Cisco Expressway Serviceability Guide

Maintain and Operate Guide

First Published: July 2015

Last Updated: June 2019

X8.11.1

Contents

Preface

Change History

Table 1 Maintain and Operate Guide Change History

Date	Change	Reason
September 2018	Updated software version from X8.11 to X8.11.1, as version X8.11 is no longer available.	Software withdrawn
July 2018	Content corrections.	Content defect
December 2016	Added information on Smart Call Home.	X8.9 release
November 2015	Added Call Detail Record information.	X8.7 release
July 2015	First published with System Metrics feature.	X8.6 release

Introducing System Metrics Collection

What is System Metrics Collection, and how does it work on Expressway?

System Metrics Collection is a feature on Expressway that publishes system performance statistics, enabling remote monitoring of performance.

The Expressway collects statistics about the performance of the hardware, OS, and the application, and publishes these statistics to a remote host (typically a data analytics server) that aggregates the data.

Where do I configure System Metrics Collection?

You can configure this feature on Expressway via the web interface or the command line. The configuration from one peer applies throughout the cluster, so we recommend that you configure it on the primary peer if you are monitoring a cluster.

There is also some configuration required on the remote server; the collectd daemon should be running on the server, and should have the collectd network plugin configured to listen on an address that can be seen by the clients. Further details depend on your monitoring environment and are beyond the scope of this information.

How can I use this data?

You can use the data to generate graphs, aggregate statistics, and analyze performance, using tools such as Circonus and Graphite.

Configure System Metrics Collection on Expressway

In the following procedure you'll use the web interface to configure the Expressway to collect statistics and publish them to a specified server. For more detailed descriptions of the options, see System Metrics Reference, page 13.

1. Log on to the Expressway and go to **Maintenance > Logging**.
2. Toggle **System Metrics Collection** to *On*.
3. Enter the **Collection server address**.

   You can use IP address, hostname or FQDN to identify the remote server.
4. Change the **Collection Interval** and **Collection server port** if necessary.

   You may need to change the port if the collection server is listening on a non-default port. You may need to change the collection interval if your policy requires finer-grained metrics than the default interval (*60*s).
5. Click **Save**.

Configure System Metrics on Remote Server

Selection and configuration of the server you choose for data analytics in your environment is beyond the scope of this document. Circonus and Graphite are applications that can handle collectd information.

Your analytics tool must support receiving data from the collectd daemon. This daemon is running on the Expressway and pushes the metrics to your analytics server, using the collectd network plugin.

The network plugin implements the collectd binary protocol for data encapsulation. The analytics server must be able to parse and present this data. Your analytics server will probably have its own UI for configuring how it collects and shows the data, which could be based on collectd or an alternative software.

If you are using collectd on the analytics server, you need to modify *collectd.conf* file so that the server:

- listens for data from the collectd clients (eg. Expressway); you need to enable the network plugin and configure the listen block with the server's IP address. For example:

  ```
  <Plugin "network">
        Listen "198.51.100.15"
  </Plugin>
  ```
- stores the data it receives in a human readable form (eg. to CSV files); you need to enable the csv plugin tell it where to write the files. For example:

  ```
  <Plugin "csv">
        DataDir "/var/lib/collectd/csv"
        StoreRates true
  </Plugin>
  ```

See also

- https://collectd.org/wiki/index.php/Networking_introduction
- https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin_network
- https://collectd.org/wiki/index.php/Binary_protocol
- https://collectd.org/wiki/index.php/Plugin:CSV
- https://collectd.org/documentation/manpages/collectd.conf.5.shtml#plugin_csv

Troubleshooting System Metrics

Is the Expressway sending data?

Take a TCP dump from the Expressway and check for packets sent to the address of your data analytics server:

Go to **Maintenance > Diagnostics > Diagnostics logging**, check the box labeled **Take tcpdump while logging**, and then start logging.

Call Detail Records (CDRs)

The system can capture CDRs if you enable the service (which is off by default), and can publish them as syslog messages if you are using remote logging.

If you select *Service only* the system keeps the CDRs for 7 days, and these CDRs can only be read via the Representational State Transfer (REST) API to the Expressway. If you select *Service and logging*, the local data is exposed in the Event Log, and the CDRs are also sent as INFO messages to your syslog host.

How to Configure CDRs

To configure CDRs on Expressway:

1. Go to **Maintenance > Logging**.
2. In the **Logging Options** section, set the **Call Detail Records** field following the below guide.

CDR Mode	Description
Off	CDRs are not logged locally (default).
Service Only	CDRs are stored locally for 7 days and then deleted. The records are not accessible via the web GUI.
Services and Logging	CDRs are stored locally for 7 days and then deleted. The records are accessible from the local event log and the external syslog server if external logging has been enabled.

APIs to access CDRs

You can use the following secure REST APIs to gather the information you require.

- get_all_records (returns all records up to seven days old).
- get_records_for_interval (returns records from during the time specified).
- get_records_for_filter (filters results using any combination).
- get_all_csv_records (returns all records up to seven days old in csv format).

To access your desired API use the following URL: https://<Expressway_IP>/api/external/callusage/<API>

Examples

- http://<Expressway_IP>/api/external/callusage/get_all_records
- http://<Expressway_IP>/api/external/callusage/get_records_for_interval?fromtime=<fromtime>&totime=<to_time> (for example https://203.0.113.17/api/external/callusage/get_records_for_interval?fromtime=2014-05-09 2000:00:00&totime=2014-05-10 2000:00:00)

  #### Input Parameters

Parameter	Description
fromtime	The start time from which the CDR records are required. It must be entered in the format 'YYYY-MM-DD HH:MI:SS' (mandatory parameter).
totime	The end time from which the CDR records are required. It must be entered in the format 'YYYY-MM-DD HH:MI:SS' (mandatory parameter).

- http://<Expressway_IP>/api/external/callusage/get_records_for_interval?fromtime=<fromtime>&totime=<to_time> (for example https://203.0.113.17/api/external/callusage/get_records_for_interval?fromtime=2014-05-09 2000:00:00&totime=2014-05-10 2000:00:00)
- http://<Expressway_IP>/api/external/callusage/get_records_for_filter?uuid=<uuid>&src_alias=<src_alias>&dest_alias=<dest_alias>&protocol=<protocol> (for example https://203.0.113.17/api/external/callusage/get_records_for_filter?uuid=6e3b5a8a-346c-421b-aa2e-f4409c43a81a&src_alias=TC149-057-h323@domain.com&dest_alias=TC149-065-h323@domain.com&protocol=H323 <-> H323)

  Input Parameters

Parameter	Description
uuid	The unique identifier of the record.
src_alias	The origin point of the call.
dest_ alias	The destination point of the call
protocol	The protocol that was used for the call (SIP, H323 etc).

- http://<Expressway_IP>/api/external/callusage/get_all_csv_records

Limitations

- The call history is only stored locally for seven days and deleted automatically.

For sample CDRs and property definitions, see the Example Call Detail Records, page 11 in the Additional Information section.

Smart Call Home (SCH)

About Smart Call Home

Smart Call Home is an embedded support capability for Expressway. It offers proactive diagnostics and real-time alerts, enabling higher network availability and increased operational efficiency.

Smart Call Home notifies users of Schedule- and Event-based notifications.

- Schedule-based: inventory, telemetry and configuration messages used to generate a Device Report and improve hardware and software quality by identifying failure trends. You can find these notifications posted on the first day of every month.
- Event-based: ad hoc events already supported by Expressway such as alarms and ACRs. You will find these notifications posted to the Smart Call Home server as and when they occur.

How to Configure Smart Call Home

1. Go to **Maintenance > Serviceability > Smart Call Home**.
2. In the **Configuration** section, select your preferred mode in the **Smart Call Home** field.

Mode	Definition
Off (Default)	The Expressway does not send information to the Smart Call Home server.
On	Turn on the Smart Call Home service. The **Replace Smart Call Home server certificate** and **Your Contact Details** options appear.
On (Anonymous)	Turn on the Smart Call Home Service in Anonymous mode. The **Replace Smart Call Home server certificate** option appears. The Expressway still sends reports to Smart Call Home, but the customer details will be kept anonymous so that Smart Call Home server will not be able to send notifications.

Note: The Expressway already has the Smart Call Home server certificate installed, so it can communicate securely with the Smart Call Home server. You do not need to replace it unless the Smart Call Home servers update their certificates.

3. If you choose to turn Smart Call Home *On*, you must enter an email address as a minimum requirement in the **Your Contact Details** section.

4. Modify the outbound transport settings for sending Smart Call Home data, if necessary:

   The default setting is to send the SCH data directly over HTTPS to the SCH servers without using a proxy. Use the table to select the right settings for your environment:

Field	Options
Transport mode	Select *HTTPS* to make external HTTPS connections. Then you can configure a proxy if you need one for HTTPS connections. Select *SMTP* to use a mail server to relay the data instead. Then you can enter details of the mail server (Address, username, password).
Proxy type	If using an HTTPS proxy, you can choose to use *This Expressway*'s forward proxy or to use an *External proxy*. You could use a different Expressway as an external proxy. In that case, enter the Expressway's address and use 8445 in the port field.
Authentication	If using an external HTTPS proxy: select *Mutual TLS* to use X.509 certificates for mutual authentication or *Basic* to enter a username and password.

5. Click **Save**.

Reference

Example Call Detail Records

Sample CDR

[{"initial_call": "false", "protocol": "SIP <-> SIP", "protocol_summary": "", "disconnect_reason": "200 OK", "licensed": "false", "tag": "b8d52a60-16a1-4bdb-be93-f5a675408811", "aside_request_uri": "", "box_call_serial_number": "22cd0e7d-c498-4068-9239-624038fe5130", "source_alias": "sip:10000005@10.196.4.82", "uuid": "800fe013-83f4-4094-a5e6-e2f9489912e2", "last_updated_timestamp": 1444725389, "details": "{\"Call\":{\"SerialNumber\": \"800fe013-83f4-4094-a5e6-e2f9489912e2\",\"BoxSerialNumber\": \"22cd0e7d-c498-4068-9239-624038fe5130\",\"Tag\": \"b8d52a60-16a1-4bdb-be93-f5a675408811\",\"State\": \"Disconnected\",\"StartTime\": \"2015-10-13 01:36:26.485636\",\"InitialCall\": \"False\",\"Licensed\": \"False\",\"LicensedAsTraversal\": \"False\",\"SourceAlias\": \"sip:10000005@10.196.4.82\",\"DestinationAlias\": \"sip:10000010@cucm-82\",\"ToLocalB2BUA\": \"False\",\"Audio\": \"False\",\"License\":{\"Traversal\": \"0\",\"NonTraversal\": \"0\",\"DemotedTraversal\": \"0\",\"CollaborationEdge\": \"0\",\"Cloud\": \"0\"},\"Duration\": \"3\",\"Legs\": [{\"Leg\":{\"Protocol\": \"SIP\",\"SIP\":{\"Address\": \"10.196.4.61:5073\",\"Transport\": \"TLS\",\"Aliases\": [{\"Alias\":{\"Type\": \"Url\",\"Origin\": \"Unknown\",\"Value\": \"sip:10000005@10.196.4.82\"}}]},\"Targets\": [{\"Target\":{\"Type\": \"Url\",\"Origin\": \"Unknown\",\"Value\": \"sip:10000010@10.196.4.116\"}}],\"BandwidthNode\": \"DefaultZone\",\"EncryptionType\": \"AES\",\"Cause\": \"200\",\"Reason\": \"OK\"}},{\"Leg\":{\"Protocol\": \"SIP\",\"SIP\":{\"Address\": \"10.196.4.71:7001\",\"Transport\": \"TLS\",\"Aliases\":[{\"Alias\":{\"Type\": \"Url\",\"Origin\": \"Unknown\",\"Value\": \"sip:10000010@cucm-82\"}}]},\"Source\":{\"Aliases\":[{\"Alias\":{\"Type\": \"Url\",\"Origin\": \"Unknown\",\"Value\": \"10000005@10.196.4.82\"}}]},\"BandwidthNode\": \"Traversal-zone\",\"EncryptionType\": \"AES\",\"Cause\": \"200\",\"Reason\": \"OK\"}}],\"Sessions\":[{\"Session\":{\"Status\": \"Completed\",\"MediaRouted\": \"False\",\"CallRouted\": \"True\",\"Participants\":{\"Leg\": \"1\",\"Leg\": \"2\",\"Incoming\":{\"Leg\": \"1\"},\"Outgoing\":{\"Leg\": \"2\"}}}}],\"EndTime\": \"2015-10-13 01:36:29.745651\"}}", "status": "Disconnected", "destination_alias": "sip:10000010@cucm-82", "licensed_as_traversal": "false", "service_uuid": "e6723fd0-5ca2-11e1-b86c-0800200c9a66", "start_time": "2015-10-13 01:36:26.485636", "traversal_license_tokens": 0, "bside_destination_alias": "", "active": "false", "media_routed": "false", "aside_destination_alias": "", "non_traversal_license_tokens": 0, "bside_request_uri": "", "end_time": "2015-10-13 01:36:29.745651", "audio": "false"}]

Note: The above sample CDR applies to all APIs with the exception of csv.

Sample csv CDR

uuid,service_uuid,active,initial_call,licensed,licensed_as_traversal,status,tag,box_call_serial_number,start_time,end_time,source_alias,destination_alias,aside_destination_alias,bside_destination_alias,aside_request_uri,bside_request_uri,protocol_summary,protocol,media_routed,audio,traversal_license_tokens,non_traversal_license_tokens,disconnect_reason,details,last_updated_timestamp

800fe013-83f4-4094-a5e6-e2f9489912e2,e6723fd0-5ca2-11e1-b86c-0800200c9a66,false,false,false,false,Disconnected,b8d52a60-16a1-4bdb-be93-f5a675408811,22cd0e7d-c498-4068-9239-624038fe5130,2015-10-13 01:36:26.485636,2015-10-13 01:36:29.745651,sip:10000005@10.196.4.82,sip:10000010@cucm-82,,,,,,SIP <-> SIP,false,false,0,0,200 OK,"{""Call"":{""SerialNumber"": ""800fe013-83f4-4094-a5e6-e2f9489912e2"",""BoxSerialNumber"": ""22cd0e7d-c498-4068-9239-624038fe5130"",""Tag"": ""b8d52a60-16a1-4bdb-be93-f5a675408811"",""State"": ""Disconnected"",""StartTime"": ""2015-10-13 01:36:26.485636"",""InitialCall"": ""False"",""Licensed"": ""False"",""LicensedAsTraversal"": ""False"",""SourceAlias"": ""sip:10000005@10.196.4.82"",""DestinationAlias"": ""sip:10000010@cucm-82"",""ToLocalB2BUA"": ""False"",""Audio"": ""False"",""License"":{""Traversal"": ""0"",""NonTraversal"": ""0"",""DemotedTraversal"": ""0"",""CollaborationEdge"": ""0"",""Cloud"": ""0""},""Duration"": ""3"",""Legs"":[{""Leg"":{""Protocol"": ""SIP"",""SIP"":{""Address"": ""10.196.4.61:5073"",""Transport"": ""TLS"",""Aliases"":[{""Alias"":{""Type"": ""Url"",""Origin"": ""Unknown"",""Value"": ""sip:10000005@10.196.4.82""}}]},""Targets"":[{""Target"":{""Type"": ""Url"",""Origin"": ""Unknown"",""Value"": ""sip:10000010@10.196.4.116""}}],""BandwidthNode"": ""DefaultZone"",""EncryptionType"": ""AES"",""Cause"": ""200"",""Reason"": ""OK""}},{""Leg"":{""Protocol"": ""SIP"",""SIP"":{""Address"": ""10.196.4.71:7001"",""Transport"": ""TLS"",""Aliases"":[{""Alias"":{""Type"": ""Url"",""Origin"":

""Unknown"",""Value"": ""sip:10000010@cucm-82""}}]},""Source"":{""Aliases"":[{""Alias"":{""Type"":
""Url"",""Origin"": ""Unknown"",""Value"": ""10000005@10.196.4.82""}}]},""BandwidthNode"": ""Traversal-
zone"",""EncryptionType"": ""AES"",""Cause"": ""200"",""Reason"": ""OK""}}],""Sessions"":[{""Session"":{""Status"":
""Completed"",""MediaRouted"": ""False"",""CallRouted"": ""True"",""Participants"":{""Leg"": ""1"",""Leg"":
""2"",""Incoming"":{""Leg"": ""1""},""Outgoing"":{""Leg"": ""2""}}}}],""EndTime"": ""2015-10-13 01:36:29.745651""}}
",1444725389

Definitions

The below table defines the properties that are visible in the CDRs.

Field	Definition
uuid	This is the ID of the CDR entry.
service_uuid	The ID used to identify whether a record is from a proxy, Lync B2BUA or Encryption B2BUA.
active	Details whether a call is a live or a historical one.
initial_call	Used internally to tie to a B2BUA call when it is a multiple-component one (involves a B2BUA hop).
licensed	This field shows you if a call used a license.
licensed_as_ traversal	This field shows you if a call used a traversal license.
status	A 200 OK message will signal that a call was successful. This field will contain an error message if the call was unsuccessful.
tag	The call ID.
box_call_serial_ number	An extra ID added to tie multiple calls together (e.g. through B2BUA).
start_time	This field shows the date and time of the call. The time zone can be set in **System > Times > Time Zone** and the date format is YYYY-MM-DD.
end_time	This field shows the end time of the call.
source_alias	This field shows the alias of the caller.
destination_alias	This field shows the alias of the callee.
aside_destination_ alias	The alias of the caller (or MS Lync client if Lync Interop).
bside_destination_ alias	This alias of the callee (or non-Lync client).
aside_request_uri	The request uri of the caller (or MS Lync client if Lync Interop).
bside_request_uri	The request uri of the callee (or non-Lync client).
protocol	This field shows if the call was SIP <-> SIP, SIP <-> H323, H323 <-> SIP, or H323 <-> H323.
protocol_summary	This field is as above but can have extra info like if a call was multi-component, DVO, etc.
media_routed	This field shows if media was sent during the call (e.g. NAT/IWF/B2BUA).
audio	This field shows if the call was an audio-only one.
traversal_license_ tokens	This field indicates if a call fork/branch took media (audio equates to 1 token and video 2).*

non_traversal_ license_tokens	This field indicates if a call fork/branch did not need to take media (audio equates to 1 token and video 2).*
disconnect_reason	This field gives reasons for a call drop such as normal call teardown or other errors i.e. last status.
details	This field gives more details of the call, including media statistics.
last_updated_ timestamp	Shows the last time that any of the above fields were updated.

* Once a call is set up only one of these entries will have a non-zero value (i.e. only for the answered fork/branch).

System Metrics Reference

What are the configuration options on the Expressway?

Table 2 Configuration commands for collectd on Expressway

What the command does	Web UI location	Example CLI command
Toggle Metrics Collection on/off	**Maintenance > Logging > System Metrics Collection**	`xconfig log SystemMetrics mode: on`
Specify the server address	**Maintenance > Logging > Collection server address**	`xconfig log SystemMetrics network address: address`
Specify the listening port	**Maintenance > Logging > Collection server port**	`xconfig log SystemMetrics network port: 25826`
Specify the collection interval	**Maintenance > Logging > Collection Interval**	`xconfig log SystemMetrics interval: 60`
Read System Metrics configuration	**Maintenance > Logging**	`xstatus SystemMetrics`

What metrics are collected from the Expressway?

The following hardware statistics are monitored:

- aggregation-cpu-sum
- aggregation-cpu-average
- Per-core CPU usage for each core in the system
- df
- disk
- load
- protocols-Tcp
- protocols-Udp
- swap
- Users
- memory
- Uptime
- Process

The following application data are monitored by the custom **exec-app** plugin for collectd:

- **gauge-active_alarms** is the count of active alarms on this Expressway
- **gauge-active_calls** is the count of calls being handled by this Expressway
- **gauge-*<service name>*** is the status of each system service.
- **gauge-*<zone name>*_ActiveCalls** counts the active calls in the named zone
- **gauge-*<zone name>*_BandwidthAllocated** measures the total bandwidth allocated to the named zone
- **gauge-*<zone name>*_BandwidthLimit**

Each of these metrics uses the collectd GAUGE data source type, which allows free-form data. On the collection server, the full collectd value name will be shown, for example **collectd*Hostname*collectd.exec-app.gauge-active_calls**.

Note that zone names are user-configurable and may thus be in conflict with the naming schema for collectd metrics. If your collection server is enforcing the schema, there is a chance that metrics from some zones will not be accepted.

What data is sent to the collection server?

The network plugin uses the collectd binary protocol to encapsulate numeric, string, and value data representing the monitored hardware resources and software processes.

The network plugin pushes the metrics data packets to the analytics server once every interval, using UDP 25826 by default. The analytics server parses and presents the data in human readable form.

If the analytics server is using the collectd network plugin and csv plugin, then the metrics are stored as small CSV files, using the metric name and timestamp to create the filename, for example **gauge-H323-2015-05-21**.

Which collectd plugins are implemented on Expressway?

Table 3 collectd plugins implemented in the Expressway application

Plugin name	Description / more information
Aggregation	Aggregates CPU values into the counters **aggregation_cpu_sum** and **aggregation_cpu_average**.
CPU	Processor information. The raw information is aggregated into **aggregation_cpu_average** and **aggregation_cpu_sum**
DF	File system information; see DF description on collectd Wiki
Disk	Hard disk performance; see Disk description on collectd Wiki
Exec-app	Customized version of **exec** that returns specific Expressway information on calls, alarms, zones, and services
Load	System load based on task queue
Memory	Memory statistics
Network	Enables publishing to a remote address. The plugin implements the collectd binary protocol for data encapsulation. The remote server must have the appropriate parsing tool
Protocols	Configurable subset of the protocols used by the Expressway

Table 3 collectd plugins implemented in the Expressway application (continued)

Plugin name	Description / more information
Process	Counts the system processes and groups them by state (e. g. running, sleeping, zombies) It also collects detailed statistics about specific processes. The plugin monitors the following processes in detail: ■ app ■ bramble ■ credentialmanagerservermain ■ cvs_main ■ erlang-beam ■ erlang-epmd ■ httpd ■ httpserver ■ ivy ■ licensemanagerservermain ■ managementconnectormain ■ managementframework ■ openssl2nss ■ policyservermain ■ syslog-ng ■ XCP
Swap	The amount of system memory written to disk
Uptime	Tracks system uptime, providing counters like average running time or maximum uptime for a particular period; see Uptime description on collectd Wiki
Users	Count of currently logged in users

Cisco Legal Information

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies are considered un-Controlled copies and the original on-line version should be referred to for latest version.

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco Trademark

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)