

Cisco 7200 Parity Error Fault Tree

Document ID: 12763

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Network Processing Engine (NPE) Parity Error Fault Tree Analysis

NPE Parity Error Detection and Messages

- Parity Errors in the NPE-300
- NPE-400 Parity/ECC Detection
- Parity Errors in the C7200 Router

Solutions

Related Information

Introduction

This document explains the steps to troubleshoot and isolate which part or component of a Cisco 7200 is failing when you identify a variety of parity error messages. We recommend that you read *Troubleshooting Router Crashes and Processor Memory Parity Errors (PMPEs)* before you proceed with this document.

Note: The information in this document is based on the Cisco 7200 Series Routers.

Prerequisites

Requirements

There are no specific prerequisites for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you work in a live network, ensure that you understand the potential impact of any command before you use it.

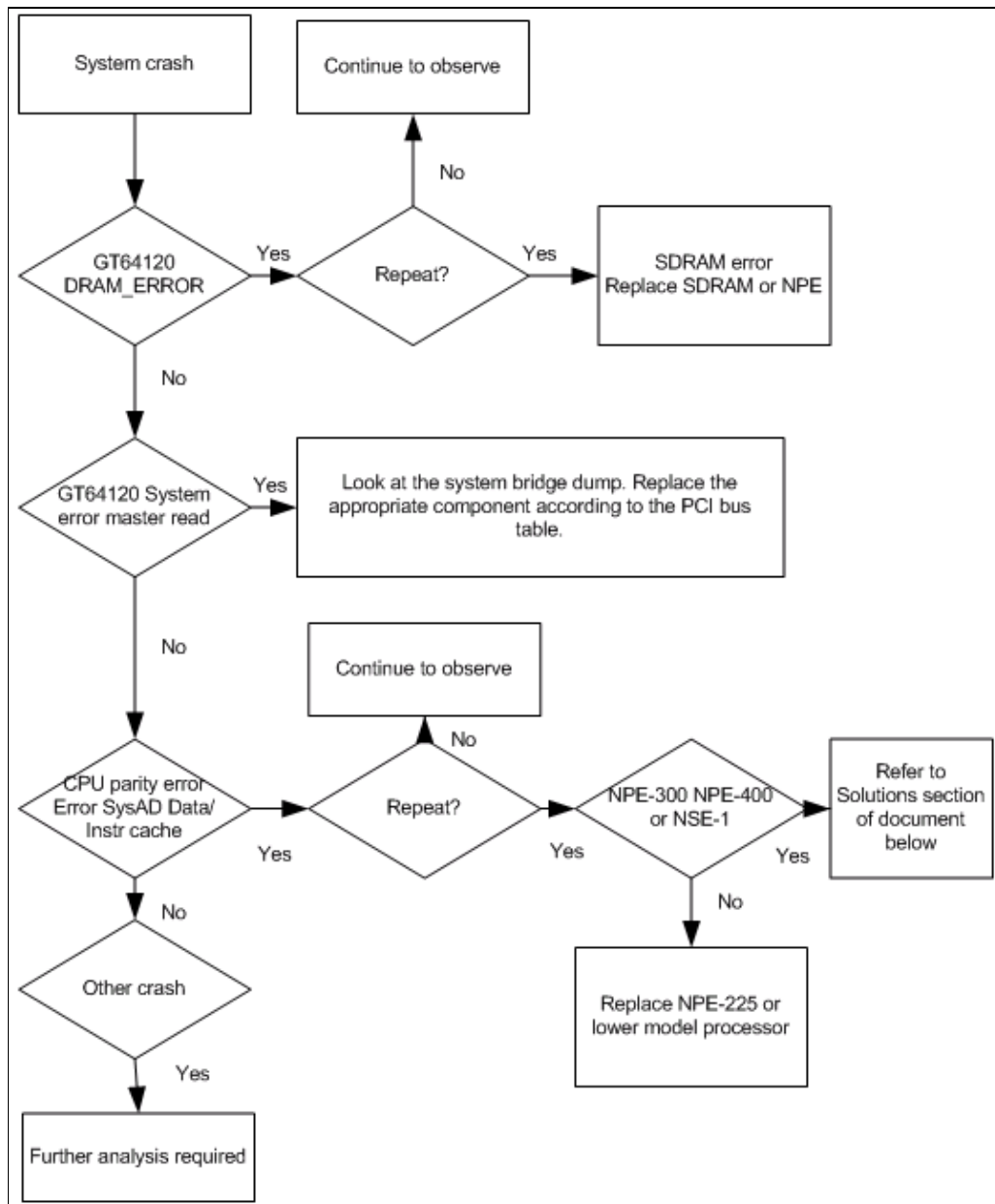
Conventions

For more information on document conventions, refer to the *Cisco Technical Tips Conventions*.

Network Processing Engine (NPE) Parity Error Fault Tree Analysis

This diagram describes the steps to determine which part or component of a Cisco 7200 is failing when you

identify a variety of parity error messages.



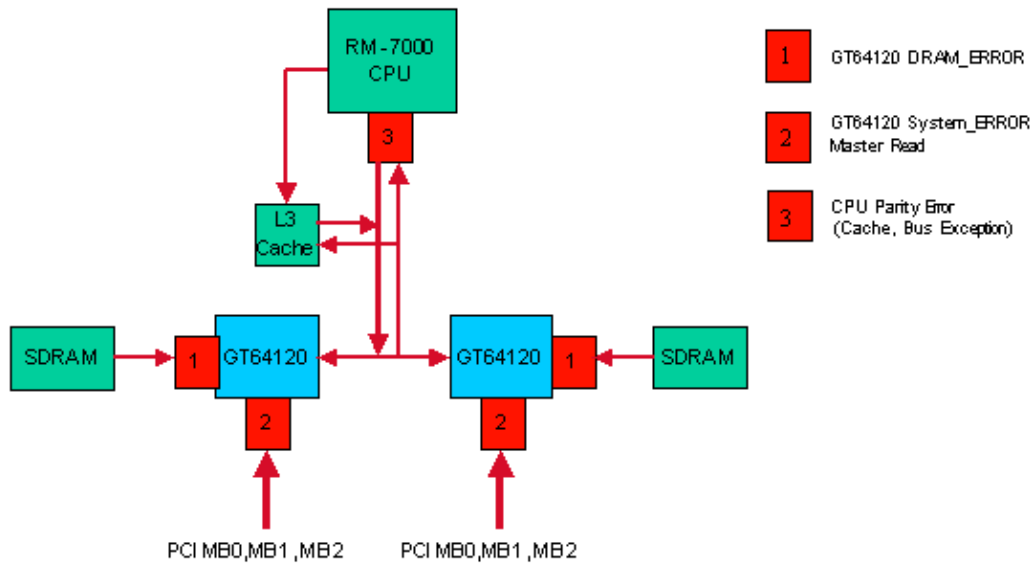
Note: Capture and record the **show tech-support** output and console logs, and collect all crashinfo files during parity error events.

NPE Parity Error Detection and Messages

This section contains block diagrams of the NPE and where these systems detect parity errors. You can find a description of each type of error message below.

Parity Errors in the NPE-300

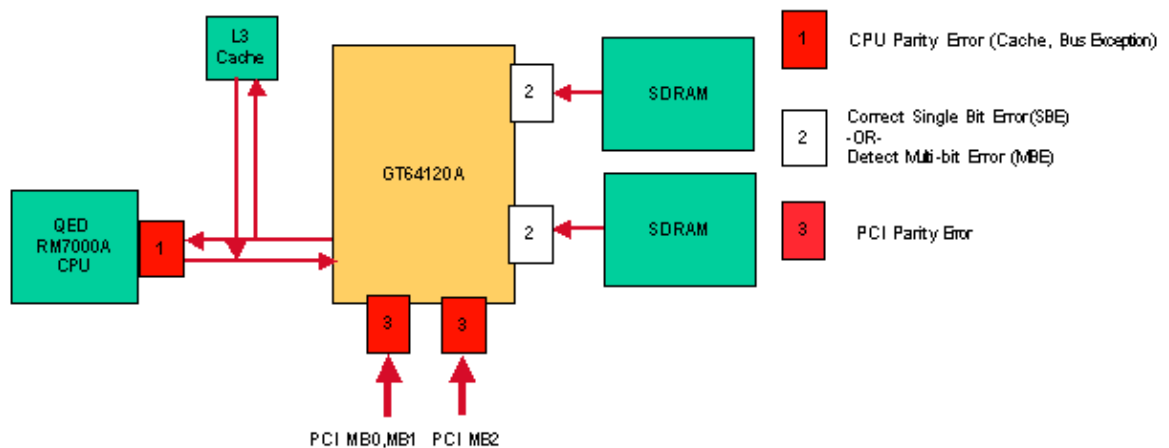
The NPE-300 has parity detection on the paths and boxes marked in red below.



The NPE-300 uses parity checking in shared memory (SDRAM), PCI Bus, and the CPU's external interface to protect the system from malfunctioning by bit errors. Parity checking is capable of detecting a single bit error by using a simple method; adding one check bit per eight bits of data. If it detects a bit error when passing the data between hardware components, the system discards the erroneous data. Single bit errors at any location in the diagram above cause the router to reset.

NPE-400 Parity/ECC Detection

The red boxes and lines indicate paths on which the NPE-400 has parity detection; the white boxes indicate detection and correction of single bit ECC errors.



The NPE-400 uses Single Bit Error Correction and Multi-bit Error Detection ECC (Error Code Correction) for shared memory (SDRAM). To increase system availability in the NPE-400, ECC corrects single bit errors in SDRAM, to allow the system to operate normally without resetting and without down time. For more information on how ECC enhances system availability, refer to the Increasing Network Availability page.

A multi-bit error in SDRAM causes the router to reset with a cache error exception or bus error. The rest of the memory and buses in the system use single bit parity detection. Single bit errors at 1 and 3 in the diagram above cause the router to reset.

Parity Errors in the C7200 Router

Several of the parity checking devices on the C7200/NPE router can report data with bad parity for any read or write operation. Here is a description of the various error messages reported on a C7200/NPE system:

GT64010/GT64120 DRAM Error

This error is reported when a GT64120 system controller detects a parity error when reading SDRAM:

```
%ERR-1-GT64120 (PCI0):Fatal error, Memory parity error (external)
GT=0xB4000000, cause=0x0100E283, mask=0x0ED01F00, real_cause=0x00000200
Bus_err_high=0x00000000, bus_err_low=0x00000000,
addr_decode_err=0x1C000000
```

Replace the SDRAM after a second failure. If the failure persists, replace the NPE.

Note: For older NPEs (NPE-100/150/200) which use the GT64010 controllers, the error looks like this:

```
%ERR-1-GT64010: Fatal error, Memory parity error (external)
cause=0x0300E283, mask=0x0CD01F00, real_cause=0x00000200
bus_err_high=0x00000000, bus_err_low=0x00000000, addr_decode_err=0x00000000
```

The GT64010 controller uses Dynamic RAM (DRAM) and not SDRAM. In this case, replace the DRAM after a second failure. If the failure persists, replace the NPE.

GT64010/GT64120 System Parity Error Master Read

A parity error in Master Read is a parity error triggered by accessing a Peripheral Component Interconnect (PCI) bridge. Here's an example of parity error output:

```
%ERR-1-GT64120 (PCI0):Fatal error, Parity error on master read
GT=B4000000, cause=0x0110E083, mask=0x0ED01F00, real_cause=0x00100000
Bus_err_high=0x00000000, bus_err_low=0x00000000,
addr_decode_err=0x00000470

%ERR-1-SERR: PCI bus system/parity error
%ERR-1-FATAL: Fatal error interrupt, No reloading
Err_stat=0x81, err_enable=0xFF, mgmt_event=0x40
```

Replace the appropriate component after a second failure. The system bridge dump indicates which component to replace.

System bridge dump:

```
Bridge 1, for PA bay 1, 3 and 5. Handle=1
DEC21150 bridge chip, config=0x0

(0x1C):sec status, io base =0x83A09141
           Detected Parity Error on secondary bus
           Data Parity Detected on secondary bus
(0x20):mem base & limit =0x4AF04880
```

These tables tell you which component has a possible problem from the error message output.

NPE–100/150/200:

Bridge number	What the bridge is for	Parity Error on Primary Bus	Parity Error on Secondary Bus
Bridge 0	Downstream MB0 to MB1 0	Replace the NPE	Replace NPE; if still present, replace chassis
Bridge 1	Upstream MB1 to MB0	Replace NPE; if still present, replace chassis	Replace the NPE
Bridge 2	Downstream MB0 to MB2	Replace the NPE	Replace NPE; if still present, replace chassis
Bridge 3	Upstream MB2 to MB0	Replace NPE; if still present, replace chassis	Replace the NPE

NPE–175/225/300/400/NSE–1:

Bridge number	What the bridge is for	Parity Error on Primary Bus	Parity Error on Secondary Bus
Bridge 0	For PA bay 0 (I/O card, PCMCIA, interfaces)	Replace the NPE	Replace NPE; if still present, replace I/O card. If still present, replace chassis
Bridge 1	For PA bay 1, 3, and 5	Replace the NPE	Replace the NPE; if still present, replace chassis
Bridge 2	For PA bay 2, 4, and 6	Replace the NPE	Replace NPE; if still present, replace chassis

All C7200s:

Bridge number	What the bridge is for	Parity Error on Primary Bus	Parity Error on Secondary Bus
Bridge 4	Port Adapter 1	Replace NPE; if still present, replace chassis	Replace PA 1; if still present, replace chassis
Bridge 5	Port Adapter 2	Replace NPE; if still present, replace chassis	Replace PA 2; if still present, replace chassis
Bridge 6	Port Adapter 3	Replace NPE; if still present,	Replace PA 3; if still present,

		replace chassis	replace chassis
Bridge 7	Port Adapter 4	Replace NPE; if still present, replace chassis	Replace PA 4; if still present, replace chassis
Bridge 8	Port Adapter 5	Replace NPE; if still present, replace chassis	Replace PA 5; if still present, replace chassis
Bridge 9	Port Adapter 6	Replace NPE; if still present, replace chassis	Replace PA 6; if still present, replace chassis

CPU Parity Error

As with all computer and networking devices, the NPE is susceptible to the rare occurrence of parity errors in processor memory. Parity errors may cause the system to reset and can be a transient Single Event Upset (SEU or soft error) or can occur multiple times (often referred to as hard errors) due to damaged hardware. For more information on SEUs, refer to the Increasing Network Availability page. A CPU parity error is reported if the CPU detects a parity error when accessing any of the processor's caches (L1, L2, or if fitted, L3).

Here are four examples of this type of error:

Example 1:

```
Error: SysAD, data cache, fields: data, 1st dword
Physical addr(21:3) 0x195BE88,
Virtual address is imprecise.
Imprecise Data Parity Error
Imprecise Data Parity Error
```

The NPE has an R7K processor with non-blocking cache. Non-blocking cache means when it executes an instruction to load data into a register and this data is not in the L1 cache, the CPU loads the data from a lower order cache or from SDRAM data. The CPU does not block execution of further instructions unless there is another cache miss or another instruction depends upon the data being loaded. This can greatly speed up the processor and improve performance, but can also lead to parity errors being imprecise. An imprecise parity error is when the CPU reads information without blocking, and later determines there was a parity error in the associated cache line. The R7K processor is unable to tell us specifically which instruction was being executed when the cache line was being loaded, and that is the reason we call it an imprecise parity error.

Even if systems use Error Code Correction (ECC), it is still possible to see an occasional parity error when more than a single error has occurred in the 64 bits of data due to a hard error in the cache.

A parity error occurs when a signal bit value is changed from its original value (0 or 1) to the opposite value. This error can occur either due to a soft or hard parity error.

Soft parity errors occur because of an external influence on the memory of the device, which changes the bit value at the current level. This type of problem is transient and does not reoccur. Hard parity errors occur when the bit value is changed by the memory itself because of damage to the memory. In that case, the problem occurs every time that area of memory is used, which means that the problem can repeat multiple times within a couple days to a week.

Example 2:

```
Error: SysAD, instr cache, fields: data, 1st dword
Physical addr(21:3) 0x000000,
virtual addr 0x6040BF60, vAddr(14:12) 0x3000
virtual address corresponds to main:text, cache word 0
```

```
          Low Data      High Data  Par  Low Data      High Data  Par
L1 Data:  0:0xAE620068 0x8C830000 0x00 1:0x50400001 0xAC600004 0x01
          2:0xAC800000 0x00000000 0x02 3:0x1600000B 0x00000000 0x01
          Low Data      High Data  Par  Low Data      High Data  Par
DRAM Data: 0:0xAE620068 0x8C830000 0x00 1:0x50400001 0xAC600004 0x01
          2:0xAC800000 0x00000000 0x02 3:0x1600000B 0x00000000 0x01
```

Example 3:

```
Cache Err Reg = 0xE4588D10
Data reference, Secondary/Sys intf cache, Data field error
Error on 1st doubleword on System interface
No errors in addition to instr error
Data phy addr that caused last parity or bus error: 0x1E84040C
```

Example 4 (NPE-300 and NPE-400 only):

```
%CERF-3-RECOVER: PC=0x604F136C, Origin=L3 Data ,PhysAddr=0x013CEFD0
```

or

```
%SYS-2-CERF_ABORT: Reason=0xEE23, PC=0x604629C8, Origin=L3 Data,
Phys Addr=0x0287A4E8
```

Both messages above are accompanied by a "Cache Error Recovery Function (CERF) report" as follows:

```
CERFa[1 ] 05:25:36 MET Tue Jul 9 2002: result=0xEE23; instr_pos=-2; rpl_off=1
CERFb[1 ] PC =604629C8; ORGN=L3 Data; PRID=00002710; PHYA=0287A4E8
CERFc[1 ] SREG=3400E105; CAUS=00000400; DEA0=0287A4E8; ECC =00000000
CERFd[1 ] CERR=E447A4EA; EPC =606361F8; DEA1=02517058; INFO=00000000
CERFe[1 ] CACHE=28FF78B4 62B36D98 02020684 00000E17 00000030 00000001 61F2934C 3EDA025D
CERFe[1 ] SDRAM=28FF78B4 62B36D98 02020684 00000E17 00000030 00000001 61F2934C 3EDA025D
CERFg[1 ] CXT =00000000; XCXT=00000000; BVAD=00000008; PFCL=00000000
CERFh[1 ] ISeq: 0045182B; 1060000E; 2C4203E9; 92430028; 38420001; 30630005
CERFi[1 ] o0 $3 ....; beq....; sltiu $2 ....; lbu $3, 0x0028($18); xori $2....; andi $3 ..
CERFj[1 ]           ;           ;           ;           6287A4E8;           ;
CERFk[1 ] ResumptionCode= 0x92430028; 0x0000000F; 0x42000018
CERFl[1 ] Instr's checked=4; diags=0x00000158,0x00040000,3600,1,0
CERFm[1 ] BaseRegLost later/off: 0/0 times; StoredValueLost: 0 times
CERFn[1 ] INFO=00000000; CNFG=5061F4BB; ICTL=00000000
```

Initial Register Values

```
CERFs00[1 ] $0=00000000 AT=61A30000 v0=00000001 v1=00000002
CERFs04[1 ] a0=28FF8728 a1=00003A98 a2=00000000 a3=00000007
CERFs08[1 ] t0=00000000 t1=3400E101 t2=606381E0 t3=FFFF00FF
CERFs12[1 ] t4=606381C8 t5=000005D4 t6=00000008 t7=61C50000
CERFs16[1 ] s0=6189C188 s1=00000000 s2=6287A4C0 s3=00003A98
CERFs20[1 ] s4=61BD57B0 s5=00000006 s6=00000000 s7=61BD6C60
CERFs24[1 ] t8=60634788 t9=00000000 k0=621A8374 k1=6063EA40
CERFs28[1 ] gp=61A33B20 sp=61E28678 s8=00000000 ra=60462CA4
```

```
1 Cache error exceptions already reported
```

You see the above logs if CERF is enabled on an NPE-300 or NPE-400 and a parity error occurs. For more information about CERF, refer to the Solutions section below.

Solutions

The following course of action is recommended when you encounter such errors:

1. Monitor the affected hardware to see if the same problem happens again. If it does not, then it was a transient Single Event Upset (SEU) and you do not need to take any action.
2. In the unlikely event that the problem does reoccur, the **cache L3 bypass/disable** command is an option that may help reduce the impact of the issue. This command is only available on the following platforms:

- ◆ 7200 with processor engine NPE-300, NPE-400, or NSE-1
- ◆ 7400 with processor engine NSE-1

Because the NPE-300 does not support ECC memory, this feature is especially important to increase system availability and handle these parity errors without service interruption. This resolves many soft parity errors. The caveat is that there is a slight performance hit to the system when L3 cache is disabled. The performance degradation is anywhere between 1% and 10% depending upon the system configuration. The syntax for using this command is dependent on the Cisco IOS software version.

- ◆ The **cache L3 disable** command can be found in Cisco IOS Software Releases 12.3(5a) and later. It will also be available in 12.1(22)E. In these versions, L3 cache is disabled by default, so no action is needed to take advantage of this feature. L3 cache can be reenabled with the command **no cache L3 disable**.
- ◆ The **cache L3 bypass** command can be found in Cisco IOS Software Releases 12.2(6)S, 12.2(6)B, 12.2(8)BC1b, 12.0(20)SP, 12.2(6)PB, 12.2(2)DD2, 12.0(20)ST3, 12.0(21)S, 12.1(11)EC, 12.2(7)T, 12.1(13), and 12.2(7) or later, and 12.1(11)E through 12.1(21)E. This command is disabled by default.

To enable L3 cache bypass, enter the following from configuration mode:

```
Router(config)#cache L3 bypass
```

To disable L3 cache bypass, enter the following from configuration mode:

```
Router(config)#no cache L3 bypass
```

The new cache setting does not take effect until the router is reloaded.

When the router boots up, system information is displayed, including information about the L3 cache. This is because the startup-config file has not yet been processed by the system. After the startup-config file is processed, the L3 cache is bypassed if the **cache L3 bypass** command is in the configuration.

To verify the L3 cache setting, you can issue the **show version** command. If the L3 cache is bypassed, there is no reference to the L3 cache in the **show version** output.

3. Another feature that helps increase system availability is the Cache Error Recovery Function (CERF). When this feature is enabled (this is the default in the latest Cisco IOS software releases, but as of February 2004, only for NPE-300 and NPE-400), the Cisco IOS software makes an attempt to resolve the parity error and keep the processor from crashing. This feature resolves around 75% of certain types of soft parity errors. By invoking this command, the system sees less than 5% performance degradation.

CERF for the NPE-300 can be found in Cisco IOS Software Releases 12.1(15), 12.1(12)EC, 12.0(22)S, 12.2(10)S, 12.2(10)T, 12.2(10), 12.2(2)XB4, 12.2(11)BC1b, and 12.1(5)XM8 or later.

CERF for the NPE–400 can be found in 12.3(3)B, 12.2(14)S3, 12.1(20)E, 12.1(19)E1, 12.3(1a), 12.2(13)T5, 12.2(18)S, 12.3(2)T, 12.2(18), 12.3(3), and 12.3(1)B1 or later.

CERF for the NPE–300 requires hardware revision 4.1 or higher. In order to identify the hardware version of your NPE–300, use the **show c7200** command.

```
Router>show c7200
...
C7206VXR CPU EEPROM:
Hardware revision 4.1 Board revision A0
...
```

CERF for the NPE–400 requires processor R7K revision 2.1 or higher. In order to identify the processor revision of your NPE–400, use the **show version** command.

```
Router>show version
...
cisco 7206VXR (NPE400) processor with 491520K/32768K bytes of memory.
R7000 CPU at 350Mhz, Implementation 39, Rev 3.2, 256KB L2, 4096KB L3
Cache 6 slot VXR midplane, Version 2.1
...
```

Note: It is important to collect all relevant crashinfo files in order to determine the root cause of the error as explained in Retrieving Information from the Crashinfo File.

If the suggestions above do not resolve the issue, then replacing the NPE may help in cases of repeated occurrences of parity errors since hard parity errors are due to damaged hardware. Hardware replacements are identical to the original NPE. Replacing the NPE does not guarantee that no further parity errors will occur since Single Event Upsets (SEUs) are inherent in any computer equipment with memory.

Related Information

- [Troubleshooting Router Crashes](#)
- [Processor Memory Parity Errors \(PMPEs\)](#)
- [Technical Support – Cisco Systems](#)

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Apr 13, 2009

Document ID: 12763
