood Interval parameters, then all further messages from that host are rejected until the rate of incoming messages and connections from the host becomes lesser than the Spam Maximum Flood Count level.

Overview of the Oracle MTA

The Oracle MTA separates the e-mail delivery and routing services into two separate processes.

SMTP_IN

The `SMTP_IN` process listens for incoming connection requests and decides if the incoming messages are for local delivery or need to be routed to another MTA for delivery elsewhere. When it receives incoming messages, it queries the Oracle Internet Directory server to find and authenticate the addresses and rewrites the addresses based on the rewriting rules. It also applies anti-spam rules. If all the steps are successful, the SMTP MTA accepts the message and inserts it into the corresponding queue based on the destination address. If SMTP process is configured to check for spam and the incoming client connection does not match the anti-spam requirements, the SMTP MTA rejects the message.

SMTP_OUT

The `SMTP_OUT` process is responsible for relaying a message to another MTA.

Some applications can insert mail directly into the Oracle Email system, without going through the `SMTP_IN` process. These applications use the PL/SQL interfaces provided by Oracle Collaboration Suite and by definition, are local and trusted by the system as an e-mail delivery mechanism.

Spam Policies

A spam policy is defined by an organization to prevent the unsolicited receipt of e-mail as defined by the RFC-2505 definition of Spam.

While Oracle Collaboration Suite enables you to configure anti-spam policies for both types of MTA services (in and out), you will spend most of your time managing and administering the `SMTP_IN` process.

If you set up news services using Oracle Collaboration Suite's NNTP server, you can protect them from spam as well. The architecture of a **Network News Transport Protocol** (NNTP) server is similar to the MTA. Managing spam for NNTP is done with the `NNTP_IN` process.

Configuring Oracle Email Anti-Spam

To configure anti-spam features on the Oracle Email MTAs, you need to have a complete understanding of how anti-spam works both generically and from the

Oracle Email perspective. You also need to have Oracle Collaboration Suite installed.

There are several parameters and a bit of logic required to check for spam. The best way to configure anti-spam is to use the Policy pages available to all administrators of Oracle Email. Once a user is set up in Oracle Collaboration Suite as an administrator, the Webmail client exposes an additional set of Administration pages, including policy pages for setting up anti-virus and anti-spam parameters. The Policy pages automatically update the anti-spam parameter values of the Enterprise Manager/Unified Messaging `SMTP_IN` process.

Note: In the current release of Oracle Email, the labeling of the anti-spam parameters in the Policy page does not match those in the Enterprise Manager/Unified Messaging `SMTP_IN` process. However, these parameters are indeed the same. This is a BUG (Bug #3228008) and will be corrected in the next release of the product. Refer to the table in the Additional Information for a list of the anti-spam parameters in the Policy page and its equivalent in the Enterprise Manager/Unified Messaging `SMTP_IN` process.

SMTP Settings

Oracle Collaboration Suite Release 2 (9.0.4) enables administrators to configure anti-spam parameters in two locations, the Oracle Enterprise Manager interface and the Policy page. Changes made in the Policy page interface are visible at Oracle Enterprise Manager's default `SMTP_IN` process level and vice-versa.

You can also create additional SMTP instances and modify the settings for an individual instance. However, changes made in Oracle Enterprise Manager at the SMTP instance level are not visible in the Policy page and will overwrite default SMTP settings at process startup. In addition, some anti-spamming settings are only accessible through the Policy page. Refer to the table of anti-spam settings in Additional Information for the differences between the two interfaces.

Follow these guidelines when administering anti-spam policies:

- Set your anti-spam parameters from the Policy page or from the default `SMTP_IN` process in Oracle Enterprise Manager.
- Clear any spam settings at the `SMTP_IN` instance level to make sure they do not overwrite the `SMTP_IN` default settings. Start up a new `SMTP_IN` instance so it inherits the default settings.
- Tune your settings at the instance level, if required. Refresh the process to activate any changes.

Turning Anti-Spam Checking On or Off

The Active parameter of the anti-spam process (located in the Webmail Administration tab of the Policy page) enables you to turn all spam checking on or off. This parameter is a list with the following options.

- If YES is chosen, then the anti-spam check takes place and all the other associated parameters are read and used
- If NO is chosen, then the anti-spam check does not occur.

In some organizations, there might be two sets of configured MTAs.

- One set is exposed to the Internet for routing external traffic into the company's intranet.
- The second set is exposed only inside the corporate network. These internal MTAs must be protected from outside access by a firewall.

Employees should be able to communicate with the internal MTAs. Hence an organization might want to turn off the anti-spam check on the internal MTAs to relieve the unnecessary burden on already protected resources.

Precedence of Spam Checking

In general, configure your `SMTP_IN` process by creating and maintaining lists of trusted and untrusted (Oracle uses the term, Reject) addresses.

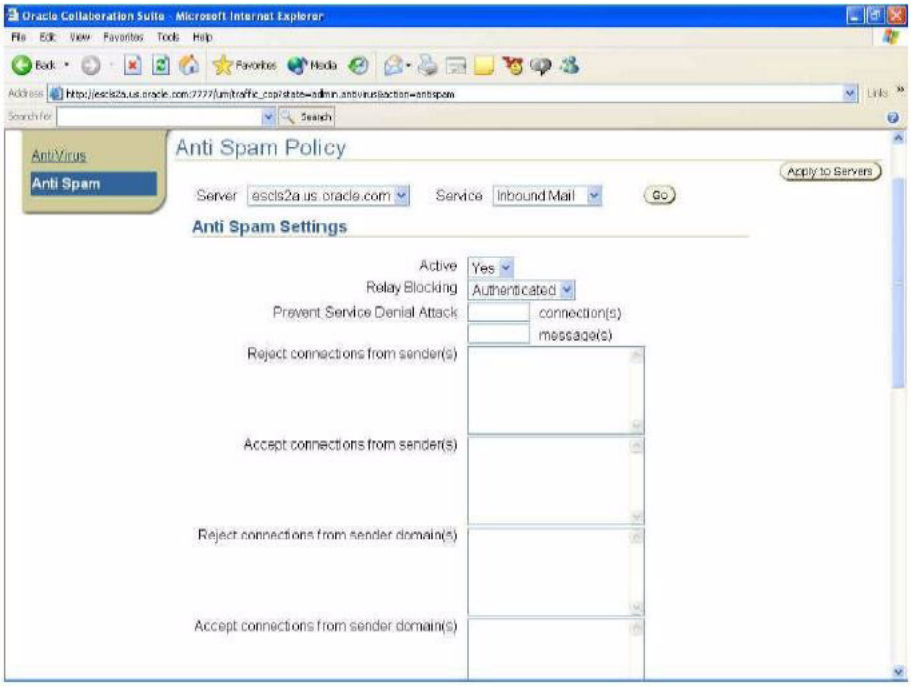
When configuring your Oracle Email installation to allow or deny message delivery, remember that the Oracle MTA takes the more restrictive path. This means if you tell the MTA processes that you both trust (Allow) and reject (Deny) a particular address, the process will not accept the address. The Oracle MTA follows the logic that Deny takes precedence over Allow actions.

For example, if `*.university.edu` is listed in both Reject and Accept connections from the Host domains parameters, all hosts that are resolved to be within the domain of `university.edu` will be rejected.

Trusted Domains are network domains you trust regardless of any further spam checks. These domains can relay even when the Relay Allowed parameter is set to false.

Trusted IPs are subnets or hosts you trust regardless of any further spam checks. These hosts can relay even when the Relay Allowed parameter is set to false.

Figure 3 Native Anti-Spamming parameters in the Anti-Spam page



Mail Flow through the Oracle MTA Process

Several actions take place when an e-mail message is routed. On each action, information is collected and decisions are made for directing the e-mail message in the appropriate way. This section describes each action and how the native anti-spam features fit into the Oracle Email process.

Action 1: A Connection is Made

DNS Hostname Lookup

- First, a client or another MTA contacts the MTA to route an e-mail message through it. The connecting IP address is first resolved from the connect request.
- The MTA calls the domain name server on the network to get the host name associated with that IP address. This process is called *DNS lookup*.
- From the host name, you can resolve the domain of the host.

At the end of this action, the `SMTP_IN` (MTA) process definitely has an IP address, but it may or may not have a domain name (if the DNS lookup fails, you have no domain).

The IP DNS host name lookup always takes place, regardless of whether or not native anti-spam is set to `TRUE`.

The Handshake

The first command that the `SMTP_IN` (MTA) process expects from a connecting MTA is a simple handshake. The computer requesting the connection sends either a `HELO` or `EHLO` command.

Along with the command, the computer must send what it thinks is its host name. This is the second source of the domain name of a host. However, the Oracle MTA ignores this value if the DNS lookup of the IP address is successful. The validated host name is always used.

The Envelope

The information that is used to decide how to route an e-mail message is the envelope information passed during the SMTP session. Think of the envelope as a virtual header consisting of three parts of address information used by Oracle Email:

- Computer Address: The connection requests information. The SMTP process captures the IP address of the computer requesting the connection. The process can also collect the domain name from the `HELO` or `EHLO` command.
- `FROM`: The e-mail address of the sender of the e-mail message.
- `RCPT TO:` The e-mail addresses of the intended recipients of the e-mail message.

Processing on computer addresses occurs prior to, and takes precedence over, both originator and recipient address checking. If the computer address is not a trusted address, then the MTA rejects the message.

Action 2: Authentication

The Authentication parameter is not really an anti-spam parameter. However, it does work with anti-spam parameters and needs to be mentioned.

The host receiving an e-mail message cannot verify the sender. Hence, it is highly desirable to verify the sender before sending the e-mail to the destination host. This is the stage where the Authentication parameter comes in. An organization can choose to configure Oracle Email to deliver e-mails through an authenticated SMTP server.

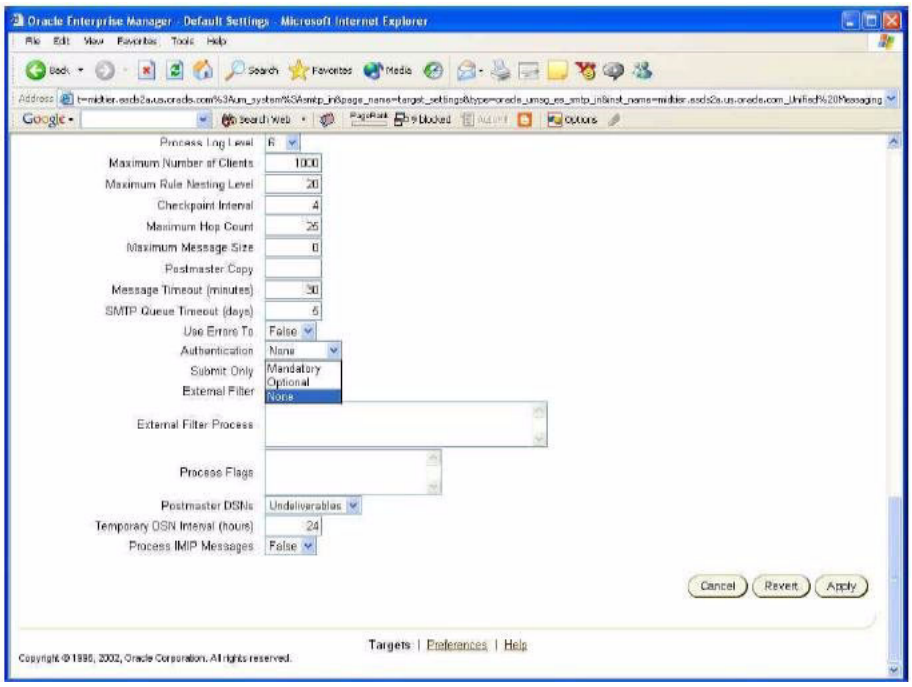
Note: The Authentication parameter is not set in the Policy page. It is set in the Administration pages of the inbound `SMTP_IN` process through Oracle Enterprise Manager.

The Authentication parameter in Oracle Email can be set to any of the following settings from a list:

- MANDATORY: The entity attempting to route an e-mail message must successfully authenticate, or the connection is terminated and the e-mail message is not routed.
- OPTIONAL: The entity attempting to route an e-mail message could authenticate if it chooses to. If the entity successfully authenticates, the e-mail message is tagged as authenticated. Otherwise, the e-mail message is still sent on, but not authenticated.
- NONE: The `SMTP_IN` process does not support or publish that it supports SMTP authentication.

There are three different settings available because MTAs route e-mail messages for clients and for other MTAs. An e-mail installation can dictate to its users that all senders must authenticate or they cannot send an e-mail. Most e-mail clients, including all the popular ones, understand and support SMTP authentication.

Figure 4 SMPTP_IN process configuration and the Authentication parameter from Oracle Enterprise Manager



As discussed earlier, internal MTAs accept external MTA connections for delivery of e-mail to users. A site has no authority over these connections. All external customers need not necessarily authenticate to send e-mail to the users of a site.

With Oracle Email, you can configure some MTAs for the internal users' clients and mandate authentication. These MTAs can route an e-mail message anywhere. Other MTAs are configured for the outside world to send e-mails to your internal users. These MTAs can optionally support authentication but only route e-mails to local users and not any external users.

Action 3: Native Anti-Spam Features

The next step is a series of address checks that is performed against those domains that you trust or do not trust. By this point, the MTA collects all the computer address information and it knows if the session is successfully authenticated or not. Before an e-mail is accepted, the connecting application needs to send host, sender, and recipient information. Once this information is accepted, then actual e-mail message data is passed.

Controlling Relay

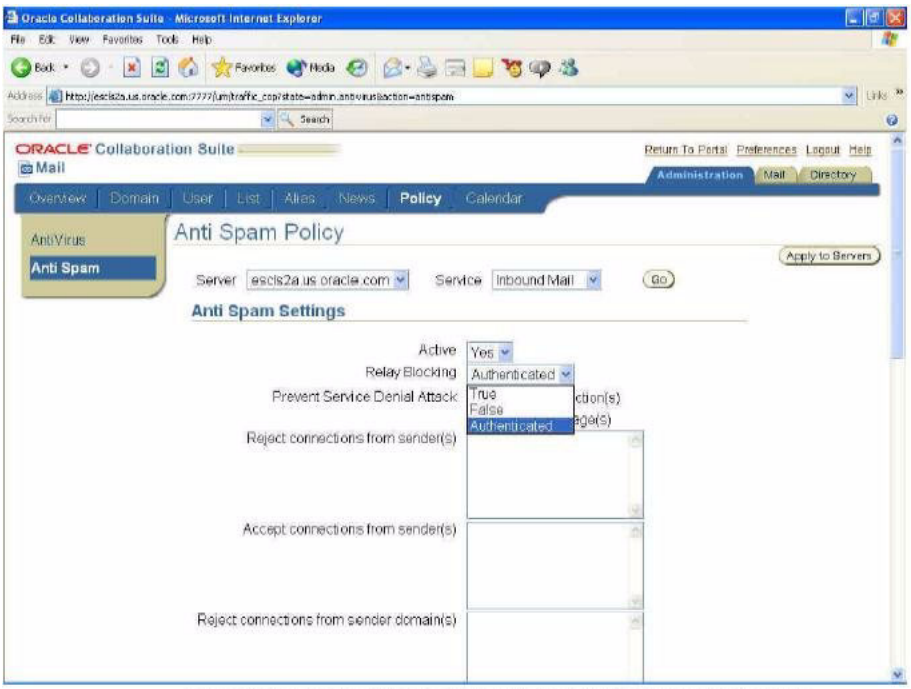
An open relay is an SMTP e-mail server that enables the processing of e-mail messages which are neither to nor from a local user. Spammers can route large volumes of e-mail messages through an open relay to disguise the source of the e-mail. It is important to control relay in order to prevent the unauthorized use of your network and computer resources.

The first anti-spam check is performed to see if the MTA enables the relay of e-mail messages. To configure the Relay Blocking parameter, go to the Policy page in the Administration tab from the Webmail client.

The Relay Blocking parameter is a list with the following options:

- TRUE: If this option is chosen, then the SMTP_IN process routes e-mail messages to outside users if both of the following conditions are true:
  - The recipient domain is listed in the Allow Relay Domains parameter
  - Other address-checking actions (reject connections from senders, and sender domains, reject connections from host domains, and reject connections from IP addresses) do not tell the process to reject the connection.
- FALSE: If this option is chosen, then the SMTP_IN process routes e-mail messages to outside users *only if* the other address-checking actions (accept connections from senders, accept connections from host domains, and accept connections from IP addresses) tell the process to trust the connection.
- AUTHENTICATED: If this option is chosen, then the SMTP_IN process routes e-mails to outside users if the other address checking actions (reject connections from senders, and sender domains, reject connections from host domains, and reject connections from IP addresses) do not tell the process to reject the connection, and the connection is successfully authenticated.

Figure 5 SMTP_IN process and the Relay Blocking parameter values.



Checking the Host Computer

The first address check is a check based on computer addresses listed in the following four parameters:

> **Note:** All of the following fields are multivalue string fields. The values are separated by a carriage return (one value in each line).

- Reject Connections from Host Domains

Domains can be prefixed with an asterisk (*) character to represent any string of one or more characters that represent a domain. The asterisk must be immediately followed by a period (.). You cannot use the asterisk to represent multiple domains that end in the same string.

- Accept Connections from Host Domains

  This uses the same syntax as the Reject connections from Host Domain field.

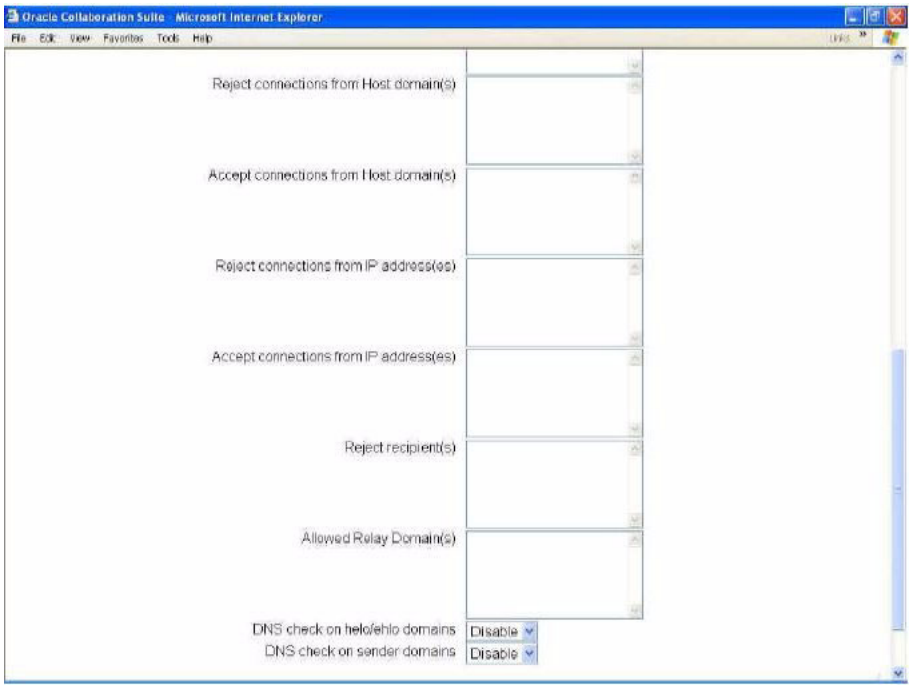
- Reject Connections from IP Addresses

  IP addresses can be suffixed with an asterisk (*) character to represent any string of one or more IP octets. The asterisk must be immediately preceded by a period (.). You cannot use the asterisk to represent multiple address that end and start with the same digit.

- Accept Connections from IP Addresses

  This uses the same syntax as the Reject connections from Host IP Address field.

Figure 6 Policy Page showing the Host and IP Accept and Reject fields.



Example of Checking IP Spam

Assume that a single MTA is used to accept incoming e-mail messages from the MTAs of other organizations as well as the e-mail clients of local users. If all the local user accounts were coming from a known bank of IP addresses, you could trust just those blocks of IP addresses for routing e-mail messages and reject all others.

For example, if there were three class B networks in the 135.55 and 144.25 ranges, you could set the Accept Connections from IP Address parameter to be:

```
135.55.200.*
135.55.230.*
```

```
144.25.*
```

These values allow any host computer on the `144.25` network, as well as any computer on the `135.55.200` and `135.55.230` subnets to route e-mail messages. The MTA refuses connection requests from all other computers.

Example of Checking Domain Spam

You can also use trusted domains to implement native anti-spam features.For example, a company can set the Accept Connections from Host Domain parameter to:

```
*.university.edu
```

This enables all hosts resolved to be somewhere within the domain of `university.edu` to route e-mail messages to recipients outside the list of local domains. All others would only be able to send e-mail messages to recipients within the local domain list.

Setting either the Reject Connections from Host Domain or Reject Connections from IP Address parameter to is the equivalent of saying *reject everyone*. Setting either Accept Connections from Host Domain or Accept Connections from IP Address parameter to * implies trust everyone except those on either of the reject lists.

Note: As discussed earlier, Reject parameters take precedence over Accept parameters.

Null values are handled as empty sets. If any of the four parameters have nothing in them, then that check is not performed. If the sender's computer address is not listed in any of the four parameters, then the Relay Blocking parameter takes affect.

Preventing Denial of Service Attacks

Preventing users of an e-mail service from using it by flooding the network with requests or spam is referred to as a DoS attack.

To prevent a DoS attack, after the Oracle MTA has checked the addressing of the host and the connection information is not rejected, there is one final check that takes place before any other anti-spam address check. The Oracle MTA keeps a count of the connections and messages sent by the connecting host within a time interval. If the total number of messages plus the number of connections exceeds the threshold level specified by the e-mail administrator in a certain time interval, then it is considered to be a network flooding scenario. In this case, the Oracle MTA rejects any further incoming messages and connections from this specific host during the remaining time interval. However, during this time, incoming messages and connections from other hosts connecting to the Oracle MTA continue to be accepted.

To configure the Prevent Service Denial Attack parameter, go to the Administration tab in the Policy page from the Webmail client as shown in Figure 7.

The parameter values for Prevent Service Denial Attack are:

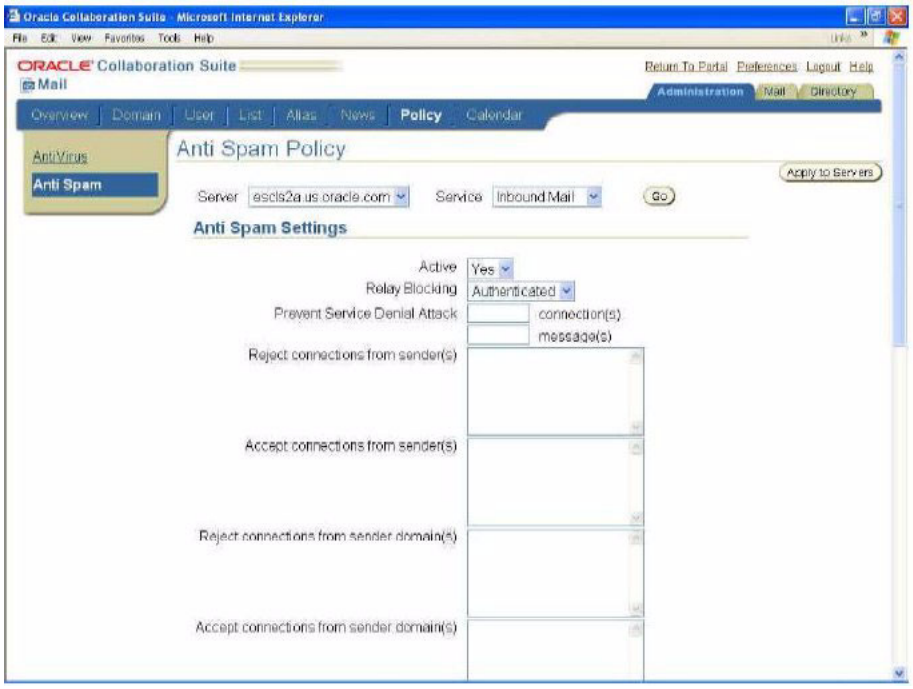
- Spam Maximum Flood Count: The number of e-mail messages plus the number of connection requests from this host within a specific time interval that is to be considered as flooding. (Default value = 10,000)
- Spam Flood Interval: The time interval, in minutes, is used in conjunction with the Spam Flood Count parameter to determine whether a host is spamming. (Default value = 10)

Note:

There is a bug (Bug #3236819) in the current release of the product according to which the parameter value labels for the Prevent Service Denial Attack parameter in the Administration tab in the Policy page are mislabeled. The parameter value currently labeled Connection refers to the Spam Flood Interval parameter. The parameter value currently labeled Message refers to Spam Maximum Flood Count.

There is a bug (Bug # 3259793) in the current release of the product whereby the default values for the Spam Maximum Flood Count and Spam Flood Interval parameters are not shown in the Administration tab in the Policy page and Enterprise Manager/Unified Messaging `SMTP_IN` process parameter page.

Figure 7 Policy Page showing Prevent Service Denial Attack parameter field.



Checking the Senders and Recipients

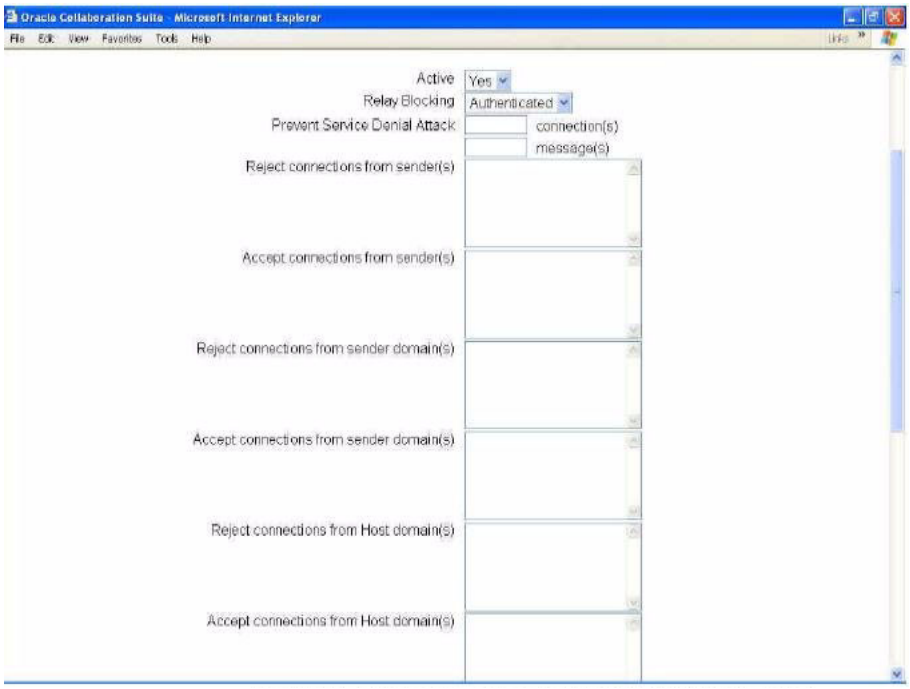
The next check is to verify the e-mail address against the Relay Blocking parameter. This checking takes place after the computer address verification.

However, this check will not take place if the computer address, domain, or IP, is trusted. Once the computer address is trusted, relaying is allowed and no further

checking takes place. An exception to this is when Relay Blocking is set to AUTHENTICATE, in which case authentication is required as well.

Sender and recipient information is obtained from the `FROM:` text and `RCPT TO:` text in an SMTP session. The concept and behavior of the Oracle MTA for the domain found in the `FROM:` address and the fully qualified e-mail address is same as that for limiting the relay of mail based on host addresses.

Figure 8 Policy Page shows Sender Accept and Reject parameters fields.



Note: All the following fields are multivalue string fields. The values are separated by a carriage return (one value in each line line).

The parameter values for checking the senders and recipients are:

- Reject Connections from Sender Domain

  Domains can be prefixed with an asterisk (*) character to represent any string of one or more characters that represent a domain. The asterisk must be immediately followed by a period (.). You cannot use the asterisk to represent multiple domains that end in the same string.

- Accept Connections from Sender Domain

  This uses the same syntax as the Reject Connections from Sender Domain field.

- Reject Connections from Sender

  A sender is a fully qualified Internet e-mail address including the user component, at (@) character, and domain component. This field does not accept any metacharacters. It is not case-sensitive.

- Accept Connections from Sender

  This uses the same syntax as the Reject Connections from Sender field.

- Reject Recipients

  This uses the same syntax as the Reject Connections from Sender field (Refer to Figure 6).

- Allowed Relay Domain

  This uses the same syntax as the Reject Connections from Sender Domain field (Refer to Figure 6).

This set of parameters enables the administrator to block e-mail messages coming from abusive domains or even specific people. An administrator can block the e-mail domain `spammers.com`, and any e-mail address originating in the `@spammers.com` domain will be rejected. This e-mail filtering can also be very granular. For example, if a site has a very specific problem with a sender at a popular domain, you can block just that one e-mail address.

An interesting innovation is the ability of the Oracle MTA to reject mail to a given set of recipients by using the Reject recipient parameter. The list in this parameter can be a list of either external or internal mail accounts.

Preventing Anti-Spam through DNS Lookup

When a computer makes a connection request to the Oracle MTA, it passes its host name. This host name is often spoofed by spammers. The Oracle MTA has the ability to verify the validity of this host name, by taking the IP address of the connection request and getting the associated host name from a name server.

There are two possible outcomes from the DNS lookup. If a host name is returned, that host name replaces whatever the host states its name is. This returned host name is used for anti-spam checking. If no host name is returned or if there is a mismatch in the domain portion of the host name, the MTA assumes that the sender is trying to spoof an IP address and the connection is refused.

Oracle Email can be configured to perform DNS checks on `HELO/EHLO` domains and sender domains. The DNS checking parameters are enabled or disabled in the Administration tab in the Policy page from the Webmail client.

- DNS check on `HELO/EHLO` domains – If this parameter is enabled, then the domain name in the `HELO/EHLO` command is checked for existence in the DNS server.
- DNS check on sender domains – If this parameter is enabled, then the domain in the sender's e-mail address is checked for existence in the DNS server.

By default, all `SMTP_IN` services of the Oracle MTA are configured to perform the DNS check. As such, hosts running the e-mail servers must have DNS lookup correctly configured.

Third-Party Anti-Spam Vendors

The Oracle MTA does not perform content-based anti-spam filtering and relies on third-party vendors to do this task. Third-party anti-spam vendors work with

Oracle much in the same way that anti-virus vendors work with our MTAs. Organizations can configure anti-spam protection by placing the third-party anti-spam protection inline in front of the Oracle MTAs.

Anti-Spam Configuration Scenarios

This section provides you with examples of various anti-spam configuration scenarios for Oracle Email. The following scenarios are provided as a reference to help you understand the concepts discussed in this document. These scenarios should not be taken as Oracle recommendations or best practices for anti-spam configuration. You should configure your Oracle Email anti-spam parameters based on your company's specific anti-spam requirements.

> **Note:** All anti-spam parameter settings referenced in the following tables are configured from the Administration tab in the Policy page from the Webmail client.

Scenario 1: Default Inbound Recipient Control

Problem:

You want the e-mail system to reject e-mail to invalid recipients.

Solution:

By default, the e-mail system rejects any mail intended for the local domain that does not have a valid recipient. The system checks that a user is valid by performing a lookup in Oracle Internet Directory.

Verification:

Verify that inbound recipient control works by connecting to the SMTP Inbound Server through telnet. An example session where `ghost` is an unknown recipient is:

```
% telnet kangaroo.wallaby.com 25
Trying 144.21.148.132...
Connected to kangaroo.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo koala.com
250 kangaroo.wallaby.com Hello host1.koala.com, pleased to meet you
% mail from:joe@abc.com250
250 2.1.0 Sender OK
% rcpt to:ghost@wallaby.com
550 5.1.1 Recipient unknown
```

Scenario 2: Inbound Recipient Control

Problem:

You want the e-mail system to reject e-mail to certain recipients even though they are valid users.

Solution:

Configure the Reject Recipient parameter to reject messages to designated recipients.

Anti-Spam Parameter Settings	What Happens
Active = YES Reject Recipient = `emu@wallaby.com` All other anti-spam parameters do not have to be changed, or can be left empty. (or null)	The `SMTP_IN` process checks the `Rcpt to` header of all incoming e-mail messages against the Reject Recipient anti-spam parameter. All e-mail messages to recipient `emu@wallaby.com` are rejected. E-mail messages to any other user of `wallaby.com` are allowed.

Verification

Verify that inbound recipient control works by connecting to the SMTP Inbound Server through telnet. An example session where `ghost` is an unknown recipient is:

```
% telnet kangaroo.wallaby.com 25
Trying 144.21.148.132...
Connected to kangaroo.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo koala.com
250 kangaroo.wallaby.com Hello host1.koala.com, pleased to meet you
% mail from:joe@abc.com250
250 2.1.0 Sender OK
% rcpt to:emu@wallaby.com
550 Spam check failed for recipient's address: emu@wallaby.com
```

Scenario 3: Controlling Relay and Trusted Domains

Problem:

You do not allow your e-mail server to be used as an open relay, except by messages coming from `wallaby.com`. This host is a trusted domain and can relay messages without any additional checking.

Solution:

Configure the Accept Connections from Host Domain parameter to accept e-mail messages from `*.wallaby.com`.

Anti-Spam Parameter Settings	What Happens
Active = YES Relay Blocking = FALSE Reject Connections from Host Domain = Accept Connections from Host Domain = `*.wallaby.com` All other anti-spam parameters do not have to be changed, or can be left empty (or null).	Oracle Email anti-spam features use reject lists and trusted lists to determine which host can relay messages. Reject lists take precedence over trusted lists. If your host is in the list of Accept Connections from Host Domain parameter and is not in the list of Reject Connections from Host Domain parameter, you can relay messages with no further spam checks. This is true even if Relay Blocking parameter is set to true.

Verification:

Verify that relay control works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the trusted domain is `wallaby.com` and the host `kangaroo.wallaby.com` is attempting to send mail:

```
% telnet koala.wallaby.com 25
Trying 144.21.148.132...
Connected to koala.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo wallaby.com
250 koala.wallaby.com Hello kangaroo.wallaby.com, pleased to meet you
% mail from:joe@abc.com
250 2.1.0 Sender OK
% rcpt to:lucy@xyz.com
250 2.1.5 Recipient ok
```

Scenario 4: Controlling Relay and Trusted IP Addresses

Problem:

You do not allow your e-mail server to be used to relay messages, except by messages coming from the trusted IP address `140.84.68.*`.

Solution:

Configure the Accept Connections from IP Address parameter to accept e-mail messages from `140.84.68.*`.

Anti-Spam Parameter Settings	What Happens
Active = YES Relay Blocking = FALSE Reject Connections from IP Address = Accept Connections from IP Address = `140.84.68.*` All other anti-spam parameters do not have to be changed, or can be left empty (or null).	Oracle Email anti-spam features use reject lists and trusted lists to determine which host can relay messages. Reject lists take precedence over trusted lists. So if your host is not the list of Reject Connections from IP Address parameter and is in the list of trusted IP addresses, you can relay messages with no further spam checks. This is true even if Relay Blocking parameter is false.

Verification:

Verify that the relay control works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the trusted subnet is `140.84.68.*` and the host `kangaroo.wallaby.com` with the IP address `140.84.68.153` is attempting to send mail:

```
% telnet koala.wallaby.com 25
Trying 144.21.148.132...
Connected to koala.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo wallaby.com
250 koala.wallaby.com Hello kangaroo.wallaby.com, pleased to meet you
% mail from:joe@abc.com
250 2.1.0 Sender OK
% rcpt to:lucy@xyz.com
250 2.1.5 Recipient ok
```

Scenario 5: Allowing Relay to Trusted Domains

Problem:

You allow your e-mail server to be used to relay messages to the `trusted.com` domain, but not to any other domains.

Solution:

Configure the Allow Relay Domains parameter to relay messages to `*.trusted.com`.

Anti-Spam Parameter Settings	What Happens
Active = YES Accept Connections from Host Domain = `*.wallaby.com` Relay Blocking = TRUE Allow Relay Domains =`*.trusted.com` All other anti-spam parameters do not have to be changed, or can be left empty (or null).	If you attempt to send e-mail from a host which is not listed in the Accept Connection from Host Domain parameter, you will only be allowed to relay to domains listed in Allow Relay Domains parameter. Hence if you are sending e-mail from `host1.wombat.com`, you will be able to send e-mail to `joe@trusted.com` but not to `joey@mail.com`.

Verification:

Verify that relay control works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the trusted domain is `wallaby.com` and the host `host1.wombat.com` is attempting to send mail:

```
% telnet koala.wallaby.com 25
Trying 144.21.148.132...
Connected to koala.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo wallaby.com
250 koala.wallaby.com Hello host1.wombat.com, pleased to meet you
% mail from:joe@test.com
250 2.1.0 Sender OK
% rcpt to: joe@yahoo.com
550 Spam check failed for recipient's address: joe@yahoo.com
rcpt to:joe@trusted.com
250 2.1.5 Recipient ok
```

Scenario 6: Accepting Connections From a Single Host

Problem:

You want your e-mail system to accept connections from a particular host or subnet only and to reject every one else.

Solution:

Set the Accept Connections from IP Address parameter to the trusted host and the Reject Connections from Sender Domain parameter to relay messages to `*`.

Anti-Spam Parameter Settings	What Happens
Active = YES Accept Connections from IP Address = `140.84.68.153` Reject Connection from Sender Domains = `*` All other anti-spam parameters do not have to be changed, or can be left empty (or null).	If you attempt to send e-mail from the trusted IP address listed in the Accept Connection from IP Address parameter, the e-mail system accepts the e-mail, even if the sender is `joe@abc`. This is because trusted IPs do not require any further spam checks. Messages from any other IP address are rejected. You can also set the value to a subnet, such as `140.84.68.*`.

Verification:

Verify that the configuration works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the trusted IP address is `140.84.68.153` with host name `host1.wombat.com` is attempting to send mail:

```
% telnet koala.wallaby.com 25
Trying 144.21.148.132...
Connected to koala.wallaby.com.
Escape character is '^]'.
220 kangaroo.wallaby.com ESMTP Oracle Email Server SMTP Inbound Server
9.0.4.0.0
Ready
% helo wallaby.com
250 koala.wallaby.com Hello host1.wombat.com, pleased to meet you
% mail from:joe@abc.com
250 2.1.0 Sender OK
% rcpt to: lucy@xyz.com
250 2.1.5 Recipient OK
```

Scenario 7: Rejecting Certain Hosts Within a Trusted Network

Problem:

You want your e-mail system to accept connections from a particular network but to reject certain hosts within that network.

Solution:

Set the Accept Connections from IP Address parameter to the network IP address and set the Reject Connection from IP Address parameter to the IP address of the host that you do not trust.

Anti-Spam Parameter Settings	What Happens
Active = YES Relay Blocking = FALSE Accept Connections from IP Address = `140.84.68.*` Reject Connection from IP Address = `140.84.68.153` All other anti-spam parameters do not have to be changed, or can be left empty (or null).	If you attempt to send e-mail from the rejected IP address listed in the Reject Connection from IP Address parameter, the e-mail system rejects the e-mail, even though it comes from a trusted network. Other hosts on the trusted network can relay messages to any destination.

Verification:

Verify that the configuration works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the rejected IP address `140.84.68.153` with host name `host1.wombat.com` is attempting to send mail:

```
% telnet koala.wallaby.com 25
Trying 144.21.148.132...
Connected to koala.wallaby.com.
Escape character is '^]'.
550 5.7.1 Spam check failed for your IP address
Connection closed by foreign host.
```

Scenario 8: Combining Relay Controls

Problem:

You want to block a host on your own network and block any spam coming from outside your network.

Solution:

Configure the Reject Connections from IP Addresses parameter to block the host on your network and the Reject Connections from Sender Domain parameter to reject any relay messages coming from outside.

Anti-Spam Parameter Settings	What Happens
Active = YES Relay Blocking = FALSE Accept Connections from IP Address = `140.84.68.*` Reject Connection from IP Address = `140.84.68.153` Reject Connection from Sender Domain = * All other anti-spam parameters do not have to be changed, or can be left empty (or null).	If you attempt to send e-mail from the rejected IP address listed in the Reject Connections from IP Address parameter, the e-mail system rejects the e-mail, even though it comes from a trusted network. Other hosts on the trusted network can relay messages to any destination. Messages coming from any other domain are rejected.

Scenario 9: Controlling Relay with Authentication

Problem:

You only want to allow valid IMAP or POP3 users belonging to a local domain to relay messages to the Internet. This is a common scenario for ISP providers.

Solution:

You can force the SMTP Inbound server to allow relay only for those users who authenticate with a valid username and password, by configuring the SMTP Inbound server with authentication.

Set the SMTP Inbound server's Authentication parameter to optional from Oracle Enterprise Manager, in addition to setting the anti-spam parameters listed in the table.

Anti-Spam Parameter Settings	What Happens
Active = YES Relay Blocking = AUTH All other anti-spam parameters do not have to be changed, or can be left empty (or null).	The precedence rules for reject lists and trusted lists apply, so a trusted domain or IP address can still relay without authentication. A rejected domain cannot relay messages.

Scenario 10: Blocking Denial of Service Attacks

Problem:

You want to prevent loss of network connectivity, system crashes, or failure of a service due to abusers' sending large amounts of e-mail to your site.

Solution:

To help prevent DoS attacks, you can configure the Spam Flood Interval and Spam Maximum Flood Count parameters. They work together to stop flooding if

the number of messages and connections from a single host exceeds the value of Spam Maximum Flood Count within the Spam Flood Interval.

Anti-Spam Parameter Settings	What Happens
Active = YES Spam Maximum Flood Count = 200 Spam Flood Interval = 2	If the number of messages plus the number of connection requests from this host exceeds 200 within 2 minutes, the SMTP Inbound server considers the host to be flooding the network and rejects any messages and connections during the remaining time interval.
All other anti-spam parameters do not have to be changed, or can be left empty (or null).	The precedence rules for reject lists and trusted lists apply, so a trusted domain or IP address can still relay without any spam checks.

Verification:

Verify that the configuration works by connecting to the SMTP Inbound Server through telnet. The following example shows a session where the IP address `140.84.68.153` with host name `host1.wombat.com` is attempting to flood the SMTP Inbound Server at `kangaroo.wallaby.com` with messages:

```
% telnet kangaroo.wallaby.com 25
Trying 144.21.148.132...
Connected to kangaroo.wallaby.com.
Escape character is '^]'.
550 5.7.1 Spam check failed for your IP address
Connection closed by foreign host.
```

Additional Information

The following table summarizes the Anti-Spam parameters in the Policy page and their equivalent in the Enterprise Manager/Unified Messaging `SMTP_IN` process.

Anti-Spam Parameters in Policy Page	Anti-Spam Parameters in Enterprise Manager /Unified Messaging SMTP_IN process	Spam Check On
Active	Native anti-spam	
Relay Blocking	Relay Allowed	
Prevent Service Denial Attack	Spam Flood Interval (minutes)	
Connection Message	Spam Maximum Flood Count	
Reject Connections from Sender	Reject Senders	E-mail
Accept Connections from Sender	**Parameter not viewable**	E-mail
Reject Connections from Sender Domain	Reject Domains	E-mail
Accept Connections from Sender Domain	**Parameter not viewable**	E-mail
Reject Connections from Host Domain	**Parameter not viewable**	Host

Anti-Spam Parameters in Policy Page	Anti-Spam Parameters in Enterprise Manager /Unified Messaging SMTP_ IN process	Spam Check On
Accept Connections from Host Domain	Trusted Domains	Host
Parameter not viewable	Local Domains	Host
Reject Connections from IP Address	Reject IPs	IP
Accept Connections from IP Address	Trusted IPs	IP
Reject Recipient	Reject Recipients	Email
Allowed Relay Domain	Relay Domains Allowed	Email
DNS Check on `HELO/EHLO` domains	**Parameter not viewable**	
DNS Check on Sender Domains	**Parameter not viewable**	

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 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/`