Debug Commands

The commands in this section are for troubleshooting the GGSN. For information about other debug commands, see the *Cisco IOS Debug Command Reference*.

Caution Because debugging output is assigned high priority in the CPU process, it can diminish the performance of the router or even render it unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

This chapter contains the following section and commands:

TID/IMSI/MSISDN-Based Conditionally Triggered Debugging

When the TID/IMSI/MSISDN-based conditionally triggered debugging feature is enabled, the GGSN generates debugging messages for PDP contexts that match a particular tunnel ID (TID), International Mobile Subscriber Identity (IMSI) value, or Mobile Station ISDN number (MSISDN) entering or leaving the GGSN. The GGSN will not generate debugging output for PDP contexts containing a different TID, IMSI, or MSISDN value.

Normally, the GGSN will generate debugging messages for every PDP context, resulting in a large number of messages that consume system resources and can make it difficult to find the specific information you need. By limiting the number of debugging messages, you can receive messages related to only to PDP contexts you want to troubleshoot.

Usage Guidelines for TID/IMSI/MSISDN-Based Conditional Debugging

Use the following guidelines when configuring TID/IMSI/MSISDN-based conditional debugging on a GGSN.

1. Before enabling a **debug gprs** command, first enable TID/IMSI/MSISDN-based debugging using the **debug condition calling** command. Ensure that the TID/IMSI or MSISDN string match the ones from the Create Request.

   For examples:

   For a create request with TID 12345678090000B0, you would enter:

   ```
   GGSN# debug condition calling 12345678090000B0
   Condition 1 set
   GGSN#
   ```

   For a create request with IMSI 21436579000000, you would enter:

   ```
   GGSN# debug condition calling 21436579000000
   Condition 2 set
   GGSN#
   ```

   For a create request with MSISDN 1112223344, you would enter:

   ```
   GGSN# debug condition calling msisdn-1112223344
   Condition 3 set
   GGSN#
   ```

   To verify the set conditions, enter:

   ```
   GGSN# show debug condition all
   Condition 1: calling 12345678090000B0 (0 flags triggered)
   Condition 2: calling 21436579000000 (0 flags triggered)
   Condition 3: calling 1112223344 (0 flags triggered)
   GGSN#
   ```

2. After turning on TID, IMSI, or MSISDN-based debugging, turn on GPRS debugging by entering the **debug gprs gtp** and/or **debug gprs charging** commands.

   Once this step is completed, when PDP Context Create Requests are received, the GGSN will display debug messages for those create requests with either a matching TID, IMSI, or MSISDN.

3. Because the **no debug all** command does not disable conditional debug flags, to ensure that you do not receive a flood of debugging messages when disabling debugging, turn off GPRS debug flags first using the **no debug all** command as follows:

```
GGSN# no debug all
All possible debugging has been turned off
GGSN#

GGSN# show debug condition all
Condition 1: calling 12345678090000B0 (1 flags triggered)
Condition 2: calling 21436579000000 (1 flags triggered)
Condition 3: calling 1112223344 (1 flags triggered)

GGSN#
```

4. Disable the conditional debug flags using the **no debug condition all** command:

```
GGSN# no debug condition all
Removing all conditions may cause a flood of debugging messages to result, unless
specified debugging flags are first removed.

Proceed with the removal of all conditions [yes/no] y
2 conditions have been removed
```

5. Verify that the conditional debug flags have been removed using the **show debug condition all** command:

```
GGSN# show debug condition all
% No conditions found
```

debug aaa coa

To display debug information for CoA processing, use the **debug aaa coa** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug aaa coa

no debug aaa coa

Syntax Description

This command has no keywords or arguments

Defaults

Debugging for POD packets is not enabled.

Command History

Release	Modification
12.4(15)XQ	This command was introduced.
12.4(24)T	This command was integrated into Cisco IOS Release 12.4(24)T.

Usage Guidelines

Use the **debug aaa coa** to display debug information for CoA processing.

Examples

The following is an example of debug information for CoA processing:

```
SAMI 5/3:  *Mar 4 23:51:02.820: COA: 10.10.10.10 request queued
SAMI 5/3:  *Mar 4 23:51:02.820: ++++++ CoA Attribute List ++++++
SAMI 5/3:  *Mar 4 23:51:02.820: 410414A8 0 00000009 string-session-id(337) 15
080808012521869
SAMI 5/3: *Mar 4 23:51:02.820: 4189D04C 0 00000009 qos-profile(507) 28
25621F9301FEFE245E1414003200
SAMI 5/3: *Mar 4 23:51:02.820:
SAMI 5/3: *Mar 4 23:51:02.820: COA: Sending ACK from port 1700 to 10.10.10.10/1700
```

debug condition calling

To limit output for some debug commands based on specified conditions, use the **debug condition** command in privileged EXEC mode. To remove the specified condition, use the **no** form of this command.

debug condition {**username** *username* | **called** *dial-string* | **caller** *dial-string* | **vcid** *vc-id* | **ip** *ip-address* | **calling** [*tid* | *imsi* | **msisdn**-*msisdn*]}

no debug condition {*condition-id* | **all**}

Syntax Description

username *username*	Generates debugging messages for interfaces with the specified username.
called *dial-string*	Generates debugging messages for interfaces with the called party number.
caller *dial-string*	Generates debugging messages for interfaces with the calling party number.
vcid *vc-id*	Generates debugging messages for the VC ID specified.
ip *ip-address*	Generates debugging messages for the IP address specified.
calling [*tid* | *imsi string* | **msisdn**-*msisdn* **]**	Displays events related to GTP processing on the GGSN based on tunnel identifier (TID), international mobile system identifier (IMSI), or Mobile Station ISDN number (MSISDN) in a PDP Context Create Request message.
condition-id	Removes the condition indicated.
all	Removes all conditional debugging conditions.

Defaults

No default behavior or values.

Command History

Release	Modification
12.3(2)XB	This command was introduced on the GGSN.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into the Cisco IOS Release 12.3(14)YU and the **msisdn** keyword option was added.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

Ensure that you enable TID/IMSI/MSISDN-based conditional debugging using the **debug condition calling** command before configuring the **debug gprs gtp** and **debug gprs charging**. In addition, ensure that you disable the **debug gprs gtp** and **debug gprs charging** commands using the **no debug all** command before disabling conditional debugging using the **no debug condition** command. This will prevent a flood of debug messages when you disable conditional debugging.

For more information on using the GGSN TID/IMSI/MSISDN-based conditional debugging, see "TID/IMSI/MSISDN-Based Conditionally Triggered Debugging" section on page 610.

Examples

Example 1

The following examples configure a conditional debug session based on a TID 12345678090000B0, IMSI 21436579000000, and MSISDN 408525823010:

```
GGSN# debug condition calling 12345678090000B0
Condition 1 set
GGSN#

GGSN# debug condition calling 21436579000000
Condition 2 set
GGSN#

GGSN# debug condition calling msisdn 408525823010
Condition 3 set
GGSN#
```

Example 2

The following example stops all conditional debugging:

```
Router# no debug conditional all
All possible debugging has been turned off
Router#
```

debug data-store

To display persistant storage device (PSD)-related debugging messages for the gateway GPRS support node (GGSN), use the **debug data-store** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug data-store

no debug data-store

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(14)YU	This command was introduced.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.
12.4(15)T	This command was integrated into Cisco IOS Release 12.4(15)T.

Usage Guidelines

This command displays PSD-related debugging messages for the GGSN.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network flows and fewer users. Debugging during these periods reduces the effect these commands have on other users on the system.

Examples

The following example configures a debugging session to check PSD-related parameters:

```
Router# debug data-store
```

debug data-store detail

To display extended details for persistent storage device (PSD)-related debugging information, use the **debug data-store detail** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug data-store detail

no debug data-store detail

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(14)YU	This command was introduced.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.
12.4(15)T	This command was integrated into Cisco IOS Release 12.4(15)T.

Usage Guidelines

This command displays PSD-related debugging messages for the GGSN.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network flows and fewer users. Debugging during these periods reduces the effect these commands have on other users on the system.

Examples

The following example configures a detailed PSD-related debugging session:

```
Router# debug data-store details
```

Related Commands

Command	Description
auto-retrieve	Configures the GGSN to automatically initiate a retrieval of G-CDRs from PSDs defined in a PSD server group.
clear data-store statistics	Clears PSD-related statistics.
show data-store	Displays the status of the PSD client and PSD server-related information.
show data-store statistics	Displays statistics related to the PSD client.

debug diameter

To display information about Diameter processing on the gateway GPRS support node (GGSN), use the **debug diameter** command in privilege EXEC mode.

debug diameter {dcca | connection | error | packet | event | fsm | failover | all}

Syntax Description

dcca	Displays Diameter Credit Control Application-related information.
connection	Displays Diameter peer connection information.
error	Displays errors related to Diameter processing.
packet	Displays Diameter packets.
event	Displays Diameter-related events.
fsm	Displays Diameter-related fault state machine messages.
failover	Displays information about DCCA server failovers.
all	Displays all Diameter-related information.

Defaults

No default behavior or values.

Command Modes

Privilege EXEC

Command History

Release	Modification
12.3(14)YQ	This command was introduced.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with Diameter processing.

Examples

The following configuration example displays Diameter-related events:

```
debug diameter event
```

debug ggsn quota-server

To display debug information related to quota server processing on the gateway GPRS support node (GGSN), use the **debug ggsn quota-server** command in privilege EXEC mode.

debug ggsn quota-server [details | packets [dump] | events | parsing | errors]

Syntax Description

details	Displays extended details about quota server operations on the GGSN.
packets	Displays packets sent between the quota server process on the GGSN and the CSG. Optionally, displays output in hexadecimal notation.
events	Displays events related to quota server processing on the GGSN.
parsing	Displays details about GTP TLV parsing between the quota server and the Content Services Gateway.
errors	Displays errors related to quota server processing on the GGSN

Defaults

No default behavior or values.

Command Modes

Privilege EXEC

Command History

Release	Modification
12.3(14)YQ	This command was introduced.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between the GGSN quota server process and the CSG.

Examples

The following example enables the display of events related to quota server processing on the GGSN:

```
Router# debug ggsn quota-server events
```

The following example enables the display of packets sent between the quota server process on the GGSN and the CSG:

```
Router# debug ggsn quota-server packets
```

The following example enables the display of detailed quota server processing debug output:

```
Router# debug ggsn quota-server details
```

debug gprs category fsm event

To display debug information related to service-aware gateway GPRS support node (GGSN) category events, and state transactions, use the **debug gprs category fsm event** command in privilege EXEC mode.

debug gprs category fsm event

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

Privilege EXEC

Command History

Release	Modification
12.3(14)YQ	This command was introduced.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with eGGSN processing.

Examples

The following example enables the display of eGGSN events and state transactions:

```
Router# debug ggsn eggsn category fsm event
```

debug gprs dcca

To display troubleshooting information about DCCA processing on the gateway GPRS support node (GGSN), use the **debug gprs dcca** command in privilege EXEC mode.

debug gprs dcca

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

Privilege EXEC

Command History

Release	Modification
12.3(14)YQ	This command was introduced.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if Diameter protocol problems are encountered on the GGSN.

Examples

The following configuration example displays information specific to DCCA processing:

```
debug gprs dcca
```

debug gprs dfp

To display debug messages for GPRS DFP weight calculation, use the **debug gprs dfp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs dfp

no debug gprs dfp

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command History

Release	Modification
12.1(9)E	This command was introduced.
12.2(4)MX	This command was integrated into Cisco IOS Release 12.2(4)MX.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

See the following caution before using **debug** commands:

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network flows and fewer users. Debugging during these periods reduces the effect these commands have on other users on the system.

This command displays debug messages for GPRS DFP weight calculation. To display debug messages for the DFP agent subsystem, use the **debug ip dfp agent** command.

Examples

The following example configures a debug session to check all GPRS DFP weight calculation:

```
Router# debug gprs dfp
GPRS DFP debugging is on
Router#
```

The following example stops all debugging:

```
Router# no debug all
All possible debugging has been turned off
Router#
```

debug gprs dhcp

To display information about Dynamic Host Configuration Protocol (DHCP) processing on the gateway GPRS support node (GGSN), use the **debug gprs dhcp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs dhcp

no debug gprs dhcp

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command History

Release	Modification
12.2(4)MX	This command was introduced.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with DHCP processing on the GGSN. To display standard debug messages between the DHCP client on the router and a DHCP server, you can also use the **debug dhcp** or **debug dhcp detail** commands with the **debug gprs dhcp** command.

Caution Because the **debug gprs dhcp** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following example shows sample output for DHCP processing on the GGSN:

```
Router# debug gprs dhcp
2d13h: GPRS:DHCP req:TID 1111111100000099, Req 1
2d13h: GPRS:Requesting IP address for pdp 1111111100000099 from server 172.16.0.8 tableid
0
2d13h: GPRS:DHCP ip allocation pass (10.88.17.43) for pdp 1111111100000099
2d13h: GPRS:Using DHCP ip address 10.88.17.43 for pdp 1111111100000099
```

The following example shows sample output for standard debug messaging for DHCP processing on the router between the DHCP client and a DHCP server:

```
2d13h: DHCP: proxy allocate request
2d13h: DHCP: new entry. add to queue
2d13h: DHCP: SDiscover attempt # 1 for entry:
2d13h: DHCP: SDiscover: sending 283 byte length DHCP packet
2d13h: DHCP: SDiscover with directed serv 172.16.0.8, 283 bytes
2d13h: DHCP: XID MATCH in dhcpc_for_us()
2d13h: DHCP: Received a BOOTREP pkt
2d13h: DHCP: offer received from 172.16.0.8
2d13h: DHCP: SRequest attempt # 1 for entry:
2d13h: DHCP: SRequest- Server ID option: 172.16.0.8
2d13h: DHCP: SRequest- Requested IP addr option: 10.88.17.43
2d13h: DHCP: SRequest placed lease len option: 604800
2d13h: DHCP: SRequest: 301 bytes
2d13h: DHCP: SRequest: 301 bytes
2d13h: DHCP: XID MATCH in dhcpc_for_us()
2d13h: DHCP: Received a BOOTREP pkt
2d13h: DHCP Proxy Client Pooling: ***Allocated IP address: 10.88.17.43
```

Related Commands

Command	Description
debug dhcp	Displays debug messages between the DHCP client on the router and a DHCP server.

debug gprs gtp

To display information about the GPRS Tunneling Protocol (GTP), use the **debug gprs gtp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs gtp {**events** | **messages** | **packets**}

no debug gprs gtp {**events** | **messages** | **packets**}

Syntax Description

events	Displays events related to GTP processing on the GGSN.
messages	Displays GTP signaling messages that are sent between the SGSN and GGSN.
packets	Displays GTP packets that are sent between the SGSN and GGSN.

Defaults

No default behavior or values.

Command History

Release	Modification
12.1(1)GA	This command was introduced.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
12.2(4)MX	This command was integrated into Cisco IOS Release 12.2(4)MX, and the **ppp** {**details**	**events**} option was added.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between the GGSN and the SGSN using GTP.

Caution Because the **debug gprs gtp** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following example enables the display of events related to GTP processing on the GGSN:

```
Router# debug gprs gtp events
```

The following example enables the display of GTP signaling messages:

```
Router# debug gprs gtp messages
```

The following example enables the display of GTP packets sent between the SGSN and GGSN:

```
Router# debug gprs gtp packets
```

The following example enables the display of GTP PPP events between the SGSN and GGSN:

```
Router# debug gprs gtp ppp events
```

The following example enables the display of detailed GTP PPP debug output along with GTP PPP events between the SGSN and GGSN:

```
Router# debug gprs gtp ppp details
Router# debug gprs gtp ppp events
```

debug gprs gtp parsing

To display information about the parsing of GPRS Tunneling Protocol (GTP) information elements (IEs) in signaling requests, use the **debug gprs gtp parsing** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs gtp parsing

no debug gprs gtp parsing

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command History

Release	Modification
12.2(4)MX	This command was introduced.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers to verify parsing of GTP IEs in signaling requests that are received by GDM or by the GGSN. If the packet is parsed successfully, you will receive a message along with the TID for the packet as shown in the following example:

```
GPRS:TID:7300000000000000:Packet Parsed successfully
```

The **debug gprs gtp parsing** command can be used to verify GDM or GGSN processing of IEs.

Caution Because the **debug gprs gtp parsing** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following example enables the display of debug messages that occur while GDM or the GGSN parses GTP IEs:

```
Router# debug gprs gtp parsing
```

debug gprs gtp ppp

To display information about PPP PDP type processing on the gateway GPRS support node (GGSN), use the **debug gprs gtp ppp** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs gtp ppp {**events** | **details**}

no debug gprs gtp ppp {**events** | **details**}

Syntax Description

events	Displays messages specific to certain conditions that are occurring during PPP PDP type processing.
details	Displays more extensive and lower-level messages related to PPP PDP type processing.

Defaults

No default behavior or values.

Command History

Release	Modification
12.2(4)MX	This command was introduced.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with PPP PDP type processing on the GGSN.

You can enable both forms of the **debug gprs gtp ppp** command at the same time, as separate command line entries. The **events** keyword generates output specific to certain conditions that are occurring, which helps qualify the output being received using the **details** option.

Caution Because the **debug gprs gtp ppp** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following debug examples provide sample output for a Create PDP Context request and clear PDP context using PPP PDP type on the GGSN. The examples show output while both debug events and details are enabled on the GGSN.

Example 1

The following example displays details and events output related to PPP PDP context processing for a Create PDP Context requested received by the GGSN:

```
Router# debug gprs gtp ppp events
GTP PPP events display debugging is on
Router# debug gprs gtp ppp details
GTP PPP details display debugging is on
7200b#
3d23h: GPRS:
3d23h: GTP-PPP Fa1/0: Create new gtp_ppp_info
3d23h: GPRS:
3d23h: GTP-PPP: domain gprs.cisco.com not in any VPDN group
3d23h: GPRS:
3d23h: GTP-PPP: aaa-group accounting not configured under APN gprs.cisco.com
3d23h: GPRS:GTP-PPP: Don't cache internally generated pak's header
3d23h: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to up
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_cstate_react changing states
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:
3d23h: GTP-PPP: Vi2: Concat names user00 & gprs.cisco.com
3d23h: GPRS:
3d23h: GTP-PPP: New username after concat: user00@gprs.cisco.com
3d23h: GPRS:
3d23h: GTP-PPP: Vi2: Concat names user00@gprs.cisco.com & gprs.cisco.com
3d23h: GPRS:
3d23h: GTP-PPP: New username after concat: user00@gprs.cisco.com
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2, changed state to
up
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_protocol_up is notified about intf UP
3d23h: GPRS:
3d23h: GTP-PPP Vi2: PDP w/ MS addr 98.102.0.1 inserted into IP radix tree
```

Example 2

The following example displays both details and events related to PPP PDP type processing after clearing PDP contexts on the GGSN:

```
Router# clear gprs gtp pdp-context all
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:GTP-PPP: pdp_entry 0x62F442A4, recv ppp data pak
3d23h: GPRS:GTP-PPP Vi2: proc_udp_input pak's linktype = 30
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_pdp_terminate shutting down the vaccess
```

```
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_pdp_shut_va shutting down intf
3d23h: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to down
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_cstate_react changing states
3d23h: GPRS:
3d23h: GTP-PPP Vi2: gtp_ppp_free_va resetting intf vectors
3d23h: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2, changed state to
down
```

debug gprs gtp ppp-regeneration

To display information about PPP regeneration processing on the GGSN, use the **debug gprs gtp ppp-regeneration** privileged EXEC command. To disable debugging output, use the **no** form of this command.

debug gprs gtp ppp-regeneration {**events** | **details**}

no debug gprs gtp ppp-regeneration {**events** | **details**}

Syntax Description

events	Displays messages specific to certain conditions that are occurring during PPP regeneration processing.
details	Displays more extensive and lower-level messages related to PPP regeneration processing.

Defaults

No default behavior or values.

Command History

Release	Modification
12.2(4)MX	This command was introduced.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between GDM and a GGSN.

You can enable both forms of the **debug gprs gtp ppp-regeneration** command at the same time, as separate command line entries. The **events** keyword generates output specific to certain conditions that are occurring, which helps qualify the output being received using the **details** option.

Caution Because the **debug gprs gtp ppp-regeneration** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following debug examples provide sample output for a create PDP context request and clear PDP context using PPP regeneration on the GGSN. The examples show output while both debug events and details are enabled on the GGSN.

Example 1

The following example displays details and events output related to PPP regeneration processing for a create PDP context requested received by the GGSN:

```
Router# debug gprs gtp ppp-regeneration details
GTP PPP regeneration details display debugging is on
Router# debug gprs gtp ppp-regeneration events
GTP PPP regeneration events display debugging is on
06:24:02: PPP-REGEN state counters: pending counter is 0
06:24:02:               State[IDLE] counter is 0
06:24:02:               State[AUTHORIZING] counter is 0
06:24:02:               State[VPDN CONNECTING] counter is 0
06:24:02:               State[PPP NEGOTIATING] counter is 0
06:24:02:               State[PPP CONNECTED] counter is 0
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: PPP-REGEN state counters: pending counter is 1
06:24:02:               State[IDLE] counter is 1
06:24:02:               State[AUTHORIZING] counter is 0
06:24:02:               State[VPDN CONNECTING] counter is 0
06:24:02:               State[PPP NEGOTIATING] counter is 0
06:24:02:               State[PPP CONNECTED] counter is 0
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: GPRS:1011111111500001:Authen: PAP username: tomy1@corporate_1.com
06:24:02: GPRS:1011111111500001:Session timer started
06:24:02: GPRS:Processing PPP regen reqQ
06:24:02: GPRS:1011111111500001:Processing Initiate PPP  regen from reqQ
06:24:02: GPRS:1011111111500001:got event [REQUEST PPP REGEN] in state [IDLE]
06:24:02: PPP-REGEN state counters: pending counter is 1
06:24:02:               State[IDLE] counter is 0
06:24:02:               State[AUTHORIZING] counter is 1
06:24:02:               State[VPDN CONNECTING] counter is 0
06:24:02:               State[PPP NEGOTIATING] counter is 0
06:24:02:               State[PPP CONNECTED] counter is 0
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: GPRS:1011111111500001:state [IDLE->AUTHORIZING] on event [REQUEST PPP REGEN]
06:24:02: GPRS:1011111111500001:Got VPN authorization info
06:24:02: GPRS:1011111111500001:got event [AUTHOR SUCCESS] in state [AUTHORIZING]
06:24:02: PPP-REGEN state counters: pending counter is 1
06:24:02:               State[IDLE] counter is 0
06:24:02:               State[AUTHORIZING] counter is 0
06:24:02:               State[VPDN CONNECTING] counter is 1
06:24:02:               State[PPP NEGOTIATING] counter is 0
06:24:02:               State[PPP CONNECTED] counter is 0
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: GPRS:1011111111500001:state [AUTHORIZING->VPDN CONNECTING] on event [AUTHOR
SUCCESS]
06:24:02: GPRS:1011111111500001:Author succeeded, establishing the tunnel
06:24:02: GPRS:1011111111500001:Create/Clone vaccess to negotiate PPP
06:24:02: GPRS:1011111111500001:no need to set NS ppp_config
06:24:02: GPRS:1011111111500001:MS no static IP addr. Get one via IPCP
06:24:02: GPRS:1011111111500001:VPDN to inform PPP regen: CONNECTED
06:24:02: GPRS:1011111111500001:got event [VPDN CONNECTED] in state [VPDN CONNECTING]
06:24:02: PPP-REGEN state counters: pending counter is 1
06:24:02:               State[IDLE] counter is 0
06:24:02:               State[AUTHORIZING] counter is 0
06:24:02:               State[VPDN CONNECTING] counter is 0
06:24:02:               State[PPP NEGOTIATING] counter is 1
06:24:02:               State[PPP CONNECTED] counter is 0
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: GPRS:1011111111500001:state [VPDN CONNECTING->PPP NEGOTIATING] on event [VPDN
CONNECTED]
06:24:02: GPRS:1011111111500001:Start PPP negotiations on vaccess
06:24:02: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to up
```

```
06:24:02: GPRS:1011111111500001:IPCP is up
06:24:02: GPRS:1011111111500001:LNS allocates 10.100.1.1 for MS
06:24:02: GPRS:1011111111500001:IP addr 10.100.1.1 is negotiated for MS
06:24:02: GPRS:1011111111500001:PPP connected
06:24:02: GPRS:1011111111500001:got event [PPP NEGOTIATED] in state [PPP NEGOTIATING]
06:24:02: PPP-REGEN state counters: pending counter is 0
06:24:02:               State[IDLE] counter is 0
06:24:02:               State[AUTHORIZING] counter is 0
06:24:02:               State[VPDN CONNECTING] counter is 0
06:24:02:               State[PPP NEGOTIATING] counter is 0
06:24:02:               State[PPP CONNECTED] counter is 1
06:24:02:               State[PPP TERMINATING] counter is 0
06:24:02: GPRS:1011111111500001:state [PPP NEGOTIATING->PPP CONNECTED] on event [PPP
NEGOTIATED]
06:24:02: GPRS:1011111111500001:PPP succeeded negotiation, session established
06:24:02: GPRS:1011111111500001:Session timer stopped
06:24:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2, changed state
to up
```

Example 2

The following example displays both details and events related to PPP regeneration processing after clearing PDP contexts on the GGSN:

```
Router# clear gprs gtp pdp-context all
06:28:05: PPP-REGEN state counters: pending counter is 0
06:28:05:               State[IDLE] counter is 0
06:28:05:               State[AUTHORIZING] counter is 0
06:28:05:               State[VPDN CONNECTING] counter is 0
06:28:05:               State[PPP NEGOTIATING] counter is 0
06:28:05:               State[PPP CONNECTED] counter is 1
06:28:05:               State[PPP TERMINATING] counter is 0
06:28:05: GPRS:1011111111500001:PPP regen current state PPP CONNECTED
06:28:05: GPRS:1011111111500001:GTP disconnecting the PPP regen session
06:28:05: GPRS:Processing PPP regen reqQ
06:28:05: GPRS:1011111111500001:Processing Disconnect PPP  regen from reqQ
06:28:05: GPRS:1011111111500001:got event [CANCEL REGEN'ED PPP] in state [PPP CONNECTED]
06:28:05: PPP-REGEN state counters: pending counter is 1
06:28:05:               State[IDLE] counter is 0
06:28:05:               State[AUTHORIZING] counter is 0
06:28:05:               State[VPDN CONNECTING] counter is 0
06:28:05:               State[PPP NEGOTIATING] counter is 0
06:28:05:               State[PPP CONNECTED] counter is 0
06:28:05:               State[PPP TERMINATING] counter is 1
06:28:05: GPRS:1011111111500001:state [PPP CONNECTED->PPP TERMINATING] on event [CANCEL
REGEN'ED PPP]
06:28:05: GPRS:1011111111500001:Cancel request after VPND tunnel is up
06:28:05: PPP-REGEN state counters: pending counter is 1
06:28:05:               State[IDLE] counter is 0
06:28:05:               State[AUTHORIZING] counter is 0
06:28:05:               State[VPDN CONNECTING] counter is 0
06:28:05:               State[PPP NEGOTIATING] counter is 0
06:28:05:               State[PPP CONNECTED] counter is 0
06:28:05:               State[PPP TERMINATING] counter is 1
06:28:05: GPRS:1011111111500001:PPP down
06:28:05: GPRS:1011111111500001:got event [PPP FAILED] in state [PPP TERMINATING]
06:28:05: PPP-REGEN state counters: pending counter is 1
06:28:05:               State[IDLE] counter is 1
06:28:05:               State[AUTHORIZING] counter is 0
06:28:05:               State[VPDN CONNECTING] counter is 0
06:28:05:               State[PPP NEGOTIATING] counter is 0
06:28:05:               State[PPP CONNECTED] counter is 0
06:28:05:               State[PPP TERMINATING] counter is 0
```

```
06:28:05: GPRS:1011111111500001:state [PPP TERMINATING->IDLE] on event [PPP FAILED]
06:28:05: GPRS:1011111111500001:LCP went down
06:28:05: GPRS:1011111111500001:VPDN disconnect
06:28:05: GPRS:1011111111500001:got event [CLEANUP CONTEXT] in state [IDLE]
06:28:05: GPRS:1011111111500001:state [IDLE->IDLE] on event [CLEANUP CONTEXT]
06:28:05: GPRS:1011111111500001:Freeing context structure
06:28:05: GPRS:1011111111500001:VPDN handle invalid, no need to free it
06:28:05: GPRS:1011111111500001:remove PPP regen context from Vi2
06:28:05: GPRS:1011111111500001:Session timer stopped
06:28:05: PPP-REGEN state counters: pending counter is 0
06:28:05:               State[IDLE] counter is 0
06:28:05:               State[AUTHORIZING] counter is 0
06:28:05:               State[VPDN CONNECTING] counter is 0
06:28:05:               State[PPP NEGOTIATING] counter is 0
06:28:05:               State[PPP CONNECTED] counter is 0
06:28:05:               State[PPP TERMINATING] counter is 0
06:28:05: GPRS:1011111111500001:PPP regen context 0x633F196C released
06:28:05: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to down
06:28:06: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2, changed state
to down
```

debug gprs iscsi

To display information about the GPRS iSCSI processing, use the **debug gprs iscsi** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs gtp {**errors** | **events** | **messages**}

no debug gprs gtp {**errors** | **events** | **messages**}

Syntax Description

errors	Displays error messages related to GPRS iSCSI processing on the GGSN.
events	Displays events related to GPRS iSCSI processing on the GGSN.
messages	Displays signaling messages related to GPRS iSCSI.

Defaults

No default behavior or values.

Command History

Release	Modification
12.4(15)XQ	This command was introduced.
12.4(24)T	This command was integrated into Cisco IOS Release 12.4(24)T.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between the GGSN and the SAN using iSCSI.

Examples

The following example displays GPRS iSCSI debugging:

```
Router#
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 246, error code = 0
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:
 ISCSI: Retrieved DTR Val is iscsi_hdr.dtr_typ_val 2
SAMI 9/3: GPRS:ISCSI: dtr_typ_val = 2 fn:send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: SAN has sent the record for a read request
SAMI 9/3: GPRS:ISCSI: ISCSI_DYNAMIC send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: gtp_msg_send_iscsi_retrieved_drt_req is called
SAMI 9/3: GPRS:retrieved cdr from ISCSI
SAMI 9/3: GPRS:Fn is gtp_msg_send_iscsi_retrieved_drt_req, pak val is 4AE35EE4
pak-datagramstart is 7C53FA18 pak->datagramsize is 232

SAMI 9/3: GPRS:ISCSI: data_len = 246, error code = 0
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:
 ISCSI: Retrieved DTR Val is iscsi_hdr.dtr_typ_val 2
SAMI 9/3: GPRS:ISCSI: dtr_typ_val = 2 fn:send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: SAN has sent the record for a read request
SAMI 9/3: GPRS:ISCSI: ISCSI_DYNAMIC send_retrieved_dtr_to_cgw
```

```
SAMI 9/3: GPRS:ISCSI: gtp_msg_send_iscsi_retrieved_drt_req is called
SAMI 9/3: GPRS:retrieved cdr from ISCSI
SAMI 9/3: GPRS:Fn is gtp_msg_send_iscsi_retrieved_drt_req, pak val is 41056464
pak-datagramstart is 7C003058 pak->datagramsize is 232

SAMI 9/3: GPRS:ISCSI: data_len = 246, error code = 0
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:
 ISCSI: Retrieved DTR Val is iscsi_hdr.dtr_typ_val 2
SAMI 9/3: GPRS:ISCSI: dtr_typ_val = 2 fn:send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: SAN has sent the record for a read request
SAMI 9/3: GPRS:ISCSI: ISCSI_DYNAMIC send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: gtp_msg_send_iscsi_retrieved_drt_req is called
SAMI 9/3: GPRS:retrieved cdr from ISCSI
SAMI 9/3: GPRS:Fn is gtp_msg_send_iscsi_retrieved_drt_req, pak val is 415563FC
pak-datagramstart is 7C53FD58 pak->datagramsize is 232

SAMI 9/3: GPRS:ISCSI: data_len = 246, error code = 0
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:
 ISCSI: Retrieved DTR Val is iscsi_hdr.dtr_typ_val 2
SAMI 9/3: GPRS:ISCSI: dtr_typ_val = 2 fn:send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: SAN has sent the record for a read request
SAMI 9/3: GPRS:ISCSI: ISCSI_DYNAMIC send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: gtp_msg_send_iscsi_retrieved_drt_req is called
SAMI 9/3: GPRS:retrieved cdr from ISCSI
SAMI 9/3: GPRS:Fn is gtp_msg_send_iscsi_retrieved_drt_req, pak val is 41056BDC
pak-datagramstart is 7C003D58 pak->datagramsize is 232

SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 1162, error code = 0
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:
 ISCSI: Retrieved DTR Val is iscsi_hdr.dtr_typ_val 1
SAMI 9/3: GPRS:ISCSI: dtr_typ_val = 1 fn:send_retrieved_dtr_to_cgw
SAMI 9/3: GPRS:ISCSI: SAN has sent the record for a read request
SAMI 9/3: GPRS:ISCSI: ISCSI_PENDING send_retrieved_dtr_to_cgw cgw_down_flags  300
SAMI 9/3: GPRS:ISCSI: gtp_msg_send_iscsi_retrieved_drt_req is called
SAMI 9/3: GPRS:retrieved cdr from ISCSI
SAMI 9/3: GPRS:Fn is gtp_msg_send_iscsi_retrieved_drt_req, pak val is 4AE3B10C
pak-datagramstart is 7C5512D8 pak->datagramsize is 1132

SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
```

```
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:ISCSI: data_len = 0, error code = 3
SAMI 9/3: GPRS:ISCSI retrieved empty record 3
SAMI 9/3: GPRS:GGSN_ISCSI_MSG
SAMI 9/3: GPRS:ISCSI_READ_ACK_RCVD
SAMI 9/3: GPRS:Empty iSCSI record was rcvd, so send leftover DTRs to CG
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
SAMI 9/3: GPRS:Fn is ggsn_iscsi_send_leftover_dtrs_to_cgw
Router#
```

debug gprs radius

To display information about Remote Access Dial-In User Service (RADIUS) processing on the gateway GPRS support node (GGSN), use the **debug gprs radius** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs radius

no debug gprs radius

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command History

Release	Modification
12.2(4)MX	This command was introduced.
12.2(8)YD	This command was integrated into Cisco IOS Release 12.2(8)YD.
12.2(8)YW	This command was integrated into Cisco IOS Release 12.2(8)YW.
12.3(2)XB	This command was integrated into Cisco IOS Release 12.3(2)XB.
12.3(8)XU	This command was integrated into Cisco IOS Release 12.3(8)XU.
12.3(11)YJ	This command was integrated into Cisco IOS Release 12.3(11)YJ.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between a RADIUS server and the GGSN.

Caution Because the **debug gprs radius** command generates a significant amount of output, use it only when traffic on the GPRS network is low, so other activity on the system is not adversely affected.

Examples

The following example enables the display of debug messages related to RADIUS processing on the GGSN:

```
Router# debug gprs radius
```

debug gprs redundancy

To display debug messages, errors, events, or packets related to GTP session redundancy (GTP-SR), use the **debug gprs redundancy** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug gprs redundancy [**debug** | **errors** | **events** | **packets**]

no debug gprs redundancy [**debug** | **errors** | **events** | **packets**]

Syntax Description

debug	Displays debug messages related to GTP-SR.
errors	Displays errors related to GTP-SR.
events	Displays events related to GTP-SR.
packets	Displays packets related to GTP-SR packets.

Defaults

Disabled.

Command Modes

Global configuration

Command History

Release	Modification
12.3(11)YJ	This command was introduced.
12.3(14)YQ	This command was integrated into Cisco IOS Release 12.3(14)YQ.
12.3(14)YU	This command was integrated into Cisco IOS Release 12.3(14)YU.
12.4(2)XB	This command was integrated into Cisco IOS Release 12.4(2)XB.

Usage Guidelines

This command displays debug level messages, errors, events, or packets for GTP-SR. It is useful for system operators and development engineers if problems are encountered with communication between the two GGSNs configured as an redundant pair and on which GTP-SR is enabled.

Examples

The following example enables the display of events related to GTP-SR processing on the GGSN:

```
Router# debug gprs redundancy
```

Related Commands

Command	Description
clear gprs redundancy statistics	Clears statistics related to GTP-SR.
gprs redundancy	Enables GTP-SR on a GGSN.
gprs redundancy charging sync-window cdr rec-seqnum	Configures the window size used to determine when the CDR record sequence number needs to be synchronized to the Standby GGSN.

Command	Description
gprs redundancy charging sync-window gtpp seqnum	Configures the window size used to determine when the GTP' sequence number needs to be synchronized to the Standby GGSN.
show gprs redundancy	Displays statistics related to GTP-SR.

debug ip iscsi

To display information about the iSCSI processing on the GGSN, use the **debug ip iscsi** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

debug ip iscsi {**all** | **error** | **event** | **packet**} [**detail**]

no debug ip iscsi {**all** | **error** | **event** | **packet**} [**detail**]

Syntax Description

all	Displays all iSCSI debug information.
error	Displays error messages related to iSCSI processing on the GGSN.
event	Displays events related to iSCSI processing on the GGSN.
packet	Displays iSCSI packets that are sent between the GGSN and SAN.
detail	(Optional) Displays detailed packet and event information.

Defaults

No default behavior or values.

Command History

Release	Modification
12.4(15)XQ	This command was introduced.
12.4(24)T	This command was integrated into Cisco IOS Release 12.4(24)T.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between the GGSN and the SAN using iSCSI.

Examples

The following example displays iSCSI debugging at the time of login:

```
===============
Router#debug ip iscsi all
iSCSI All debugging is on

Router#show debug
iSCSI:
  iSCSI Events debugging is on
  iSCSI Events Detailed debugging is on
  iSCSI Packets debugging is on
  iSCSI Packets Detailed debugging is on
  iSCSI Error debugging is on

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#gprs iscsi LINUX
Router(config)#end
Router#
SAMI 9/3: iSCSI Event: iSCSI Connection Event (0), State Change from(0) -> To(1)
SAMI 9/3: iSCSI Event: Socket Connect Success
SAMI 9/3: iSCSI Event: iSCSI Connection Event (4), State Change from(1) -> To(2)
SAMI 9/3: iSCSI Event: Send CONN Up Msg to RX
```

```
SAMI 9/3: INTR->TGT (HEADER + DATA):
493DEE20:                   43810000 00000092          C.......
493DEE30: 30303030 31000000 00000000 00000000  00001...........
493DEE40: 00000001 00000000 00000000 00000000  ................
493DEE50: 00000000 00000000 496E6974 6961746F  ........Initiato
493DEE60: 724E616D 653D6971 6E2E3139 38372D30  rName=iqn.1987-0
493DEE70: 372E636F 6D2E6369 73636F3A 6D777462  7.com.cisco:mwtb
493DEE80: 6732352D 7375702D 30392D33 00546172  g25-sup-09-3.Tar
493DEE90: 6765744E 616D653D 69716E2E 32303032  getName=iqn.2002
493DEEA0: 2D31302E 6564752E 756E682E 696F6C2E  -10.edu.unh.iol.
493DEEB0: 69736373 692E6472 61667432 302D7461  iscsi.draft20-ta
493DEEC0: 72676574 3A310053 65737369 6F6E5479  rget:1.SessionTy
493DEED0: 70653D4E 6F726D61 6C004175 74684D65  pe=Normal.AuthMe
493DEEE0: 74686F64 3D4E6F6E 65000000           thod=None...
SAMI 9/3: iSCSI Event: Starting Login Timer (5)
SAMI 9/3: iSCSI Event: New Connection Event - 0
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 23810000 00000027 30303030 31000000  #......'00001...
4B5A7260: 00000000 00000000 00000001 00000001  ................
4B5A7270: 00000005 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: TGT->INTR:Data:
493E6E50:                   41757468 4D657468          AuthMeth
493E6E60: 6F643D4E 6F6E6500 54617267 6574506F  od=None.TargetPo
493E6E70: 7274616C 47726F75 70546167 3D310000  rtalGroupTag=1..
493E6E80:
SAMI 9/3: iSCSI Event: Data-In: Read (40) bytes of Data Segment
SAMI 9/3: INTR->TGT (HEADER + DATA):
493DEE20:                   43870000 00000133          C......3
493DEE30: 30303030 31000000 00000000 00000000  00001...........
493DEE40: 00000001 00000002 00000000 00000000  ................
493DEE50: 00000000 00000000 48656164 65724469  ........HeaderDi
493DEE60: 67657374 3D4E6F6E 65004461 74614469  gest=None.DataDi
493DEE70: 67657374 3D4E6F6E 65004D61 78526563  gest=None.MaxRec
493DEE80: 76446174 61536567 6D656E74 4C656E67  vDataSegmentLeng
493DEE90: 74683D33 32373638 00446566 61756C74  th=32768.Default
493DEEA0: 54696D65 32576169 743D3500 44656661  Time2Wait=5.Defa
493DEEB0: 756C7454 696D6532 52657461 696E3D35  ultTime2Retain=5
493DEEC0: 0049464D 61726B65 723D4E6F 004F464D  .IFMarker=No.OFM
493DEED0: 61726B65 723D4E6F 00457272 6F725265  arker=No.ErrorRe
493DEEE0: 636F7665 72794C65 76656C3D 3000496E  coveryLevel=0.In
493DEEF0: 69746961 6C523254 3D596573 00496D6D  itialR2T=Yes.Imm
493DEF00: 65646961 74654461 74613D59 6573004D  ediateData=Yes.M
493DEF10: 61784275 7273744C 656E6774 683D3136  axBurstLength=16
493DEF20: 33383400 46697273 74427572 73744C65  384.FirstBurstLe
493DEF30: 6E677468 3D313633 3834004D 61784F75  ngth=16384.MaxOu
493DEF40: 74737461 6E64696E 67523254 3D31004D  tstandingR2T=1.M
493DEF50: 6178436F 6E6E6563 74696F6E 733D3100  axConnections=1.
493DEF60: 44617461 50445549 6E4F7264 65723D59  DataPDUInOrder=Y
493DEF70: 65730044 61746153 65717565 6E636549  es.DataSequenceI
493DEF80: 6E4F7264 65723D59 65730000           nOrder=Yes..
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 23870000 000000C2 30303030 31000F53  #......B00001..S
4B5A7260: 00000000 00000000 00000002 00000001  ................
4B5A7270: 00000005 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: TGT->INTR:Data:
493E6E50:                   48656164 65724469          HeaderDi
493E6E60: 67657374 3D4E6F6E 65004461 74614469  gest=None.DataDi
493E6E70: 67657374 3D4E6F6E 65004465 6661756C  gest=None.Defaul
493E6E80: 7454696D 65325761 69743D35 00446566  tTime2Wait=5.Def
493E6E90: 61756C74 54696D65 32526574 61696E3D  aultTime2Retain=
493E6EA0: 35004572 726F7252 65636F76 6572794C  5.ErrorRecoveryL
493E6EB0: 6576656C 3D300049 6D6D6564 69617465  evel=0.Immediate
```

```
493E6EC0: 44617461 3D596573 004D6178 4F757473  Data=Yes.MaxOuts
493E6ED0: 74616E64 696E6752 32543D31 004D6178  tandingR2T=1.Max
493E6EE0: 436F6E6E 65637469 6F6E733D 31004669  Connections=1.Fi
493E6EF0: 72737442 75727374 4C656E67 74683D31  rstBurstLength=1
493E6F00: 36333834 004D6178 42757273 744C656E  6384.MaxBurstLen
493E6F10: 6774683D 31363338 34000000           gth=16384...
SAMI 9/3: iSCSI Event: Data-In: Read (196) bytes of Data Segment
SAMI 9/3: iSCSI Event: iSCSI Connection Event (6), State Change from(2) -> To(3)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: iSCSI Event: iSCSI Session Event (0), State Change from(0) -> To(1)
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00000000 00000000 00000001 00000000  ................
493DEE40: 00000001 00000003 00000000 00000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 00000001 00000000 00000003 00000002  ................
4B5A7270: 00000006 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Test unit ready command successful
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00000000 00000000 00000002 000000FF  ................
493DEE40: 00000002 00000004 A0000000 00000000  ........ ........
493DEE50: 00FF0000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000030 00000000 00000000  %......0........
4B5A7260: 00000002 FFFFFFFF 00000000 00000003  ................
4B5A7270: 00000006 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 2, cmnd 0xA0, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
414F59E0: 00000028 00000000 00000000 00000000  ...(............
414F59F0: 00010000 00000000 00020000 00000000  ................
414F5A00: 00030000 00000000 00040000 00000000  ................
414F5A10:
SAMI 9/3: iSCSI Event: Data-In: Read (48) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 00000002 00000000 00000004 00000003  ................
4B5A7270: 00000007 00000001 00000000 000000CF  ...............O
4B5A7280:
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00000000 00000000 00000003 000000FF  ................
493DEE40: 00000003 00000005 12000000 FF000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 000000FF 00000000 00000000  %...............
4B5A7260: 00000003 FFFFFFFF 00000000 00000004  ................
4B5A7270: 00000007 00000000 00000000 00000000  ................
4B5A7280:
```

```
SAMI 9/3: iSCSI Event: recv_data for itt 3, cmnd 0x12, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
493D6960:          00000402 1F008000 554E482D      ........UNH-
493D6970: 494F4C20 66696C65 2D6D6F64 65207461  IOL file-mode ta
493D6980: 72676574 312E3220 00000000 00000000  rget1.2 ........
493D6990: 00000000 00000000 00000000 00000000  ................
493D69A0: 00000000 00000000 00000000 00000000  ................
493D69B0: 00000000 00000000 00000000 00000000  ................
493D69C0: 00000000 00000000 00000000 00000000  ................
493D69D0: 00000000 00000000 00000000 00000000  ................
493D69E0: 00000000 00000000 00000000 00000000  ................
493D69F0: 00000000 00000000 00000000 00000000  ................
493D6A00: 00000000 00000000 00000000 00000000  ................
493D6A10: 00000000 00000000 00000000 00000000  ................
493D6A20: 00000000 00000000 00000000 00000000  ................
493D6A30: 00000000 00000000 00000000 00000000  ................
493D6A40: 00000000 00000000 00000000 00000000  ................
493D6A50: 00000000 00000000 00000000 00000000  ................
493D6A60: 00000000                             ....
SAMI 9/3: iSCSI Event: Data-In: Read (256) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 00000003 00000000 00000005 00000004  ................
4B5A7270: 00000008 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Processing inquire LUN response
SAMI 9/3: SCSI Event: Calling Device Add - 414F59E0
SAMI 9/3: SCSI Event: scsi add device
SAMI 9/3: SCSI Event: lun_in_inquiry 1
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00010000 00000000 00000004 000000FF  ................
493DEE40: 00000004 00000006 12000000 FF000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 000000FF 00000000 00000000  %...............
4B5A7260: 00000004 FFFFFFFF 00000000 00000005  ................
4B5A7270: 00000008 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 4, cmnd 0x12, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
493A3D40: 00000402 1F008000 554E482D 494F4C20  ........UNH-IOL
493A3D50: 66696C65 2D6D6F64 65207461 72676574  file-mode target
493A3D60: 312E3220 00000000 00000000 00000000  1.2 ............
493A3D70: 00000000 00000000 00000000 00000000  ................
493A3D80: 00000000 00000000 00000000 00000000  ................
493A3D90: 00000000 00000000 00000000 00000000  ................
493A3DA0: 00000000 00000000 00000000 00000000  ................
493A3DB0: 00000000 00000000 00000000 00000000  ................
493A3DC0: 00000000 00000000 00000000 00000000  ................
493A3DD0: 00000000 00000000 00000000 00000000  ................
493A3DE0: 00000000 00000000 00000000 00000000  ................
493A3DF0: 00000000 00000000 00000000 00000000  ................
493A3E00: 00000000 00000000 00000000 00000000  ................
493A3E10: 00000000 00000000 00000000 00000000  ................
493A3E20: 00000000 00000000 00000000 00000000  ................
493A3E30: 00000000 00000000 00000000 00000000  ................
```

```
493A3E40:
SAMI 9/3: iSCSI Event: Data-In: Read (256) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 00000004 00000000 00000006 00000005  ................
4B5A7270: 00000009 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Processing inquire LUN response
SAMI 9/3: SCSI Event: Calling Device Add - 41E1B98C
SAMI 9/3: SCSI Event: scsi add device
SAMI 9/3: SCSI Event: lun_in_inquiry 2
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00020000 00000000 00000005 000000FF  ................
493DEE40: 00000005 00000007 12000000 FF000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 000000FF 00000000 00000000  %...............
4B5A7260: 00000005 FFFFFFFF 00000000 00000006  ................
4B5A7270: 00000009 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 5, cmnd 0x12, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4B643390:          00000402 1F008000 554E482D      ........UNH-
4B6433A0: 494F4C20 66696C65 2D6D6F64 65207461  IOL file-mode ta
4B6433B0: 72676574 312E3220 00000000 00000000  rget1.2 ........
4B6433C0: 00000000 00000000 00000000 00000000  ................
4B6433D0: 00000000 00000000 00000000 00000000  ................
4B6433E0: 00000000 00000000 00000000 00000000  ................
4B6433F0: 00000000 00000000 00000000 00000000  ................
4B643400: 00000000 00000000 00000000 00000000  ................
4B643410: 00000000 00000000 00000000 00000000  ................
4B643420: 00000000 00000000 00000000 00000000  ................
4B643430: 00000000 00000000 00000000 00000000  ................
4B643440: 00000000 00000000 00000000 00000000  ................
4B643450: 00000000 00000000 00000000 00000000  ................
4B643460: 00000000 00000000 00000000 00000000  ................
4B643470: 00000000 00000000 00000000 00000000  ................
4B643480: 00000000 00000000 00000000 00000000  ................
4B643490: 00000000                             ....
SAMI 9/3: iSCSI Event: Data-In: Read (256) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 00000005 00000000 00000007 00000006  ................
4B5A7270: 0000000A 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Processing inquire LUN response
SAMI 9/3: SCSI Event: Calling Device Add - 4B63DC5C
SAMI 9/3: SCSI Event: scsi add device
SAMI 9/3: SCSI Event: lun_in_inquiry 3
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00030000 00000000 00000006 000000FF  ................
493DEE40: 00000006 00000008 12000000 FF000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
```

```
4B5A7250: 25800000 000000FF 00000000 00000000  %...............
4B5A7260: 00000006 FFFFFFFF 00000000 00000007  ................
4B5A7270: 0000000A 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 6, cmnd 0x12, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4198DBD0: 00000402 1F008000 554E482D 494F4C20  ........UNH-IOL
4198DBE0: 66696C65 2D6D6F64 65207461 72676574  file-mode target
4198DBF0: 312E3220 00000000 00000000 00000000  1.2 ............
4198DC00: 00000000 00000000 00000000 00000000  ................
4198DC10: 00000000 00000000 00000000 00000000  ................
4198DC20: 00000000 00000000 00000000 00000000  ................
4198DC30: 00000000 00000000 00000000 00000000  ................
4198DC40: 00000000 00000000 00000000 00000000  ................
4198DC50: 00000000 00000000 00000000 00000000  ................
4198DC60: 00000000 00000000 00000000 00000000  ................
4198DC70: 00000000 00000000 00000000 00000000  ................
4198DC80: 00000000 00000000 00000000 00000000  ................
4198DC90: 00000000 00000000 00000000 00000000  ................
4198DCA0: 00000000 00000000 00000000 00000000  ................
4198DCB0: 00000000 00000000 00000000 00000000  ................
4198DCC0: 00000000 00000000 00000000 00000000  ................
4198DCD0:
SAMI 9/3: iSCSI Event: Data-In: Read (256) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 0000
Router#0000 00000000  !................
4B5A7260: 00000006 00000000 00000008 00000007  ................
4B5A7270: 0000000B 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Processing inquire LUN response
SAMI 9/3: SCSI Event: Calling Device Add - 4B63C60C
SAMI 9/3: SCSI Event: scsi add device
SAMI 9/3: SCSI Event: lun_in_inquiry 4
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00040000 00000000 00000007 000000FF  ................
493DEE40: 00000007 00000009 12000000 FF000000  ................
493DEE50: 00000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 000000FF 00000000 00000000  %...............
4B5A7260: 00000007 FFFFFFFF 00000000 00000008  ................
4B5A7270: 0000000B 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 7, cmnd 0x12, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4B63C720: 00000402 1F008000 554E482D 494F4C20  ........UNH-IOL
4B63C730: 66696C65 2D6D6F64 65207461 72676574  file-mode target
4B63C740: 312E3220 00000000 00000000 00000000  1.2 ............
4B63C750: 00000000 00000000 00000000 00000000  ................
4B63C760: 00000000 00000000 00000000 00000000  ................
4B63C770: 00000000 00000000 00000000 00000000  ................
4B63C780: 00000000 00000000 00000000 00000000  ................
4B63C790: 00000000 00000000 00000000 00000000  ................
4B63C7A0: 00000000 00000000 00000000 00000000  ................
4B63C7B0: 00000000 00000000 00000000 00000000  ................
4B63C7C0: 00000000 00000000 00000000 00000000  ................
```

```
4B63C7D0: 00000000 00000000 00000000 00000000  ................
4B63C7E0: 00000000 00000000 00000000 00000000  ................
4B63C7F0: 00000000 00000000 00000000 00000000  ................
4B63C800: 00000000 00000000 00000000 00000000  ................
4B63C810: 00000000 00000000 00000000 00000000  ................
4B63C820:
SAMI 9/3: iSCSI Event: Data-In: Read (256) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 00000007 00000000 00000009 00000008  ................
4B5A7270: 0000000C 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event: Processing inquire LUN response
SAMI 9/3: SCSI Event: Calling Device Add - 493A3378
SAMI 9/3: SCSI Event: scsi add device
SAMI 9/3: SCSI Event: max= 5 lun_in_inquiry= 5
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00000000 00000000 00000008 000000FF  ................
493DEE40: 00000008 0000000A 25000000 00000000  ........%.......
493DEE50: 00000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000008 00000000 00000000  %...............
4B5A7260: 00000008 FFFFFFFF 00000000 00000009  ................
4B5A7270: 0000000C 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 8, cmnd 0x25, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
493D65D0:                   003FFFFF 00000200          .?......
493D65E0:
SAMI 9/3: iSCSI Event: Data-In: Read (8) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 00000008 00000000 0000000A 00000009  ................
4B5A7270: 0000000D 00000001 00000000 000000F7  ................w
4B5A7280:
SAMI 9/3: SCSI Event: Processing read capacity response
SAMI 9/3: SCSI Event: max= 5 lun= 1
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00010000 00000000 00000009 000000FF  ................
493DEE40: 00000009 0000000B 25000000 00000000  ........%.......
493DEE50: 00000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000008 00000000 00000000  %...............
4B5A7260: 00000009 FFFFFFFF 00000000 0000000A  ................
4B5A7270: 0000000D 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 9, cmnd 0x25, bufflen 255, offset 0 exp offset 0,
flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
41637830:                           003FFFFF              .?..
41637840: 00000200                              ....
SAMI 9/3: iSCSI Event: Data-In: Read (8) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
```

```
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 00000009 00000000 0000000B 0000000A  ................
4B5A7270: 0000000E 00000001 00000000 000000F7  ................w
4B5A7280:
SAMI 9/3: SCSI Event: Processing read capacity response
SAMI 9/3: SCSI Event: max= 5 lun= 2
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00020000 00000000 0000000A 000000FF  ................
493DEE40: 0000000A 0000000C 25000000 00000000  ........%.......
493DEE50: 00000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000008 00000000 00000000  %...............
4B5A7260: 0000000A FFFFFFFF 00000000 0000000B  ................
4B5A7270: 0000000E 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 10, cmnd 0x25, bufflen 255, offset 0 exp offset
0, flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4ADE19D0:                           003FFFFF              .?..
4ADE19E0: 00000200                                ....
SAMI 9/3: iSCSI Event: Data-In: Read (8) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 0000000A 00000000 0000000C 0000000B  ................
4B5A7270: 0000000F 00000001 00000000 000000F7  ................w
4B5A7280:
SAMI 9/3: SCSI Event: Processing read capacity response
SAMI 9/3: SCSI Event: max= 5 lun= 3
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00030000 00000000 0000000B 000000FF  ................
493DEE40: 0000000B 0000000D 25000000 00000000  ........%.......
493DEE50: 00000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000008 00000000 00000000  %...............
4B5A7260: 0000000B FFFFFFFF 00000000 0000000C  ................
4B5A7270: 0000000F 00
Router#000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 11, cmnd 0x25, bufflen 255, offset 0 exp offset
0, flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4ADE1B10: 003FFFFF 00000200                     .?......
SAMI 9/3: iSCSI Event: Data-In: Read (8) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 0000000B 00000000 0000000D 0000000C  ................
4B5A7270: 00000010 00000001 00000000 000000F7  ................w
4B5A7280:
SAMI 9/3: SCSI Event: Processing read capacity response
SAMI 9/3: SCSI Event: max= 5 lun= 4
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
```

```
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00040000 00000000 0000000C 000000FF  ................
493DEE40: 0000000C 0000000E 25000000 00000000  ........%.......
493DEE50: 00000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000008 00000000 00000000  %...............
4B5A7260: 0000000C FFFFFFFF 00000000 0000000D  ................
4B5A7270: 00000010 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 12, cmnd 0x25, bufflen 255, offset 0 exp offset
0, flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4B642580:                           0003FFFF              ....
4B642590: 00000200                              ....
SAMI 9/3: iSCSI Event: Data-In: Read (8) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21820000 00000000 00000000 00000000  !...............
4B5A7260: 0000000C 00000000 0000000E 0000000D  ................
4B5A7270: 00000011 00000001 00000000 000000F7  ................w
4B5A7280:
SAMI 9/3: SCSI Event: Processing read capacity response
SAMI 9/3: SCSI Event: Max= 5 lun= 5
SAMI 9/3: SCSI Event: device discovery completed
SAMI 9/3: SCSI Event:
Creating File System on sda0
SAMI 9/3: SCSI Event: Read command, lba(0), nblocks(1)
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00000000 00000000 0000000D 00000200  ................
493DEE40: 0000000D 0000000F 28000000 00000000  ........(.......
493DEE50: 01000000 00000000                     ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000200 00000000 00000000  %...............
4B5A7260: 0000000D FFFFFFFF 00000000 0000000E  ................
4B5A7270: 00000011 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 13, cmnd 0x28, bufflen 512, offset 0 exp offset
0, flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4B5A8B00: 00000000 00000000 00000000 00000000  ................
4B5A8B10: 00000000 00000000 00000000 00000000  ................
4B5A8B20: 00000000 00000000 00000000 00000000  ................
4B5A8B30: 00000000 00000000 00000000 00000000  ................
4B5A8B40: 00000000 00000000 00000000 00000000  ................
4B5A8B50: 00000000 00000000 00000000 00000000  ................
4B5A8B60: 00000000 00000000 00000000 00000000  ................
4B5A8B70: 00000000 00000000 00000000 00000000  ................
4B5A8B80: 00000000 00000000 00000000 00000000  ................
4B5A8B90: 00000000 00000000 00000000 00000000  ................
4B5A8BA0: 00000000 00000000 00000000 00000000  ................
4B5A8BB0: 00000000 00000000 00000000 00000000  ................
4B5A8BC0: 00000000 00000000 00000000 00000000  ................
4B5A8BD0: 00000000 00000000 00000000 00000000  ................
4B5A8BE0: 00000000 00000000 00000000 00000000  ................
4B5A8BF0: 00000000 00000000 00000000 00000000  ................
4B5A8C00: 00000000 00000000 00000000 00000000  ................
4B5A8C10: 00000000 00000000 00000000 00000000  ................
4B5A8C20: 00000000 00000000 00000000 00000000  ................
4B5A8C30: 00000000 00000000 00000000 00000000  ................
```

```
4B5A8C40: 00000000 00000000 00000000 00000000  ................
4B5A8C50: 00000000 00000000 00000000 00000000  ................
4B5A8C60: 00000000 00000000 00000000 00000000  ................
4B5A8C70: 00000000 00000000 00000000 00000000  ................
4B5A8C80: 00000000 00000000 00000000 00000000  ................
4B5A8C90: 00000000 00000000 00000000 00000000  ................
4B5A8CA0: 00000000 00000000 00000000 00000000  ................
4B5A8CB0: 00000000 00000000 E6F06A79 00000000  ........fpjy....
4B5A8CC0: 00000000 00000000 00000000 00000000  ................
4B5A8CD0: 00000000 00000000 00000000 00000000  ................
4B5A8CE0: 00000000 00000000 00000000 00000000  ................
4B5A8CF0: 00000000 00000000 00000000 000055AA  ..............U*
4B5A8D00:
SAMI 9/3: iSCSI Event: Data-In: Read (512) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 21800000 00000000 00000000 00000000  !...............
4B5A7260: 0000000D 00000000 0000000F 0000000E  ................
4B5A7270: 00000012 00000001 00000000 00000000  ................
4B5A7280:
SAMI 9/3: SCSI Event:
Creating File System on sda1
SAMI 9/3: SCSI Event: Read command, lba(0), nblocks(1)
SAMI 9/3: iSCSI Event-Det: handle scsi cmd req
SAMI 9/3: iSCSI Event-Det: run pending queue
SAMI 9/3: iSCSI Event-Det: send scsi command
SAMI 9/3: INTR->TGT HEAD:
493DEE20:                   01C00000 00000000          .@......
493DEE30: 00010000 00000000 0000000E 00000200  ................
493DEE40: 0000000E 00000010 28000000 00000000  ........(.......
493DEE50: 01000000 00000000                    ........
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 25800000 00000200 00000000 00000000  %...............
4B5A7260: 0000000E FFFFFFFF 00000000 0000000F  ................
4B5A7270: 00000012 00000000 00000000 00000000  ................
4B5A7280:
SAMI 9/3: iSCSI Event: recv_data for itt 14, cmnd 0x28, bufflen 512, offset 0 exp offset
0, flags 0x80 datasn 0

SAMI 9/3: TGT->INTR:Data:
4B5A8B00: 33C08ED0 BC007CFB 5007501F FCBE1B7C  3@.P<.|{P.P.|>.|
4B5A8B10: BF1B0650 57B9E501 F3A4CBBD BE07B104  ?..PW9e.s$K=>.1.
4B5A8B20: 386E007C 09751383 C510E2F4 CD188BF5  8n.|.u..E.btM..u
4B5A8B30: 83C61049 7419382C 74F6A0B5 07B4078B  .F.It.8,tv 5.4..
4B5A8B40: F0AC3C00 74FCBB07 00B40ECD 10EBF288  p,<.t|;..4.M.kr.
4B5A8B50: 4E10E846 00732AFE 4610807E 040B740B  N.hF.s*~F..~..t.
4B5A8B60: 807E040C 7405A0B6 0775D280 46020683  .~..t. 6.uR.F...
4B5A8B70: 46080683 560A00E8 21007305 A0B607EB  F...V..h!.s. 6.k
4B5A8B80: BC813EFE 7D55AA74 0B807E10 0074C8A0  <.>~}U*t..~..tH
4B5A8B90: B707EBA9 8BFC1E57 8BF5CBBF 05008A56  7.k).|.W.uK?...V
4B5A8BA0: 00B408CD 1372238A C1243F98 8ADE8AFC  .4.M.r#.A$?..^.|
4B5A8BB0: 43F7E38B D186D6B1 06D2EE42 F7E23956  Cwc.Q.V1.RnBwb9V
4B5A8BC0: 0A772372 05394608 731CB801 02BB007C  .w#r.9F.s.8..;.|
4B5A8BD0: 8B4E028B 5600CD13 73514F74 4E32E48A  .N..V.M.sQOtN2d.
4B5A8BE0: 5600CD13 EBE48A56 0060BBAA 55B441CD  V.M.kd.V.`;*U4AM
4B5A8BF0: 13723681 FB55AA75 30F6C101 742B6160  .r6.{U*u0vA.t+a`
4B5A8C00: 6A006A00 FF760AFF 76086A00 68007C6A  j.j..v..v.j.h.|j
4B5A8C10: 016A10B4 428BF4CD 13616173 0E4F740B  .j.4B.tM.aas.Ot.
4B5A8C20: 32E48A56 00CD13EB D661F9C3 496E7661  2d.V.M.kVayCInva
4B5A8C30: 6C696420 70617274 6974696F 6E207461  lid partition ta
4B5A8C40: 626C6500 4572726F 72206C6F 6164696E  ble.Error loadin
4B5A8C50: 67206F70 65726174 696E6720 73797374  g operating syst
4B5A8C60: 656D004D 69737369 6E67206F 70657261  em.Missing opera
4B5A8C70: 74696E67 20737973 74656D00 00000000  ting system.....
4B5A8C80: 00000000 00000000 00000000 00000000  ................
```

```
4B5A8C90: 00000000 00000000 00000000 00000000  ................
4B5A8CA0: 00000000 00000000 00000000 00000000  ................
4B5A8CB0: 00000000 002C4463 656289D3 00000000  .....,Dceb.S....
4B5A8CC0: 00000000 00000000 00000000 00000000  ................
4B5A8CD0: 00000000 00000000 00000000 00000000  ................
4B5A8CE0: 00000000 00000000 00000000 00000000  ................
4B5A8CF0: 00000000 00000000 00000000 000055AA  ..............U*
4B5A8D00:
SAMI 9/3: iSCSI Event: Data-In: Read (512) bytes of Data Segment
SAMI 9/3: TGT->INTR:Header:
4B5A7250:
SAMI 9/3: %SYS-5-CONFIG_I: Configured from console by console
SAMI 9/3: %RSM-4-UNEXPECTED: Error: Drive sda4 unusable (Invalid DOS media or no media in
slot) -Process= "RSM Process", ipl= 0, pid= 193,  -Traceback= 0x446E45DC 0x442AD9BC
0x442AB94C 0x442A6318 0x442A648C 0x442AB41C 0x442A3B28 0x45602878 0x45605C50
SAMI 9/3: %GPRSISCSIFLTMG-4-GPRS_ISCSI_OPEN_SUCCESS: Succeeded to establish connection
with SAN with session id 13
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: FFFFFFFF 0000001C 0000001E 0000001C  ................
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: FFFFFFFF 0000001C 0000001C 0000001E  ................
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 0000001C FFFFFFFF 0000001C 0000001E  ................
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 0000001C FFFFFFFF 0000001E 0000001C  ................
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 0000001D FFFFFFFF 0000001C 0000001F  ................
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 0000001D FFFFFFFF 0000001F 0000001C  ................
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 0000001E FFFFFFFF 0000001C 00000020  ................
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 0000001E FFFFFFFF 00000020 0000001C  ........... ....
```

```
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 0000001F FFFFFFFF 0000001C 00000021  ...............!
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ...............
4B5A7260: 0000001F FFFFFFFF 00000021 0000001C  ...........!....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000020 FFFFFFFF 0000001C 00000022  ... ...........".
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ...............
4B5A7260: 00000020 FFFFFFFF 00000022 0000001C  ... .......".....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000021 FFFFFFFF 0000001C 00000023  ...!...........#
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ...............
4B5A7260: 00000021 FFFFFFFF 00000023 0000001C  ...!.......#....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000022 FFFFFFFF 0000001C 00000024  ..."...........$
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ...............
4B5A7260: 00000022 FFFFFFFF 00000024 0000001C  ..."....... $....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000023 FFFFFFFF 0000001C 00000025  ...#...........%
4B5A5B70: 00000000 00000000 00000000 00000000  ................
Router#
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
```

```
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 00000023 FFFFFFFF 00000025 0000001C  ...#.......%....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000024 FFFFFFFF 0000001C 00000026  ...$...........&
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5
Router#
Router#
Router#A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 00000024 FFFFFFFF 00000026 0000001C  ...$.......&....
4B5A7270: 00000020 00000000 00000000 00000000  ... ............
4B5A7280:
Router#
Router#
SAMI 9/3: iSCSI Event-Det: Connection timer event (0)
SAMI 9/3: iSCSI Event: FFP Timeout Event Active Tasks(0)
SAMI 9/3: iSCSI Event: Starting Full Feature Phase Timer (5)
SAMI 9/3: INTR->TGT (HEADER + DATA):
4B5A5B50: 40800000 00000000 00000000 00000000  @...............
4B5A5B60: 00000025 FFFFFFFF 0000001C 00000027  ...%...........'
4B5A5B70: 00000000 00000000 00000000 00000000  ................
4B5A5B80:
SAMI 9/3: TGT->INTR:Header:
4B5A7250: 20800000 00000000 00000000 00000000   ................
4B5A7260: 00000025 FFFFFFFF 00000027 0000001C  ...%.......'....
4B5A7270:un al 00000020 00000000 00000000 00000000  ... ............
4B5A7280:                                                       1
All possible debugging has been turned off
Router#sh ip iscsi session
ID        TARGET             STATE                CONNECTIONS
-----------------------------------------------------------------
13        LINUX              Logged In            1

=================================================
```

debug record-storage-module

To display debugging information related to the record storage module (RSM), use the **debug record-storage-module** command in privileged EXEC model. To disable debugging output, use the **no** form of this command.

debug record-storage-module [**all** | **dsm** | **error** | **event**]

no debug record-storage-module [**all** | **dsm** | **error** | **event**]

Syntax Description

all	Displays all RSM flags.
dsm	Displays data store manager debug information.
error	Displays RSM-related errors.
event	Displays RSM-related events.

Defaults

No default behavior or values.

Command History

Release	Modification
12.4(15)XQ	This command was introduced.
12.4(24)T	This command was integrated into Cisco IOS Release 12.4(24)T.

Usage Guidelines

This command is useful for system operators and development engineers if problems are encountered with communication between the GGSN and the SCSI target.

Examples

The following example displays RSM-related debugging at the time of the write process:

```
Router#
SAMI 9/3: %GPRSFLTMG-4-CHARGING: GSN: 32.0.0.2, TID: 0000000000000000, APN: NULL, Reason:
3, GSN GTP' Transfer Failure
Router#
SAMI 9/3: RSM-Event-Det: Write by appl GGSN for profile LINUX
SAMI 9/3: RSM-FUNC: Write Handler
SAMI 9/3: RSM-DSM-DET: Allocate write buffer
SAMI 9/3: RSM-DSM-DET: rem_len= 260966, bytes= 1178
SAMI 9/3: RSM-DSM: Write to file now
SAMI 9/3: RSM-DSM-DET: sda3:/root/00000001/00000001.dat exists
SAMI 9/3: RSM-DSM: Size of sda3:/root/00000001/00000001.dat is 780686
SAMI 9/3: RSM-DSM-DET: Write to sda3:/root/00000001/00000001.dat
SAMI 9/3: RSM-DSM-DET: sda3:/root/00000001/00000001.dat size is 781864 bytes
SAMI 9/3: RSM-DSM-DET: Call the write response handler
Router#show debug
Record Storage Module:
  RSM DSM debugging is on
  RSM DSM DETAIL debugging is on
  RSM EVENT DETAIL debugging is on
  RSM EVENT debugging is on
  RSM ERROR debugging is on
```

The following example displays RSM-related debugging at the time of the read process:

```
Router#
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Allocate read buffer
SAMI 9/3: RSM-DSM-DET: Data buffer empty, read from disk
SAMI 9/3: RSM-DSM: Read from file sda3:/root/00000001/00000001.dat
SAMI 9/3: RSM-DSM-DET: Read fd is illegal in drive sda3
SAMI 9/3: RSM-DSM-DET: sda3:/root/00000001/00000001.dat exists
SAMI 9/3: RSM-DSM-DET: Read from off = 778460
SAMI 9/3: RSM-FUNC: Read in buffer
SAMI 9/3: RSM-DSM-DET: Read 262144 byte from sda3:/root/00000001/00000001.dat
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 262
SAMI 9/3: RSM-Event-Det: Read record= 246 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 524
SAMI 9/3: RSM-Event-Det: Read record= 246 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 786
SAMI 9/3: RSM-Event-Det: Read record= 246 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 1048
SAMI 9/3: RSM-Event-Det: Read record= 246 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 2226
SAMI 9/3: RSM-Event-Det: Read record= 1162 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Complete Record, next rec offset= 3404
SAMI 9/3: RSM-Event-Det: Read record= 1162 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Next Record is not in buffer
SAMI 9/3: RSM-FUNC: Copy partial record to next buffer
SAMI 9/3: RSM-DSM-DET: copy= 0 bytes from offset= 3404 to offset= 2016
SAMI 9/3: RSM-DSM-DET: Data buffer empty, read from disk
SAMI 9/3: RSM-DSM: Read from file sda3:/root/00000001/00000001.dat
SAMI 9/3: RSM-FUNC: Read in buffer
SAMI 9/3: RSM-DSM-DET: Read 262144 byte from sda3:/root/00000001/00000001.dat
SAMI 9/3: RSM-DSM-DET: Chk if more data exists
SAMI 9/3: RSM-DSM-DET: Get next read file
SAMI 9/3: RSM-DSM-DET: sda3:/root/00000001/00000002.dat (File not found)
SAMI 9/3: RSM-DSM-DET: Get next read dir
SAMI 9/3: RSM-DSM-DET: sda3:/root/00000002/ does not exist
SAMI 9/3: RSM-DSM: Check next read drive sda3
SAMI 9/3: RSM-DSM: file sda3:/root/00000001/00000001.dat is the file currently read
SAMI 9/3: RSM-Error: Disk is empty
SAMI 9/3: RSM-DSM: Zero bytes read
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
```

```
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
SAMI 9/3: RSM-Event-Det: Read by appl GGSN for profile LINUX
SAMI 9/3: RSM-DSM-DET: Bytes in write buffer = 0
SAMI 9/3: RSM-Event: Disk is empty-No more records to Read
SAMI 9/3: RSM-Event-Det: Read record= 0 bytes
```