Logging

This chapter contains the following sections:

Overview

Understanding Log Files and Log Subscriptions

Logs are a compact, efficient method of gathering critical information about the email operations of AsyncOS. These logs record information regarding activity on your appliance . The information will vary depending upon the log you view, for example, Bounce logs or Delivery logs.

Most logs are recorded in plain text (ASCII) format; however, delivery logs are formatted in binary for resource efficiency. The ASCII text information is readable in any text editor.

Cisco offers the M-Series Content Security Management appliance for centralized reporting and tracking tool for logs from multiple appliances . See your Cisco representative for more information.

A log subscription associates a log type with a name, logging level, and other constraints such as size and destination information; multiple subscriptions for the same log type are permitted.

Log Types

The log type indicates what information will be recorded within the generated log such as message data, system statistics, binary or textual data. You select the log type when creating a log subscription. See Log Subscriptions, on page 59 for more information.

AsyncOS generates the following log types:

Table 1: Log Types

Log	Description
Text Mail Logs	Text mail logs record information regarding the operations of the email system. For example, message receiving, message delivery attempts, open and closed connections, bounces, TLS connections, and others.
qmail Format Mail Logs	qmail format delivery logs record the same information regarding the operations of the email system as delivery logs following, but stored in qmail format.
Delivery Logs	Delivery logs record critical information about the email delivery operations of the appliance — for example, information regarding each recipient delivery and bounce at the time of the delivery attempt. The log messages are "stateless," meaning that all associated information is recorded in each log message and users need not reference previous log messages for information about the current delivery attempt. Delivery logs are recorded in a binary format for resource efficiency. Delivery Log files must be post-processed using a provided utility to convert them to XML or CSV (comma-separated values) format. The conversion tools are located at: https://supportforums.cisco.com/document/33721/cisco-ironport-systems-contributed-tools
Bounce Logs	Bounce logs record information about bounced recipients. The information recorded for each bounced recipient includes: the message ID, the recipient ID, the Envelope From address, the Envelope To address, the reason for the recipient bounce, and the response code from the recipient host. In addition, you can choose to log a fixed amount of each bounced recipient message. This amount is defined in bytes and the default is zero.
Status Logs	This log file records system statistics found in the CLI status commands, including status detail and dnsstatus . The period of recording is set using the setup subcommand in logconfig . Each counter or rate reported in status logs is the value since the last time the counter was reset.
Domain Debug Logs	Domain debug logs record the client and server communication during an SMTP conversation between the appliance and a specified recipient host. This log type can be used to debug issues with specific recipient hosts. You must specify the total number of SMTP sessions to record in the log file. As sessions are recorded, this number decreases. You can stop domain debug before all sessions have been recorded by deleting or editing the log subscription.
Injection Debug Logs	Injection debug logs record the SMTP conversation between the appliance and a specified host connecting to the system. Injection debug logs are useful for troubleshooting communication problems between the Email Security appliance and a host on the Internet.
System Logs	System logs record the following: boot information, virtual appliance license expiration alerts, DNS status information, and comments users typed using commit command. System logs are useful for troubleshooting the basic state of the appliance .
CLI Audit Logs	The CLI audit logs record all CLI activity on the system.

Log	Description
FTP Server Logs	FTP logs record information about the FTP services enabled on the interface. Connection details and user activity are recorded.
GUI Logs	See HTTP Logs.
HTTP Logs	HTTP logs record information about the HTTP and/or secure HTTP services enabled on the interface. Because the graphical user interface (GUI) is accessed via HTTP, the HTTP logs are ostensibly the GUI equivalent of the CLI Audit logs. Session data (new session, session expired) and pages accessed in the GUI are recorded. These logs also include information about SMTP transactions, for example information about scheduled reports emailed from the appliance .
NTP Logs	NTP logs record the conversation between the appliance and any NTP (Network Time Protocol) servers configured. For more information, see “Editing the Network Time Protocol (NTP) Configuration (Time Keeping Method)” in the “System Administration” chapter.
LDAP Debug Logs	LDAP debug logs are meant for debugging LDAP installations. (See the “LDAP Queries” chapter.) Useful information about the queries that the appliance is sending to the LDAP server are recorded here.
Anti-Spam Logs	Anti-spam logs record the status of the anti-spam scanning feature of your system, including the status on receiving updates of the latest anti-spam rules. Also, any logs related to the Context Adaptive Scanning Engine are logged here.
Anti-Spam Archive	If you enabled an Anti-Spam scanning feature, messages that are scanned and associated with the “archive message” action are archived here. The format is an mbox-format log file. For more information about anti-spam engines, see the “Anti-Spam” chapter.
Graymail Engine Logs	Contains information about the graymail engine, status, configuration, and so on. Most information is at Info or Debug level.
Graymail Archive	Contains archived messages (the messages that are scanned and associated with the “archive message” action). The format is an mbox-format log file.
Anti-Virus Logs	AntiVirus logs record the status of the anti-virus scanning feature of your system, including the status on receiving updates of the latest anti-virus identity files.
Anti-Virus Archive	If you enabled an anti-virus engine, messages that are scanned and associated with the “archive message” action are archived here. The format is an mbox-format log file. For more information, see the “Anti-Virus” chapter.
AMP Engine Logs	The AMP Engine logs record the status of the Advanced Malware Protection features of the system. For more information, see File Reputation Filtering and File Analysis

Log	Description
AMP Archive	If you have configured mail policies to archive messages that Advanced Malware Protection engine has found to have attachments that are unscannable or contain malware, those messages are archived here. The format is an mbox-format log file.
Scanning Logs	The scanning log contains all LOG and COMMON messages for scanning engines (see Alerts). This is typically application faults, alert sent, alert failed, and log error messages. This log does not apply to system-wide alerts.
Spam Quarantine Logs	Spam Quarantine logs record actions associated with the Spam Quarantine processes.
Spam Quarantine GUI Logs	Spam Quarantine logs record actions associated with the Spam Quarantine including configuration via the GUI, end user authentication, and end user actions (releasing email, etc.).
SMTP Conversation Logs	The SMTP conversation log records all parts of incoming and outgoing SMTP conversations.
Safe/Block Lists Logs	Safelist/blocklist logs record data about the safelist/blocklist settings and database.
Reporting Logs	Reporting logs record actions associated with the processes of the centralized reporting service.
Reporting Query Logs	Reporting query logs record actions associated with the reporting queries that are run on the appliance .
Updater Logs	The updater log records events related to updates for system services, such as McAfee Anti-Virus definition updates.
Tracking Logs	Tracking logs record actions associated with the processes of the tracking service. Tracking logs are a subset of the mail logs.
Authentication Logs	The authentication log records successful user logins and unsuccessful login attempts.
Configuration History Logs	Configuration history logs record the following information: What changes were made on the appliance , and when were the changes made? A new configuration history log is created each time a user commits a change.
Upgrade Logs	Status information about upgrade download and installation.
API Logs	API logs record various events related to the AsyncOS API for the appliance , for example: • API has started or stopped • Connection to the API failed or closed (after providing response) • Authentication succeeded or failed • Request contains errors • Error while communicating network configuration changes with AsyncOS API

Log	Description
Consolidated Event Logs	The Consolidated Event Logs summarizes each message event in a single log line. Using this log type you can reduce the number of bytes of data (log information) sent to a Security Information and Event Management (SIEM) vendor or application for analysis. The logs are in the Common Event Format (CEF) log message format that is widely used by most SIEM vendors.
CSN Logs	The CSN logs contain details about the CSN data uploads. The CSN data (appliance and feature usage details can be seen at the trace level.
Advanced Phishing Protection Logs	The Advanced Phishing Protection logs contain information related to Cisco Advanced Phishing Protection Cloud Service. Most information is at the Info or Critical level.

Log Type Characteristics

The following table summarizes the different characteristics of each log type.

Table 2: Log Type Comparison

						Contains								
	Transactional	Stateless	Recorded as text	Recorded as mbox file	Recorded as binary	Periodic Status Information	Message Receiving Information	Delivery Information	Individual Hard Bounces	Individual Soft Bounces	Injection SMTP Conversation	Header Logging	Delivery SMTP Conversation	Configuration Information
Mail Logs	•		•			•	•	•	•	•		•		
qmail Format Delivery Logs		•			•		•	•	•			•		
Delivery Log		•			•		•	•	•			•		
Bounce Logs	•		•						•	•		•		
Status Logs		•	•			•								
Domain Debug Logs	•		•					•	•	•			•	

						Contains								
	Transactional	Stateless	Recorded as text	Recorded as mbox file	Recorded as binary	Periodic Status Information	Message Receiving Information	Delivery Information	Individual Hard Bounces	Individual Soft Bounces	Injection SMTP Conversation	Header Logging	Delivery SMTP Conversation	Configuration Information
Injection Debug Logs	•		•				•				•			
System Logs	•		•			•								
CLI Audit Logs	•		•			•								
FTP Server Logs	•		•			•								
HTTP Logs	•		•			•								
NTP Logs	•		•			•								
LDAP Logs	•		•											
Anti-spam Logs	•		•			•								
Anti-Spam Archive				•										
Graymail Engine Logs	•		•			•								
Graymail Archive				•										
Anti-virus Logs	•		•			•								
Anti-Virus Archive				•										

						Contains								
	Transactional	Stateless	Recorded as text	Recorded as mbox file	Recorded as binary	Periodic Status Information	Message Receiving Information	Delivery Information	Individual Hard Bounces	Individual Soft Bounces	Injection SMTP Conversation	Header Logging	Delivery SMTP Conversation	Configuration Information
AMP Engine Logs	•		•			•								
AMP Archive				•										
Scanning Logs	•		•			•								•
Spam Quarantine	•		•			•								
Spam Quarantine GUI	•		•			•								
Safe/Block Lists Logs	•		•			•								
Reporting Logs	•		•		•									
Reporting Query Logs	•		•		•									
Updater Logs			•											
Tracking Logs	•				•	•	•	•	•	•		•		
Authentication Logs	•		•											
Configuration History Logs	•		•											•
API Logs	•		•											

						Contains								
	Transactional	Stateless	Recorded as text	Recorded as mbox file	Recorded as binary	Periodic Status Information	Message Receiving Information	Delivery Information	Individual Hard Bounces	Individual Soft Bounces	Injection SMTP Conversation	Header Logging	Delivery SMTP Conversation	Configuration Information
Consolidated Event Logs	•		•				•	•						
CSN Logs	•		•			•								•
Advanced Phishing Protection Logs	•		•											

Log Retrieval Methods

Log files can be retrieved based upon one of the following file transfer protocols. You set the protocol while creating or editing the log subscription in the GUI or via the logconfig command during the log subscription process.

Note When using a Log Push method on a particular log, that log will be locally unavailable for troubleshooting or searching via the CLI.

Table 3: Log Transfer Protocols

Manually Download	This method lets you access log files at any time by clicking a link to the log directory on the Log Subscriptions page, then clicking the log file to access. Depending on your browser, you can view the file in a browser window, or open or save it as a text file. This method uses the HTTP(S) protocol and is the default retrieval method. **Note** Using this method, you cannot retrieve logs for any computer in a cluster, regardless of level (machine, group, or cluster), even if you specify this method in the CLI.
FTP Push	This method periodically pushes log files to an FTP server on a remote computer. The subscription requires a username, passphrase, and destination directory on the remote computer. Log files are transferred based on a rollover schedule set by you.

SCP Push	This method periodically pushes log files to an SCP server on a remote computer. This method requires an SSH SCP server on a remote computer using the SSH1 or SSH2 protocol. The subscription requires a username, SSH key, and destination directory on the remote computer. Log files are transferred based on a rollover schedule set by you.
Syslog Push	This method sends log messages to a remote syslog server. This method conforms to RFC 3164. You must submit a hostname for the syslog server and choose to use either UDP or TCP for log transmission. The port used is 514. A facility can be selected for the log; however, a default for the log type is pre-selected in the dropdown menu. Only text-based logs can be transferred using syslog push.
[Only for Consolidated Event Logs] AWS S3 Push	This method periodically pushes log files to the Amazon Simple Storage Service (S3) Bucket available on the Amazon Web Services (AWS) public cloud. The subscription requires an S3 bucket name, access key, and a secret key to access the Amazon S3 bucket. You can set a rollover schedule to transfer the log files. **Note** Make sure that you have a valid AWS S3 bucket to use this retrieval method. For more information, refer to the AWS user documentation at https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_access-keys.html.

Log Filenames and Directory Structure

AsyncOS creates a directory for each log subscription based on the log subscription name. The actual name of the log file in the directory is composed of the log filename specified by you, the timestamp when the log file was started, and a single-character status code. The filename of logs are made using the following formula:

```
/LogSubscriptionName/LogFilename.@timestamp.statuscode
```

Status codes may be .current or .s (signifying saved). You should only transfer or delete log files with the saved status.

Log Rollover and Transfer Schedule

Log files are created by log subscriptions, and are rolled over (and transferred, if a push-based retrieval option is selected) based on the first user-specified condition reached: maximum file size or scheduled rollover. Use the logconfig command in the CLI or the Log Subscriptions page in the GUI to configure both the maximum file size and time interval for scheduled rollovers. You can also use the **Rollover Now** button in the GUI or the rollovernow command in the CLI to rollover selected log subscriptions. See Rolling Over Log Subscriptions, on page 63 for more information on scheduling rollovers.

Logs retrieved using manual download are saved until they reach the maximum number you specify (the default is 10 files) or until the system needs more space for log files.

Logs Enabled by Default

Your appliance is pre-configured with many log subscriptions enabled by default (other logs may be configured depending on which license keys you have applied). By default, the retrieval method is “Manually Download.”

All pre-configured log subscriptions have a Log Level of 3, except for error_logs which is set at 1 so that it will contain only errors. See Log Levels, on page 60 for more information. For information about creating new log subscriptions, or modifying existing ones, see Log Subscriptions, on page 59.

Log Types

Timestamps in Log Files

The following log files include the begin and end date of the log itself, the version of AsyncOS, and the GMT offset (provided in seconds, and only at the beginning of the log):

- Anti-Virus log
- LDAP log
- System log
- Mail log

Using Text Mail Logs

They contain details of email receiving, email delivery and bounces. These logs are a useful source of information to understand delivery of specific messages and to analyze system performance.

These logs do not require any special configuration. However, you must configure the system properly to view attachment names, and attachment names may not always be logged. For information, see Enabling Message Tracking and Message Tracking Overview.

Information displayed in text mail logs is shown in the following table:

Table 4: Text Mail Log Statistics

Statistic	Description
ICID	Injection Connection ID. This is a numerical identifier for an individual SMTP connection to the system, over which 1 to thousands of individual messages may be sent.
DCID	Delivery Connection ID. This is a numerical identifier for an individual SMTP connection to another server, for delivery of 1 to thousands of messages, each with some or all of their RIDs being delivered in a single message transmission.
RCID	RPC Connection ID. This is a numerical identifier for an individual RPC connection to the Spam quarantine. It is used to track messages as they are sent to and from the Spam Quarantine.
MID	Message ID: Use this to track messages as they flow through the logs.
RID	Recipient ID: Each message recipient is assigned an ID.
New	New connection initiated.
Start	New message started.

Interpreting a Text Mail Log

Use the following sample as a guide to interpret log files.

Note Individual lines in log files are NOT numbered. They are numbered here only for sample purposes.

Table 5: Text Mail Log Detail

1	`Mon Apr 17 19:56:22 2003 Info:`
	`New SMTP ICID 5 interface Management (10.1.1.1)`
	`address 10.1.1.209 reverse dns host remotehost.com verified yes`

2	`Mon Apr 17 19:57:20 2003 Info:` `Start MID 6 ICID 5`
3	`Mon Apr 17 19:57:20 2003 Info:` `MID 6 ICID 5 From: <sender@remotehost.com>`

4	`Mon Apr 17 19:58:06 2003 Info:` `MID 6 ICID 5 RID 0 To: <mary@yourdomain.com>`
5	`Mon Apr 17 19:59:52 2003 Info:` `MID 6 ready 100 bytes from <sender@remotehost.com>`
6	`Mon Apr 17 19:59:59 2003 Info: ICID 5 close`
7	`Mon Mar 31 20:10:58 2003 Info:` `New SMTP DCID 8 interface 192.168.42.42 address 10.5.3.25`
8	`Mon Mar 31 20:10:58 2003 Info:` `Delivery start DCID 8 MID 6 to RID [0]`
9	`Mon Mar 31 20:10:58 2003 Info:` `Message done DCID 8 MID 6 to RID [0]`
10	`Mon Mar 31 20:11:03 2003 Info: DCID 8 close`

Use the following table as a guide to reading the preceding log file.

Table 6: Detail of Text Mail Log Example

Line Number	Description
1	A new connection is initiated into the system and assigned an Injection ID (ICID) of "5." The connection was received on the Management IP interface and was initiated from the remote host at 10.1.1.209.
2	The message was assigned a Message ID (MID) of "6" after the MAIL FROM command is issued from the client.
3	The sender address is identified and accepted.
4	The recipient is identified and assigned a Recipient ID (RID) of "0."
5	MID 5 is accepted, written to disk, and acknowledged.
6	Receiving is successful and the receiving connection closes.
7	Next the message delivery process starts. It is assigned a Delivery Connection ID (DCID) of "8" from 192.168.42.42 and to 10.5.3.25.
8	The message delivery starts to RID "0."
9	Delivery is successful for MID 6 to RID "0."
10	The delivery connection closes.

Examples of Text Mail Log Entries

Following are some sample log entries based on various situations.

Message Injection and Delivery

A message is injected into the appliance for a single recipient. The message is successfully delivered.

```
Wed Jun 16 21:42:34 2004 Info: New SMTP ICID 282204970 interface mail.example.com
(1.2.3.4) address 2.3.4.5 reverse dns host unknown verified no

Wed Jun 16 21:42:34 2004 Info: ICID 282204970 SBRS None

Wed Jun 16 21:42:35 2004 Info: Start MID 200257070 ICID 282204970

Wed Jun 16 21:42:35 2004 Info: MID 200257070 ICID 282204970 From: <someone@foo.com>

Wed Jun 16 21:42:36 2004 Info: MID 200257070 ICID 282204970 RID 0 To: <user@example.com>

Wed Jun 16 21:42:38 2004 Info: MID 200257070 Message-ID
'<37gva9$5uvbhe@mail.example.com>'

Wed Jun 16 21:42:38 2004 Info: MID 200257070 Subject 'Hello'

Wed Jun 16 21:42:38 2004 Info: MID 200257070 ready 24663 bytes from <someone@foo.com>

Wed Jun 16 21:42:38 2004 Info: MID 200257070 antivirus negative

Wed Jun 16 21:42:38 2004 Info: MID 200257070 queued for delivery

Wed Jun 16 21:42:38 2004 Info: New SMTP DCID 2386069 interface 1.2.3.4 address 1.2.3.4

Wed Jun 16 21:42:38 2004 Info: Delivery start DCID 2386069 MID 200257070 to RID [0]

Wed Jun 16 21:42:38 2004 Info: ICID 282204970 close

Wed Jun 16 21:42:38 2004 Info: Message done DCID 2386069 MID 200257070 to RID [0]
[('X-SBRS', 'None')]

Wed Jun 16 21:42:38 2004 Info: MID 200257070 RID [0] Response 2.6.0
<37gva9$5uvbhe@mail.example.com> Queued mail for delivery

Wed Jun 16 21:42:43 2004 Info: DCID 2386069 close
```

Successful Message Delivery

```
Mon Mar 31 20:10:58 2003 Info: New SMTP DCID 5 interface 172.19.0.11 address
63.251.108.110

Mon Mar 31 20:10:58 2003 Info: Delivery start DCID 5 MID 4 to RID [0]

Mon Mar 31 20:10:58 2003 Info: Message done DCID 5 MID 4 to RID [0]

Mon Mar 31 20:11:03 2003 Info: DCID 5 close
```

Unsuccessful Message Delivery (Hard Bounce)

A message with two recipients is injected into the appliance . Upon delivery, the destination host returns a 5 XX error, which indicates that the message cannot be delivered to either recipient. The appliance notifies the sender and removes the recipients from the queue.

```
Mon Mar 31 20:00:23 2003 Info: New SMTP DCID 3 interface 172.19.0.11 address
64.81.204.225

Mon Mar 31 20:00:23 2003 Info: Delivery start DCID 3 MID 4 to RID [0, 1]
Mon Mar 31 20:00:27 2003 Info: Bounced: DCID 3 MID 4 to RID 0 - 5.1.0 - Unknown address
error ('550', ['<george@yourdomain.com>... Relaying denied']) []
Mon Mar 31 20:00:27 2003 Info: Bounced: DCID 3 MID 4 to RID 1 - 5.1.0 - Unknown address
error ('550', ['<jane@yourdomain.com>... Relaying denied']) []
Mon Mar 31 20:00:32 2003 Info: DCID 3 close
```

Soft Bounce Followed by Successful Delivery

A message is injected into the appliance . On the first delivery attempt, the message soft bounces and is queued for future delivery. On the second attempt, the message is successfully delivered.

```
Mon Mar 31 20:10:58 2003 Info: New SMTP DCID 5 interface 172.19.0.11 address
63.251.108.110

Mon Mar 31 20:00:23 2003 Info: Delivery start DCID 3 MID 4 to RID [0, 1]

Mon Mar 31 20:00:23 2003 Info: Delayed: DCID 5 MID 4 to RID 0 - 4.1.0 - Unknown address
error ('466', ['Mailbox temporarily full.'])[]

Mon Mar 31 20:00:23 2003 Info: Message 4 to RID [0] pending till Mon Mar 31 20:01:23
2003

Mon Mar 31 20:01:28 2003 Info: DCID 5 close

Mon Mar 31 20:01:28 2003 Info: New SMTP DCID 16 interface PublicNet address 172.17.0.113

Mon Mar 31 20:01:28 2003 Info: Delivery start DCID 16 MID 4 to RID [0]

Mon Mar 31 20:01:28 2003 Info: Message done DCID 16 MID 4 to RID [0]

Mon Mar 31 20:01:33 2003 Info: DCID 16 close
```

Message Scanning Results for the scanconfig Command

You can use the scanconfig command to determine the system behavior when a message can not be deconstructed into its component parts (when removing attachments). The Options are Deliver , Bounce , or Drop .

The following example shows the Text Mail log with `scanconfig` set to Deliver .

```
Tue Aug 3 16:36:29 2004 Info: MID 256 ICID 44784 From: <test@virus.org>

Tue Aug 3 16:36:29 2004 Info: MID 256 ICID 44784 RID 0 To: <joe@example.com>

Tue Aug 3 16:36:29 2004 Info: MID 256 Message-ID '<137398.@virus.org>'

Tue Aug 3 16:36:29 2004 Info: MID 256 Subject 'Virus Scanner Test #22'

Tue Aug 3 16:36:29 2004 Info: MID 256 ready 1627 bytes from <test@virus.org>

Tue Aug 3 16:36:29 2004 Warning: MID 256, Message Scanning Problem: Continuation line
seen before first header


Tue Aug 3 16:36:29 2004 Info: ICID 44784 close

Tue Aug 3 16:36:29 2004 Info: MID 256 antivirus positive 'EICAR-AV-Test'
```

```
Tue Aug 3 16:36:29 2004 Info: Message aborted MID 256 Dropped by antivirus

Tue Aug 3 16:36:29 2004 Info: Message finished MID 256 done
```

The following example shows the Text Mail log with `scanconfig` set to drop .

```
Tue Aug 3 16:38:53 2004 Info: Start MID 257 ICID 44785

Tue Aug 3 16:38:53 2004 Info: MID 257 ICID 44785 From: test@virus.org

Tue Aug 3 16:38:53 2004 Info: MID 257 ICID 44785 RID 0 To: <joe@example.com>
Tue Aug 3 16:38:53 2004 Info: MID 257 Message-ID '<392912.@virus.org>'
Tue Aug 3 16:38:53 2004 Info: MID 25781 Subject 'Virus Scanner Test #22'
Tue Aug 3 16:38:53 2004 Info: MID 257 ready 1627 bytes from <test@virus.org>
Tue Aug 3 16:38:53 2004 Warning: MID 257, Message Scanning Problem: Continuation line
seen before first header
Tue Aug 3 16:38:53 2004 Info: Message aborted MID 25781 Dropped by filter 'drop_zip_c'
Tue Aug 3 16:38:53 2004 Info: Message finished MID 257 done
Tue Aug 3 16:38:53 2004 Info: ICID 44785 close
```

Message with Attachment

In this example, a content filter with condition “Message Body Contains” has been configured to enable identification of attachment names:

```
Sat Apr 23 05:05:42 2011 Info: New SMTP ICID 28 interface Management (192.0.2.10)
address 224.0.0.10 reverse dns host test.com verified yes

Sat Apr 23 05:05:42 2011 Info: ICID 28 ACCEPT SG UNKNOWNLIST match sbrs[-1.0:10.0]
SBRS 0.0

Sat Apr 23 05:05:42 2011 Info: Start MID 44 ICID 28
Sat Apr 23 05:05:42 2011 Info: MID 44 ICID 28 From: <sender1@example.com>
Sat Apr 23 05:05:42 2011 Info: MID 44 ICID 28 RID 0 To: <recipient1@example.org>
Sat Apr 23 05:05:42 2011 Info: MID 44 Message-ID '<000001cba32e$f24ff2e0$d6efd8a0$@com>'
Sat Apr 23 05:05:42 2011 Info: MID 44 Subject 'Message 001'
Sat Apr 23 05:05:42 2011 Info: MID 44 ready 240129 bytes from <sender1@example.com>
Sat Apr 23 05:05:42 2011 Info: MID 44 matched all recipients for per-recipient
policy DEFAULT in the inbound table
Sat Apr 23 05:05:42 2011 Info: ICID 28 close
Sat Apr 23 05:05:42 2011 Info: MID 44 interim verdict using engine: CASE
spam negative
Sat Apr 23 05:05:42 2011 Info: MID 44 using engine: CASE spam negative
Sat Apr 23 05:05:43 2011 Info: MID 44 attachment 'Banner.gif'
Sat Apr 23 05:05:43 2011 Info: MID 44 attachment '=D1=82=D0=B5=D1=81=D1=82.rst'
Sat Apr 23 05:05:43 2011 Info: MID 44 attachment 'Test=20Attachment.docx'
Sat Apr 23 05:05:43 2011 Info: MID 44 queued for delivery
```

Note that the second of the three attachments is Unicode. On terminals that cannot display Unicode, these attachments are represented in quoted-printable format.

Successful Message Delivery with DANE Support

A message reaches the appliance for a single recipient. The appliance requests for a secure DNS MX record, DNS A record, and a TLSA record from the DNS server. If you select DANE as "Mandatory", the TLSA record is validated against the X.509 certificate value of the recipient's domain. If the TLSA record validation is successful, the message is delivery to the recipient.

```
Tue Nov 13 12:13:33 2018 Debug: Trying DANE MANDATORY for example.org
Tue Nov 13 12:13:33 2018 Debug: SECURE MX record(mail.example.org) found for example.org
```

```
Tue Nov 13 12:13:33 2018 Debug: DNS query: Q('mail.example.org', 'CNAME')
Tue Nov 13 12:13:33 2018 Debug: DNS query: QN('mail.example.org', 'CNAME',
'recursive_nameserver0.parent')
Tue Nov 13 12:13:33 2018 Debug: DNS query: QIP  ('mail.example.org','CNAME','8.8.8.8',60)
Tue Nov 13 12:13:33 2018 Debug: DNS query: Q  ('mail.example.org', 'CNAME', '8.8.8.8')
Tue Nov 13 12:13:34 2018 Debug: DNSSEC Response data([], , 0, 1799)
Tue Nov 13 12:13:34 2018 Debug: Received NODATA for domain mail.example.org type CNAME
Tue Nov 13 12:13:34 2018 Debug: No CNAME record(NoError) found for domain(mail.example.org)
Tue Nov 13 12:13:34 2018 Debug: SECURE A record (4.31.198.44) found for
MX(mail.example.org) in example.org
Tue Nov 13 12:13:34 2018 Info: New SMTP DCID 92 interface 10.10.1.191 address 4.31.198.44
port 25
Tue Nov 13 12:13:34 2018 Info: ICID 13 lost
Tue Nov 13 12:13:34 2018 Info: ICID 13 close
Tue Nov 13 12:13:34 2018 Debug: DNS query: Q('_25._tcp.mail.example.org', 'TLSA')
Tue Nov 13 12:13:34 2018 Debug: DNS query: QN('_25._tcp.mail.example.org', 'TLSA',
'recursive_nameserver0.parent')
Tue Nov 13 12:13:34 2018 Debug: DNS query: QIP
('_25._tcp.mail.example.org','TLSA','8.8.8.8',60)
Tue Nov 13 12:13:34 2018 Debug: DNS query: Q  ('_25._tcp.mail.example.org', 'TLSA', '8.8.8.8')
Tue Nov 13 12:13:35 2018 Debug: DNSSEC Response data(['0301010c72ac70b745ac19998811b13
1d662c9ac69dbdbe7cb23e5b514b56664c5d3d6'], secure, 0, 1799)
Tue Nov 13 12:13:35 2018 Debug: DNS encache (_25._tcp.mail.example.org, TLSA,
[(2550119024205761L, 0,
'SECURE', '0301010c72ac70b745ac19998811b131d662c9ac69dbdbe7cb23e5b514b56664c5d3d6')])
Tue Nov 13 12:13:35 2018 Debug: SECURE TLSA Record found for MX(mail.example.org) in
example.org
Tue Nov 13 12:13:36 2018 Info: DCID 92 Certificate verification successful
Tue Nov 13 12:13:36 2018 Info: DCID 92 TLS success protocol TLSv1.2 cipher
Tue Nov 13 12:13:36 2018 Info: DCID 92 TLS success protocol TLSv1.2 cipher
ECDHE-RSA-AES256-GCM-SHA384 for example.org
Tue Nov 13 12:13:36 2018 Info: Delivery start DCID 92 MID 23 to RID [0]
```

Message Delivery Failed due to Certificate Verification Failure

A message reaches the appliance for a single recipient. The appliance requests for a secure DNS MX record, DNS A record, and a TLSA record from the DNS server. If you select DANE as "Mandatory", the TLSA record is validated against the X.509 certificate value of the recipient's domain. If the certificate verification fails, the message is delivered at a later time. If secure TLSA record is not found, the message is bounced.

```
Wed Nov 14 05:52:08 2018 Debug: DNS query: QN('server1.example.net', 'CNAME',
'recursive_nameserver0.parent')
Wed Nov 14 05:52:08 2018 Debug: DNS query: QIP
('server1.example.net','CNAME','10.10.2.184',60)
Wed Nov 14 05:52:08 2018 Debug: DNS query: Q  ('server1.example.net', 'CNAME', '10.10.2.184')
Wed Nov 14 05:52:08 2018 Debug: DNSSEC Response data([], , 0, 284)
Wed Nov 14 05:52:08 2018 Debug: Received NODATA for domain server1.example.net type CNAME
Wed Nov 14 05:52:08 2018 Debug: No CNAME record(NoError) found for domain(server1.example.net)
Wed Nov 14 05:52:08 2018 Debug: Secure CNAME(server1.example.net) found for
MX(someone.cs2.example.net)
in example.net
Wed Nov 14 05:52:08 2018 Debug: SECURE A record (10.10.1.198) found for
MX(someone.cs2.example.net)
in example.net
Wed Nov 14 05:52:08 2018 Info: New SMTP DCID 102 interface 10.10.1.191 address 10.10.1.198
 port 25
Wed Nov 14 05:52:08 2018 Debug: Fetching TLSA records with CNAME(server1.example.net) for
MX(someone.cs2.example.net) in example.net
Wed Nov 14 05:52:08 2018 Debug: DNS query: Q('_25._tcp.server1.example.net', 'TLSA')
Wed Nov 14 05:52:08 2018 Debug: SECURE TLSA Record found for MX(server1.example.net) in
example.net
Wed Nov 14 05:52:08 2018 Debug: DCID 102 All TLSA records failed for certificate not trusted
Wed Nov 14 05:52:08 2018 Debug: Fetching TLSA records with initial
name(someone.cs2.example.net)
```

```
in example.net
Wed Nov 14 05:52:08 2018 Debug: DNS query: Q('_25._tcp.someone.cs2.example.net', 'TLSA')
Wed Nov 14 05:52:08 2018 Debug: SECURE TLSA Record found for MX(someone.cs2.example.net)
in example.net
Wed Nov 14 05:52:08 2018 Info: DCID 102 Certificate verification successful
Wed Nov 14 05:52:08 2018 Info: DCID 102 TLS success protocol TLSv1.2 cipher
DHE-RSA-AES128-SHA256
for example.net
Wed Nov 14 05:52:08 2018 Info: Delivery start DCID 102 MID 26 to RID [0]
Wed Nov 14 05:52:08 2018 Info: Message done DCID 102 MID 26 to RID [0]
Wed Nov 14 05:52:08 2018 Info: MID 26 RID [0] Response 'ok:  Message 31009 accepted'
Wed Nov 14 05:52:08 2018 Info: Message finished MID 26 done

Wed Nov 14 06:36:22 2018 Debug: Trying DANE MANDATORY for example.net
Wed Nov 14 06:36:22 2018 Debug: SECURE MX record(someone.cs2.example.net) found for
example.net
Wed Nov 14 06:36:22 2018 Debug: DNS query: Q('someone.cs2.example.net', 'CNAME')
Wed Nov 14 06:36:22 2018 Debug: DNS query: QN('someone.cs2.example.net', 'CNAME',
'recursive_nameserver0.parent')
Wed Nov 14 06:36:22 2018 Debug: DNS query: QIP
('someone.cs2.example.net','CNAME','10.10.2.184',60)
Wed Nov 14 06:36:22 2018 Debug: DNS query: Q  ('someone.cs2.example.net', 'CNAME',
'10.10.2.184')
Wed Nov 14 06:36:22 2018 Debug: DNSSEC Response data(['mail.example2.net.'], secure, 0,
3525)
Wed Nov 14 06:36:22 2018 Debug: DNS encache (someone.cs2.example.net, CNAME,
[(2692348132363369L, 0,
'SECURE', 'mail.example2.net')])
Wed Nov 14 06:36:22 2018 Debug: DNS query: Q('mail.example2.net', 'CNAME')
Wed Nov 14 06:36:22 2018 Debug: DNS query: QN('mail.example2.net', 'CNAME',
'recursive_nameserver0.parent')
Wed Nov 14 06:36:22 2018 Debug: DNS query: QIP  ('mail.example2.net','CNAME','10.10.2.184',60)
Wed Nov 14 06:36:22 2018 Debug: DNS query: Q  ('mail.example2.net', 'CNAME', '10.10.2.184')
Wed Nov 14 06:36:22 2018 Debug: DNSSEC Response data([], , 0, 225)
Wed Nov 14 06:36:22 2018 Debug: Received NODATA for domain mail.example2.net type CNAME
Wed Nov 14 06:36:22 2018 Debug: No CNAME record(NoError) found for domain(mail.example2.net)
Wed Nov 14 06:36:22 2018 Debug: Secure CNAME(mail.example2.net) found for
MX(someone.cs2.example.net)
in example.net
Wed Nov 14 06:36:22 2018 Debug: INSECURE A record (10.10.1.197) found for
MX(someone.cs2.example.net)
in example.net
Wed Nov 14 06:36:22 2018 Debug: Fetching TLSA records with initial
name(someone.cs2.example.net) in example.net
Wed Nov 14 06:36:22 2018 Info: New SMTP DCID 104 interface 10.10.1.191 address 10.10.1.197
 port 25
Wed Nov 14 06:36:36 2018 Debug: DNS query: Q('_25._tcp.someone.cs2.example.net', 'TLSA')
Wed Nov 14 06:36:36 2018 Debug: SECURE TLSA Record found for MX(someone.cs2.example.net)
in example.net
Wed Nov 14 06:36:36 2018 Debug: DCID 104 All TLSA records failed for certificate not trusted
Wed Nov 14 06:36:36 2018 Info: MID 27 DCID 104 DANE failed for the domain example.net:
DANE Certificate verification failed
Wed Nov 14 06:36:36 2018 Info: Failed for all MX hosts in example.net
```

Message Delivery Failed due to Invalid TLSA Record

A message reaches the appliance for a single recipient. The appliance requests for a secure DNS MX record, DNS A record, and a TLSA record from the DNS server. If you select DANE as "Mandatory", the TLSA record is validated against the X.509 certificate value of the recipient's domain. If an invalid TLSA record is found, the message delivery is attempted at a later time or the message is bounced.

```
Tue Aug 7 05:15:18 2018 Debug: Trying DANE MANDATORY for example-dane.net
Tue Aug 7 05:15:18 2018 Debug: SECURE MX record (someone.example-dane.net) found for
test-tlsabogus.net
```

```
Tue Aug 7 05:15:18 2018 Debug: DNS query: Q ('someone.example-dane.net', 'CNAME')
Tue Aug 7 05:15:18 2018 Debug: DNS query: QN ('someone.example-dane.net', 'CNAME',
'recursive_nameserver0.parent')
Tue Aug 7 05:15:18 2018 Debug: DNS query: QIP
('someone.example-dane.net','CNAME','10.10.2.183', 60)
Tue Aug 7 05:15:18 2018 Debug: DNS query: Q ('someone.example-dane.net', 'CNAME',
'10.10.2.183')
Tue Aug 7 05:15:18 2018 Debug: DNSSEC Response data ([], , 0, 300)
Tue Aug 7 05:15:18 2018 Debug: SECURE A record (10.10.1.198) found for MX
(someone.example-dane.net)
in example-dane.net
Tue Aug 7 05:15:18 2018 Info: ICID 32 close
Tue Aug 7 05:15:18 2018 Info: New SMTP DCID 61 interface 10.10.1.194 address 10.10.1.198
port 25
Tue Aug 7 05:15:18 2018 Debug: DNS query: Q ('_25._tcp.someone.example-dane.net', 'TLSA')
Tue Aug 7 05:15:18 2018 Debug: DNS query: QN ('_25._tcp.someone.example-dane.net', 'TLSA',

'recursive_nameserver0.parent')
Tue Aug 7 05:15:18 2018 Debug: DNS query: QIP
('_25._tcp.someone.example-dane.net','TLSA','10.10.2.183', 60)
Tue Aug 7 05:15:18 2018 Debug: DNS query: Q ('_25._tcp.someone.example-dane.net', 'TLSA',
'10.10.2.183')
Tue Aug 7 05:15:18 2018 Debug: DNSSEC Response data
(['03010160b3f16867357cdfef37bb6acd687af54f
225e3bfa945e1d37bfd37bd4eb6020'], bogus, 0, 60)
Tue Aug 7 05:15:18 2018 Debug: DNS encache (_25._tcp.someone.example-dane.net, TLSA,
[(11065394975822091L,
 0, 'BOGUS', '03010160b3f16867357cdfef37bb6acd687af54f225e3bfa945e1d37bfd37bd4eb6020')])
Tue Aug 7 05:15:18 2018 Debug: BOGUS TLSA Record is found for MX (someone.example-dane.net)

in example-dane.net
Tue Aug 7 05:15:18 2018 Debug: Trying next MX record in example-dane.net
Tue Aug 7 05:15:18 2018 Info: MID 44 DCID 61 DANE failed: TLSA record BOGUS
Tue Aug 7 05:15:18 2018 Debug: Failed for all MX hosts in example-dane.net
```

Rolling Back to Opportunistic TLS as TLSA Record Not Found

A message reaches the appliance for a single recipient. The appliance requests for a secure DNS MX record, DNS A record, and a TLSA record from the DNS server. If you select DANE as "Opportunistic", the TLSA record is validated against the X.509 certificate value of the recipient's domain. If TLSA record is not found for the recipient's domain, opportunistic TLS is used for encrypting SMTP conversations.

```
Wed Sep 12 06:51:32 2018 Debug: Trying DANE OPPORTUNISTIC for example-dane.com
Wed Sep 12 06:51:32 2018 Debug: SECURE MX record (mx.example-dane.com) found for
digitalhellion.com
Wed Sep 12 06:51:32 2018 Debug: DNS query: Q ('mx.example-dane.com', 'CNAME')
Wed Sep 12 06:51:32 2018 Debug: DNS query: QN ('mx.example-dane.com', 'CNAME',
'recursive_nameserver0.parent')
Wed Sep 12 06:51:32 2018 Debug: DNS query: QIP ('mx.example-dane.com', 'CNAME','8.8.8.8',60)
Wed Sep 12 06:51:32 2018 Debug: DNS query: Q ('mx.example-dane.com', 'CNAME', '8.8.8.8')
Wed Sep 12 06:51:32 2018 Debug: DNSSEC Response data ([], , 0, 1799)
Wed Sep 12 06:51:32 2018 Debug: Received NODATA for domain mx.example-dane.com type CNAME
Wed Sep 12 06:51:32 2018 Debug: No CNAME record (NoError) found for domain
(mx.example-dane.com)
Wed Sep 12 06:51:32 2018 Debug: SECURE A record (162.213.199.115) found for MX
(mx.example-dane.com)
in example-dane.com
Wed Sep 12 06:51:32 2018 Info: ICID 1 lost
Wed Sep 12 06:51:32 2018 Info: ICID 1 close
Wed Sep 12 06:51:33 2018 Info: New SMTP DCID 2 interface 10.10.1.173 address 162.213.199.115
 port 25
Wed Sep 12 06:51:33 2018 Debug: DNS query: Q ('_25._tcp.mx.example-dane.com', 'TLSA')
Wed Sep 12 06:51:33 2018 Debug: DNS query: QN ('_25._tcp.mx.example-dane.com', 'TLSA',
'recursive_nameserver0.parent')
```

```
Wed Sep 12 06:51:33 2018 Debug: DNS query: QIP
('_25._tcp.mx.example-dane.com','TLSA','8.8.8.8', 60)
Wed Sep 12 06:51:33 2018 Debug: DNS query: Q ('_25._tcp.mx.example-dane.com', 'TLSA',
'8.8.8.8')
Wed Sep 12 06:51:34 2018 Debug: DNSSEC Response data ([], , 3, 1798)
Wed Sep 12 06:51:34 2018 Debug: Received NXDomain for domain _25._tcp.mx.example-dane.com'
 type TLSA
Wed Sep 12 06:51:34 2018 Debug: No TLSA record (NXDomain) found for MX (mx.example-dane.com)
Wed Sep 12 06:51:34 2018 Debug: Falling back to conventional TLS for MX (mx.example-dane.com)

in example-dane.com
Wed Sep 12 06:51:34 2018 Info: MID 1 DCID 2 DANE failed for the domain example-dane.com:
No TLSA Record
Wed Sep 12 06:51:34 2018 Info: DCID 2 TLS success protocol TLSv1.2 cipher
ECDHE-RSA-AES256-GCM-SHA384
Wed Sep 12 06:51:35 2018 Info: Delivery start DCID 2 MID 1 to RID [0]
```

Message received based on Sender's Country of Origin

In this example, the log shows a message received based on the country of origin of a particular sender group.

```
Thu Apr  6 06:50:18 2017 Info: ICID 73 ACCEPT SG ALLOWED_LIST match country[us] SBRS -10.0
 country United States
```

Maximum URLs in Message Attachments Exceeds URL Scan Limit

In this example, the log shows the number of URLs in the message attachments that exceeded the URL scan limit

```
Wed Nov 8 13:35:48 2017 Info:  MID $mid not completely scanned for URL Filtering. Error:
$error
```

Maximum URLs in Message Body Exceeds URL Scan Limit

In this example, the log shows the number of URLs in the message body that exceeded the URL scan limit.

```
Wed Nov 8 13:37:42 2017 Info: MID 976 not completely scanned for URL Filtering.
Error: The number of URLs in the message body exceeded the URL scan limit.
```

Malicious Shortened URL redirected to Cisco Proxy Server

In this example, the log shows a shortened URl that is marked as malicious due to a URL reputation score of -3, and redirected to the Cisco Security Proxy server.

```
Tue Nov 7 10:42:41 2017 Info: MID 9 having URL: http://ow.ly/Sb6O30fJvVn has been expanded
 to http://bit.ly/2frAl1x
Tue Nov 7 10:42:42 2017 Info: MID 9 having URL: http://bit.ly/2frAl1x has been expanded to
 http://thebest01.wayisbetter.cn/?cMFN
Tue Nov 7 10:42:42 2017 Info: MID 9 URL http://thebest01.wayisbetter.cn/?cMFN has reputation
 -3.854 matched Action: URL redirected to Cisco Security proxy
Tue Nov 7 10:42:42 2017 Info: MID 9 rewritten to MID 10 by
url-reputation-proxy-redirect-action filter 'aa'
```

Unable to Expand Shortened URL in Message

In this example, the log shows that the shortened URL in the message could not be expanded to the actual URL.

```
Mon Oct 30 10:58:59 2017 Info: MID 36 having URL: http://ow.ly/P0Kw30fVst3 has been expanded
 to http://bit.ly/2ymYWPR
```

```
Mon Oct 30 10:59:00 2017 Info: MID 36 having URL: http://bit.ly/2ymYWPR has been expanded
to http://ow.ly/cTS730fVssH
Mon Oct 30 10:59:01 2017 Info: MID 36 having URL: http://ow.ly/cTS730fVssH has been expanded
 to http://bit.ly/2xK8PD9
Mon Oct 30 10:59:01 2017 Info: MID 36 having URL: http://bit.ly/2xK8PD9 has been expanded
to http://ow.ly/lWOi30fVssl
Mon Oct 30 10:59:02 2017 Info: MID 36 having URL: http://ow.ly/lWOi30fVssl has been expanded
 to http://bit.ly/2ggHv9e
Mon Oct 30 10:59:03 2017 Info: MID 36 having URL: http://bit.ly/2ggHv9e has been expanded
to http://ow.ly/4fSO30fVsqx
Mon Oct 30 10:59:04 2017 Info: MID 36 having URL: http://ow.ly/4fSO30fVsqx has been expanded
 to http://bit.ly/2hKEFcW
Mon Oct 30 10:59:05 2017 Info: MID 36 having URL: http://bit.ly/2hKEFcW has been expanded
to http://ow.ly/NyH830fVsq6
Mon Oct 30 10:59:06 2017 Info: MID 36 having URL: http://ow.ly/NyH830fVsq6 has been expanded
 to http://bit.ly/2ysnsNi
Mon Oct 30 10:59:06 2017 Info: MID 36 having URL: http://bit.ly/2ysnsNi has been expanded
to http://ow.ly/JhUN30fVsnL
Mon Oct 30 10:59:07 2017 Info: MID 36 having URL: http://ow.ly/JhUN30fVsnL has been expanded
 to http://bit.ly/2hKQmAe
Mon Oct 30 10:59:07 2017 Info: MID 36 URL http://bit.ly/2hKQmAe is marked malicious due to
 : URL depth exceeded
Mon Oct 30 11:04:48 2017 Warning: MID 40 Failed to expand URL http://mail1.example.com/abcd
 Reason: Error while trying to retrieve expanded URL
Mon Oct 30 11:04:48 2017 Info: MID 40 not completely scanned for URL Filtering. Error:
Message has a shortened URL that could not be expanded
```

Log Entry for Malicious URL in Message Attachment

In this example, the log shows a URLin the message attachment that is malicious with a reputation score of -9.5.

```
Mon Nov 6 06:50:18 2017 Info: MID 935 Attachment file_1.txt URL http://jrsjvysq.net has
reputation -9.5 matched
Condition: URL Reputation Rule
```

Message marked as Unscannable due to Extraction Failure

In this example, the log shows a message that is not scanned by the Content Scanner due to an attachment extraction failure.

```
Tue Oct 24 08:28:58 2017 Info: Start MID 811 ICID 10
Tue Oct 24 08:28:58 2017 Info: MID 811 ICID 10 From: <sender@example.com>
Tue Oct 24 08:28:58 2017 Info: MID 811 ICID 10 RID 0 To: <recipient@example.com>
Tue Oct 24 08:28:58 2017 Info: MID 811 Message-ID '<example@cisco.com>'
Tue Oct 24 08:28:58 2017 Info: MID 811 Subject 'Test mail'
Tue Oct 24 08:28:58 2017 Info: MID 811 ready 5242827 bytes from <user2@sender.com>
Tue Oct 24 08:28:58 2017 Info: MID 811 matched all recipients for per-recipient policy
DEFAULT in the inbound table
Tue Oct 24 08:28:59 2017 Info: MID 811 attachment 'gzip.tar.gz'
Tue Oct 24 08:28:59 2017 Info: MID 811 was marked as unscannable due to extraction failures.
 Reason: Error in extraction process - Decoding Errors.
Tue Oct 24 08:28:59 2017 Info: ICID 10 close
Tue Oct 24 08:28:59 2017 Info: MID 811 quarantined to "Policy" (Unscannable: due to Extraction
 Failure)
Tue Oct 24 08:28:59 2017 Info: Message finished MID 811 done
```

Message marked as Unscannable due to RFC Violation

In this example, the log shows a message that is not scanned by the Content Scanner due to an RFC violation.

```
Tue Oct 24 08:23:26 2017 Info: Start MID 807 ICID 6
Tue Oct 24 08:23:26 2017 Info: MID 807 ICID 6 From: <sender@example.com>
Tue Oct 24 08:23:26 2017 Info: MID 807 ICID 6 RID 0 To: <recipient@example.com>
Tue Oct 24 08:23:26 2017 Info: MID 807 Subject 'Test Mail'
Tue Oct 24 08:23:26 2017 Info: MID 807 ready 427 bytes from <sender@example.com>
Tue Oct 24 08:23:26 2017 Info: MID 807 matched all recipients for per-recipient policy
DEFAULT in the inbound table
Tue Oct 24 08:23:26 2017 Info: MID 807 was marked as unscannable due to an RFC violation.
Reason: A Unix-From header was found in the middle of a header block.
Tue Oct 24 08:23:26 2017 Info: MID 807 queued for delivery
Tue Oct 24 08:23:26 2017 Info: ICID 6 close
```

Log Entries for Generated or Re-Written Messages

Some functions, such as rewrite/redirect actions (alt-rcpt-to filters, anti-spam rcpt rewrite, bcc() actions, anti-virus redirections, etc.), create new messages. When looking through the logs, you might need to check the results and add in further MIDs and possibly DCIDs. Entries such as these are possible:

```
Tue Jun  1 20:02:16 2004 Info: MID 14 generated based on MID 13 by bcc filter 'nonetest'
```

or:

```
Tue Jan 6 15:03:18 2004 Info: MID 2 rewritten to 3 by antispam

Fri May 14 20:44:43 2004 Info: MID 6 rewritten to 7 by alt-rcpt-to-filter filter
'testfilt'
```

An interesting point to note about ‘rewritten’ entries is that they can appear after lines in the log indicating use of the new MID.

Messages Sent to the Spam Quarantine

When you send a message to the quarantine, the mail logs track the movement to and from the quarantine using the RCID (RPC connection ID) to identify the RPC connection. In the following mail log, a message is tagged as spam, and sent to the Spam Quarantine:

```
Wed Feb 14 12:11:40 2007 Info: Start MID 2317877 ICID 15726925

Wed Feb 14 12:11:40 2007 Info: MID 2317877 ICID 15726925 From: <HLD@chasehf.bfi0.com>
Wed Feb 14 12:11:40 2007 Info: MID 2317877 ICID 15726925 RID 0 To:
<stevel@healthtrust.org>
Wed Feb 14 12:11:40 2007 Info: MID 2317877 Message-ID
'<W1TH05606E5811BEA0734309D4BAF0.323.14460.pimailer44.DumpShot.2@email.chase.com>'
Wed Feb 14 12:11:40 2007 Info: MID 2317877 Subject 'Envision your dream home - Now make
it a reality'

Wed Feb 14 12:11:40 2007 Info: MID 2317877 ready 15731 bytes from <HLD@chasehf.bfi0.com>

Wed Feb 14 12:11:40 2007 Info: MID 2317877 matched all recipients for per-recipient
policy DEFAULT in the inbound table

Wed Feb 14 12:11:41 2007 Info: MID 2317877 using engine: CASE spam suspect

Wed Feb 14 12:11:41 2007 Info: EUQ: Tagging MID 2317877 for quarantine
Wed Feb 14 12:11:41 2007 Info: MID 2317877 antivirus negative
Wed Feb 14 12:11:41 2007 Info: MID 2317877 queued for delivery
Wed Feb 14 12:11:44 2007 Info: RPC Delivery start RCID 756814 MID 2317877 to local
IronPort Spam Quarantine


Wed Feb 14 12:11:45 2007 Info: EUQ: Quarantined MID 2317877
```

```
Wed Feb 14 12:11:45 2007 Info: RPC Message done RCID 756814 MID 2317877

Wed Feb 14 12:11:45 2007 Info: Message finished MID 2317877 done
```

Example of External Threat Feeds Mail Logs

The Mail Logs contain information about threats detected in incoming messages and actions taken on such messages. Most information is at the Info or Debug level.

```
Thu Jun 7 20:48:10 2018 Info: MID 91 Threat feeds source 'S1' detected malicious URL:
'http://digimobil.mobi/' in attachment(s): malurl.txt. Action: Attachment stripped
```

Examples of SDR Filtering Log Entries

The SDR filtering information is posted to the Mail Logs. Most information is at the Info or Debug level.

- Sender Domain Reputation Authentication Failure
- Sender Domain Reputation Request Timeout
- Sender Domain Reputation Invalid Host
- Sender Domain Reputation General Errors

Sender Domain Reputation Authentication Failure

In this example, the log shows a message that was not filtered based on SDR because of an authentication failure when connecting to the SDR service.

```
Mon Jul 2 08:57:18 2018 Info: New SMTP ICID 3 interface Management (192.0.2.10) address
224.0.0.10 reverse dns host unknown verified no
Mon Jul 2 08:57:18 2018 Info: ICID 3 ACCEPT SG UNKNOWNLIST match ipr[none] ipr not enabled
 country not enabled
Mon Jul 2 08:57:18 2018 Info: Start MID 3 ICID 3
Mon Jul 2 08:57:18 2018 Info: MID 3 ICID 3 From: <sender1@example.com>
Mon Jul 2 08:57:18 2018 Info: MID 3 ICID 3 RID 0 To: <recipient1@example.com>
Mon Jul 2 08:57:18 2018 Info: MID 3 Message-ID '<000001cba32e$f24ff2e0$d6efd8a0$@com>' Mon
 Jul 2 08:57:18 2018 Info: MID 3 Subject 'Message 001'
Mon Jul 2 08:57:19 2018 Info: MID 3 SDR: Message was not scanned for Sender Domain Reputation.
 Reason: Authentication failure.
```

Solution

Use the `sdradvancedconfig` command in the CLI to configure the required parameters when connecting your email gateway to the SDR service.

Sender Domain Reputation Request Timeout

In this example, the log shows a message that was not filtered based on SDR because of a request timeout error when communicating with the SDR service.

```
Mon Jul 2 09:00:13 2018 Info: New SMTP ICID 4 interface Management (192.0.2.10) address
224.0.0.10 reverse dns host unknown verified no
Mon Jul 2 09:00:13 2018 Info: ICID 4 ACCEPT SG UNKNOWNLIST match ipr[none] ipr not enabled
 country not enabled
Mon Jul 2 09:00:13 2018 Info: Start MID 4 ICID 4
Mon Jul 2 09:00:13 2018 Info: MID 4 ICID 4 From: <sender1@example.com>
Mon Jul 2 09:00:13 2018 Info: MID 4 ICID 4 RID 0 To: <recipient1@example.com >
Mon Jul 2 09:00:13 2018 Info: MID 4 Message-ID '<000001cba32e$f24ff2e0$d6efd8a0$@com>'
```

```
Mon Jul 2 09:00:13 2018 Info: MID 4 Subject 'Message 001'
Mon Jul 2 09:00:13 2018 Info: MID 4 SDR: Message was not scanned for Sender Domain Reputation.
 Reason: Request timed out.
```

Solution

When an SDR request times out, the message is marked as unscannable, and the configured actions are applied to the message.

Sender Domain Reputation Invalid Host

In this example, the log shows a message that was not filtered based on SDR because an invalid SDR service host was configured on your email gateway.

```
Mon Jul 2 09:04:08 2018 Info: ICID 7 ACCEPT SG UNKNOWNLIST match ipr[none] ipr not enabled
 country not enabled
Mon Jul 2 09:04:08 2018 Info: Start MID 7 ICID 7
Mon Jul 2 09:04:08 2018 Info: MID 7 ICID 7 From: <sender1@example.com >
Mon Jul 2 09:04:08 2018 Info: MID 7 ICID 7 RID 0 To: <recipient1@example.com >
Mon Jul 2 09:04:08 2018 Info: MID 7 Message-ID '<000001cba32e$f24ff2e0$d6efd8a0$@com>' Mon
 Jul 2 09:04:08 2018 Info: MID 7 Subject 'Message 001'
Mon Jul 2 09:04:08 2018 Info: MID 7 SDR: Message was not scanned for Sender Domain Reputation.
 Reason: Invalid host configured.
```

Solution

Use the `sdradvancedconfig` command in the CLI to configure the required parameters when connecting your email gateway to the SDR service.

Sender Domain Reputation General Errors

In this example, the log shows a message that was not filtered based on SDR because of an unknown error.

```
Mon Jul 2 09:00:13 2018 Info: New SMTP ICID 4 interface Management (192.0.2.10) address
224.0.0.10 reverse dns host unknown verified no
Mon Jul 2 09:00:13 2018 Info: ICID 4 ACCEPT SG UNKNOWNLIST match ipr[none] ipr not enabled
 country not enabled
Mon Jul 2 09:00:13 2018 Info: Start MID 4 ICID 4
Mon Jul 2 09:00:13 2018 Info: MID 4 ICID 4 From: <sender1@example.com >
Mon Jul 2 09:00:13 2018 Info: MID 4 ICID 4 RID 0 To: <recipient1@example.com >
Mon Jul 2 09:00:13 2018 Info: MID 4 Message-ID '<000001cba32e$f24ff2e0$d6efd8a0$@com>'
Mon Jul 2 09:00:13 2018 Info: MID 4 Subject 'Test mail'
Mon Jul 2 09:00:13 2018 Info: MID 4 SDR: Message was not scanned for Sender Domain Reputation.
 Reason: Unknown error.
```

Solution

When an unknown error occurs, the message is marked as unscannable, and the configured actions are applied to the message.

Cisco Advanced Phishing Protection Cloud Service Expired

In this example, the log shows that the Cisco Advanced Phishing Protection Cloud Service expired.

```
Wed May 6 11:47:45 2020 Critical: The Cisco Advanced
Phishing Protection Cloud Service has expired and is disabled. Contact
your Cisco Account Manager to renew the service and enable it.
```

Solution: You need to contact your Cisco Account Manager to renew the service and enable it.

Reminder about Cisco Advanced Phishing Protection Cloud Service Expiry Date

In this example, the log shows that the Cisco Advanced Phishing Protection Cloud Service will expire on a particular date.

```
 Fri May 8 04:50:26 2020 Info: Cisco Advanced
Phishing Protection Cloud Service expires on
2020-05-10 07:00:00. You need to contact your Cisco Account
Manager to renew the service
```

Solution: You need to contact your Cisco Account Manager to renew the service.

No API Access UID and API Access Secret Key

In this example, the log shows that the appliance was unable to poll for the Cisco Advanced Phishing Protection Cloud Service expiry date because of no API Access UID and API Access Secret key.

```
 Wed May 6 17:52:52 2020 Critical: Failed to poll
for the Cisco Advanced Phishing Protection Cloud Service
expiry date. You need to add the API Access UID and API Access
secret key.
```

Solution: You need to add the API Access UID and API Access secret key.

Invalid API Access UID or API Access Secret Key

In this example, the log shows that the appliance was unable to poll for the Cisco Advanced Phishing Protection Cloud Service expiry date because of an nvalid API Access UID and API Access Secret key.

```
Wed May 6 17:52:52 2020 Critical: Failed to poll
for the Cisco Advanced Phishing Protection Cloud Service
expiry date because the API Access Key is invalid. You need
to re-configure the API Access UID and secret key
```

Solution: You need to re-configure the API Access UID and secret key.

Using Delivery Logs

Delivery logs record critical information about the email delivery operations of AsyncOS. The log messages are “stateless,” meaning that all associated information is recorded in each log message and users need not reference previous log messages for information about the current delivery attempt.

The delivery log records all information pertaining to email delivery operations for each recipient. All information is laid out in a logical manner and is human-readable after conversion using a utility provided by Cisco. The conversion tools are located at: https://supportforums.cisco.com/document/33721/cisco-ironport-systems-contributed-tools

Delivery logs are recorded and transferred in a binary format for resource efficiency. Information recorded in delivery logs is shown in the following table:

Table 7: Delivery Log Statistics

Statistic	Description
Delivery status	Success (message was successfully delivered) or bounce (message was hard bounced)
Del_time	Delivery time

Statistic	Description
Inj_time	Injection time. del_time - inj_time = time the recipient message stayed in the queue
Bytes	Message size
Mid	Message ID
Ip	Recipient host IP. The IP address of the host that received or bounced the recipient message
From	Envelope From, also known as Envelope Sender or MAIL FROM
Source_ip	Source host IP. The IP address of the host of the incoming message
Code	SMTP response code from recipient host
Reply	SMTP response message from recipient host
Rcpt Rid	Recipient ID. Recipient ID starts with <0>, messages with multiple recipients will have multiple recipient IDs
To	Envelope To
Attempts	Number of delivery attempts

If the delivery status was bounce, this additional information appears in the delivery log:

Table 8: Delivery Log Bounce Information

Statistic	Description
Reason	RFC 1893 Enhanced Mail Status Code interpretation of the SMTP response during the delivery
Code	SMTP response code from recipient host
Error	SMTP response message from recipient host

If you have set up logheaders (see Logging Message Headers, on page 62), the header information appears after the delivery information:

Table 9: Delivery Log Header Information

Statistic	Description
Customer_data	XML tag marking the beginning of logged headers
Header Name	Name of the header
Value	Contents of the logged header

Examples of Delivery Log Entries

The examples in this section show a variety of Delivery Log entries.

Successful Message Delivery

```
Mon Mar 31 20:10:58 2003 Info: New SMTP DCID 5 interface 172.19.0.11 address
63.251.108.110

Mon Mar 31 20:10:58 2003 Info: Delivery start DCID 5 MID 4 to RID [0]

Mon Mar 31 20:10:58 2003 Info: Message done DCID 5 MID 4 to RID [0]

Mon Mar 31 20:11:03 2003 Info: DCID 5 close
```

Delivery Status Bounce

```
<bounce del_time="Sun Jan 05 08:28:33.073 2003" inj_time="Mon Jan 05 08:28:32.929 2003"

bytes="4074" mid="94157762" ip="0.0.0.0" from="campaign1@yourdomain.com"

source_ip="192.168.102.1 "reason="5.1.0 - Unknown address error" code="550"

error="["Requested action not taken: mailbox unavailable"]">


<rcpt rid="0" to="user@sampledomain.com" attempts="1" />

</bounce>
```

Delivery Log Entry with Logheaders

```
<success del_time="Tue Jan 28 15:56:13.123 2003" inj_time="Tue Jan 28 15:55:17.696 2003"
bytes="139" mid="202" ip="10.1.1.13" from="campaign1@yourdomain.com"
source_ip="192.168.102.1" code="250" reply="sent">


<rcpt rid="0" to="user@sampledomain.com" attempts="1" />

<customer_data>

<header name="xname" value="sh"/>

</customer_data>

</success>
```

Using Bounce Logs

The bounce log records all information pertaining to each bounced recipient. Information recorded in bounce logs is shown in the following table:.

Table 10: Bounce Log Statistics

Statistic	Description
Timestamp	The time of the bounce event
Log level	The level of detail in this bounce log
Bounce type	Bounced or delayed (for example, hard or soft-bounce)

Statistic	Description
MID/RID	Message ID and recipient ID
From	Envelope From
To	Envelope To
Reason	RFC 1893 Enhanced Mail Status Code interpretation of the SMTP response during the delivery
Response	SMTP response code and message from recipient host

In addition, if you have specified message size to log or setup **logheaders** (see Logging Message Headers, on page 62), the message and header information will appear after the bounce information:

Table 11: Bounce Log Header Information

Header	The header name and content in the header
Message	Content of the message logged

Examples of Bounce Log Entries

Soft-Bounced Recipient (Bounce Type = Delayed)

```
Thu Dec 26 18:37:00 2003 Info: Delayed: 44451135:0
From:<campaign1@yourdomain.com> To:<user@sampledomain.com>


Reason: "4.1.0 - Unknown address error" Response: "('451',
['<user@sampledomain.com> Automated block triggered by suspicious
activity from your IP address (10.1.1.1). Have your system administrator
send e-mail to postmaster@sampledomain.com if you believe this block is
in error'])"
```

Hard-Bounced Recipient (Bounce Type = Bounced)

```
Thu Dec 26 18:36:59 2003 Info: Bounced: 45346670:0 From:<campaign1@yourdomain.com>
To:<user2@sampledomain.com>


Reason: "5.1.0 - Unknown address error" Response: "('550', ['There is no such active
account.'])"
```

Bounce Log with Message Body and Logheaders

```
Wed Jan 29 00:06:30 2003 Info: Bounced: 203:0 From:<campaign1@yourdomain.com>
To:<user@sampledomain.com>


Reason:"5.1.2 - Bad destination host" Response: "('000', [])" Headers: ['xname:
userID2333']' Message: Message-Id:



<1u5jak$6b@yourdomain.com>\015\012xname: userID2333\015\012subject:
Greetings.\015\012\015\012Hi Tom:'
```

Note The text string \015\012 represents a line break (for example, CRLF).

Using Status Logs

Status logs record system statistics found in the CLI status commands, including status , status detail , and dnsstatus . The period of recording is set using the setup subcommand in logconfig . Each counter or rate reported in status logs is the value since the last time the counter was reset.

Reading Status Logs

The following table shows the status log labels and the matching system statistics.

Table 12: Status Log Statistics

Statistic	Description
CPULd	CPU Utilization
DskIO	Disk I/O Utilization
RAMUtil	RAM Utilization
QKUsd	Queue Kilobytes Used
QKFre	Queue Kilobytes Free
CrtMID	Message ID (MID)
CrtICID	Injection Connection ID (ICID)
CRTDCID	Delivery Connection ID (DCID)
InjBytes	Total Injected Message Size in Bytes
InjMsg	Injected Messages
InjRcp	Injected Recipients
GenBncRcp	Generated Bounce Recipients
RejRcp	Rejected Recipients
DrpMsg	Dropped Messages
SftBncEvnt	Soft Bounced Events
CmpRcp	Completed Recipients
HrdBncRcp	Hard Bounced Recipients
DnsHrdBnc	DNS Hard Bounces

Statistic	Description
5XXHrdBnc	5XX Hard Bounces
FltrHrdBnc	Filter Hard Bounces
ExpHrdBnc	Expired Hard Bounces
OtrHrdBnc	Other Hard Bounces
DlvRcp	Delivered Recipients
DelRcp	Deleted Recipients
GlbUnsbHt	Global Unsubscribe Hits
ActvRcp	Active Recipients
UnatmptRcp	Unattempted Recipients
AtmptRcp	Attempted Recipients
CrtCncIn	Current Inbound Connections
CrtCncOut	Current Outbound Connections
DnsReq	DNS Requests
NetReq	Network Requests
CchHit	Cache Hits
CchMis	Cache Misses
CchEct	Cache Exceptions
CchExp	Cache Expired
CPUTTm	Total CPU time used by the application
CPUETm	Elapsed time since the application started
MaxIO	Maximum disk I/O operations per second for the mail process
RamUsd	Allocated memory in bytes
SwIn	Memory swapped in.
SwOut	Memory swapped out.
SwPgIn	Memory paged in.
SwPgOut	Memory paged out.
MMLen	Total number of messages in the system
DstInMem	Number of destination objects in memory

Statistic	Description
ResCon	Resource conservation tarpit value. Acceptance of incoming mail is delayed by this number of seconds due to heavy system load
WorkQ	This is the number of messages currently in the work queue
QuarMsgs	Number of individual messages in policy, virus, or outbreak quarantine (messages present in multiple quarantines are counted only once)
QuarQKUsd	KBytes used by policy, virus, and outbreak quarantine messages
LogUsd	Percent of log partition used
SophLd	Percent CPU used by Sophos anti-virus scanning
McafLd	Percent CPU used by McAfee anti-virus scanning
CASELd	Percent CPU used by CASE scanning
TotalLd	Total CPU consumption
LogAvail	Amount of disk space available for log files
EuQ	Estimated number of messages in the Spam quarantine
EuqRls	Estimated number of messages in the Spam quarantine release queue
RptLD	CPU load during the Reporting process
QtnLd	CPU load during the Quarantine process
EncrQ	Messages in the Encryption Queue

Status Log Example

```
Fri Feb 28 12:11:48 2020 Info: Status: CPULd 45 DskIO 22 RAMUtil 22 QKUsd 6676975
QKFre 1711633 CrtMID 6130195 CrtICID 722770 CrtDCID 54 InjMsg 4572789 InjRcp
4575323 GenBncRcp 255536 RejRcp 20388 DrpMsg 469642 SftBncEvnt 0 CmpRcp 3650806 HrdBncRcp
255536
DnsHrdBnc 23 5XXHrdBnc 28 FltrHrdBnc 255485 ExpHrdBnc 0
OtrHrdBnc 0 DlvRcp 3394965 DelRcp 305 GlbUnsbHt 0 ActvRcp 65 UnatmptRcp 65 AtmptRcp 0
CrtCncIn 9
CrtCncOut 0 DnsReq 7756744 NetReq 7769130 CchHit 8373490 CchMis
1989637 CchEct 1625236 CchExp 1569329 CPUTTm 37 CPUETm 62 MaxIO 465600 RAMUsd 1473355956
MMLen 54782
DstInMem 11 ResCon 0 WorkQ 54710 QuarMsgs 375
QuarQKUsd 145096 LogUsd 26 SophLd 15 BMLd 0 CASELd 0 TotalLd 100 LogAvail 116G EuQ 64 EuqRls
 0 CmrkLd 0
McafLd 9 SwIn 122 SwOut 5295 SwPgIn 368 SwPg Out 63639
SwapUsage 4% RptLd 0 QtnLd 19 EncrQ 0 InjBytes 516664777890
```

Using Domain Debug Logs

Domain debug logs record the client and server communication during an SMTP conversation between the appliance and a specified recipient host. This log type is primarily used to debug issues with specific recipient hosts.

Table 13: Domain Debug Log Statistics

Statistic	Description
Timestamp	The time of the bounce event
Log level	The level of detail in this bounce log
From	Envelope From
To	Envelope To
Reason	RFC 1893 Enhanced Mail Status Code interpretation of the SMTP response during the delivery
Response	SMTP response code and message from recipient host

Domain Debug Log Example

```
Sat Dec 21 02:37:22 2003 Info: 102503993 Sent: 'MAIL FROM:<daily@dailyf-y-i.net>'

Sat Dec 21 02:37:23 2003 Info: 102503993 Rcvd: '250 OK'

Sat Dec 21 02:37:23 2003 Info: 102503993 Sent: 'RCPT TO:<LLLSMILE@aol.com>'

Sat Dec 21 02:37:23 2003 Info: 102503993 Rcvd: '250 OK'

Sat Dec 21 02:37:23 2003 Info: 102503993 Sent: 'DATA'

Sat Dec 21 02:37:24 2003 Info: 102503993 Rcvd: '354 START MAIL INPUT, END WITH "." ON A
LINE BY ITSELF'

Sat Dec 21 02:37:24 2003 Info: 102503993 Rcvd: '250 OK'
```

Using Injection Debug Logs

Injection debug logs record the SMTP conversation between the appliance and a specified host connecting to the system. Injection debug logs are useful for troubleshooting communication problems between the appliance and a client initiating a connection from the Internet. The log records all bytes transmitted between the two systems and classifies them as “Sent to” the connecting host or “Received from” the connecting host.

You must designate the host conversations to record by specifying an IP address, an IP range, hostname, or partial hostname. Any connecting IP address within an IP range will be recorded. Any host within a partial domain will be recorded. The system performs reverse DNS lookups on connecting IP addresses to convert to hostnames. IP addresses without a corresponding PTR record in DNS will not match hostnames.

You must also specify the number of sessions to record.

Each line within an Injection Debug log contains the following information in the following table.

Table 14: Injection Debug Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted

Statistic	Description
ICID	The Injection Connection ID is a unique identifier that can be tied to the same connection in other log subscriptions
Sent/Received	Lines marked with “Sent to” are the actual bytes sent to the connecting host. Lines marked with “Received from” are the actual bytes received from the connecting host
IP Address	IP address of the connecting host

Injection Debug Log Example

```
Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '220 postman.example.com
ESMTP\015\012'

Wed Apr 2 14:30:04 2003 Info: 6216 Rcvd from '172.16.0.22': 'HELO
mail.remotehost.com\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '250
postman.example.com\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Rcvd from '172.16.0.22': 'MAIL
FROM:<sender@remotehost.com>\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '250 sender
<sender@remotehost.com> ok\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Rcvd from '172.16.0.22': 'RCPT
TO:<recipient@example.com>\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '250 recipient
<recipient@example.com> ok\015\012'
Wed Apr 2 14:30:04 Info: 6216 Rcvd from '172.16.0.22': 'DATA\015\012'

Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '354 go ahead\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Rcvd from '172.16.0.22': 'To:
recipient@example.com\015\012Date: Apr 02 2003 10:09:44\015\012Subject: Test
Subject\015\012From: Sender <sender@remotehost.com>\015\012'

Wed Apr 2 14:30:04 2003 Info: 6216 Rcvd from '172.16.0.22': 'This is the content of the
message'
Wed Apr 2 14:30:04 Info: 6216 Sent to '172.16.0.22': '250 ok\015\012'

Wed Apr 2 14:30:04 Info: 6216 Rcvd from '172.16.0.22': 'QUIT\015\012'
Wed Apr 2 14:30:04 2003 Info: 6216 Sent to '172.16.0.22': '221
postman.example.com\015\012'
```

Using System Logs

Table 15: System Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The logged event

System Log Example

In this example, the System log shows some commit entries, including the name of the user issuing the commit and the comment entered.

```
Wed Sep 8 18:02:45 2004 Info: Version: 4.0.0-206 SN: XXXXXXXXXXXX-XXX

Wed Sep 8 18:02:45 2004 Info: Time offset from UTC: 0 seconds

Wed Sep 8 18:02:45 2004 Info: System is coming up

Wed Sep 8 18:02:49 2004 Info: bootstrapping DNS cache

Wed Sep 8 18:02:49 2004 Info: DNS cache bootstrapped

Wed Sep 8 18:13:30 2004 Info: PID 608: User admin commit changes: SSW:Password

Wed Sep 8 18:17:23 2004 Info: PID 608: User admin commit changes: Completed Web::SSW

Thu Sep 9 08:49:27 2004 Info: Time offset from UTC: -25200 seconds

Thu Sep 9 08:49:27 2004 Info: PID 1237: User admin commit changes: Added a second CLI
log for examples

Thu Sep 9 08:51:53 2004 Info: PID 1237: User admin commit changes: Removed example CLI
log.
```

Using CLI Audit Logs

Table 16: CLI Audit Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
PID	Process ID for the particular CLI session in which the command was entered
Message	The message consists of the CLI command entered, the CLI output (including menus, lists, etc.), and the prompt that is displayed

CLI Audit Log Example

In this example, the CLI Audit log shows that, for PID 16434, the following CLI commands were entered: who , textconfig .

```
Thu Sep 9 14:35:55 2004 Info: PID 16434: User admin entered 'who'; prompt was
'\nmail3.example.com> '


Thu Sep 9 14:37:12 2004 Info: PID 16434: User admin entered 'textconfig'; prompt was
'\nUsername Login Time Idle Time Remote Host What\n======== ========== =========
========== ====\nadmin Wed 11AM 3m 45s 10.1.3.14 tail\nadmin 02:32PM
0s 10.1.3.14 cli\nmail3.example.com> '


Thu Sep 9 14:37:18 2004 Info: PID 16434: User admin entered ''; prompt was '\nThere are
no text resources currently defined.\n\n\nChoose the operation you want to perform:\n-
```

```
NEW - Create a new text resource.\n- IMPORT - Import a text resource from a file.\n[]> '
```

Using FTP Server Logs

Table 17: FTP Server Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
ID	Connection ID. A separate ID for each FTP connection
Message	The message section of the log entry can be logfile status information, or FTP connection information (login, upload, download, logout, etc.)

FTP Server Log Example

In this example, the FTP Server log records a connection (ID:1). The IP address of the incoming connection is shown, as well as the activity (uploading and downloading files) and the logout.

```
Wed Sep 8 18:03:06 2004 Info: Begin Logfile

Wed Sep 8 18:03:06 2004 Info: Version: 4.0.0-206 SN: 00065BF3BA6D-9WFWC21

Wed Sep 8 18:03:06 2004 Info: Time offset from UTC: 0 seconds

Wed Sep 8 18:03:06 2004 Info: System is coming up

Fri Sep 10 08:07:32 2004 Info: Time offset from UTC: -25200 seconds

Fri Sep 10 08:07:32 2004 Info: ID:1 Connection from 10.1.3.14 on 172.19.0.86

Fri Sep 10 08:07:38 2004 Info: ID:1 User admin login SUCCESS

Fri Sep 10 08:08:46 2004 Info: ID:1 Upload wording.txt 20 bytes

Fri Sep 10 08:08:57 2004 Info: ID:1 Download words.txt 1191 bytes

Fri Sep 10 08:09:06 2004 Info: ID:1 User admin logout
```

Using HTTP Logs

Table 18: HTTP Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
ID	Session ID
req	IP address of machine connecting
user	Username of user connecting

Statistic	Description
Message	Information regarding the actions performed. May include GET or POST commands or system status, etc.

HTTP Log Example

In this example, the HTTP log shows the admin user's interaction with the GUI (running the System Setup Wizard, etc.).

```
Wed Sep 8 18:17:23 2004 Info: http service on 192.168.0.1:80 redirecting to https port 443

Wed Sep 8 18:17:23 2004 Info: http service listening on 192.168.0.1:80

Wed Sep 8 18:17:23 2004 Info: https service listening on 192.168.0.1:443

Wed Sep 8 11:17:24 2004 Info: Time offset from UTC: -25200 seconds

Wed Sep 8 11:17:24 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg POST
/system_administration/system_setup_wizard HTTP/1.1 303

Wed Sep 8 11:17:25 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg GET
/system_administration/ssw_done HTTP/1.1 200

Wed Sep 8 11:18:45 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg GET
/monitor/incoming_mail_overview HTTP/1.1 200

Wed Sep 8 11:18:45 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg GET
/monitor/mail_flow_graph?injector=&width=365&interval=0&type=recipientsin&height=190
HTTP/1.1 200

Wed Sep 8 11:18:46 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg GET
/monitor/classification_graph?injector=&width=325&interval=0&type=recipientsin&height=19
0 HTTP/1.1 200

Wed Sep 8 11:18:49 2004 Info: req:10.10.10.14 user:admin id:iaCkEh2h5rZknQarAecg GET
/monitor/quarantines HTTP/1.1 200
```

Using NTP Logs

Table 19: NTP Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of either a Simple Network Time Protocol (SNTP) query to the server, or an adjust: message

NTP Log Example

In this example, the NTP log shows the appliance polling the NTP host twice.

```
Thu Sep 9 07:36:39 2004 Info: sntp query host 10.1.1.23 delay 653 offset -652

Thu Sep 9 07:36:39 2004 Info: adjust: time_const: 8 offset: -652us next_poll: 4096
```

```
Thu Sep 9 08:44:59 2004 Info: sntp query host 10.1.1.23 delay 642 offset -1152

Thu Sep 9 08:44:59 2004 Info: adjust: time_const: 8 offset: -1152us next_poll: 4096
```

Using Scanning Logs

The scanning log contains all LOG and COMMON messages for the appliance's scanning engines. See the Alerts section of the “System Administration” chapter for a list of available the COMMON and LOG alert messages.

Table 20: Scanning Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of an application fault, sent alert, failed alert, or log error message for one of the scanning engines.

Scanning Log Example

In this example, the log shows the history of an appliance sending a warning alert concerning Sophos anti-virus.

```
Wed Feb 23 22:05:48 2011 Info: Internal SMTP system attempting to send a message to
alerts@example.com with subject 'Warning <Anti-Virus> mail3.example.com: sophos
antivirus - The Anti-Virus database on this system is...' (attempt #0).


Wed Feb 23 22:05:48 2011 Info: Internal SMTP system successfully sent a message to
alerts@example.com with subject 'Warning <Anti-Virus> mail3.example.com: sophos
antivirus - The Anti-Virus database on this system is...'.


Wed Feb 23 22:05:48 2011 Info: A Anti-Virus/Warning alert was sent to alerts@example.com
with subject "Warning <Anti-Virus> mail3.example.com: sophos antivirus - The Anti-Virus
database on this system is...".
```

Using Anti-Spam Logs

Table 21: Anti-Spam Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of the check for the anti-spam updates, as well as the results (whether an update of the engine or the anti-spam rules was needed, etc.)

Anti-Spam Log Example

In this example, the anti-spam log shows the anti-spam engine checking for updates to spam definitions and CASE updates:

```
Fri Apr 13 18:59:47 2007 Info: case antispam - engine (19103) : case-daemon: server
successfully spawned child process, pid 19111

Fri Apr 13 18:59:47 2007 Info: case antispam - engine (19111) : startup: Region profile:
Using profile global

Fri Apr 13 18:59:59 2007 Info: case antispam - engine (19111) : fuzzy: Fuzzy plugin v7
successfully loaded, ready to roll

Fri Apr 13 19:00:01 2007 Info: case antispam - engine (19110) : uribllocal: running URI
blocklist local

Fri Apr 13 19:00:04 2007 Info: case antispam - engine (19111) : config: Finished loading
configuration
```

Using Graymail Logs

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message contains information about the graymail engine, status, configuration, and so on.

Graymail Log Example

```
Tue Mar 24 08:56:45 2015 Info: graymail [BASE] Logging at DEBUG level

Tue Mar 24 08:56:45 2015 Info: graymail [HANDLER] Initializing request handler

Tue Mar 24 08:56:50 2015 Info: graymail [ENGINE] Loaded graymail scanner library

Tue Mar 24 08:56:50 2015 Info: graymail [ENGINE] Created graymail scanner instance

Tue Mar 24 08:56:50 2015 Info: graymail [HANDLER] Debug mode disabled on graymail process

Tue Mar 24 08:56:50 2015 Info: graymail [HANDLER] Starting thread WorkerThread_0
```

Using Anti-Virus Logs

Table 22: AntiVirus Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of the check for the anti-virus update, as well as the results (whether an update of the engine or the virus definitions was needed, etc.)

Anti-Virus Log Example

In this example, the Anti-Virus log shows the Sophos anti-virus engine checking for updates to virus definitions (IDE) and the engine itself.

```
Thu Sep 9 14:18:04 2004 Info: Checking for Sophos Update

Thu Sep 9 14:18:04 2004 Info: Current SAV engine ver=3.84. No engine update needed

Thu Sep 9 14:18:04 2004 Info: Current IDE serial=2004090902. No update needed.
```

You can temporarily set this to DEBUG level to help diagnose why the anti-virus engine returns a particular verdict for a given message. The DEBUG logging information is verbose; use with caution.

Using AMP Engine Logs

The AMP Engine logs contain details of:

- File reputation query sent to the file reputation server and response received from the file reputation server.
- File analysis, if the file is uploaded to file analysis server. The status of the file analysis is recorded periodically until a response is received from the file analysis server.

Examples of AMP Engine Log Entries

Following are sample AMP Engine log entries based on certain scenarios:

- Initialization of File Reputation and File Analysis Servers, on page 38
- File Reputation Server Not Configured, on page 38
- Initialization of File Reputation Query, on page 38
- Response Received for File Reputation Query from File Reputation Server, on page 39
- File Uploaded for Analysis and File Analysis Process, on page 40
- File Not Uploaded for Analysis, on page 41
- File Upload Skipped for File Analysis due to File Upload Limit , on page 41
- File Upload Skipped for File Analysis due to File Analysis Server Error, on page 42
- File Retrospective Verdict Received, on page 42

Initialization of File Reputation and File Analysis Servers

```
Wed Oct 5 15:17:31 2016 Info: File reputation service initialized successfully
Wed Oct 5 15:17:31 2016 Info: The following file type(s) can be sent for File Analysis:
Microsoft Windows / DOS Executable, Microsoft Office 97-2004 (OLE), Microsoft Office 2007+
 (Open XML), Other potentially malicious file types, Adobe Portable Document Format (PDF).
 To allow analysis of new file type(s), go to Security Services > File Reputation and
Analysis.
Wed Oct 5 15:17:31 2016 Info: File Analysis service initialized successfully
```

File Reputation Server Not Configured

```
Tue Oct 4 23:15:24 2016 Warning: MID 12 reputation query failed for attachment 'Zombies.pdf'
 with error "Cloud query failed"
```

Initialization of File Reputation Query

```
Fri Oct 7 09:44:04 2016 Info: File reputation query initiating. File Name = 'mod-6.exe',
MID = 5, File Size = 1673216 bytes,
File Type = application/x-dosexec
```

Statistic	Description
File Name	The name of the file whose SHA-256 hash identifier is sent to the file reputation server. If the file name is not available, it is termed as unknown .
MID	The Message ID used to track messages that flow through the email pipeline.
File Size	The size of the file whose SHA-256 hash identifier is sent to the file reputation server.
File Type	The type of the file whose SHA-256 hash identifier is sent to the file reputation server. Following are the supported file types: • Microsoft Windows / DOS Executable • Microsoft Office 97-2004 (OLE) • Microsoft Office 2007+ (Open XML) • Other potentially malicious file types • Adobe Portable Document Format (PDF)

Response Received for File Reputation Query from File Reputation Server

```
Fri Oct 7 09:44:06 2016 Info: Response received for file reputation query from Cloud. File
Name = 'mod-6.exe', MID = 5, Disposition = MALICIOUS, Malware = W32.061DEF69B5-100.SBX.TG,
Reputation Score = 73, sha256 =
061def69b5c100e9979610fa5675bd19258b19a7ff538b5c2d230b467c312f19, upload_action = 2
```

Statistic	Description
File Name	The name of the file whose SHA-256 hash identifier is sent to the file reputation server. If the file name is not available, it is termed as unknown .
MID	The message ID used to track messages that flow through the email pipeline.
Disposition	The file reputation disposition values are: • MALICIOUS • CLEAN • FILE UNKNOWN - When the reputation score is zero. • VERDICT UNKNOWN - When the disposition is FILE UNKNOWN and score is non-zero. • LOW RISK - When no dynamic content is found in a file after file analysis, the verdict is Low Risk. The file is not sent for file analysis, and the message continues through the email pipeline.
Malware	The name of the malware threat.

Statistic	Description
Reputation score	The reputation score assigned to the file by the file reputation server. If the file disposition is **VERDICT UNKNOWN**, the appliance adjusts the file reputation verdict based on the reputation score and the threshold value.
Upload Action	The upload action value recommended by the file reputation server to take on the given file: • 0 - Need not send for upload • 1 - Send file for upload. **Note** The appliance uploads the file when the upload action value is '1.' • 2 - Do not send file for upload • 3 - Send only metadata for upload

File Uploaded for Analysis and File Analysis Process

```
Wed Sep 28 11:31:58 2016 Info: File uploaded for analysis. SHA256:
e7ae35a8227b380ca761c0317e814e4aaa3d04f362c6b913300117241800f0ea

Wed Sep 28 11:36:58 2016 Info: File Analysis is running for SHA:
e7ae35a8227b380ca761c0317e814e4aaa3d04f362c6b913300117241800f0ea

Fri Oct 7 07:39:13 2016 Info: File Analysis complete. SHA256:
16454aff5082c2e9df43f3e3b9cdba3c6ae1766416e548c30a971786db570bfc, Submit Timestamp:
1475825466, Update Timestamp: 1475825953, Disposition: 3 Score: 100, run_id: 194926004
Details: Analysis is completed for the File
SHA256[16454aff5082c2e9df43f3e3b9cdba3c6ae1766416e548c30a971786db570bfc]
Spyname:[W32.16454AFF50-100.SBX.TG]
```

Statistic	Description
SHA256	The SHA-256 hash identifier for the corresponding file.
Submit Timestamp	The date and time at which the file is uploaded to the file analysis server by the appliance .
Update Timestamp	The date and time at which the file analysis for the file is complete
Disposition	The file reputation disposition values are. • 1 - No malware detected • 2 - Clean • 3 - Malware
Score	The analysis score assigned to the file by the file analysis server.
Run ID	The numeric value (ID) assigned to the file by the file analysis server for a particular file analysis.
Details	Additional information if errors are reported during file analysis, otherwise it indicates that the final analysis is complete for the file.

Statistic	Description
Spy Name	The name of the threat, if a malware is found in the file during file analysis.

File Not Uploaded for Analysis

```
Wed Sep 14 12:27:52 2016 Info: File not uploaded for analysis. MID = 0 File
SHA256[a5f28f1fed7c2fe88bcdf403710098977fa12c32d13bfbd78bbe27e95b245f82] file
mime[text/plain] Reason: No active/dynamic contents exists
```

Statistic	Description
MID	The message ID used to track messages that flow through the email pipeline.
File MIME	The MIME type of the file.
Reason	Following are one of the reason values for file not uploaded to the file analysis server even when the upload_action is set to '1': • File already uploaded by another node - The file is already uploaded to the file analysis server via another appliance . • File analysis in progress - File is already selected for upload which is in progress. • File already uploaded to File Analysis server • Not a supported File type • File size is out of bounds - The upload file size exceeds the threshold limit set by the file analysis server. • Upload queue was full • File Analysis server error • No active/dynamic contents exists • Generic/Unknown Error

File Upload Skipped for File Analysis due to File Upload Limit

```
Tue Jun 20 13:22:56 2017 Info: File analysis upload skipped. SHA256:
b5c7e26491983baa713c9a2910ee868efd891661c6a0553b28f17b8fdc8cc3ef,Timestamp[1454782976]
details[File SHA256[b5c7e26491983baa713c9a2910ee868efd891661c6a0553b28f17b8fdc8cc3ef] file
mime[application/pdf], upload priority[Low] not uploaded, re-tries[3], backoff[986]
discarding ...]
Tue Jun 20 13:22:56 2017 Critical: The attachment could not be uploaded to the
File Analysis server because the appliance exceeded the upload limit
```

Statistic	Description
SHA256	The SHA-256 hash identifier for the corresponding file.
Timestamp	The date and time at which the file failed to upload to the file analysis server.
Details	Details of the File Analysis server error.
File MIME	The MIME type of the file.

Statistic	Description
Upload priority	Upload priority values are: • High - For all selected file types, except PDF file type. • Low - For only PDF file types
Re-tries	The number of upload attempts performed on a given file. **Note** A maximum of three upload attempts can be performed on a given file.
Backoff (x)	The number of (x) seconds before the appliance needs to wait before it makes an attempt to upload the file to the file analysis server. This occurs when the appliance reaches the daily upload limit.
Critical (Reason)	The attachment could not be uploaded to the File Analysis server because the appliance exceeded the upload limit.

File Upload Skipped for File Analysis due to File Analysis Server Error

```
Sat Feb 6 13:22:56 2016 Info:SHA256:
69e17e213732da0d0cbc48ae7030a4a18e0c1289f510e8b139945787f67692a5,Timestamp[1454959409]
details[Server Response HTTP code:[502]]
```

Statistic	Description
SHA256	The SHA-256 hash identifier for the corresponding file.
Timestamp	The date and time at which an attempt is made to upload the file to the file analysis server.
Details	Information about the File Analysis server error.

File Retrospective Verdict Received

```
Fri Oct 7 07:39:13 2016 Info: Retrospective verdict received. SHA256:
16454aff5082c2e9df43f3e3b9cdba3c6ae1766416e548c30a971786db570bfc, Timestamp: 1475832815.7,
Verdict: MALICIOUS, Reputation Score: 0, Spyname: W32.16454AFF50-100.SBX.
```

Statistic	Description
SHA256	The SHA-256 hash identifier for the corresponding file.
Timestamp	The date and time at which a file retrospective verdict is received from the file analysis server.
Verdict	The file retrospective verdict value is malicious or clean .
Reputation Score	The reputation score assigned to the file by the file reputation server.
Spyname	The name of the threat, if a malware is found in the file during file analysis.

Using Spam Quarantine Logs

Table 23: Spam Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of actions taken (messages quarantined, released from quarantine, etc.).

Spam Quarantine Log Example

In this example, the log shows a message (MID 8298624) being released from the quarantine to admin@example.com.

```
Mon Aug 14 21:41:47 2006 Info: ISQ: Releasing MID [8298624, 8298625] for all

Mon Aug 14 21:41:47 2006 Info: ISQ: Delivering released MID 8298624 (skipping work
queue)

Mon Aug 14 21:41:47 2006 Info: ISQ: Released MID 8298624 to admin@example.com

Mon Aug 14 21:41:47 2006 Info: ISQ: Delivering released MID 8298625 (skipping work
queue)

Mon Aug 14 21:41:47 2006 Info: ISQ: Released MID8298625 to admin@example.com
```

Using Spam Quarantine GUI Logs

Table 24: Spam GUI Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	The message consists of actions taken, including user authentication, etc.

Spam Quarantine GUI Log Example

In this example, the log shows a successful authentication, login and logout:

```
Fri Aug 11 22:05:28 2006 Info: ISQ: Serving HTTP on 192.168.0.1, port 82

Fri Aug 11 22:05:29 2006 Info: ISQ: Serving HTTPS on 192.168.0.1, port 83

Fri Aug 11 22:08:35 2006 Info: Authentication OK, user admin

Fri Aug 11 22:08:35 2006 Info: logout:- user:pqufOtL6vyI5StCqhCfO session:10.251.23.228

Fri Aug 11 22:08:35 2006 Info: login:admin user:pqufOtL6vyI5StCqhCfO
session:10.251.23.228

Fri Aug 11 22:08:44 2006 Info: Authentication OK, user admin
```

Using LDAP Debug Logs

Table 25: LDAP Debug Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted
Message	LDAP Debug message

LDAP Debug Log Example

Note Individual lines in log files are NOT numbered. They are numbered here only for sample purposes

1	`Thu Sep  9 12:24:56 2004 Begin Logfile`
2	`Thu Sep  9 12:25:02 2004 LDAP:` `Masquerade query sun.masquerade address employee@routing.qa to employee@mail.qa`
3	`Thu Sep  9 12:25:02 2004 LDAP:` `Masquerade query sun.masquerade address employee@routing.qa to employee@mail.qa`
4	`Thu Sep  9 12:25:02 2004 LDAP:` `Masquerade query sun.masquerade address employee@routing.qa to employee@mail.qa`
5	`Thu Sep  9 12:28:08 2004 LDAP: Clearing LDAP cache`
6	`Thu Sep  9 13:00:09 2004 LDAP:` `Query '(&(ObjectClass={g})(mailLocalAddress={a}))' to server sun (sun.qa:389)`

7	`Thu Sep  9 13:00:09 2004 LDAP:`
	`After substitute, query is '(&(ObjectClass=inetLocalMailRecipient)`
	`(mailLocalAddress=rroute.d00002b.loc@ldap.route.local.add00002.qa))'`

8	`Thu Sep  9 13:00:09 2004 LDAP: connecting to server`
9	`Thu Sep  9 13:00:09 2004 LDAP: connected`

10	`Thu Sep  9 13:00:09 2004 LDAP:`
	`Query (&(ObjectClass=inetLocalMailRecipient)`
	`(mailLocalAddress=rroute.d00002b.loc@ldap.route.local.add00002.qa)) returned 1`
	` results`

11	`Thu Sep  9 13:00:09 2004 LDAP: returning: [<LDAP:>]`

Use as a guide to reading the preceding log file.

Table 26: Detail of LDAP Debug Log Example

Line Number	Description
1	The log file is initialized.
2 3 4	The listener is configured to use LDAP for masquerading, specifically with the LDAP query named “sun.masquerade.” The address employee@routing.qa is looked up in the LDAP server, a match is found, and the resulting masquerade address is employee@mail.qa, which will be written to the message headers and/or the envelope from, depending on the masquerade configuration.
5	The user has manually run ldapflush .
6	A query is about to be sent to sun.qa, port 389. The query template is: (&(ObjectClass={g})(mailLocalAddress={a})). The {g} will be replaced by the groupname specified in the calling filter, either a rcpt-to-group or mail-from-group rule. The {a} will be replaced by the address in question.
7	Now the substitution (described previously) takes place, and this is what the query looks like before it is sent to the LDAP server.
8	The connection to the server is not yet established, so make a connection.
9	The data that is sent to the server.
10	The result is an empty positive, meaning one record was returned, but since the query didn't ask for any fields, there is no data to report. These are used for both group and accept queries when the query checks to see if there is a match in the database.

Using Safelist/Blocklist Logs

The following table shows the statistics recorded in safelist/blocklist logs.

Table 27: Safelist/Blocklist Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted.
Message	The message consists of actions taken, including user authentication, and so forth.

Safelist/Blocklist Log Example

In this example, the safelist/blocklist log shows the appliance creating database snapshots every two hours. It also shows when senders were added to the database.

```
Fri Sep 28 14:22:33 2007 Info: Begin Logfile Fri Sep 28 14:22:33 2007 Info: Version:
6.0.0-425 SN: XXXXXXXXXXXX-XXX Fri Sep 28 14:22:33 2007 Info: Time offset from UTC:
10800 seconds Fri Sep 28 14:22:33 2007 Info: System is coming up.
```

```
Fri Sep 28 14:22:33 2007 Info: SLBL: The database snapshot has been created.

Fri Sep 28 16:22:34 2007 Info: SLBL: The database snapshot has been created.

Fri Sep 28 18:22:34 2007 Info: SLBL: The database snapshot has been created.

Fri Sep 28 20:22:34 2007 Info: SLBL: The database snapshot has been created.

Fri Sep 28 22:22:35 2007 Info: SLBL: The database snapshot has been created.

.......................

Mon Oct 1 14:16:09 2007 Info: SLBL: The database snapshot has been created.

Mon Oct 1 14:37:39 2007 Info: SLBL: The database snapshot has been created.

Mon Oct 1 15:31:37 2007 Warning: SLBL: Adding senders to the database failed.

Mon Oct 1 15:32:31 2007 Warning: SLBL: Adding senders to the database failed.

Mon Oct 1 16:37:40 2007 Info: SLBL: The database snapshot has been created.
```

Using Reporting Logs

The following table shows the statistics recorded in reporting logs.

Table 28: Reporting Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted.
Message	The message consists of actions taken, including user authentication, and so forth.

Reporting Log Example

In this example, the Reporting log shows the appliance set at the information log level.

```
Wed Oct 3 13:39:53 2007 Info: Period minute using 0 (KB)

Wed Oct 3 13:39:53 2007 Info: Period month using 1328 (KB)

Wed Oct 3 13:40:02 2007 Info: Update 2 registered appliance at 2007-10-03-13-40

Wed Oct 3 13:40:53 2007 Info: Pages found in cache: 1304596 (99%). Not found: 1692

Wed Oct 3 13:40:53 2007 Info: Period hour using 36800 (KB)

Wed Oct 3 13:40:53 2007 Info: Period day using 2768 (KB)

Wed Oct 3 13:40:53 2007 Info: Period minute using 0 (KB)

Wed Oct 3 13:40:53 2007 Info: Period month using 1328 (KB)

Wed Oct 3 13:40:53 2007 Info: HELPER checkpointed in 0.00580507753533 seconds

Wed Oct 3 13:41:02 2007 Info: Update 2 registered appliance at 2007-10-03-13-41
```

```
Wed Oct 3 13:41:53 2007 Info: Pages found in cache: 1304704 (99%). Not found: 1692

Wed Oct 3 13:41:53 2007 Info: Period hour using 36800 (KB)

Wed Oct 3 13:41:53 2007 Info: Period day using 2768 (KB)

Wed Oct 3 13:41:53 2007 Info: Period minute using 0 (KB)

Wed Oct 3 13:41:53 2007 Info: Period month using 1328 (KB)

Wed Oct 3 13:42:03 2007 Info: Update 2 registered appliance at 2007-10-03-13-42
```

Using Reporting Query Logs

The following table shows the statistics recorded in reporting query logs.

Table 29: Reporting Query Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted.
Message	The message consists of actions taken, including user authentication, and so forth.

Reporting Query Log Example

In this example, the reporting query log shows the appliance running a daily outgoing email traffic query for the period from August 29 to October 10, 2007.

```
Tue Oct 2 11:30:02 2007 Info: Query: Closing interval handle 811804479.

Tue Oct 2 11:30:02 2007 Info: Query: Closing interval handle 811804480.

Tue Oct 2 11:30:02 2007 Info: Query: Closing query handle 302610228.

Tue Oct 2 11:30:02 2007 Info: Query: Merge query with handle 302610229 for
['MAIL_OUTGOING_TRAFFIC_SUMMARY.

DETECTED_SPAM', 'MAIL_OUTGOING_TRAFFIC_SUMMARY.DETECTED_VIRUS',
'MAIL_OUTGOING_TRAFFIC_SUMMARY.THREAT_CONTENT_FILTER',
'MAIL_OUTGOING_TRAFFIC_SUMMARY.TOTAL_CLEAN_RECIPIENTS',
'MAIL_OUTGOING_TRAFFIC_SUMMARY.TOTAL_RECIPIENTS_PROCESSED'] for rollup period "day" with
interval range 2007-08-29 to 2007-10-01
with key constraints

None sorting on ['MAIL_OUTGOING_TRAFFIC_SUMMARY.DETECTED_SPAM'] returning results from
0 to 2 sort_ascending=False.

Tue Oct 2 11:30:02 2007 Info: Query: Closing query handle 302610229.

Tue Oct 2 11:30:02 2007 Info: Query: Merge query with handle 302610230 for
['MAIL_OUTGOING_TRAFFIC_SUMMARY.

TOTAL_HARD_BOUNCES', 'MAIL_OUTGOING_TRAFFIC_SUMMARY.TOTAL_RECIPIENTS_DELIVERED',
'MAIL_OUTGOING_TRAFFIC_SUMMARY.TOTAL_RECIPIENTS'] for rollup period "day" with interval
range 2007-08-29 to
2007-10-01 with key constraints None sorting on
['MAIL_OUTGOING_TRAFFIC_SUMMARY.TOTAL_HARD_BOUNCES'] returning
results from 0 to 2 sort_ascending=False.
```

```
Tue Oct 2 11:30:02 2007 Info: Query: Closing query handle 302610230.
```

Using Updater Logs

Table 30: Updater Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted.
Message	The message consists of system service update information, as well as AsyncOS checking for updates and the scheduled date and time of the next update.

Updater Log Example

In this example, the logs show the appliance being updated with new McAfee Anti-Virus definitions.

```
Fri Sep 19 11:07:51 2008 Info: Starting scheduled update

Fri Sep 19 11:07:52 2008 Info: Acquired server manifest, starting update 11

Fri Sep 19 11:07:52 2008 Info: Server manifest specified an update for mcafee

Fri Sep 19 11:07:52 2008 Info: mcafee was signalled to start a new update

Fri Sep 19 11:07:52 2008 Info: mcafee processing files from the server manifest

Fri Sep 19 11:07:52 2008 Info: mcafee started downloading files

Fri Sep 19 11:07:52 2008 Info: mcafee downloading remote file
"http://stage-updates.ironport.com/mcafee/dat/5388"

Fri Sep 19 11:07:52 2008 Info: Scheduled next update to occur at Fri Sep 19 11:12:52
2008

Fri Sep 19 11:08:12 2008 Info: mcafee started decrypting files

Fri Sep 19 11:08:12 2008 Info: mcafee decrypting file
"mcafee/dat/5388" with method "des3_cbc"

Fri Sep 19 11:08:17 2008 Info: mcafee started decompressing files

Fri Sep 19 11:08:17 2008 Info: mcafee started applying files

Fri Sep 19 11:08:17 2008 Info: mcafee applying file "mcafee/dat/5388"

Fri Sep 19 11:08:18 2008 Info: mcafee verifying applied files

Fri Sep 19 11:08:18 2008 Info: mcafee updating the client manifest

Fri Sep 19 11:08:18 2008 Info: mcafee update completed

Fri Sep 19 11:08:18 2008 Info: mcafee waiting for new updates

Fri Sep 19 11:12:52 2008 Info: Starting scheduled update

Fri Sep 19 11:12:52 2008 Info: Scheduled next update to occur at Fri Sep 19 11:17:52
2008
```

```
Fri Sep 19 11:17:52 2008 Info: Starting scheduled update

Fri Sep 19 11:17:52 2008 Info: Scheduled next update to occur at Fri Sep 19 11:22:52
2008
```

Updater Log Example

In this example, the logs show the automatic updates being disabled and backup being applied to the Sophos Anti-Virus definitions.

```
Fri Mar 10 15:05:55 2017 Debug: Skipping update request for "postx"

Fri Mar 10 15:05:55 2017 Debug: postx updates disabled

Fri Mar 10 15:05:55 2017 Debug: Skipping update request for "postx"

Fri Mar 10 15:05:55 2017 Trace: command session starting

Fri Mar 10 15:05:55 2017 Info: Automatic updates disabled for engine Sophos engine

Fri Mar 10 15:05:55 2017 Info: Sophos: Backup update applied successfully

Fri Mar 10 15:05:55 2017 Info: Internal SMTP system attempting to send a message to
abshastr@ironport.com
with subject 'Automatic updates are now disabled for sophos' attempt #0).

Fri Mar 10 15:05:55 2017 Debug: amp feature key disabled

Fri Mar 10 15:05:55 2017 Debug: Skipping update request for "amp"

Fri Mar 10 15:05:55 2017 Debug: amp feature key disabled
```

Understanding Tracking Logs

Tracking logs record information about the email operations of AsyncOS. The log messages are a subset of the messages recorded in the mail logs.

The tracking logs are used by the appliance's message tracking component to build the message tracking database. Because the log files are consumed in the process of building the database, the tracking logs are transient. The information in tracking logs is not designed to be read or analyzed by humans.

You can also view tracking information from multiple appliances using the Cisco Security Management appliance .

Using Authentication Logs

The authentication log records successful user logins and unsuccessful login attempts.

Table 31: Authentication Log Statistics

Statistic	Description
Timestamp	Time that the bytes were transmitted.
Message	The message consists of the username of a user who attempted to log in to the appliance and whether the user was authenticated successfully.

Authentication Log Example

In this example, the log shows the log in attempts by users “admin,” “joe,” and “dan.”

```
Wed Sep 17 15:16:25 2008 Info: Begin Logfile

Wed Sep 17 15:16:25 2008 Info: Version: 6.5.0-262 SN: XXXXXXX-XXXXX

Wed Sep 17 15:16:25 2008 Info: Time offset from UTC: 0 seconds

Wed Sep 17 15:18:21 2008 Info: User admin was authenticated successfully.

Wed Sep 17 16:26:17 2008 Info: User joe failed authentication.

Wed Sep 17 16:28:28 2008 Info: User joe was authenticated successfully.

Wed Sep 17 20:59:30 2008 Info: User admin was authenticated successfully.

Wed Sep 17 21:37:09 2008 Info: User dan failed authentication.
```

Using Configuration History Logs

A configuration history log consists of a configuration file with an additional section listing the name of the user, a description of where in the configuration the user made changes, and the comment the user entered when committing the change. Each time a user commits a change, a new log is created containing the configuration file after the change.

Configuration History Log Example

In this example, the configuration history log shows that the user (admin) added a guest user to the table that defines which local users are allowed to log in to the system.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE config SYSTEM "config.dtd">

<!--

XML generated by configuration change.

Change comment: added guest user

User: admin

Configuration are described as:

This table defines which local users are allowed to log into the system.

Product: Cisco IronPort M160 Messaging Gateway(tm) Appliance

Model Number: M160

Version: 6.7.0-231

Serial Number: 000000000ABC-D000000

Number of CPUs: 1

Memory (GB): 4
```

```
Current Time: Thu Mar 26 05:34:36 2009

Feature "Cisco IronPort Centralized Configuration Manager": Quantity = 10, Time
Remaining = "25 days"

Feature "Centralized Reporting": Quantity = 10, Time Remaining = "9 days"

Feature "Centralized Tracking": Quantity = 10, Time Remaining = "30 days"

Feature "Centralized Spam Quarantine": Quantity = 10, Time Remaining = "30 days"

Feature "Receiving": Quantity = 1, Time Remaining = "Perpetual"

-->

<config>
```

Using External Threat Feeds Engine Logs

The ETF Logs contain information about the ETF engine, status, configuration, and so on. Most information is at the Info or Debug level.

Example of External Threat Feeds Engine Logs

```
Thu Jun 7 04:54:15 2018 Info: THREAT_FEEDS: Job failed with exception: Invalid URL or Port
 Thu Jun 7 05:04:13 2018 Info: THREAT_FEEDS: A delta poll is scheduled for the source: S1
Thu Jun 7 05:04:13 2018 Info: THREAT_FEEDS: A delta poll has started for the source: S1,
domain: s1.co, collection: sss
Thu Jun 7 05:04:13 2018 Info: THREAT_FEEDS: Observables are being fetched from the source:
 S1 between 2018-06-07 04:34:13+00:00 and 2018-06-07 05:04:13.185909+00:00
Thu Jun 7 05:04:13 2018 Info: THREAT_FEEDS: 21 observables were fetched from the source:
S1
Thu Jun 7 05:19:14 2018 Info: THREAT_FEEDS: A delta poll is scheduled for the source: S1
Thu Jun 7 05:19:14 2018 Info: THREAT_FEEDS: A delta poll has started for the source: S1,
domain: s1.co, collection: sss
```

ETF Source Configuration Failure - Invalid Collection Name

In this example, the log shows that your appliance was not able to fetch threat feeds from an external threat feed source because of an invalid collection name.

```
Info: THREAT_FEEDS: [TaxiiClient] Failed to poll threat feeds from following source:
hailataxii.com, cause of failure: Invalid Collection name
```

Solution

Go to *Mail Policies > External Threat Feeds Manager* page in the web interface or use the `threatfeedsconfig > sourceconfig` subcommand in the CLI and enter the correct collection name for the configured external threat feed source.

ETF Source Configuration Failure - HTTP Error

In this example, the log shows that your appliance was not able to fetch threat feeds from an external threat feed source because of an HTTP error.

```
Info: THREAT_FEEDS: [TaxiiClient] Failed to poll threat feeds from following source:
hailataxii.com , cause of failure: HTTP Error
```

Solution

Go to *Mail Policies* > *External Threat Feeds Manager* page in the web interface or use the `threatfeedsconfig > sourceconfig` subcommand in the CLI and enter the correct polling path or user authentication credentials for the configured external threat feed source.

ETF Source Configuration Failure - Invalid URL

In this example, the log shows that your appliance was not able to fetch threat feeds from an external threat feed source because of an invalid URL.

```
Info: THREAT_FEEDS: [TaxiiClient] Failed to poll threat feeds from following source:
hailataxii.com , cause of failure: HTTP Error
```

Solution

Go to *Mail Policies* > *External Threat Feeds Manager* page in the web interface or use the `threatfeedsconfig > sourceconfig` subcommand in the CLI and enter the correct hostname or port number for the configured external threat feed source.

Using Consolidated Event Logs

When configuring a log subscription with the log type as Consolidated Event Logs, use the 'Log Fields' option if you want to include specific message attributes in a single log line output.

The following log fields are selected by default when you configure a log subscription with the log type as Consolidated Event Log:

- ICID
- DCID
- Serial Number
- MID

Note You cannot remove any of the default log fields from the Selected Log Fields list.

Example of Consolidated Event Logs

In this example, the log shows all the available fields selected when you configure a log subscription with the log type as Consolidated Event Logs.

```
Sun Aug 25 12:37:08 2019: CEF:0|Cisco|C100V Email Security Virtual Appliance|13.0.0-
283|ESA_CONSOLIDATED_LOG_EVENT|Consolidated Log Event|5|cs6Label=SDRRepScore
cs6=Weak deviceExternalId=42157574DD75FA3BD343-C964FC856529 ESAMID=144137
startTime=Sun Aug 25 12:35:39 2019 deviceInboundInterface=IncomingMail
ESADMARCVerdict=Skipped dvc=10.10.2.10 ESAAttachmentDetails={'MSOLE2msword.docx': {'AMP':
{'Verdict': 'FILE UNKNOWN', 'fileHash':
'917a35e8ffdd121c35b47a937dd4399539f0aa5b52a60fd038e0c4fdea78d357'}, 'BodyScanner': {}}}
ESAFriendlyFrom=ec@tester.com deviceDirection=0 ESAMailFlowPolicy=ACCEPT
suser=ec@tester.com cs1Label=MailPolicy cs1=DEFAULT act=QUARANTINED
ESAFinalActionDetails=To POLICY cs4Label=ExternalMsgID
cs4='<20190825173813.6679.31096@vm21bsd0008.cs21>' duser=aroma@mar-
esa.com ESAHeloIP=10.10.4.8 cfp1Label=SBRSScore cfp1=None ESASDRDomainAge=23 years 6 months
```

```
19 days cs3Label=SDRThreatCategory cs3=N/A ESASPFVerdict=None sourceHostName=unknown
ESASenderGroup=SUSPECTLIST sourceAddress=10.10.4.8 ESAICID=190746
cs5Label=ESAMsgLanguage cs5=English
msg=[Cousin\=20Domain][SUSPECTED\=20SPAM]\=20asdfdsaf
```

Log Field	CEF Field Name	CEF Field Value
Prefix Fields		
	CEF format version	Example: 0
	Appliance vendor	Example: Cisco
	Appliance product	Example: C100V Email Security Virtual Appliance
	Appliance version	Example: 13.0.0-234
	Event Class ID	Example: ESA_CONSOLIDATED_LOG_EVENT
	Event Name	Example: Consolidated Log Event
	Severity	Example: 5
GUI Fields		
Serial Number	deviceExternalId	Example: 42156AC79142E979C5CD-02DE66639E9C
ICID Timestamp	startTime	Example: Mon Jul 29 11:22:22 2019
ICID	ESAICID	Example: 199
Listener Name	deviceInboundInterface(for incoming mails) deviceOutboundInterface(for outgoing mails)	Example: Inbound Example: Outbound
Sender IP	sourceAddress	Example: 10.10.2.75
Sender Domain	sourceHostName	Example: demo.cisco.com
Mail Direction	deviceDirection	Example: 0 0 -> incoming 1 -> outgoing
Mail Language	cs5	Example: cs5Label=ESAMsgLanguage cs5=English
SBRS Score	cfp1	Example: cfp1Label=SBRSScore, cfp1=1.1

Log Field	CEF Field Name	CEF Field Value
Data IP	dvc	Example: 10.10.2.75
Mail Sender Geo Location	cs2	Example: cs2Label=GeoLocation cs2=India
Message Too Big from Sender	ESAMsgTooBigFromSender	Example: true Possible Values: true/false
Rate Limited IP	ESARateLimitedIP	Example: 10.10.2.75
Mail Policy Name	cs1	Example: cs1Label=MailPolicy cs1=default
Mail Flow Policy Name	ESAMailFlowPolicy	Example: ACCEPT
Sender Group Name	ESASenderGroup	Example: UNKNOWNLIST
DHA IP	ESADHASource	Example: 10.10.2.75
Recipients	duser	Example: demo@test.com
Remote IP/Helo Domain IP	ESAHeloIP	Example: 10.10.2.75
Remote Host/ Helo Domain	ESAHeloDomain	Example: test.com
TLS Outgoing Connection Status	ESATLSOutConnStatus	Example: Success Possible Values: Success/Failure
TLS Outgoing Protocol	ESATLSOutProtocol	Example: TLSv1.2
TLS Outgoing Cipher	ESATLSOutCipher	Example: ECDHE-RSA-AES128-GCM-SHA256
TLS Incoming Connection Status	ESATLSInConnStatus	Example: Success Possible Values: Success/Failure
TLS Incoming Protocol	ESATLSInProtocol	Example: TLSv1.2
TLS Incoming Cipher	ESATLSInCipher	Example: ECDHE-RSA-AES128-GCM-SHA256
DMARC Verdict	ESADMARCVerdict	Example: Success Possible Values: PermFailure/TempFailure/ Reject/Success

Log Field	CEF Field Name	CEF Field Value
DKIM Verdict	ESADKIMVerdict	Example: Pass Possible Values: Pass/Neutral/TempError/ PermError/HardFail/None
SPF Verdict	ESASPFVerdict	Example: Pass Possible Values: Pass/Neutral/SoftFail/Fail/ TempError/PermError
Friendly From	ESAFriendlyFrom	Example: demo@test.com
Mail From	suser	Example: demo@test.com
Reply-To	ESAReplyTo	Example: demo@test.com
Subject	msg	Example: This is a sample subject
MID	ESAMID	Example: 101
Message ID	cs4	Example: cs1Label=ExternalMsgID cs1=20190729112221.42958.40626 @vm21esa0075.cs21
SDR Reputation Score	cs6	Example: cs6Label= SDRRepScore cs6=Tainted
SDR Consolidated Domain Age	ESASDRDomainAge	Example: 1 year 21 days
SDR Consolidated Threat Category	cs3	Example: cs3Label= SDRThreatCategory cs3=mal
Message Filters Verdict	Message Filters Verdict	Example: MATCH Possible Values: NOT EVALUATED/MATCH/NO MATCH
AS Verdict	ESAASVerdict	Example: POSITIVE Possible Values: Not EVALUATED/NEGATIVE/SUSPECT/ BULK_MAIL/SOCIAL_MAIL/MARKE TING_MAIL/POSITIVE

Log Field	CEF Field Name	CEF Field Value
AV Verdict	ESAAVVerdict	Example: POSITIVE Possible Values: NOT EVALUATED/NEGATIVE/REPAIRED /ENCRYPTED/UNSCANNABLE/POSITIVE
AMP Verdict	ESAAMPVerdict	Example: UNKNOWN Possible Values: NOT EVALUATED/CLEAN/FA_PENDING/ UNKNOWN/SKIPPED/ UNSCANNABLE /LOW_RISK/MALICIOUS
Graymail Verdict	ESAGMVerdict	Example: POSITIVE Possible Values: NOT EVALUATED/POSITIVE/NEGATIVE
Content Filters Verdict	ESACFVerdict	Example: MATCH Possible Values: NOT EVALUATED/MATCH/NO MATCH
Outbreak Filters Verdict	ESAOFVerdict	Example: NEGATIVE Possible Values: NOT EVALUATED/POSITIVE/NEGATIVE
DLP Verdict	ESADLPVerdict	Example: VIOLATION Possible Values: NOT EVALUATED/NO TRIGGER/VIOLATION/NO VIOLATION

Log Field	CEF Field Name	CEF Field Value
URL Details	ESAURLDetails	Example: {url1:{expanded_url:<>, category:<>, wbrs_score:<>, in_attachment:<>, Attachment_with_url:<>,},url2:{…}} **Note** A URL is truncated if it contains more than 255 characters
File Details	ESAAttachmentDetails	Example: {name1:{source: {<>hash:<>, verdicts:<>}}} **Note** A filename is truncated if it contains more than 255 characters.
Mailbox Auto-Remediation Details	ESAMARAction	Example: {action:<>;succesful_rcpts=<>;failed_recipients=<>;filename=<>}
DCID	ESADCID	Example: 199
DCID Timestamp	EndTime	Example: Mon Jul 29 09:55:07 2019
DANE Status	ESADaneStatus	Example: success Possible Values: success/failure
DANE Host	ESADaneHost	Example: testdomain.com
Message Final Action	act	Example: act=DELIVERED Possible Values: DROPPED/BOUNCED/DELIVERED - if the message is not quarantined. QUARANTINED - if the message is quarantined. DQ - if the message is sent to Delayed Quarantine. This is an exception and not a quarantine type.

Log Field	CEF Field Name	CEF Field Value
Message Final Action Details	ESAFinalActionDetails	Example: act=DROPPED ESAFinalActionDetails= By AMP act=QUARANTINED ESAFinalActionDetails=To SPAM

Note If there is no value for a selected log field (for example, 'DKIMVerdict' because DKIM is not enabled on your appliance), the log field is not included in the log message.

Using CSN Logs

The CSN logs contain details about the CSN data uploads. The CSN data (appliance and feature usage details can be seen at the trace level.

Examples of CSN Data Log Entries:

- In this example, the log shows that the appliance was not able to send the CSN data to Cisco because the appliance smart license was not registered with Cisco Smart Software Manager (CSSM).

  ```
  Tue Apr 7 12:52:47 2020 Warning: Device is not
  registered with CSSM. Skipping upload of CSN data
  ```

  Solution: Make sure that you register your appliance smart license with Cisco Smart Software Manager (CSSM).

- In this example, the log shows that the appliance was not able to send the CSN data to Cisco because of a Cisco Security Services Exchange (SSE) connectivity error.

  ```
  Thu Apr 9 13:32:46 2020 Warning: The appliance
  failed to upload CSN data. reason for failure:
  SSE error: HTTP Error 503: Service Unavailable
  ```

  Solution: Make sure that you disable CSN and enable it again on your appliance .

Using Advanced Phishing Protection Logs

The Advanced Phishing Protection logs contain information related to Cisco Advanced Phishing Protection Cloud Service. Most information is at the Info or Critical level.

Examples of Advanced Phishing Protection Data Log Entries:

- In this example, the log shows that the appliance was not able to forward the message headers to Cisco Advanced Phishing Protection Cloud Service because the service expired.

  ```
  Wed May 6 18:21:40 2020 Info: eaas : You cannot
  forward the MID [877] Message Headers to Cisco Advanced
  Phishing Protection Cloud Service as the service has
  expired
  ```

- In this example, the log shows that the Cisco Advanced Phishing Protection Cloud Service has expired and is disabled in your appliance .

```
Wed May 6 18:21:40 2020 Info: eaas : Cisco
Advanced Phishing Protection Cloud Service has expired
and is disabled. Contact your Cisco Account manager to
renew the service and then enable it.
```

Solution: Contact your Cisco Account manager to renew the service and then enable it.

- In this example, the log shows that the Cisco Advanced Phishing Protection Cloud Service will expire on a particular date.

```
Fri May 8 04:50:26 2020 Info: eaas : Cisco
Advanced Phishing Protection Cloud Service expires on
2020-05-10 07:00:00. You need to contact your Cisco Account
manager to renew the service.
```

Solution: Contact your Cisco Account manager to renew the service.

Log Subscriptions

Configuring Log Subscriptions

It is recommended that you avoid deleting preexisting log subscriptions on appliances .

Use the Log Subscriptions page on the System Administration menu (or the **logconfig** command in the CLI) to configure a log subscription. Log subscriptions create log files that store information about AsyncOS activity, including errors. A log subscription is either retrieved or delivered (pushed) to another computer. Generally, log subscriptions have the following attributes:

Table 32: Log File Attributes

Attribute	Description
Log type	Defines the type of information recorded and the format of the logs subscriptions. See *Table: Log Types* for more information.
Log Name	Nickname for the log subscription to be used for your future reference.
Log Fields	Select the required log fields to include in the consolidated event log line for a given message. **Note** The Serial Number and MID log fields are selected by default, and you cannot deselect these fields. **Note** This field is only applicable when you are configuring a log subscription with the log type as Consolidated Event Logs.

Attribute	Description
File Name	Used for the physical name of the file when written to disk. If multiple appliances are being used, the log filename should be unique to identify the system that generated the log file.
Rollover by File Size	The maximum size the file can reach before rolling over.
Rollover by Time	Sets the time interval for file rollovers.
Rate Limit	Sets the maximum number of logged events in the log file, within the specified time range (in seconds). The default time range value is 10 seconds.
Log level	Sets the level of detail for each log subscription.
Retrieval method	Defines how the log subscription will be obtained from the appliance .

Log Levels

Log levels determine the amount of information delivered in a log. Logs can have one of five levels of detail. A more detailed setting creates larger log files and puts more drain on system performance. More detailed settings include all the messages contained in less detailed settings, plus additional messages. As the level of detail increases, system performance decreases.

Note Log levels may be selected for all mail log types.

Table 33: Log Levels

Log Level	Description
Critical	The least detailed setting. Only errors are logged. Using this setting will not allow you to monitor performance and other important activities; however, the log files will not reach their maximum size as quickly. This log level is equivalent to the syslog level “Alert.”
Warning	All errors and warnings created by the system. Using this setting will not allow you to monitor performance and other important activities. This log level is equivalent to the syslog level “Warning.”
Information	The information setting captures the second-by-second operations of the system. For example, connections opened or delivery attempts. The Information level is the recommended setting for logs. This log level is equivalent to the syslog level “Info.”
Debug	Use the Debug log level when you are trying to discover the cause of an error. Use this setting temporarily, and then return to the default level. This log level is equivalent to the syslog level “Debug.”

Log Level	Description
Trace	The Trace log level is recommended only for developers. Using this level causes a serious degradation of system performance and is not recommended. This log level is equivalent to the syslog level “Debug.”

Creating a Log Subscription in the GUI

Procedure

Step 1 Choose **System Administration > Log Subscriptions**.

Step 2 Click **Add Log Subscription**.

Step 3 Select a log type and enter the log name (for the log directory) as well as the name for the log file itself.

Step 4 [Only for Consolidated Event Logs] Select the required log fields to include in the log line for a given message.

Step 5 Specify the maximum file size before AsyncOS rolls over the log file as well as a time interval between rollovers. See Rolling Over Log Subscriptions, on page 63 for more information on rolling over log files.

Step 6 Select the log level. The available options are Critical, Warning, Information, Debug, or Trace.

Step 7 Configure the log retrieval method.

Step 8 Submit and commit your changes.

Editing Log Subscriptions

Procedure

Step 1 Choose **System Administration > Log Subscriptions**.

Step 2 Click the name of the log in the Log Settings column.

Step 3 Make changes to the log subscription.

Step 4 Submit and commit your changes.

Configuring Global Settings for Logging

The system periodically records system measurements within the Text Mail Logs and the Status Logs. Use the **Edit Settings** button in the Global Settings section of the **System Administration > Log Subscriptions** page (or the `logconfig -> setup` command in the CLI) to configure:

- System metrics frequency. This is the amount of time, in seconds, that the system waits between recording measurements.
- Whether to record the Message-ID headers.
- Whether to record the remote response status code.
- Whether to record the subject header of the original message.
- A list of headers that should be logged for each message.

All logs optionally include the following three pieces of data:

1. Message-ID

   When this option is configured, every message will have its Message ID header logged, if it is available. Note that this Message-ID may have come from the received message or may have been generated by AsyncOS itself. For example:

   ```
   Tue Apr 6 14:38:34 2004 Info: MID 1 Message-ID Message-ID-Content
   ```

2. Remote Response

   When this option is configured, every message will have its remote response status code logged, if it is available. For example:

   ```
   Tue Apr 6 14:38:34 2004 Info: MID 1 RID [0] Response 'queued as 9C8B425DA7'
   ```

   The remote response string is the human-readable text received after the response to the DATA command during the delivery SMTP conversation. In this example, the remote response after the connection host issued the data command is “queued as 9C8B425DA7.”

   ```
   [...]

   250 ok hostname

   250 Ok: queued as 9C8B425DA7
   ```

   Whitespace, punctuation, (and in the case of the 250 response, the OK characters) are stripped from the beginning of the string. Only whitespace is stripped from the end of the string. For example, appliances , by default, respond to the DATA command with this string: 250 Ok: Message MID accepted. So, the string “Message MID accepted” would be logged if the remote host were another appliance .

3. Original Subject Header

   When this option is enabled, the original subject header of each message is included in the log.

   ```
   Tue May 31 09:20:27 2005 Info: Start MID 2 ICID 2

   Tue May 31 09:20:27 2005 Info: MID 2 ICID 2 From: <mary@example.com>

   Tue May 31 09:20:27 2005 Info: MID 2 ICID 2 RID 0 To: <joe@example.com>

   Tue May 31 09:20:27 2005 Info: MID 2 Message-ID '<44e4n$2@example.com>'

   Tue May 31 09:20:27 2005 Info: MID 2 Subject 'Monthly Reports Due'
   ```

Logging Message Headers

In some cases, it is necessary to record the presence and contents of a message’s headers as they pass through the system. You specify the headers to record in the Log Subscriptions Global Settings page (or via the logconfig -> logheaders subcommand in the CLI). The appliance records the specified message headers in the Text Mail Logs, the Delivery Logs, and the Bounce Logs. If the header is present, the system records the name of the header and the value. If a header is not present, nothing is recorded in the logs.

Note The system evaluates all headers that are present on a message, at any time during the processing of the message for recording, regardless of the headers specified for logging.

The RFC for the SMTP protocol is located at http://www.faqs.org/rfcs/rfc2821.html and defines user-defined headers.

If you have configured headers to log via the logheaders command, the header information appears after the delivery information:

Table 34: Log Headers

Header name	Name of the header
Value	Contents of the logged header

For example, specifying “date, x-subject” as headers to be logged will cause the following line to appear in the mail log:

```
Tue May 31 10:14:12 2005 Info: Message done DCID 0 MID 3 to RID [0]
[('date', 'Tue, 31 May 2005 10:13:18 -0700'), ('x-subject', 'Logging this header')]
```

Configuring Global Settings for Logging Using the GUI

Procedure

Step 1 Choose **System Administration** > **Log Subscriptions**.

Step 2 Scroll down to the **Global Settings** section.

Step 3 Click **Edit Settings**.

Step 4 Specify information including the system measurement frequency, whether to include Message-ID headers in mail logs, whether to include the remote response, and whether to include the original subject header of each message.

Step 5 Enter any other headers you wish to include in the logs.

Step 6 Submit and commit your changes.

Rolling Over Log Subscriptions

To prevent log files on the appliance from becoming too large, AsyncOS performs a “rollover” and archives a log file when it reaches a user-specified maximum file size or time interval and creates a new file for incoming log data. Based on the retrieval method defined for the log subscription, the older log file is stored on the appliance for retrieval or delivered to an external computer. See Log Retrieval Methods, on page 8 for more information on how to retrieve log files from the appliance .

When AsyncOS rolls over a log file, it performs the following actions:

- Renames the current log file with the timestamp of the rollover and a letter "**s**" extension signifying saved.
- Creates a new log file and designates the file as current with the "**current**" extension.
- Transfers the newly saved log file to a remote host (if using the push-based retrieval method).
- Transfers any previously unsuccessful log files from the same subscription (if using the push-based retrieval method).
- Deletes the oldest file in the log subscription if the total number of files to keep on hand has been exceeded (if using the poll-based retrieval method).

  You define a log subscription's rollover settings when creating or editing the subscription using the **System Administration > Log Subscriptions** page in the GUI or the logconfig command in the CLI. The two settings available for triggering a log file rollover are:

- A maximum file size.
- A time interval.

Rollover By File Size

AsyncOS rolls over log files when they reach a maximum file size to prevent them from using too much disk space. When defining a maximum file size for rollovers, use the suffix m for megabytes and k for kilobytes. For example, enter 10m if you want AsyncOS to roll over the log file when it reaches 10 megabytes.

Rollover By Time

If you want to schedule rollovers to occur on a regular basis, you can select one of the following time intervals:

- **None.** AsyncOS only performs a rollover when the log file reaches the maximum file size.
- **Custom Time Interval.** AsyncOS performs a rollover after a specified amount of time has passed since the previous rollover. To create a custom time interval for scheduled rollovers, enter the number of days, hours, and minutes between rollovers using d , h , and m as suffixes.
- **Daily Rollover.** AsyncOS performs a rollover every day at a specified time. If you choose a daily rollover, enter the time of day you want AsyncOS to perform the rollover using the 24-hour format (HH:MM).

  Only the GUI offers the Daily Rollover option. If you want to configure a daily rollover using the logconfig command in the CLI, choose the Weekly Rollover option and use an asterisk (*) to specify that AsyncOS should perform the rollover on every day of the week.

- **Weekly Rollover.** AsyncOS performs a rollover on one or more days of the week at a specified time. For example, you can set up AsyncOS to rollover the log file every Wednesday and Friday at midnight. To configure a weekly rollover, choose the days of the week to perform the rollover and the time of day in the 24-hour format (HH:MM).

  If you are using the CLI, you can use a dash (-) to specify a range of days, an asterisk (*) to specify every day of the week, or a comma (,) to separate multiple days and times.

  The following table shows how to use the CLI to roll over the files for a log subscription on Wednesday and Friday at midnight (00:00).

Table 35: Weekly Log Rollover Settings in the CLI

`Do you want to configure time-based log files rollover? [N]> y`
`Configure log rollover settings:`

```
1. Custom time interval.
2. Weekly rollover.
[1]> 2
1. Monday
2. Tuesday
3. Wednesday
4. Thursday
5. Friday
6. Saturday
7. Sunday
Choose the day of week to roll over the log files.  Separate multiple days with comma,

or use "*" to specify every day of a week.
Also you can use dash to specify a range like "1-5":
[]> 3, 5
Enter the time of day to rollover log files in 24-hour format (HH:MM).
You can specify hour as "*" to match every hour, the same for minutes.
Separate multiple times of day with comma:
[]> 00:00
```

Rolling Over Log Subscriptions on Demand

To roll over log subscriptions immediately using the GUI:

Procedure

Step 1 On the System Administration > Log Subscriptions page, mark the checkbox to the right of the logs you wish to roll over.

Step 2 Optionally, you can select all logs for rollover by marking the All checkbox.

Step 3 Once one or more logs have been selected for rollover, the **Rollover Now** button is enabled. Click the **Rollover Now** button to roll over the selected logs.

Viewing Recent Log Entries in the GUI

Before You Begin

You must have the HTTP or HTTPS service enabled on the Management interface in order to view logs via the GUI.

Procedure

Step 1 Select **System Administration > Log Subscriptions**.

Step 2 Select the log subscription in the **Log Files** column of the table.

Step 3 Sign in.

Step 4 Select a log file to view it in your browser or to save it to disk.

Viewing Recent Log Entries in the CLI (tail Command)

AsyncOS supports a tail command, which shows the latest entries of configured logs on the appliance . Issue the tail command and select the number of a currently configured log to view it. Use Ctrl-C to exit from the tail command.

Example

In the following example, the tail command is used to view the system log. (This log tracks user comments from the commit command, among other things.) The tail command also accepts the name of a log to view as a parameter: tail mail_logs .

```
mail3.example.com> tail

Currently configured logs:


1. "antispam" Type: "Anti-Spam Logs" Retrieval: Manual Download

2. "antivirus" Type: "Anti-Virus Logs" Retrieval: Manual Download

3. "asarchive" Type: "Anti-Spam Archive" Retrieval: Manual Download

4. "authentication" Type: "Authentication Logs" Retrieval: Manual Download

5. "avarchive" Type: "Anti-Virus Archive" Retrieval: Manual Download

6. "bounces" Type: "Bounce Logs" Retrieval: Manual Download

7. "cli_logs" Type: "CLI Audit Logs" Retrieval: Manual Download

8. "encryption" Type: "Encryption Logs" Retrieval: Manual Download

9. "error_logs" Type: "IronPort Text Mail Logs" Retrieval: Manual Download
```

```
10. "euq_logs" Type: "IronPort Spam Quarantine Logs" Retrieval: Manual Download

11. "euqgui_logs" Type: "IronPort Spam Quarantine GUI Logs" Retrieval: Manual Download

12. "ftpd_logs" Type: "FTP Server Logs" Retrieval: Manual Download

13. "gui_logs" Type: "HTTP Logs" Retrieval: Manual Download

14. "mail_logs" Type: "IronPort Text Mail Logs" Retrieval: Manual Download

15. "reportd_logs" Type: "Reporting Logs" Retrieval: Manual Download

16. "reportqueryd_logs" Type: "Reporting Query Logs" Retrieval: Manual Download

17. "scanning" Type: "Scanning Logs" Retrieval: Manual Download

18. "slbld_logs" Type: "Safe/Block Lists Logs" Retrieval: Manual Download

19. "sntpd_logs" Type: "NTP logs" Retrieval: Manual Download

20. "status" Type: "Status Logs" Retrieval: Manual Download

21. "system_logs" Type: "System Logs" Retrieval: Manual Download

22. "trackerd_logs" Type: "Tracking Logs" Retrieval: Manual Download

23. "updater_logs" Type: "Updater Logs" Retrieval: Manual Download


Enter the number of the log you wish to tail.

[]> 19

Press Ctrl-C to stop.

Mon Feb 21 12:25:10 2011 Info: PID 274: User system commit changes: Automated Update for
Quarantine Delivery Host

Mon Feb 21 23:18:10 2011 Info: PID 19626: User admin commit changes:

Mon Feb 21 23:18:10 2011 Info: PID 274: User system commit changes: Updated filter logs
config

Mon Feb 21 23:46:06 2011 Info: PID 25696: User admin commit changes: Receiving
suspended.

^Cmail3.example.com>
```

Configuring Host Keys

Use the `logconfig -> hostkeyconfig` subcommand to manage host keys for use with SSH when pushing logs to other servers from the appliance . SSH servers must have a pair of host keys, one private and one public. The private host key resides on the SSH server and cannot be read by remote machines. The public host key is distributed to any client machine that needs to interact with the SSH server.

Note To manage user keys, see Managing Secure Shell (SSH) Keys.

The `hostkeyconfig` subcommand performs the following functions:

Table 36: Managing Host Keys - List of Subcommands

Command	Description
New	Add a new key.
Edit	Modify an existing key.
Delete	Delete an existing key.
Scan	Automatically download a host key.
Print	Display a key.
Host	Display system host keys. This is the value to place in the remote system's ‘known_hosts’ file.
Fingerprint	Display system host key fingerprints.
User	Display the public key of the system account that pushes the logs to the remote machine. This is the same key that is displayed when setting up an SCP push subscription. This is the value to place in the remote system's 'authorized_keys' file.

In the following example, AsyncOS scans for host keys and add them for the host:

```
mail3.example.com> logconfig

Currently configured logs:

[ list of logs ]

Choose the operation you want to perform:

- NEW - Create a new log.

- EDIT - Modify a log subscription.

- DELETE - Remove a log subscription.

- SETUP - General settings.

- LOGHEADERS - Configure headers to log.

- HOSTKEYCONFIG - Configure SSH host keys.

[]> hostkeyconfig

Currently installed host keys:

1. mail3.example.com ssh-dss [ key displayed ]

Choose the operation you want to perform:

- NEW - Add a new key.

- EDIT - Modify a key.

- DELETE - Remove a key.
```

```
- SCAN - Automatically download a host key.
- PRINT - Display a key.
- HOST - Display system host keys.
- FINGERPRINT - Display system host key fingerprints.
- USER - Display system user keys.
[]> scan
Please enter the host or IP address to lookup.
[]> mail3.example.com
Choose the ssh protocol type:
1. SSH1:rsa
2. SSH2:rsa
3. SSH2:dsa
4. All
[4]>
SSH2:dsa
mail3.example.com ssh-dss
[ key displayed ]
SSH2:rsa
mail3.example.com ssh-rsa
[ key displayed ]
SSH1:rsa
mail3.example.com 1024 35
[ key displayed ]

Add the preceding host key(s) for mail3.example.com? [Y]>
Currently installed host keys:
1. mail3.example.com ssh-dss [ key displayed ]
2. mail3.example.com ssh-rsa [ key displayed ]
3. mail3.example.com 1024 35 [ key displayed ]
Choose the operation you want to perform:
- NEW - Add a new key.
- EDIT - Modify a key.
- DELETE - Remove a key.
```

```
- SCAN - Automatically download a host key.

- PRINT - Display a key.

- HOST - Display system host keys.

- FINGERPRINT - Display system host key fingerprints.

- USER - Display system user keys.

[]>

Currently configured logs:

[ list of configured logs ]

Choose the operation you want to perform:

- NEW - Create a new log.

- EDIT - Modify a log subscription.

- DELETE - Remove a log subscription.

- SETUP - General settings.

- LOGHEADERS - Configure headers to log.

- HOSTKEYCONFIG - Configure SSH host keys.

[]>
```