



デバッグ コマンド

この章のコマンドは、Cisco MDS 9000 ファミリのマルチレイヤディレクタおよびファブリック スイッチに対応しています。すべてのデバッグ コマンドは EXEC モードで実行されます。ここでは、デバッグ コマンドはアルファベット順に記載されています。詳細については、『*Cisco MDS 9000 Family Troubleshooting Guide*』および『*Cisco MDS 9000 Family System Messages Guide*』を参照してください。

CLI (コマンドライン インターフェイス) を使用することで、各スイッチのデバッグ モードをイネーブルにできます。また、制御プロトコル処理におけるアクティビティ ログの更新をリアルタイムで確認できます。各ログのエントリは、タイムスタンプされ、時系列にリストされていきます。デバッグ機能へのアクセスは、CLI のロール メカニズムによって制限したり、ロールに基づいて許可したりすることができます。

debug aaa

ブート変数のデバッグをイネーブルにするには、EXEC モードで **debug aaa** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug aaa {all | conf-events | errors | events | mts}
```

```
no debug aaa {all | conf-events | errors | events | mts}
```

シンタックスの説明

all	すべての AAA デバッグ オプションをイネーブルにします。
conf-events	AAA 設定イベント デバッグをイネーブルにします。
errors	AAA エラーに対するデバッグをイネーブルにします。
events	AAA イベントに対するデバッグをイネーブルにします。
mts	Messaging and Transaction Service (MTS) パケット デバッグの AAA 送受信をイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(1)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug aaa conf-events** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug aaa conf-events
Nov 20 06:29:52 aaa: aaa_cleanup_session
Nov 20 06:29:52 aaa: mts_drop of request msg
Nov 20 06:29:52 aaa: Configured method local Succeeded
Nov 20 06:29:58 aaa: Src: 0x00000101/10886 Dst: 0x00000101/0 ID: 0x003
ize: 197 [REQ] Opc: 8402 (MTS_OPC_AAA_REQ) RR: 0x003A48F7 HA_SEQNO: 0x0
TS: 0x9FC1C1234E7C REJ:0 SYNC:0
Nov 20 06:29:58 aaa: 01 01 0C 00 00 00 00 00 00 00 00 00 00 02 01
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 06 08 00 03 05 00 00 00
Nov 20 06:29:58 aaa: 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Nov 20 06:29:58 aaa: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

関連コマンド

コマンド	説明
aaa authentication login	ログインするための認証モードを作成します。
no debug all	すべてのデバッグをディセーブルにします。
show aaa authentication	設定された認証方式を表示します。

debug all

スイッチ上のすべての機能のデバッグをイネーブルにするには、EXEC モードで **debug all** コマンドを使用します。このコマンドの **no** 形式を使うことで、このコマンドをディセーブルにして、すべてのデバッグを終了させることができます。

debug all

no debug all

シンタックスの説明 このコマンドには、引数またはキーワードはありません。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン **no debug all** コマンドはすべての診断結果を終了します。**no debug all** コマンドを使用すると、誤ってデバッグコマンドを終了し忘れるということがなくなるので便利です。



注意

他のネットワーク トラフィックの処理よりデバッグの出力が優先されるため、また、**debug all** コマンドは他 **debug** のコマンドよりも出力が多いため、スイッチのパフォーマンスが著しく落ちる、または使用できなくなる可能性があります。不測の事態に備え、**debug** コマンドは特定して使用することを推奨します。

例 次に、**debug all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug all
```

関連コマンド	コマンド	説明
	show debug	スイッチに設定されているデバッグ コマンドを表示します。

debug biosd

bios_daemon のデバッグを設定するには、EXEC モードで **debug biosd** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug biosd all
```

```
no debug biosd all
```

シンタックスの説明	all	すべての bios_daemon デバッグ オプションをイネーブルにします。
デフォルト		ディセーブル
コマンドモード		EXEC モード
コマンド履歴	リリース	変更内容
	2.1(1)	このコマンドが導入されました。
使用上のガイドライン		なし
例		次に、 debug biosd コマンドが実行された場合に表示される、システムの実行結果の例を示します。 switch# debug biosd
関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug bootvar

ブート変数のデバッグをイネーブルにするには、EXEC モードで **debug bootvar** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug bootvar {all | errors | events | info | pss}
```

```
no debug bootvar {all | errors | events | info | pss}
```

シンタックスの説明

all	すべてのブート変数のデバッグ オプションをイネーブルにします。
errors	ブート変数エラーに対するデバッグをイネーブルにします。
events	ブート変数イベントに対するデバッグをイネーブルにします。
info	ブート変数情報に対するデバッグをイネーブルにします。
pss	ブート変数 PSS 操作に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug bootvar all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug bootvar all
```

関連コマンド

コマンド	説明
debug all	スイッチ上のすべての機能のデバッグをイネーブルにします。
show boot	モジュールのブート変数を表示します。

debug callhome

Call Home 機能のデバッグをイネーブルにするには、EXEC モードで **debug callhome** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug callhome {all | events | mts}
```

```
no debug callhome {all | events | mts}
```

シンタックスの説明

all	すべての Call Home 機能に対するデバッグをイネーブルにします。
events	すべての Call Home イベントに対するデバッグをイネーブルにします。
mts	すべての Messaging Transaction Service (MTS) の Call Home tx/rx パケットに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

debug callhome コマンドを **all** パラメータと一緒に使用すると、Call Home 機能が受信する Call Home のイベント追跡および MTS メッセージのダンプのトラブルシューティングが表示されます。



(注)

Call Home のデバッグ機能は、成功または失敗した Call Home 電子メール送信に対するイベントの追跡を表示します。

例 次に、**debug callhome events** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug callhome events
2005-03-09T05:37:21 2005 Mar 9 05:37:21 callhome: filling in name field with Test
2005 Mar 9 05:37:21 callhome: filling in the header list
2005 Mar 9 05:37:21 callhome: filling up the chassis list
2005 Mar 9 05:37:21 callhome: filling up the main body list
2005 Mar 9 05:37:21 callhome: filling up the fru list 2005 Mar 9 05:37:21 callhome:
Entering function do_event_correlation
2005 Mar 9 05:37:21 callhome: getting dest profiles for alert group test
2005 Mar 9 05:37:21 callhome: getting dest profiles for alert group cisco-tac
2005 Mar 9 05:37:21 callhome: Applying the event rule for destination profile
full_txt
2005 Mar 9 05:37:21 callhome: Applying the event rule for destination profile
short_txt
2005 Mar 9 05:37:21 callhome: Applying the event rule for destination profile xml
2005 Mar 9 05:37:21 callhome: Applying the event rule for destination profile basu
2005 Mar 9 05:37:21 callhome: Exiting function do_event_correlation
2005 Mar 9 05:37:21 callhome: running cli commands for alert name : Test, message id
: 1540383426
2005 Mar 9 05:37:21 callhome: process scheduled for running cli commands for alert
Test, message id 1540383426, destination profile basu
2005 Mar 9 05:37:21 callhome: process scheduled for running cli commands for alert
Test, message id 1540383426, destination profile xml
2005 Mar 9 05:37:21 callhome: process scheduled for running cli commands for alert
Test, message id 1540383426, destination profile short_txt
.
.
.
```

次に、**debug callhome mts** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug callhome mts
Apr 8 13:09:42 callhome: Src: 0x00000501/4067 Dst: 0x00000501/66 ID: 0x0004FA
0D Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x0004FA0D HA_SEQNO:
0x00000000 TS: 0x86708AFE37B REJ:0
Apr 8 13:09:42 callhome: 00 00 00 00 00 06 00 00 00 00 00 00 00 00 00 00
Apr 8 13:09:42 callhome: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 13:09:42 callhome: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
...
Apr 8 13:09:42 callhome: Src: 0x00000501/4067 Dst: 0x00000501/66 ID: 0x0004FA
10 Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x0004FA10 HA_SEQNO:
0x00000000 TS: 0x86708D6A974 REJ:0
Apr 8 13:09:42 callhome: 00 00 00 00 05 00 00 00 00 00 00 00 00 00 00 00
Apr 8 13:09:42 callhome: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 13:09:42 callhome: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
...
.
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show callhome	スイッチに設定された Call Home 情報を表示します。

debug cert-enroll

証明書登録デーモンに対するデバッグをイネーブルにするには、EXEC モードで **debug cert-enroll** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

debug cert-enroll {all | config | config-lowlevel | request | request-lowlevel}

no debug cert-enroll {all | config | config-lowlevel | request | request-lowlevel}

シンタックスの説明	
all	すべてのデバッグフラグをイネーブルにします。
config	証明書登録コンフィギュレーションに対するデバッグをイネーブルにします。
config-lowlevel	証明書登録コンフィギュレーションに対する低レベルのデバッグをイネーブルにします。
request	証明登録要求に対するデバッグをイネーブルにします。
request-lowlevel	証明登録要求に対する低レベルのデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	3.0(1)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug cert-enroll all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug cert-enroll all
2006 Jan 21 00:44:52.875125 cert_enroll: cert_en_debug_conf_open: entering...
2006 Jan 21 00:44:52.875602 cert_enroll: cert_en_debug_conf_open: exiting
2006 Jan 21 00:44:52.876284 cert_enroll: cert_en_conf_close: entering...
2006 Jan 21 00:44:52.876349 cert_enroll: cert_en_conf_close: returning 0
2006 Jan 21 00:44:52.876400 cert_enroll: cert_en_enable_info_config: entering for
Cert-enroll Daemon debug
2006 Jan 21 00:44:52.876428 cert_enroll: cert_en_debug_conf_open: entering...
2006 Jan 21 00:44:52.876679 cert_enroll: cert_en_debug_conf_open: exiting
sw-46-180# 2006 Jan 21 00:44:52.876712 cert_enroll: cert_en_enable_info_config:
SET_REQ for Cert-enroll Daemon debug with 1
2006 Jan 21 00:44:52.876857 cert_enroll: cert_en_enable_info_config: SET_REQ done for
Cert-enroll Daemon debug with 1
2006 Jan 21 00:44:52.876896 cert_enroll: cert_en_enable_info_config: got back the
return value of configuration operation:success
2006 Jan 21 00:44:52.876922 cert_enroll: cert_en_debug_conf_close: entering...
2006 Jan 21 00:44:52.876965 cert_enroll: cert_en_debug_conf_close: returning 0
2006 Jan 21 00:44:52.876991 cert_enroll: cert_en_enable_info_config: exiting for
Cert-enroll Daemon debug...
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show crypto ca certificates	設定済トラストポイント証明書を表示します。

debug cdp

Cisco Discovery Protocol (CDP) 機能のデバッグをイネーブルにするには、EXEC モードで **debug cdp** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug cdp {all | errors | events {mts | packets | pss}} [interface {gigabitethernet slot/port | mgmt 0}]
```

```
no debug cdp {all | errors | events {mts | packets | pss}} [interface {gigabitethernet slot/port | mgmt 0}]
```

シンタックスの説明

all	すべての CDP 機能に対するデバッグをイネーブルにします。
errors	CDP のエラー状態に対するデバッグをイネーブルにします。
events	CDP イベントに対するデバッグをイネーブルにします。
mts	CDP tx/rx MTS パケットに対するデバッグをイネーブルにします。
packets	CDP tx/rx CDP パケットに対するデバッグをイネーブルにします。
pss	PSS に関連したすべての CDP イベントに対するデバッグをイネーブルにします。
interface	特定のインターフェイスに対するデバッグを指定します。
gigabitethernet slot/port	ギガビット イーサネット インターフェイスのスロットおよびポートを指定します。
mgmt 0	管理インターフェイスを指定します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.1(1)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug cdp events packets** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug cdp events packets
Apr  8 21:22:34 cdp: Sent CDP packet, interface 0x2380000
Apr  8 21:22:34 cdp: Sent CDP packet, interface 0x2381000
Apr  8 21:22:35 cdp: Sent CDP packet, interface 0x2382000
Apr  8 21:22:35 cdp: Sent CDP packet, interface 0x2383000
Apr  8 21:22:51 cdp: Received CDP packet, interface 0x5000000
Apr  8 21:23:01 cdp: Sent CDP packet, interface 0x5000000
Apr  8 21:23:34 cdp: Sent CDP packet, interface 0x2380000
Apr  8 21:23:34 cdp: Sent CDP packet, interface 0x2381000
Apr  8 21:23:35 cdp: Sent CDP packet, interface 0x2382000
...
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show cdp	グローバルまたは特定のインターフェイスに対して設定した CDP パラメータを表示します。

debug cfs

Cisco Fabric Services (CFS) に対するデバッグをイネーブルにするには、EXEC モードで **debug cfs** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug cfs {all | errors | events {db [vsan vsan-id] | fc2 [vsan vsan-id] | fsm-action [vsan vsan-id] |
  fsm-trans [sap sap-id] | mts [vsan vsan-id] | pss [vsan vsan-id]} | fsm {ha | trans} | merge}
```

```
no debug cfs {all | errors | events {db [vsan vsan-id] | fc2 [vsan vsan-id] | fsm-action [vsan vsan-id] |
  fsm-trans [sap sap-id] | mts [vsan vsan-id] | pss [vsan vsan-id]} | fsm {ha | trans} | merge}
```

シンタックスの説明

all	すべての CFS デバッグをイネーブルにします。
errors	CFS のエラー状態に対するデバッグをイネーブルにします。
events	CFS イベントに対するデバッグをイネーブルにします。
db	CFS データベース イベントに対するデバッグをイネーブルにします。
vsan vsan-id	指定 VSAN (仮想 SAN) ID へのデバッグを制限します。有効範囲は 1 ~ 4093 です。
fc2	CFS FC2 イベントに対するデバッグをイネーブルにします。
fsm-action	CFS FSM アクション イベントに対するデバッグをイネーブルにします。
fsm-trans	CFS FSM 移行イベントに対するデバッグをイネーブルにします。
sap sap-id	指定 SAP ID へのデバッグを制限します。有効範囲は 0 ~ 2147483647 です。
mts	CFS MTS イベントに対するデバッグをイネーブルにします。
pss	CFS PSS イベントに対するデバッグをイネーブルにします。
fsm	CFS FSM イベントに対するデバッグをイネーブルにします。
ha	CFS FSM ハイ アベイラビリティ イベントに対するデバッグをイネーブルにします。
trans	CFS FSM 移行イベントに対するデバッグをイネーブルにします。
merge	CFS 結合イベントに対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug cfs all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug cfs all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show cfs	CFS 情報を表示します。

debug cimserver

Common Information Model (CIM) 管理アプリケーション機能のデバッグをイネーブルにするには、EXEC モードで **debug cimserver** コマンドを使用します。debug コマンドをディセーブルにするには、このコマンドの no 形式を使用するか、no debug all コマンドを使用してすべてのデバッグを終了させます。

```
debug cimserver {all | errors | events | mts | trace}
```

```
no debug cimserver {all | errors | events | mts | trace}
```

シンタックスの説明

all	すべての CIM 機能に対するデバッグをイネーブルにします。
errors	CIM のエラー状態に対するデバッグをイネーブルにします。
events	CIM イベントに対するデバッグをイネーブルにします。
mts	CIM tx/rx MTS パケットに対するデバッグをイネーブルにします。
trace	CIM トレースに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(1)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug cimserver all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug cimserver all
2004 Mar 29 20:05:22 cimsrvprov: cim_mts_dispatch(): Opcode is 182
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show cimserver	CIM 設定を表示します。

debug cloud

クラウド検出のデバッグをイネーブルにするには、EXEC モードで **debug cloud** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug cloud {all | bypass ficon_mgr | cloud | conditional | demux vsan vsan-id | deque | discovery |
error | event vsan vsan-id | ha vsan vsan-id | init | member | memory | messages | remotesync |
trace [detail vsan vsan-id | vsan vsan-id] | warning [vsan-id] | xipc | xipc}
```

```
no debug cloud {all | bypass ficon_mgr | cloud | conditional | demux vsan vsan-id | deque | discovery |
error | event vsan vsan-id | ha vsan vsan-id | init | member | memory | messages | remotesync |
trace [detail vsan vsan-id | vsan vsan-id] | warning [vsan-id] | xipc | xipc}
```

シンタックスの説明

all	クラウドのすべての機能のデバッグをイネーブルにします。
bypass	デバッグ中に、クラウドの実行において一部のコンポーネントを回避できます。
ficon_mgr	デバッグ中に FICON マネージャを回避できます。
cloud	すべてのクラウド コマンドのデバッグをイネーブルにします。
conditional	クラウド検出コンディショナル サービスに対するデバッグをイネーブルにします。
demux	クラウド メッセージ デマルチプレクサに対するデバッグをイネーブルにします。
vsan vsan-id	指定 VSAN (仮想 SAN) ID へのデバッグを制限します。有効範囲は 1 ~ 4094 です。
deque	クラウド メッセージ デキューに対するデバッグをイネーブルにします。
discovery	検出プロセスに対するデバッグをイネーブルにします。
error	クラウド エラーに対するデバッグをイネーブルにします。
event	クラウドの Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	クラウドのハイ アベイラビリティ (HA) に対するデバッグをイネーブルにします。
init	クラウド検出の初期化に対するデバッグをイネーブルにします。
member	クラウドのメンバー変更に対するデバッグをイネーブルにします。
memory	クラウドのメモリ割り当てに対するデバッグをイネーブルにします。
messages	クラウド検出の MTS メッセージに対するデバッグをイネーブルにします。
remotesync	検出のリモート同期に対するデバッグをイネーブルにします。
trace	クラウド トレースに対するデバッグをイネーブルにします。
detail	クラウドの詳細トレースに対するデバッグをイネーブルにします。
warning	クラウド警告のデバッグをイネーブルにします。
xipc	XIPC メッセージに対するデバッグをイネーブルにします。
xipc	クラウドのデータ シリアライゼーションに対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴	リリース	変更内容
	3.0(1)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug cloud all** コマンドによるシステムの出力結果の例を示します。

```
switch# debug cloud all
1980 Feb 15 22:03:41.650721 cloud: fu_fsm_execute_all: match_msg_id(0), log_alre
ady_open(0)
1980 Feb 15 22:03:41.650874 cloud: fu_fsm_execute_all: null fsm_event_list
1980 Feb 15 22:03:41.650956 cloud: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 1302150) dropped
1980 Feb 15 22:03:41.651000 cloud: cloud_deque
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show cloud discovery	クラウド検出情報を表示します。
	show cloud membership	クラウドのメンバーに関する情報を表示します。

debug core

コア デーモンのデバッグをイネーブルにするには、EXEC モードで **debug core** コマンドを使用します。debug コマンドをディセーブルにするには、このコマンドの no 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

debug core {error | flow}

no debug core {error | flow}

シンタックスの説明	error	flow
	コア デーモンのエラー状態に対するデバッグをイネーブルにします。	コア デーモン フローに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug core flow** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug core flow
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show cores	現在アクティブ スーパーバイザからアップロード可能なすべてのコアを表示します。

debug device-alias

デバイスエイリアスに対するデバッグをイネーブルにするには、EXEC モードで **debug device-alias** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug device-alias {all | database {detail | errors | events} | fsm | ha | import {errors | events} |
merge {errors | events | packets} | pss {errors | events} | session {errors | events | packets} | trace}
no debug device-alias {all | database {detail | errors | events} | fsm | ha | import {errors | events} |
merge {errors | events | packets} | pss {errors | events} | session {errors | events | packets} | trace}
```

シンタックスの説明

all	すべてのデバイスエイリアスのデバッグをイネーブルにします。
database	デバイスエイリアス データベース イベントに対するデバッグをイネーブルにします。
detail	デバイスエイリアス データベース イベントに対する詳細デバッグをイネーブルにします。
errors	デバイスエイリアスのエラー状態に対するデバッグをイネーブルにします。
events	デバイスエイリアス イベントに対するデバッグをイネーブルにします。
fsm	デバイスエイリアス Finite State Machine (FSM) イベントに対するデバッグをイネーブルにします。
ha	デバイスエイリアス HA (ハイアベイラビリティ) イベントに対するデバッグをイネーブルにします。
import	デバイスエイリアス インポートに対するデバッグをイネーブルにします。
merge	デバイスエイリアス結合に対するデバッグをイネーブルにします。
packets	デバイスエイリアス パケットに対するデバッグをイネーブルにします。
pss	デバイスエイリアス PSS に対するデバッグをイネーブルにします。
session	デバイスエイリアス セッションに対するデバッグをイネーブルにします。
trace	デバイスエイリアス トレースに対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug device-alias all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug device-alias all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show device-alias	デバイスエイリアス情報を表示します。

debug dpvm

Dynamic Port VSAN Membership (DPVM) のデバッグをイネーブルにするには、EXEC モードで **debug dpvm** コマンドを使用します。 **debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug dpvm {all | cfs-events | change-events | db-events | errors | ftrace | merge-event | mts-events |
pss-events | session-events | snmp-events | sys-events}
```

```
no debug dpvm {all | cfs-events | change-events | db-events | errors | ftrace | merge-event | mts-events
| pss-events | session-events | snmp-events | sys-events}
```

シンタックスの説明

all	すべての DPVM に対するデバッグをイネーブルにします。
cfs-events	Cisco Fabric Services (CFS) に対するデバッグをイネーブルにします。
change-events	変更イベントに対するデバッグをイネーブルにします。
db-events	データベース イベントに対するデバッグをイネーブルにします。
errors	エラーに対するデバッグをイネーブルにします。
ftrace	機能トレースに対するデバッグをイネーブルにします。
merge-event	結合イベントに対するデバッグをイネーブルにします。
mts-events	MTS イベントに対するデバッグをイネーブルにします。
pss-events	PSS イベントに対するデバッグをイネーブルにします。
session-events	セッション イベントに対するデバッグをイネーブルにします。
snmp-events	SNMP (簡易ネットワーク管理プロトコル) イベントに対するデバッグをイネーブルにします。
sys-events	システム イベントに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

このコマンドを使用するには、**dpvm enable** コマンドを使用して DPVM をイネーブルにする必要があります。

例

次に、**debug dpvm all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug dpvm all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show dpvm	DPVM データベース情報を表示します。

debug dstats

デルタ統計情報のデバッグをイネーブルにするには、EXEC モードで **debug dstats** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug dstats {error | flow}
```

```
no debug dstats {error | flow}
```

シンタックスの説明	error	デルタ統計情報のエラー状態に対するデバッグをイネーブルにします。
	flow	デルタ統計情報フローに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug dstats flow** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug dstats flow
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug ethport

イーサネット ポートのデバッグをイネーブルにするには、EXEC モードで **debug ethport** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ethport {all | error | event [interface gigabitethernet slot/port | module slot] |
  ha [interface gigabitethernet slot/port | module slot] |
  trace [interface gigabitethernet slot/port | module slot]}
```

```
no debug ethport {all | error | event [interface gigabitethernet slot/port | module slot] |
  ha [interface gigabitethernet slot/port | module slot] |
  trace [interface gigabitethernet slot/port | module slot]}
```

シンタックスの説明

all	すべてのイーサネット ポート機能に対するデバッグをイネーブルにします。
error	イーサネット ポートのエラー状態に対するデバッグをイネーブルにします。
event	イーサネット ポート イベントに対するデバッグをイネーブルにします。
ha	ポートのハイ アベイラビリティに対するデバッグをイネーブルにします。
trace	イーサネット ポートのトレースに対するデバッグをイネーブルにします。
interface gigabitethernet slot/port	ギガビット イーサネット インターフェイスのスロットおよびポートを指定します。
module slot	デバッグするモジュールのスロット番号を指定します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug ethport all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ethport all
1981 May  5 07:28:59 ethport: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
1981 May  5 07:28:59 ethport: fu_fsm_execute_all: null_fsm_event_list
1981 May  5 07:28:59 ethport: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 52343) dropped
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug exceptionlog

例外ログデバッグ機能をイネーブルにするには、EXECモードで **debug exceptionlog** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug exceptionlog {demux | deque | error | flow | info}
```

```
no debug exceptionlog {demux | deque | error | flow | info}
```

シンタックスの説明

demux	例外ログ デマルチプレクサ機能に対するデバッグをイネーブルにします。
deque	例外ログ デキュー機能に対するデバッグをイネーブルにします。
error	例外ログ エラーに対するデバッグをイネーブルにします。
flow	例外ログ フローに対するデバッグをイネーブルにします。
info	例外ログ情報に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug exceptionlog** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug exceptionlog
7), credit(3), empty
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug fabric-binding

ファブリック バインディング機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fabric-binding** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fabric-binding {all | efmd {db-events | errors | merge {errors | events | packets}} | mts-events
| pss-events} | errors [vsan vsan-id] | events [vsan vsan-id] | mts-events | pss-events | snmp-events
| trace [vsan vsan-id]}
```

```
no debug fabric-binding {all | efmd {db-events | errors | merge {errors | events | packets}} |
mts-events | pss-events} | errors [vsan vsan-id] | events [vsan vsan-id] | mts-events | pss-events |
snmp-events | trace [vsan vsan-id]}
```

シンタックスの説明

all	すべてのファブリック バインディング機能に対するデバッグをイネーブルにします。
efmd	Exchange Fabric Membership Data (EFMD) プロトコルに対するデバッグをイネーブルにします。
db-events	EFMD プロトコル データベース イベントに対するデバッグをイネーブルにします。
merge	EFMD プロトコル結合に対するデバッグをイネーブルにします。
packets	EFMD プロトコル パケットに対するデバッグをイネーブルにします。
errors	ファブリック バインディング エラーに対するデバッグをイネーブルにします。
vsan vsan-id	VSAN (仮想 SAN) ID を指定します。有効範囲は 1 ~ 4093 です。
events	ファブリック バインディング イベントに対するデバッグをイネーブルにします。
mts-events	ファブリック バインディング Messaging and Transaction Service (MTS) イベントに対するデバッグをイネーブルにします。
pss-events	ファブリック バインディング PSS イベントに対するデバッグをイネーブルにします。
snmp-events	ファブリック バインディング SNMP (簡易ネットワーク管理プロトコル) イベントに対するデバッグをイネーブルにします。
trace	ファブリック バインディング トレースに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、**debug fabric-binding all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fabric-binding all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fabric-binding	設定したファブリック バインディングの情報を表示します。

debug fc-tunnel

ファイバチャネルトンネル機能に対するデバッグをイネーブルにするには、EXECモードで **debug fc-tunnel** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fc-tunnel {all | errors | external-events | ha | label-update | mts {pkt | pkthdr} {both | rx | tx} |
pss | route-update [vsan vsan-id] | rsvp-messages [tunnel tunnel-id | vsan vsan-id] | state-machine}
```

```
no debug fc-tunnel {all | errors | external-events | ha | label-update | mts {pkt | pkthdr} {both | rx |
tx} | pss | route-update [vsan vsan-id] | rsvp-messages [tunnel tunnel-id | vsan vsan-id] |
state-machine}
```

シンタックスの説明

all	すべての Fibre Channel (FC) トンネル機能に対するデバッグをイネーブルにします。
errors	FC トンネルエラーに対するデバッグをイネーブルにします。
external-events	FC トンネルイベントに対するデバッグをイネーブルにします。
ha	FC トンネル HA (ハイアベイラビリティ) イベントに対するデバッグをイネーブルにします。
label-update	FC トンネルラベル更新に対するデバッグをイネーブルにします。
mts	FC トンネル Messaging and Transaction Service (MTS) イベントに対するデバッグをイネーブルにします。
pkt	パケットに対するデバッグを指定します。
pkthdr	ヘッダーに対するデバッグを指定します。
both	送受信の両方向に対するデバッグを指定します。
tx	送信方向に対するデバッグを指定します。
rx	受信方向に対するデバッグを指定します。
pss	FC トンネル PSS イベントに対するデバッグをイネーブルにします。
route-update	FC トンネルルート更新に対するデバッグをイネーブルにします。
vsan vsan-id	VSAN (仮想 SAN) ID を指定します。有効範囲は 1 ~ 4093 です。
rsvp-messages	FC トンネル SNMP (簡易ネットワーク管理プロトコル) イベントに対するデバッグをイネーブルにします。
tunnel tunnel-id	トンネル ID を指定します。有効範囲は 1 ~ 255 です。
state-machine	FC トンネルトレースに対するデバッグをイネーブルにします。
node	受信方向のパケットのノードを指定します。
opcode	受信方向のパケットの opcode を指定します。
sap	受信方向のパケットの sap を指定します。
range	1 ~ 4096 までの整数の範囲を指定します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、**debug fc-tunnel all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug fc-tunnel all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fc-tunnel	設定された FC トンネル情報を表示します。

debug fc2

FC2 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fc2** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fc2 {credit |
  error [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]flag |
  flow [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]] (interface
  fc type number | vsan vsan-id) |frame |loopback |
  pkt {both | tx | rx} [bytes bytes | fcid fcid [bytes bytes | interface {fc slot/port | fcip port} [bytes
  bytes | pkts pkts [bytes bytes]]] | pkts pkts [bytes bytes] | vsan vsan-id [bytes bytes | interface {fc
  slot/port | fcip port} [bytes bytes | pkts pkts [bytes bytes]]] |
  pkthdr {both | tx | rx} [bytes bytes | fcid fcid [bytes bytes | interface {fc slot/port | fcip port} [bytes
  bytes | pkts pkts [bytes bytes]]] | pkts pkts [bytes bytes] | vsan vsan-id [bytes bytes | interface {fc
  slot/port | fcip port} [bytes bytes | pkts pkts [bytes bytes]]] | rdl |
  rxhdrhistory [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port |
  fcip port}]]] | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  txhdrhistory [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port |
  fcip port}]]] | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]]

no debug fc2 {credit |
  error [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]flag |
  flow [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]] (interface
  fc type number | vsan vsan-id) |frame |loopback |
  pkt {both | tx | rx} [bytes bytes | fcid fcid [bytes bytes | interface {fc slot/port | fcip port} [bytes
  bytes | pkts pkts [bytes bytes]]] | pkts pkts [bytes bytes] | vsan vsan-id [bytes bytes | interface {fc
  slot/port | fcip port} [bytes bytes | pkts pkts [bytes bytes]]] |
  pkthdr {both | tx | rx} [bytes bytes | fcid fcid [bytes bytes | interface {fc slot/port | fcip port} [bytes
  bytes | pkts pkts [bytes bytes]]] | pkts pkts [bytes bytes] | vsan vsan-id [bytes bytes | interface {fc
  slot/port | fcip port} [bytes bytes | pkts pkts [bytes bytes]]] | rdl |
  rxhdrhistory [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port |
  fcip port}]]] | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]
  txhdrhistory [fcid fcid [interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port |
  fcip port}]]] | interface {fc slot/port | fcip port} | vsan vsan-id [interface {fc slot/port | fcip port}]]]
```

シンタックスの説明

credit	FC2 クレジットに対するデバッグをイネーブルにします。
error	FC2 エラーに対するデバッグをイネーブルにします。
fcid fcid	指定の Fibre Channel ID (FCID) に対するデバッグを制限します。
interface	指定のインターフェイスにデバッグを制限します。
fc slot/port	指定のインターフェイスにデバッグを制限します。
fcip port	指定のインターフェイスにデバッグを制限します。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
flag	FC2 フラグに対するデバッグをイネーブルにします。
flow	FC2 フローに対するデバッグをイネーブルにします。
frame	FC2 フレームに対するデバッグをイネーブルにします。
loopback	FC2 ループバックに対するデバッグをイネーブルにします。

pkt	FC パケットに対するデバッグをイネーブルにします。
both	送受信の両方向に対するデバッグをイネーブルにします。
tx	送信方向のデバッグをイネーブルにします。
rx	受信方向のデバッグをイネーブルにします。
bytes bytes	表示バイト数を指定します。
pkts pkts	表示パケット数を指定します。
pkthdr	FC ヘッダーに対するデバッグをイネーブルにします。
rdl	FC2 Remote Digital Loopback (RDL) に対するデバッグをイネーブルにします。
rxhdrhistory	FC2 の受信ヘッダーの履歴に対するデバッグをイネーブルにします。
txhdrhistory	FC2 の送信ヘッダーの履歴に対するデバッグをイネーブルにします。

デフォルト デイセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン Fabric Shortest Path First (FSPF) が不正な FC2 パケットを受信した場合、**debug fc2 pkt** コマンドの出力結果を分析します。

例 次に、**debug fc2 error vsan 1** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch1# debug fc2 error vsan 1
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをデイセーブルにします。
	show fc2	FC2 情報を表示します。

debug fc2d

FC2 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fc2** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fc2 {all | bypass ficon_mgr | demux [vsan vsan-id] | deque | error | event [vsan vsan-id] | ha
[vsan vsan-id] | trace [detail] [vsan vsan-id] | warning [vsan vsan-id]}
```

```
no debug fc2 {all | bypass ficon_mgr | demux [vsan vsan-id] | deque | error | event [vsan vsan-id] | ha
[vsan vsan-id] | trace [detail] [vsan vsan-id] | warning [vsan vsan-id]}
```

シンタックスの説明

all	すべての FC2D デバッグ フラグをイネーブルにします。
bypass	FC2D の実行において一部のコンポーネントを回避できます。
ficon_mgr	FC2D の実行において Fibre Connection (FICON) Manager を回避できます。
demux	FC2D メッセージ デマルチプレクサに対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
deque	FC2D メッセージ デキューに対するデバッグをイネーブルにします。
error	FC2D エラーに対するデバッグをイネーブルにします。
event	FC2D Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	FC2D HA (ハイ アベイラビリティ) に対するデバッグをイネーブルにします。
trace	FC2D トレースに対するデバッグをイネーブルにします。
detail	FC2D トレースに対する詳細デバッグをイネーブルにします。
warning	FC2D 警告に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(4)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug fc2d all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch1# debug fc2d all
2004 Mar 29 22:57:25 fc2d: fu_fsm_execute_all: match_msg_id(0), log_already_open (0)
2004 Mar 29 22:57:25 fc2d: fu_fsm_execute_all: null_fsm_event_list
2004 Mar 29 22:57:25 fc2d: fu_fsm_engine_post_event_processing: mts_msg MTS_OPC_
DEBUG_WRAP_MSG(msg_id 6894921) dropped
```

関連コマンド

コマンド	説明
debug all	FC2 機能に対するデバッグをイネーブルにします。
no debug all	すべてのデバッグをディセーブルにします。
show fc2	FC2 情報を表示します。

debug fcc

Fibre Channel Congestion (FCC) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fcc** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcc {all | error [module slot] | event [module slot] |
  mts [pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | tx | rx [numpkt
  range]}] | trace [module slot]}

no debug fcc {all | error [module slot] | event [module slot] |
  mts [pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | tx | rx [numpkt
  range]}] | trace [module slot]}
```

シンタックスの説明

all	すべての FCC 機能に対するデバッグをイネーブルにします。
errors	Fibre Channel Congestion (FCC) のエラー状態に対するデバッグをイネーブルにします。
events	FCC イベントに対するデバッグをイネーブルにします。
mts	FCC tx/rx MTS パケットに対するデバッグをイネーブルにします。
trace	FCC トレースに対するデバッグをイネーブルにします。
module slot	デバッグするモジュールのスロット番号を指定します。
pkt	FCC tx/rx FCC パケットに対するデバッグをイネーブルにします。
pkthdr	FCC tx/rx FCC ヘッダーに対するデバッグをイネーブルにします。
numpkt	必要なパケット数を指定します。
both	送受信の両方向に対するデバッグを指定します。
tx	送信方向に対するデバッグを指定します。
rx	受信方向に対するデバッグを指定します。
node	受信方向のパケットのノードを指定します。
opcode	受信方向のパケットの opcode を指定します。
sap	受信方向のパケットの sap を指定します。
range	1 ~ 4096 までの整数の範囲を指定します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

■ debug fcc

使用上のガイドライン なし

例 次に、**debug fcc all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcc all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fcc	FCC 設定を表示します。

debug fcdomain

fcdomain 機能のデバッグをイネーブルにするには、EXEC モードで **debug fcdomain** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcdomain {all | critical | error |
  fc {pkt | pkthdr} {both | rx | tx} [interface type number [vsan vsan-id] | vsan vsan-id] |
  ipc {pkt | pkthdr} {both | rx [node range | opcode range | sap range] | tx} |
  memory | notify | phase}
no debug fcdomain {all | critical | error |
  fc {pkt | pkthdr} {both | rx | tx} [interface type number [vsan vsan-id] | vsan vsan-id] |
  ipc {pkt | pkthdr} {both | rx [node range | opcode range | sap range] | tx} |
  memory | notify | phase}
```

シンタックスの説明

all	すべての fcdomain パラメータのデバッグをイネーブルにします。
critical	致命的な操作に対するデバッグをイネーブルにします。
error	エラー操作に対するデバッグをイネーブルにします。
fc	ファイバ チャネル パケットおよびヘッダーに対するデバッグをイネーブルにします。
fcip	ファイバ チャネル IP パケットおよびヘッダーに対するデバッグをイネーブルにします。
port-channel	ポートチャネル パケットおよびヘッダーに対するデバッグをイネーブルにします。
pkt	パケットに対するデバッグをイネーブルにします。
pkthdr	ヘッダーに対するデバッグをイネーブルにします。
both	送受信の両方向に対するデバッグをイネーブルにします。
rx	受信方向のデバッグをイネーブルにします。
interface type number	デバッグするインターフェイスを指定します。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
tx	送信方向のデバッグをイネーブルにします。
memory	メモリ操作に対するデバッグをイネーブルにします。
notify	通知に対するデバッグをイネーブルにします。
phase	グローバル フェーズに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、**debug fcdomain critical** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcdomain critical
Jan 27 07:04:31 fcdomain: Src: 0x00000501/6243 Dst: 0x00000501/14 ID: 0x0005BF
41 Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x0005BF41 HA_SEQNO:
0x00000000 TS: 0x183C4D027F4A3
Jan 27 07:04:31 fcdomain: 00 00 00 00 68 00 00 00 00 00 00 00 00 00 00 00
Jan 27 07:04:31 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Jan 27 07:04:31 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
Jan 27 07:04:31 fcdomain: 2F 64 65 76 2F 70 74 73 2F 30 00 00 00 00 00 00
Jan 27 07:04:31 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
...
```

次に、**debug fcdomain error** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcdomain error
Jan 27 07:05:29 fcdomain: Src: 0x00000501/6245 Dst: 0x00000501/14 ID: 0x0005BF
7E Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x0005BF7E HA_SEQNO:
0x00000000 TS: 0x183D5E63C081A
Jan 27 07:05:29 fcdomain: 00 00 00 00 64 00 00 00 00 00 00 00 00 00 00 00
Jan 27 07:05:29 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Jan 27 07:05:29 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
Jan 27 07:05:29 fcdomain: 2F 64 65 76 2F 70 74 73 2F 30 00 00 00 00 00 00
...
```

次に、**debug fcdomain ipc pkthdr both** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcdomain ipc pkthdr both
Apr 8 20:44:38 fcdomain: Src: 0x00000501/3883 Dst: 0x00000501/14 ID: 0x00038E
1D Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x00038E1D HA_SEQNO:
0x00000000 TS: 0x5DD9B14EA3AA REJ:0
Apr 8 20:44:38 fcdomain: 00 00 00 00 08 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
Apr 8 20:44:38 fcdomain: 2F 64 65 76 2F 70 74 73 2F 30 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
...
Apr 8 20:44:38 fcdomain: Src: 0x00000501/3883 Dst: 0x00000501/14 ID: 0x00038E
20 Size: 252 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x00038E20 HA_SEQNO:
0x00000000 TS: 0x5DD9B186CCEB REJ:0
Apr 8 20:44:38 fcdomain: 00 00 00 00 07 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF
Apr 8 20:44:38 fcdomain: 2F 64 65 76 2F 70 74 73 2F 30 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 8 20:44:38 fcdomain: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
...
```

関連コマンド

コマンド	説明
show fcdomain domain-list	ファブリックにおける現在のドメインを表示します。
fcdomain	fcdomain 機能をイネーブルにします。

debug fcfwd

ファイバチャネル転送機能に対するデバッグをイネーブルにするには、EXECモードで **debug fcfwd** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcfwd {flogimap | idxmap | pemap | sfib | spanmap} {error | event | trace} [module slot | vsan
vsan-id [module slot]]
```

```
no debug fcfwd {flogimap | idxmap | pemap | sfib | spanmap} {error | event | trace} [module slot |
vsan vsan-id [module slot]]
```

シンタックスの説明

flogimap	flogimap に対するデバッグをイネーブルにします。
idxmap	idxmap に対するデバッグをイネーブルにします。
pemap	pemap に対するデバッグをイネーブルにします。
sfib	sfib に対するデバッグをイネーブルにします。
spanmap	spanmap に対するデバッグをイネーブルにします。
error	Fibre Channel Congestion (FCC) のエラー状態に対するデバッグをイネーブルにします。
event	FCC イベントに対するデバッグをイネーブルにします。
trace	FCC トレースに対するデバッグをイネーブルにします。
module slot	デバッグするモジュールのスロット番号を指定します。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug fcfwd error** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcfwd error
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fcfwd	設定した fcfwd テーブルおよび統計情報を表示します。

debug fcns

ネーム サーバの登録に対するデバッグをイネーブルにするには、EXEC モードで **debug fcns** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcns {all | errors | events {mts | query | register}} [vsan vsan-id]
```

```
no debug fcns {all | errors | events {mts | query | register}} [vsan vsan-id]
```

シンタックスの説明

all	すべてのネーム サーバ機能のデバッグをイネーブルにします。
errors	ネーム サーバのエラー状態に対するデバッグをイネーブルにします。
events	すべてのネーム サーバ イベントに対するデバッグをイネーブルにします。
mts	ネーム サーバ tx/rx MTS パケットに対するデバッグをイネーブルにします。
query	ネーム サーバ tx/rx CDP パケットに対するデバッグをイネーブルにします。
register	ネーム サーバの PSS 関連のイベントに対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug fcns events register vsan 99** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcns events register vsan 99
Feb 17 04:42:54 fcns: vsan 99: Got Entry for port-id 27800
Feb 17 04:42:54 fcns: vsan 99: Registered port-name 36a4078be0000021 for port-id 780200
Feb 17 04:42:54 fcns: vsan 99: Registered node-name 36a4078be0000020 for port-id 780200
...
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fcns database	指定の VSAN またはすべての VSAN に対する検出結果またはネーム サーバ データベースを表示します。
show fcns statistics	指定の VSAN またはすべての VSAN に対する統計情報を表示します。

debug fcs

Fabric Configuration Server (FCS) に対するデバッグをイネーブルにするには、EXEC モードで **debug fcs** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcs {all | discovery events | errors [vsan vsan-id] | ess-events [vsan vsan-id] |
  mts events {brief | detail} | pss events | queries events [vsan vsan-id] |
  registrations events [vsan vsan-id] | rscn events [vsan vsan-id] | snmp events}

no debug fcs {all | discovery events | errors [vsan vsan-id] | ess-events [vsan vsan-id] |
  mts events {brief | detail} | pss events | queries events [vsan vsan-id] |
  registrations events [vsan vsan-id] | rscn events [vsan vsan-id] | snmp events}
```

シンタックスの説明

all	すべての FCS 機能に対するデバッグをイネーブルにします。
discovery events	FCS 検出イベントに対するデバッグをイネーブルにします。
errors	FCS のエラー状態に対するデバッグをイネーブルにします。
mts events	FCS tx/rx MTS イベントに対するデバッグをイネーブルにします。
pss events	FCS に対するデバッグをイネーブルにします。
brief	各イベントに簡易情報を提供します。
detail	各イベントに詳細情報を提供します。
queries events	FCS tx/rx イベントに対するデバッグをイネーブルにします。
registration events	FCS PSS 関連のイベントに対するデバッグをイネーブルにします。
rscn events	FCS Registered State Change Notification (RSCN) イベントに対するデバッグをイネーブルにします。
snmp events	FCS SNMP (簡易ネットワーク管理プロトコル) イベントに対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug fcs all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fcs all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show fcs	ファブリック設定のステータスを表示します。

debug fcsp-mgr

Fibre Channel Security Protocol (FC-SP) マネージャに対するデバッグをイネーブルにするには、EXEC モードで **debug fcsp-mgr** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fcsp-mgr {all | critical | datastructure | dhchap | error | event-gen | fc2 | fsm | general | ha | init
| level1 | level2 | level3 | level4 | level5 | message | mts | notify | trace}
```

```
no debug fcsp-mgr {all | critical | datastructure | dhchap | error | event-gen | fc2 | fsm | general | ha |
init | level1 | level2 | level3 | level4 | level5 | message | mts | notify | trace}
```

シンタックスの説明

all	すべての FC-SP 機能に対するデバッグをイネーブルにします。
critical	FC-SP クリティカル エラーのデバッグをイネーブルにします。
datastructure	FC-SP データ構造のデバッグをイネーブルにします。
dhchap	Diffie-Hellman Challenge Handshake Authentication Protocol (DHCHAP) のデバッグをイネーブルにします。
error	FC-SP エラーのデバッグをイネーブルにします。
event-gen	FC-SP イベント生成のデバッグをイネーブルにします。
fc2	FC-SP FC2 メッセージのデバッグをイネーブルにします。
fsm	FC-SP イベントのデバッグをイネーブルにします。
general	FC-SP の一般的なデバッグをイネーブルにします。
ha	FC-SP ハイ アベイラビリティのデバッグをイネーブルにします。
init	FC-SP 初期化のデバッグをイネーブルにします。
level1	FC-SP Mgr のデバッグ レベルを 1 に設定します。
level2	FC-SP Mgr のデバッグ レベルを 2 に設定します。
level3	FC-SP Mgr のデバッグ レベルを 3 に設定します。
level4	FC-SP Mgr のデバッグ レベルを 4 に設定します。
level5	FC-SP Mgr のデバッグ レベルを 5 に設定します。
message	FC-SP メッセージのデバッグをイネーブルにします。
mts	FC-SP Messaging and Transaction Service (MTS) メッセージのデバッグをイネーブルにします。
notify	デバッグ レベルを通知に設定します。
trace	FC-SP 機能の開始 / 終了のデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、`debug fcsp-mgr all` コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug fcsp-mgr all
2004 Mar 29 23:33:56 fcsp-mgr: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
2004 Mar 29 23:33:56 fcsp-mgr: fu_fsm_execute_all: null_fsm_event_list
2004 Mar 29 23:33:56 fcsp-mgr: fu_fsm_engine_post_event_processing: mts msg MTS_
OPC_DEBUG_WRAP_MSG(msg_id 7061762) dropped
```

関連コマンド

コマンド	説明
<code>no debug all</code>	すべてのデバッグをディセーブルにします。
<code>show fcsp</code>	FC-SP 設定のステータスを表示します。

debug fdmi

Fabric-Device Management Interface (FDMI) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fdmi** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fdmi {all | errors | fdmi-messages [vsan vsan-id] | ha | mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | tx | rx [numpkt range]}} | pss | trace}
```

```
no debug fdmi {all | errors | fdmi-messages [vsan vsan-id] | ha | mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | tx | rx [numpkt range]}} | pss | trace}
```

シンタックスの説明

all	すべての FDMI 機能に対するデバッグをイネーブルにします。
errors	FDMI のエラー状態に対するデバッグをイネーブルにします。
fdmi-messages	FDMI Protocol Data Unit (PDU) のダンプをイネーブルにします。
ha	HA (ハイアベイラビリティ) 同期メッセージのダンプをイネーブルにします。
mts	FDMI tx/rx MTS イベントに対するデバッグをイネーブルにします。
pkt	FCC tx/rx FCC パケットに対するデバッグをイネーブルにします。
both	送受信の両方向に対するデバッグを指定します。
tx	送信方向に対するデバッグを指定します。
rx	受信方向に対するデバッグを指定します。
node	受信方向のパケットのノードを指定します。
range	1 ~ 4096 までの整数の範囲を指定します。
opcode	受信方向のパケットの opcode を指定します。
sap	受信方向のパケットの sap を指定します。
pkthdr	FCC tx/rx FCC ヘッダーに対するデバッグをイネーブルにします。
numpkt	必要なパケット数を指定します。
pss	FDMI PSS に対するデバッグをイネーブルにします。
trace	FDMI トレースに対するデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、`debug fdmi all` コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fdmi all
2005 Mar 10 02:37:28 fdmi: 00 00 00 02 00 00 00 1C 04 19 65 08 00 82 39 08
2005 Mar 10 02:37:28 fdmi: C4 16 65 08 44 19 65 08 E4 87 39 08 04 17 65 08
2005 Mar 10 02:37:28 fdmi: 84 19 65 08 4C 8D 39 08 44 17 65 08 C4 19 65 08
2005 Mar 10 02:37:28 fdmi: B4 92 39 08 00 17 65 08 04 1A 65 08 1C 98 39 08
2005 Mar 10 02:37:28 fdmi: C4 17 65 08 44 1A 65 08 84 9D 39 08 04 18 65 08
2005 Mar 10 02:37:28 fdmi: 84 1A 65 08 EC A2 39 08 44 18 65 08 C4 1A 65 08
2005 Mar 10 02:37:28 fdmi: 54 A8 39 08 84 18 65 08 04 1B 65 08 BC AD 39 08
2005 Mar 10 02:37:28 fdmi: 00 00 00 02 00 00 0B B8 00 00 00 00 00 00 00
2005 Mar 10 02:37:28 fdmi: 00 00 00 00 00 00 00 00 00 00 00 00
2005 Mar 10 02:37:28 fdmi: Src: 0x00000601/27 Dst: 0x00000601/105 ID: 0x0069E217
Size: 140 [REQ] Opc: 7804 (MTS_OPC_FDMI_SNMP) RR: 0x0069E217 HA_SEQNO: 0x00000000
TS: 0x25218CC5A40E3 REJ:0 SYNC:0
```

関連コマンド

コマンド	説明
<code>no debug all</code>	すべてのデバッグをディセーブルにします。
<code>show fdmi</code>	FDMI データベース情報を表示します。

debug ficon

FI-bre Con-nection (FICON; ファイバ接続) インターフェイスに対するデバッグをイネーブルにするには、EXEC モードで **debug ficon** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ficon {all | bypass {acl | esa | file | pm | postcheck | precheck} |
control-device {all | bypass ficon_mgr | demux [vsan vsan-id] | deque | error | event [vsan vsan-id]
| ficon_mgr | ha [vsan vsan-id] | demux [vsan vsan-id] | sb3 {error | flow} trace [detail] [vsan
vsan-id] | warning [vsan vsan-id]} |
error | event | file-trace | ha | max-port-number ports | pss-trace |
stat {all | demux [vsan vsan-id] | deque | error | event [vsan vsan-id] | ha [vsan vsan-id] | trace
[detail] [vsan vsan-id] | warning [vsan vsan-id]} | timer | trace}

no debug ficon {all | bypass {acl | esa | file | pm | postcheck | precheck} |
control-device {all | bypass ficon_mgr | demux [vsan vsan-id] | deque | error | event [vsan vsan-id]
| ficon_mgr | ha [vsan vsan-id] | demux [vsan vsan-id] | sb3 {error | flow} trace [detail] [vsan
vsan-id] | warning [vsan vsan-id]} |
error | event | file-trace | ha | max-port-number port | pss-trace |
stat {all | demux [vsan vsan-id] | deque | error | event [vsan vsan-id] | ha [vsan vsan-id] | trace
[detail] [vsan vsan-id] | warning [vsan vsan-id]} | timer | trace}
```

シンタクスの説明

all	すべての FICON 機能に対するデバッグをイネーブルにします。
bypass	FICON のエラー状態に対するバイパス フラグをイネーブルにします。
acl	Access Control List (ACL; アクセス コントロール リスト) マネージャの実行を回避します。
esa	Encryption Services Adapter (ESA) の実行を回避します。
file	ファイル操作の実行を回避します。
pm	ポート マネージャの実行を回避します。
postcheck	VSAN (仮想 SAN) をイネーブルにするための事後確認の実行を回避します。
precheck	VSAN をイネーブルにするための事前確認の実行を回避します。
control-device	FICON 制御デバイスのダンプをイネーブルにします。
all	FICON 制御デバイスのすべてのデバッグ フラグを指定します。
bypass ficon-mgr	FICON マネージャを回避します。
demux	FICON 制御デバイス メッセージ デマルチプレクサのデバッグを設定します。
deque	FICON 制御デバイス メッセージ デキューのデバッグを設定します。
error	FICON 制御デバイス エラーのデバッグを設定します。
event	FICON 制御デバイス Finite State Machine (FSM) およびイベントのデバッグを設定します。
ficon_mgr	FICON マネージャ制御デバイスのデバッグを設定します。
ha	FICON 制御デバイス HA (ハイ アベイラビリティ) のデバッグを設定します。
sb3	SB3 ライブラリのデバッグを設定します。
trace	FICON 制御デバイス トレースのデバッグを設定します。
warning	FICON 制御デバイス警告のデバッグを設定します。
error	FICON エラーに対するデバッグをイネーブルにします。
event	FICON イベントに対するデバッグをイネーブルにします。

file-trace	FICON ファイルフローのデバッグをイネーブルにします。
ha	HA 同期メッセージのデバッグをイネーブルにします。
max-port-number <i>ports</i>	最大ポート数を指定します。
pss-trace	FICON PSS フローのデバッグをイネーブルにします。
stat	FICON 統計情報のデバッグをイネーブルにします。
all	FICON 統計情報のすべてのデバッグフラグを指定します。
demux	FICON 統計情報メッセージデマルチプレクサを指定します。
deque	FICON 統計情報メッセージデキューを指定します。
error	FICON 統計情報エラーを指定します。
event	FICON 統計情報 FSM およびイベントを指定します。
ha	FICON 統計情報 HA を指定します。
trace	FICON 統計情報トレースを指定します。
warning	FICON 統計情報警告を指定します。
timer	FICON タイマーメッセージのデバッグをイネーブルにします。
trace	FICON フローのデバッグをイネーブルにします。

デフォルト デイセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.3(2)	このコマンドが導入されました。

使用上のガイドライン このコマンドを使用するには、スイッチ上で FICON がイネーブルになっている必要があります。

例 次に、**debug ficon all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ficon all
2005 Mar 10 02:38:58 ficon: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
2005 Mar 10 02:38:58 ficon: fu_fsm_execute_all: null fsm_event_list
2005 Mar 10 02:38:58 ficon: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 6943776) dropped
switch# undebug all
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをデイセーブルにします。
	show ficon	設定された FICON 情報を表示します。

debug flogi

Fabric Login (FLOGI) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug flogi** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug flogi {action [interface type number | vsan vsan-id] |
all | bypass {acl | dm | dpvm | fcsp | lcp | npiv | ns | pl | pm | pmvc | rib| vsan_mgr | zs} |
demux [interface type number | vsan vsan-id] |error |event [interface type number | vsan vsan-id] |
ha [interface type number | vsan vsan-id] |init [interface type number | vsan vsan-id] |
timers [interface type number | vsan vsan-id] |trace [interface type number | vsan vsan-id] |
warning}
```

シンタックスの説明

action	すべての FLOGI デバッグ機能をイネーブルにします。
all	すべての FLOGI デバッグ オプションをイネーブルにします。
bypass	FLOGI の実行において一部のコンポーネントを回避します。
acl	Access Control List (ACL; アクセス制御リスト) の実行を回避します。
dm	ドメイン マネージャの実行を回避します。
dpvm	Dynamic Port VSAN Membership (DPVM) の実行を回避します。
fcsp	Fibre Channel Security Protocol (FC-SP) の実行を回避します。
lcp	Link Control Protocol (LCP) の実行を回避します。
npiv	NPIV の実行を回避します。
ns	ネーム サーバの実行を回避します。
pl	ポート ロックの実行を回避します。
pm	ポート マネージャの実行を回避します。
pmvc	PM VSAN (仮想 SAN) 変更の実行を回避します。
rib	Routing Information Base (RIB) の実行を回避します。
vsan_mgr	VSAN マネージャの実行を回避します。
zs	ゾーン サーバの実行を回避します。
demux	FLOGI のデマルチプレクサをイネーブルにします。
error	FLOGI のエラー状態に対するデバッグをイネーブルにします。
event	FLOGI Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	FLOGI のハイ アベイラビリティに対するデバッグをイネーブルにします。
init	FLOGI 追加、削除、および初期化に対するデバッグをイネーブルにします。
timer	FLOGI のメッセージ タイマーに対するデバッグをイネーブルにします。
trace	FLOGI トレースに対するデバッグをイネーブルにします。
warning	FLOGI の警告に対するデバッグをイネーブルにします。
interface type number	指定のインターフェイスにデバッグを制限します。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug flogi all** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug flogi all
Apr  9 22:44:08 flogi: fs_demux: msg consumed by sdwrap_process msg
Apr  9 22:44:08 flogi: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr  9 22:44:08 flogi: fu_fsm_execute_all: null_fsm_event_list
Apr  9 22:44:08 flogi: fu_fsm_engine: mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 67690)
dropped
```

次に、**debug flogi event** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug flogi event
Apr 10 00:07:16 flogi: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr 10 00:07:16 flogi: fu_fsm_execute_all: null_fsm_event_list
Apr 10 00:07:16 flogi: fu_fsm_engine: mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 71314)
dropped
```

次に、**debug flogi trace** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug flogi trace
Apr 10 00:42:36 flogi: fs_genport_vsan_hash_fn: key: 0x1 index: 0x1
Apr 10 00:42:36 flogi: fs_mts_hdlr_fs_flogo: FLOGI_HOLD(0x8122144) refcnt:3
Apr 10 00:42:36 flogi: fs_clear_all_outstanding_responses_for_flogi: FLOGI_FREE(
a07e00300500252b) refcnt:3
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show flogi database	すべての VSAN におけるすべてのインターフェイスを介するすべての FLOGI セッションを表示します。

debug fm

機能マネージャのデバッグをイネーブルにするには、EXEC モードで **debug fm** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fm {error | flow}
```

```
no debug fm {error | flow}
```

シンタックスの説明	error	機能マネージャのエラー状態に対するデバッグをイネーブルにします。
	flow	機能マネージャ フローに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug fm flow** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug fm flow
switch# 2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: ----- EVENT START
2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: received MTS message:
2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: Src: 0x00000601/27 Dst:
0x00000601/121 ID: 0x006A0FC4 Size: 160 [REQ] Opc: 8922
(MTS_OPC_FM_CMI_GET_FEATURE_OP) RR: 0x006A0FC4 HA_SEQNO: 0x00000000 TS:
0x2524B48D52B53 REJ:0 SYNC:0
2005 Mar 10 02:40:19 feature-mgr: fm_handle_cmi_get_feature_op: Get feature (1) op
request
2005 Mar 10 02:40:19 feature-mgr: fm_handle_cmi_get_feature_op: Reply to get feature
ivr op request: op 2, op_state 0, result 0x0 (success)
2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: ----- EVENT START
2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: received MTS message:
2005 Mar 10 02:40:19 feature-mgr: fm_event_loop: Src: 0x00000601/27 Dst:
0x00000601/121 ID: 0x006A0FC6 Size: 160 [REQ] Opc: 8922
(MTS_OPC_FM_CMI_GET_FEATURE_OP) RR: 0x006A0FC6 HA_SEQNO: 0x00000000 TS:
0x2524B48EBF55D REJ:0 SYNC:0
2005 Mar 10 02:40:19 feature-mgr: fm_handle_cmi_get_feature_op: Get feature (1) op
request
2005 Mar 10 02:40:19 feature-mgr: fm_handle_cmi_get_feature_op: Reply to get feature
ivr op request: op 2, op_state 0, result 0x0 (success)
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug fspf

Fabric Shortest Path First (FSPF) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug fspf** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug fspf {all [interface type number] [vsan vsan-id] | database [interface type number] [vsan vsan-id] |
error [event [interface type number] [vsan vsan-id] |
fc {pkt | pkthdr} {both | tx | rx} [interface type number] [vsan vsan-id] |
flood [interface type number] [vsan vsan-id] | ha [interface type number] [vsan vsan-id] |
mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | rx [numpkt
range] | tx}} | retrans [interface type number] [vsan vsan-id] | route | timer}
```

```
no debug fspf {all [interface type number] [vsan vsan-id] |
database [interface type number] [vsan vsan-id] | error [event [interface type number] [vsan vsan-id] |
fc {pkt | pkthdr} {both | tx | rx} [interface type number] [vsan vsan-id] |
flood [interface type number] [vsan vsan-id] | ha [interface type number] [vsan vsan-id] |
mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | rx [numpkt
range] | tx}} | retrans [interface type number] [vsan vsan-id] | route | timer}
```

シンタックスの説明

all	すべての FSPF 機能に対するデバッグをイネーブルにします。
database	FSPF データベースに対するデバッグをイネーブルにします。
error	FSPF のエラー状態に対するデバッグをイネーブルにします。
events	FSPF イベントに対するデバッグをイネーブルにします。
fc	ファイバ チャネル パケットおよびヘッダーに対するデバッグをイネーブルにします。
fc-tunnel	ファイバ チャネル トンネル インターフェイスに対するデバッグをイネーブルにします。
fcip	ファイバ チャネル IP パケットおよびヘッダーに対するデバッグをイネーブルにします。
fv	ファイバ チャネル仮想化インターフェイスに対するデバッグをイネーブルにします。
gigabitethernet slot/port	ギガビット イーサネット インターフェイスのスロットおよびポートを指定します。
ipc	IPC パケットおよびヘッダーに対するデバッグをイネーブルにします。
mgmt 0	管理インターフェイスを指定します。
port-channel	ポートチャネル パケットおよびヘッダーに対するデバッグをイネーブルにします。
sup-fc	帯域内インターフェイスに対するデバッグをイネーブルにします。
pkt	FCC tx/rx FCC パケットに対するデバッグをイネーブルにします。
both	送受信の両方向に対するデバッグを指定します。
tx	送信方向に対するデバッグを指定します。
rx	受信方向に対するデバッグを指定します。
node	受信方向のパケットのノードを指定します。
range	1 ~ 4096 までの整数の範囲を指定します。
opcode	受信方向のパケットの opcode を指定します。
sap	受信方向のパケットの sap を指定します。
pkthdr	FCC tx/rx FCC ヘッダーに対するデバッグをイネーブルにします。
numpkt	必要なパケット数を指定します。
flood	FSPF のフラッディング イベントのデバッグをイネーブルにします。

ha	FSPF のハイ アベイラビリティに対するデバッグをイネーブルにします。
mts	FSPF tx/rx MTS イベントに対するデバッグをイネーブルにします。
retrans	FSPF の再送信に対するデバッグをイネーブルにします。
route	FSPF のルート計算に対するデバッグをイネーブルにします。
timer	FSPF タイマーに対するデバッグをイネーブルにします。
interface type number	指定のインターフェイスにデバッグを制限します。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト デイセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン インターフェイス上で不正なパケットを受信した場合に、**debug fc pkt** コマンドを使用します。

VSAN 内のインターフェイス上で、パケットの処理に関するエラーを受信した場合、**debug fspf error** を実行して、詳細情報を得ることができます。ただし、インターフェイスの両側で FSPF パラメータの設定に問題がないことを確認してください。また、**debug fspf fc pkt** コマンドを実行して、インターフェイスを指定することもできます。

VSAN 内のローカル Label Switch Router (LSR) のフラッディングに関するエラーを受信した場合、**debug fspf flood** コマンドおよび **debug fspf error** コマンドを実行します。エラーが送信パケットで報告された場合、インターフェイスがアクティブであるかどうかを確認して、**debug fc2 error** コマンドを実行してください。

VSAN 内のインターフェイスのタイマー イベント処理に関するエラーを受信した場合、**debug fspf error** コマンドを実行します。

間違った Messaging and Transaction Service (MTS) メッセージによる処理のエラーを受信した場合、**debug fspf mts pkt** コマンドおよび **debug fspf error** コマンドを実行します。

Routing Information Base (RIB) との相互通信時にエラーを受信した場合、RIB デバッグ トレースに従い、**debug fspf route** コマンドを使用します。

VSAN のルート計算中にエラーを受信した場合、**debug fspf error** コマンドおよび **debug fspf route** コマンドを実行します。

インターフェイスが FULL 以外の状態で停止したためにエラーを受信した場合、関連するインターフェイス上で、**debug fspf event** コマンドおよび **debug fspf fc pkt** コマンドを実行します。

例 次に、**debug fspf all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch1# debug fspf all
Apr 5 11:50:01 fspf: Wrong hello interval for packet on interface 100f000 in VSAN 1
Apr 5 11:50:04 fspf: Error in processing hello packet , error code = 4
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show fspf	グローバル FSPF 情報を表示します。

debug hardware arbiter

ハードウェア アービター ドライバに対するデバッグをイネーブルにするには、EXEC モードで **debug hardware artiber** コマンドを使用します。 **debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug hardware arbiter {error | flow} [group number]
```

```
no debug hardware arbiter {error | flow} [group number]
```

シンタックスの説明	error	説明
	error	ハードウェア アービター カーネル エラーに対するデバッグをイネーブルにします。
	flow	ハードウェア アービター カーネル フローに対するデバッグをイネーブルにします。
	group number	指定のグループにデバッグを制限します。有効範囲は 0 ~ 17 です。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug hardware arbiter error group** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug hardware arbiter error group 1
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show hardware	スイッチ ハードウェア インベントリの詳細を表示します。

debug idehsd

IDE ホット スワップ ハンドラのデバッグをイネーブルにするには、EXEC モードで **debug idehsd** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug idehsd {cmd dbglevel [debug-level] | error | flow}
```

```
no debug idehsd {cmd dbglevel [debug-level] | error | flow}
```

シンタックスの説明	cmd dbglevel	IDE ホット スワップ ハンドラに対するデバッグをイネーブルにします。
	<i>debug-level</i>	0 ~ 8 までのデバッグ レベルを指定します。
	error	IDE ホット スワップ ハンドラのエラー状態に対するデバッグをイネーブルにします。
	flow	IDE ホット スワップ ハンドラのフローに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンド モード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug idehsd cmd dbglevel** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug idehsd cmd dbglevel 5
set debug level to 5 succeeded
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug ike

Internet Key Exchange (IKE) プロトコルのデバッグをイネーブルにするには、EXEC モードで **debug ike** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ike {all | error | event | message | mts | protocol | verbose | warning}
```

```
no debug ike {all | error | event | message | mts | protocol | verbose | warning}
```

シンタックスの説明

all	IKE のすべてのデバッグ フラグをイネーブルにします。
error	IKE エラーに対するデバッグをイネーブルにします。
event	IKE イベント生成に対するデバッグをイネーブルにします。
message	IKE メッセージに対するデバッグをイネーブルにします。
mts	Messaging and Transaction Service (MTS) 関連の IKE アクティビティに対するデバッグをイネーブルにします。
protocol	IKE プロトコル関連処理に対するデバッグをイネーブルにします。
verbose	IKE プロトコル関連処理に対する詳細デバッグをイネーブルにします。
warning	IKE 警告に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

このコマンドを使用するには、**crypto ike enable** コマンドを使用して IKE をイネーブルにする必要があります。

例

次に、**debug ike all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug ike all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show crypto ike domain ipsec	IKE プロトコル情報を表示します。

debug ilc_helper

Intelligent Line Card (ILC) ヘルパーのデバッグをイネーブルにするには、EXEC モードで **debug ilc_helper** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ilc_helper {all | errors | events | info}
```

```
no debug ilc_helper {all | errors | events | info}
```

シンタックスの説明		
all		すべての ILC ヘルパー機能に対するデバッグをイネーブルにします。
errors		ILC ヘルパーのエラー状態に対するデバッグをイネーブルにします。
events		ILC ヘルパー イベントに対するデバッグをイネーブルにします。
info		ILC ヘルパー情報に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug ilc_helper all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ilc_helper all
For Application :125, sdwrap:mts_send : Broken pipe
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug ipacl

IP Access Control List (ACL; アクセス コントロール リスト) のデバッグをイネーブルにするには、EXEC モードで **debug ipacl** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ipacl {all | error | event | trace}
```

```
no debug ipacl {all | error | event | trace}
```

シンタックスの説明

all	すべての IP ACL 機能に対するデバッグをイネーブルにします。
error	IP ACL のエラー状態に対するデバッグをイネーブルにします。
event	IP ACL イベントに対するデバッグをイネーブルにします。
trace	IP ACL トレースに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug ipacl all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug ipacl all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show ip access-list	現在アクティブになっている IP ACL を表示します。

debug ipconf

IP 設定のデバッグをイネーブルにするには、EXEC モードで **debug ipconf** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ipconf {all | errors | events | info | pss}
```

```
no debug ipconf {all | errors | events | info | pss}
```

シンタックスの説明

all	すべての IP 設定機能に対するデバッグをイネーブルにします。
errors	IP 設定のエラー状態に対するデバッグをイネーブルにします。
events	IP 設定 tx/rx MTS イベントに対するデバッグをイネーブルにします。
info	IP 設定情報に対するデバッグをイネーブルにします。
pss	IP 設定 PSS 操作に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug ipconf all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ipconf all
switch# 2005 Mar 10 02:45:30 ipconf: Received MTS message
2005 Mar 10 02:45:30 ipconf: MTS message received opcode 862 source 0x00000601/27
2005 Mar 10 02:45:30 ipconf: Getting ip addresses on interface 5000000
2005 Mar 10 02:45:30 ipconf: Received MTS message
2005 Mar 10 02:45:30 ipconf: MTS message received opcode 862 source 0x00000601/27
2005 Mar 10 02:45:30 ipconf: Getting ip addresses on interface 5000000
2005 Mar 10 02:45:30 ipconf: Received MTS message
2005 Mar 10 02:45:30 ipconf: MTS message received opcode 862 source 0x00000601/27
2005 Mar 10 02:45:30 ipconf: Getting ip addresses on interface 5000000
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug ipfc

IP over Fibre Channel (IPFC) のデバッグをイネーブルにするには、EXEC モードで **debug ipfc** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ipfc {all | errors | events | info | kernel {errors | events}}
```

シンタックスの説明	all	すべての IPFC 機能に対するデバッグをイネーブルにします。
	errors	IPFC のエラー状態に対するデバッグをイネーブルにします。
	events	IPFC tx/rx MTS イベントに対するデバッグをイネーブルにします。
	info	IPFC 情報に対するデバッグをイネーブルにします。
	kernel	IPFC カーネル操作に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug ipfc kernel errors** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ipfc kernel errors
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug ips

IP Storage Services (IPS) モジュールに対するデバッグをイネーブルにするには、EXEC モードで **debug ips** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ips {acl {flow | flow-detail} | all | demux | error | flow {ethernet | fcip} | fsm | ha | init | iscsi
           {config | config-detail | flow | flow-detail | msgs} | islb {cfs {config | config-detail | error | flow |
           flow-detail} | config | config-detail | flow | flow-detail | vrrp {error | flow | flow-detail}} | isns
           {config | config-detail | error | flow | flow-detail | msgs | packet} | show_all | upgrade}
```

```
no debug ips {acl {flow | flow-detail} | all | demux | error | flow {ethernet | fcip} | fsm | ha | init | iscsi
             {config | config-detail | flow | flow-detail | msgs} | islb {cfs {config | config-detail | error | flow |
             flow-detail} | config | config-detail | flow | flow-detail | vrrp {error | flow | flow-detail}} | isns
             {config | config-detail | error | flow | flow-detail | msgs | packet} | show_all | upgrade}
```

シンタックスの説明

acl	Access Control List (ACL; アクセス コントロール リスト) に対するデバッグをイネーブルにします。
flow	IPS フローに対するデバッグをイネーブルにします。
flow-detail	IPS フローに対する詳細デバッグをイネーブルにします。
all	すべての IPS デバッグ オプションをイネーブルにします。
demux	IPS デマルチプレクサに対するデバッグをイネーブルにします。
error	IPS のエラー状態に対するデバッグをイネーブルにします。
ethernet	イーサネットフローにデバッグを制限します。
fcip	Fibre Channel over IP (FCIP) フローにデバッグを制限します。
fsm	IPS FSM およびイベントに対するデバッグをイネーブルにします。
ha	IPS のハイ アベイラビリティに対するデバッグをイネーブルにします。
init	IPS 追加、削除、および初期化に対するデバッグをイネーブルにします。
iscsi	iSCSI のデバッグをイネーブルにします。
config	iSCSI 設定のデバッグをイネーブルにします。
config-detail	iSCSI 設定の詳細デバッグをイネーブルにします。
msgs	送受信された iSCSI メッセージのデバッグをイネーブルにします。
islb	iSLB のデバッグをイネーブルにします。
cfs	iSLB CFS のデバッグをイネーブルにします。
config	iSLB CFS 設定のデバッグをイネーブルにします。
config-detail	iSLB CFS 設定に対する詳細デバッグをイネーブルにします。
error	iSLB CFS のエラー状態に対するデバッグをイネーブルにします。
flow	iSLB CFS フローに対するデバッグをイネーブルにします。
flow-detail	iSLB CFS フローに対する詳細デバッグをイネーブルにします。
config	iSLB 設定のデバッグをイネーブルにします。
config-detail	iSLB 設定に対する詳細デバッグをイネーブルにします。
flow	iSLB フローに対するデバッグをイネーブルにします。
flow-detail	iSLB フローに対する詳細デバッグをイネーブルにします。
vrrp	iSLB VRRP のデバッグをイネーブルにします。
config	iSNS 設定のデバッグをイネーブルにします。
config-detail	iSNS 設定に対する詳細デバッグをイネーブルにします。
error	iSNS のエラー状態に対するデバッグをイネーブルにします。
flow	iSNS フローに対するデバッグをイネーブルにします。
flow-detail	iSNS フローに対する詳細デバッグをイネーブルにします。

msgs	送受信された iSNS メッセージのデバッグをイネーブルにします。
packet	iSNS パケットのデバッグをイネーブルにします。
show_all	すべてのデバッグ IPS マネージャ フラグをイネーブルにします。
upgrade	アップグレードに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.1(1)	このコマンドが導入されました。
	3.0(1)	iSLB オプションおよび iSNS オプションが追加されました。

使用上のガイドライン なし

例 次に、**debug ips show_all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ips show_all
IPS Manager:
iSCSI Trace Detail debugging is on
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show ips stats	IP ストレージの統計を表示します。
	show ips status	IP ストレージのステータスを表示します。

debug ipsec

Internet Protocol Security (IPSec) のデバッグをイネーブルにするには、EXEC モードで **debug ipsec** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ipsec {all | bypass ficon_mgr | config | config-detail | demux | deque | error | event | flow |
            flow-detail | ha | trace [detail] | warning}
```

```
no debug ipsec {all | bypass ficon_mgr | config | config-detail | demux | deque | error | event | flow |
              flow-detail | ha | trace [detail] | warning}
```

シンタックスの説明

all	すべての IPSec デバッギングをイネーブルにします。
bypass ficon_mgr	Fibre Connection (FICON) マネージャを回避します。
config	IPSec 設定に対するデバッグをイネーブルにします。
config-detail	詳細 IPSec 設定に対するデバッグをイネーブルにします。
demux	IPSec メッセージのデマルチプレクサに対するデバッグをイネーブルにします。
deque	IPSec メッセージのデキューに対するデバッグをイネーブルにします。
error	IPSec エラーに対するデバッグをイネーブルにします。
event	IPSec FSM およびイベントに対するデバッグをイネーブルにします。
flow	IPSec フローに対するデバッグをイネーブルにします。
flow-detail	詳細 IPSec フローに対するデバッグをイネーブルにします。
ha	IPSec のハイ アベイラビリティに対するデバッグをイネーブルにします。
trace	IPSec トレースに対するデバッグをイネーブルにします。
detail	詳細トレースを指定します。
warning	IPSec 警告に対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

このコマンドを使用するには、**crypto ipsec enable** コマンドを使用して IPSec をイネーブルにする必要があります。

例

次に、**debug ipsec config** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ipsec config
```

関連コマンド

コマンド	説明
crypto ipsec enable	IPSec をイネーブルにします。
no debug all	すべてのデバッグをディセーブルにします。

debug isns

Internet Storage Name Services (iSNS; インターネットストレージネームサービス) に対するデバッグをイネーブルにするには、EXEC モードで **debug isns** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug isns {all | bypass ficon_mgr | cloud | db | deque | error | event [vsan vsan-id] |
fabric distribute | ha [vsan vsan-id] | prot | trace [detail] | warning [vsan vsan-id]}
no debug isns {all | bypass ficon_mgr | cloud | db | deque | error | event [vsan vsan-id] |
fabric distribute | ha [vsan vsan-id] | prot | trace [detail] | warning [vsan vsan-id]}
```

シンタックスの説明

all	すべての iSNS デバッグをイネーブルにします。
bypass ficon-mgr	Fibre Connection (FICON) マネージャ実行の回避をイネーブルにします。
cloud	iSNS クラウド検出に対するデバッグをイネーブルにします。
db	iSNS データベースに対するデバッグをイネーブルにします。
deque	iSNS メッセージのデキューに対するデバッグをイネーブルにします。
error	iSNS エラーに対するデバッグをイネーブルにします。
event	iSNS イベントに対するデバッグをイネーブルにします。
vsan vsan-id	指定 VSAN (仮想 SAN) ID へのデバッグを制限します。有効範囲は 1 ~ 4093 です。
fabric distribute	iSNS ファブリック配信に対するデバッグをイネーブルにします。
ha	iSNS のハイ アベイラビリティに対するデバッグをイネーブルにします。
prot	iSNS プロトコルに対するデバッグをイネーブルにします。
trace	iSNS トレースに対するデバッグをイネーブルにします。
detail	詳細 iSNS トレースをイネーブルにします。
warning	iSNS 警告に対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

このコマンドを使用するには、**isns-server enable** コマンドを使用して iSNS をイネーブルにする必要があります。

例

次に、**debug isns error** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug isns error
```

関連コマンド

コマンド	説明
isns-server enable	iSNS サーバをイネーブルにします。
no debug all	すべてのデバッグをディセーブルにします。
show isns	iSNS 情報を表示します。

debug ivr

Inter-VSAN Routing (IVR) のデバッグをイネーブルにするには、EXEC モードで **debug ivr** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug {all | demux | dep | dep-detail | dequeue | drav-fsm | drav-fsm-detail | errors | fcid-rewrite |
fcid-rewrite-detail | ficon | ficon-detail | ha | pnat | pv | pv-detail | state-machine [vsan vsan-id]
| test | trace | trace-detail | tu-fsm | tu-fsm-detail | zone-distrib-errors | zone-distrib-events |
zone-fsm | zone-fsm-detail}
```

```
no debug {all | demux | dep | dep-detail | dequeue | drav-fsm | drav-fsm-detail | errors | fcid-rewrite
| fcid-rewrite-detail | ficon | ficon-detail | ha | pnat | pv | pv-detail | state-machine [vsan vsan-id]
| test | trace | trace-detail | tu-fsm | tu-fsm-detail | zone-distrib-errors | zone-distrib-events |
zone-fsm | zone-fsm-detail}
```

シンタックスの説明

all	IVR デバッグのすべてのフィルタをイネーブルにします。
demux	IVR イベント逆多重化に対するデバッグをイネーブルにします。
dep	IVR DEP に対するデバッグをイネーブルにします。
dep-detail	IVR DEP の詳細なデバッグをイネーブルにします。
dequeue	IVR イベント デキューに対するデバッグをイネーブルにします。
drav-fsm	IVR DRAV Finite State Machine (FSM) に対するデバッグをイネーブルにします。
drav-fsm-detail	IVR DRAV FSM の詳細なデバッグをイネーブルにします。
errors	IVR エラーに対するデバッグをイネーブルにします。
fcid-rewrite	IVR Fibre Channel ID (FCID) リライトに対するデバッグをイネーブルにします。
fcid-rewrite-detail	IVR FCID リライトの詳細なデバッグをイネーブルにします。
ficon	IVR FICON に対するデバッグをイネーブルにします。
ficon-detail	IVR FICON の詳細に対するデバッグをイネーブルにします。
ha	IVR のハイ アベイラビリティに対するデバッグをイネーブルにします。
pfcid	IVR 固定 FCID モジュールに対するデバッグをイネーブルにします。
pfcid-detail	IVR 固定 FCID モジュールに対する詳細デバッグをイネーブルにします。
pnat	IVR ペイロード Network Address Translation (NAT) のデバッグをイネーブルにします。
pv	IVR PV ステート マシンに対するデバッグをイネーブルにします。
pv-detail	IVR PV ステート マシンの詳細なデバッグをイネーブルにします。
state-machine	FSM に対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
test	IVR テスト機能に対するデバッグをイネーブルにします。
trace	IVR トレースに対するデバッグをイネーブルにします。
trace-detail	IVR の詳細トレースに対するデバッグをイネーブルにします。
tu-fsm	IVR TU FSM に対するデバッグをイネーブルにします。
tu-fsm-detail	IVR TU FSM の詳細なデバッグをイネーブルにします。
zone-distrib-errors	IVR ゾーン配信エラーに対するデバッグをイネーブルにします。
zone-distrib-events	IVR ゾーン配信イベントに対するデバッグをイネーブルにします。
zone-fsm	IVR ゾーン FSM に対するデバッグをイネーブルにします。
zone-fsm-detail	IVR ゾーン FSM の詳細なデバッグをイネーブルにします。

デフォルト デイセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	2.1(1)	このコマンドが導入されました。
	3.0(1)	<ul style="list-style-type: none"> • ficon オプションおよび ficon-detail オプションが追加されました。 • pfcid オプションおよび pfcid-detail オプションが追加されました。

使用上のガイドライン なし

例 次に、**debug ivr all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ivr all
2005 Mar 10 01:27:27 ivr: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
2005 Mar 10 01:27:27 ivr: fu_fsm_execute_all: null fsm_event_list
2005 Mar 10 01:27:27 ivr: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 6774251) dropped
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをデイセーブルにします。
	show ivr	IVR 設定を表示します。

debug klm

カーネルロード可能モジュールパラメータのデバッグをイネーブルにするには、EXECモードで **debug klm** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug klm {fc2 {cpuhog seconds | flag flags} | scsi-target {driver | error [vsan vsan-id] [fcid fc-id] |
flag flags | flow [vsan vsan-id] [fcid fc-id] | snmp | syscall} | sdip {all | error | flow | warning}}
```

```
no debug klm {fc2 {cpuhog seconds | flag flags} | scsi-target {driver | error [vsan vsan-id] [fcid fc-id]
| flag flags | flow [vsan vsan-id] [fcid fc-id] | snmp | syscall} | sdip {all | error | flow | warning}}
```

シンタックスの説明

fc2	FC2 ドライバデバッグパラメータに対するデバッグをイネーブルにします。
cpuhog seconds	FC2 CPU Hog の値を指定します。有効値は 0 ~ 10000 秒です。
flag flags	フラグ値を指定します。有効範囲は、0x0 ~ 0xffffffff です。
scsi-target	SCSI ターゲットドライバに対するデバッグをイネーブルにします。
driver	SCSI ターゲットドライバフラグに対するデバッグをイネーブルにします。
error	ドライバのエラー状態に対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
fcid fc-id	指定の Fibre Channel ID (FCID) インターフェイスにデバッグを制限します。
flow	SCSI ターゲットフローに対するデバッグをイネーブルにします。
snmp	SCSI ターゲットの SNMP (簡易ネットワーク管理プロトコル) 要求に対するデバッグをイネーブルにします。
syscall	SCSI ターゲットシステム呼び出し要求に対するデバッグをイネーブルにします。
sdip	SDIP ドライバに対するデバッグをイネーブルにします。
all	SCSI ターゲットドライバに対するデバッグをイネーブルにします。
flow	ドライバフローに対するデバッグをイネーブルにします。
warning	ドライバ警告に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug klm scsi-target driver** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug klm scsi-target driver
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug license

ライセンス デバッグをイネーブルにするには、EXEC モードで **debug license** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug license {all | errors | event s | mts}
```

```
no debug license {all | errors | events | mts}
```

シンタックスの説明	
all	すべてのライセンス機能に対するデバッグをイネーブルにします。
errors	ライセンスのエラー状態に対するデバッグをイネーブルにします。
events	ライセンス イベントに対するデバッグをイネーブルにします。
mts	MTS の Tx/Rx パケットに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug license all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug license all
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show license	ライセンス情報を表示します。

debug logfile

指定のファイルにデバッグ コマンドの結果を出力するには、EXEC モードで **debug logfile** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug logfile filename [size bytes]
```

シンタックスの説明

<i>filename</i>	ログ ファイル名を割り当てます。最大 80 文字まで可能です。
<i>size bytes</i>	ログ ファイルのサイズをバイトで指定します。有効範囲は 4096 ~ 4194304 です。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

このコマンドを使用して、特定のログ ファイルにデバッグ メッセージを記録します。ファイルに出力する方法は、デバッグの結果をコンソールに出力するより安全で簡単です。

debug logfile コマンドを使用してログ ファイルを作成すると、自動的にそのファイルがログ（特に指定しなければスーパーバイザ モジュールのディレクトリ）に作成されます。

たとえば、**debug logfile** コマンドを使用して **captureDebug** という名前のログ ファイルを作成すると、**dir log://sup-local/?** コマンドを入力してその作成したログ ファイルを見つける必要があります。次に、作成したログ ファイルを見つける方法を示します。

```
switch# debug logfile captureDebug
switch# dir log://sup-local/?
log:                               Enter URL "log:[/]/<module-number>/<filename>"
log://sup-local/dmesg
log://sup-local/messages
→ log://sup-local/captureDebug

switch# dir log://sup-local/
```

例

次に、*sample* という名前のファイルにデバッグ コマンドの出力結果をリダイレクトする例を示します。

```
switch# debug logfile sample
```

次に、*sample* という名前のファイルにログ ファイルのサイズを割り当てる例を示します。

```
switch# debug logfile sample size 410000
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show logging	現在のメッセージ ロギングの設定を表示します。

debug mcast

マルチキャスト定義に対するデバッグをイネーブルにするには、EXEC モードで **debug mcast** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug mcast {all | error [vsan vsan-id] [interface fc slot/port] | event [vsan vsan-id] [interface fc slot/port] | mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | rx [numpkt range] | tx}} | trace [vsan vsan-id] [interface fc slot/port]}
```

```
no debug mcast {all | error [vsan vsan-id] [interface fc slot/port] | event [vsan vsan-id] [interface fc slot/port] | mts {pkt {both | rx [node range | opcode range | sap range] | tx} | pkthdr {both | rx [numpkt range] | tx}} | trace [vsan vsan-id] [interface fc slot/port]}
```

シンタックスの説明

all	すべてのマルチキャスト定義に対するデバッグをイネーブルにします。
error	マルチキャストエラーに対するデバッグをイネーブルにします。
event	マルチキャストイベントに対するデバッグをイネーブルにします。
mts	マルチキャスト tx/rx MTS イベントに対するデバッグをイネーブルにします。
trace	マルチキャスト トレースに対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
interface fc slot/port	指定のインターフェイスにデバッグを制限します。
pkt	パケットに対するデバッグを指定します。
pkthdr	ヘッダーに対するデバッグを指定します。
numpkt	必要なパケット数を指定します。
both	送受信の両方向に対するデバッグを指定します。
tx	送信方向に対するデバッグを指定します。
rx	受信方向に対するデバッグを指定します。
node	受信方向のパケットのノードを指定します。
opcode	受信方向のパケットの opcode を指定します。
sap	受信方向のパケットの sap を指定します。
range	1 ~ 4096 までの整数の範囲を指定します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug mcast all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug mcast all
```

■ debug mip

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show mcast	マルチキャスト情報を表示します。

debug mip

Multiple IP (MIP) カーネル ドライバに対するデバッグをイネーブルにするには、EXEC モードで **debug mip** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug mip {errors | events}
```

```
no debug mip {errors | events}
```

シンタックスの説明	errors	events
	MIP のエラー状態に対するデバッグをイネーブルにします。	MIP イベントに対するデバッグをイネーブルにします。

デフォルト	ディセーブル
-------	--------

コマンドモード	EXEC モード
---------	----------

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン	なし
------------	----

例 次に、**debug mip errors** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug mip errors
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug module

スイッチングまたはサービス モジュールに対するデバッグをイネーブルにするには、EXEC モードで **debug module** コマンドを使用します。 **debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、 **no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug module {all | error [module slot] | event | ha | no-powerdown | trace [module slot]}
```

```
no debug module {all | error [module slot] | event | ha | no-powerdown | trace [module slot]}
```

シンタックスの説明

all	すべてのモジュール機能に対するデバッグをイネーブルにします。
error	モジュールのエラー状態に対するデバッグをイネーブルにします。
event	モジュール イベントに対するデバッグをイネーブルにします。
ha	モジュールのハイ アベイラビリティに対するデバッグをイネーブルにします。
no-powerdown	モジュールのパワー サイクル機能をディセーブルにします。
trace	モジュールのトレース フローに対するデバッグをイネーブルにします。
module slot	指定のモジュールにデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、 **debug module all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug module all
2005 Mar 10 02:51:01 module: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
2005 Mar 10 02:51:01 module: fu_fsm_execute_all: null_fsm_event_list
2005 Mar 10 02:51:01 module: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 6986564) dropped
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show module	モジュールのステータスを表示します。

debug ntp

Network Time Protocol (NTP) モジュールに対するデバッグをイネーブルにするには、EXEC モードで **debug ntp** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ntp {errors | info}
```

```
no debug ntp {errors | info}
```

シンタックスの説明	errors	NTP のエラー状態に対するデバッグをイネーブルにします。
	info	NTP 情報およびイベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug ntp info** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ntp info
2005 Mar 10 03:00:42 ntp: Dropping msg_ref with rr_token [7002722]
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show ntp	設定された NTP サーバおよびピア アソシエーションを表示します。

debug npv

スイッチで N Port Virtualization (NPV; N ポート仮想化) コンフィギュレーションのデバッグをイネーブルにするには、**debug npv** コマンドを使用します。

debug npv

シンタックスの説明	このコマンドには、引数またはキーワードはありません。
-----------	----------------------------

デフォルト	なし
-------	----

コマンドモード	EXEC モード
---------	----------

コマンド履歴	リリース	変更内容
	3.2(1)	このコマンドが導入されました。

使用上のガイドライン	なし
------------	----

例	次に、スイッチに設定されているすべての NPV デバッグ コマンドを表示する例を示します。
---	---

```
switch# show debug npv
N_port Virtualizer:
  FC Receive Packets debugging is on
  FC Transmit Packets debugging is on
  FC Receive Packet header debugging is on
  FC Transmit Packet header debugging is on
  MTS Receive Packets debugging is on
  MTS Transmit Packets debugging is on
  MTS Receive Packet header/payload debugging is on
  MTS Transmit Packet header/payload debugging is on
  High Availability debugging is on
  FSM Transitions debugging is on
  Error debugging is on
  Warning debugging is on
  Trace debugging is on
  Trace Detail debugging is on
  Demux debugging is on
  Dequeue debugging is on
  Packets debugging is on
  Database debugging is on
  Timers debugging is on
  External Interface FSM Events debugging is on
  External Interface FSM Errors debugging is on
  External Interface FSM Trace debugging is on
  FLOGI FSM Events debugging is on
  FLOGI FSM Errors debugging is on
  FLOGI FSM Trace debugging is on
  Server Interface FSM Events debugging is on
  Server Interface FSM Errors debugging is on
  Server Interface FSM Trace debugging is on
  Events debugging is on
```

関連コマンド	コマンド	説明
	show debug npv	スイッチに設定されている NPV デバッグ コマンドを表示します。

debug obfl

Onboard Failure Logging (OBFL; オンボード障害ロギング) のデバッグをイネーブルにするには、EXEC モードで **debug obfl** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug obfl {error | trace}
```

```
no debug obfl {error | trace}
```

シンタックスの説明	error	OBFL のエラー状態に対するデバッグをイネーブルにします。
	info	OBFL イベントに対するデバッグをイネーブルにします。

デフォルト	ディセーブル
-------	--------

コマンドモード	EXEC モード
---------	----------

コマンド履歴	リリース	変更内容
	3.0(1)	このコマンドが導入されました。

使用上のガイドライン	なし
------------	----

例 次に、**debug obfl error** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug obfl error
2006 Jan 23 21:30:59.573503 obfl: obfl_process_mts_msgs(): OBFL received mts mes
sage: opc:182
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show logging onboard	OBFL 情報を表示します。

debug platform

プラットフォーム マネージャに対するデバッグをイネーブルにするには、EXEC モードで **debug platform** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug platform {all [fc_id fc-id] | error [module slot] | flow [module slot] | fsm | ha | hitless |
mts {pkt | pkthdr} {tx | rx} | nopowerdown | supervisor-reset}
```

```
no debug platform {all [fc_id fc-id] | error [module slot] | flow [module slot] | fsm | ha | hitless |
mts {pkt | pkthdr} {tx | rx} | nopowerdown | supervisor-reset}
```

シンタックスの説明

all	すべてのプラットフォーム機能に対するデバッグをイネーブルにします。
error	プラットフォーム関連のエラー状態に対するデバッグをイネーブルにします。
flow	プラットフォーム関連のフローに対するデバッグをイネーブルにします。
fsm	プラットフォーム関連の Finite State Machine (FSM) に対するデバッグをイネーブルにします。
ha	プラットフォーム関連のハイ アベイラビリティに対するデバッグをイネーブルにします。
hitless	スイッチが hitless モードの間、プラットフォーム ローディング機能をイネーブルにします。
mts	プラットフォーム関連の tx/rx MTS イベントに対するデバッグをイネーブルにします。
nopowerdown	電源切断モジュールをイネーブルにします。
supervisor-reset	ローカル スーパーバイザをリセットします。
fc_id fc-id	指定 FCID モジュール番号にデバッグを制限します。有効範囲は 0 ~ 2147483647 です。
pkt	パケットに対するデバッグをイネーブルにします。
pkthdr	ヘッダーに対するデバッグをイネーブルにします。
tx	送信方向のデバッグをイネーブルにします。
rx	受信方向のデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、**debug platform all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug platform all
2005 Mar 10 03:01:56 platform: fu_fsm_execute_all: match_msg_id(0),
log_already_open(0)
2005 Mar 10 03:01:56 platform: fu_fsm_execute_all: null fsm_event_list
2005 Mar 10 03:01:56 platform: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 7004045) dropped
v-185# 2005 Mar 10 03:01:56 platform: env_chg_none: ps 0 old 1 new 1
2005 Mar 10 03:01:57 platform: env_chg_none: ps 0 old 1 new 1
2005 Mar 10 03:01:58 platform: env_chg_none: ps 0 old 1 new 1
v-185# debug platform all
2005 Mar 10 03:01:59 platform: fu_priority_select: - setting fd[7] for select call
2005 Mar 10 03:01:59 platform: fu_priority_select_select_queue: round credit(5)
2005 Mar 10 03:01:59 platform: curr_q - FU_PSEL_Q_CAT_CQ, usr_q_info(0),
priority(1), credit(0), empty
2005 Mar 10 03:01:59 platform: fu_priority_select: returning FU_PSEL_Q_CAT_FD queue,
fd(7), usr_q_info(1)
2005 Mar 10 03:01:59 platform: fu_fsm_engine: line[2139]
.
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug plog

永続的ロギング（PLOG）に対するデバッグをイネーブルにするには、EXEC モードで **debug plog** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug plog {error | trace}
```

```
no debug plog {error | trace}
```

シンタックスの説明	error	PLOG のエラー状態に対するデバッグをイネーブルにします。
	trace	PLOG イベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	3.0(1)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug plog** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug plog
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug port

ポートのデバッグをイネーブルにするには、EXEC モードで **debug port** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug port {all | bypass {acl_manager | domain_manager | fcsp | ficon | fport_server | lcp |
loopback_diag | port_channel_mgr | port_lock | qos_mgr | span | switch_wnn | vsan_mgr |
wnn_mgr | xbar_mgr | zone_server} | error | event [interface type number | module slot] | ha
[interface type number | module slot] | trace [interface type number | module slot]}
```

```
no debug port {all | bypass {acl_manager | domain_manager | fcsp | ficon | fport_server | lcp |
loopback_diag | port_channel_mgr | port_lock | qos_mgr | span | switch_wnn | vsan_mgr |
wnn_mgr | xbar_mgr | zone_server} | error | event [interface type number | module slot] | ha
[interface type number | module slot] | trace [interface type number | module slot]}
```

シンタックスの説明

all	すべてのポート デバッグ オプションをイネーブルにします。
bypass	ポートの実行において一部のコンポーネントを回避します。
error	ポートのエラー状態に対するデバッグをイネーブルにします。
event	ポート Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	ポートのハイ アベイラビリティに対するデバッグをイネーブルにします。
trace	ポート トレースに対するデバッグをイネーブルにします。
acl_manager	Access Control List (ACL; アクセス コントロール リスト) マネージャの実行を回避します。
domain_manager	ドメイン マネージャの実行を回避します。
fcsp	Fibre Channel Security Protocol (FC-SP) の実行を回避します。
ficon	Fibre Connection (FICON) の実行を回避します。
fport_server	FPort サーバの実行を回避します。
lcp	Link Control Protocol (LCP) の実行を回避します。
loopback_diag	ループバック 診断の実行を回避します。
port_channel_mgr	ポート チャネル マネージャの実行を回避します。
port_lock	ポート ロックの実行を回避します。
qos_mgr	Quality of Service (QoS; サービス品質) マネージャの実行を回避します。
span	Switched Port Analyzer (SPAN; スイッチド ポート アナライザ) の実行を回避します。
switch_wnn	スイッチ WWN の使用と ELP での VSAN WWN の使用を回避します。
vsan_mgr	VSAN (仮想 SAN) マネージャの実行を回避します。
wnn_mgr	World Wide Name (WWN) マネージャの実行を回避します。
xbar_mgr	Crossbar (XBAR) マネージャの実行を回避します。
zone_mgr	ゾーン マネージャの実行を回避します。
interface type number	指定のインターフェイスにデバッグを制限します。
module slot	指定のモジュールにデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug port all** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug port all
Apr 10 00:49:38 port: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr 10 00:49:38 port: fu_fsm_execute_all: null_fsm_event_list
Apr 10 00:49:38 port: fu_fsm_engine: mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 40239)
dropped
```

次に、**debug port event** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug port event
Apr 10 15:30:35 port: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr 10 15:30:35 port: fu_fsm_execute_all: null_fsm_event_list
Apr 10 15:30:35 port: fu_fsm_engine: mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 7002)
dropped
switch# Apr 10 15:30:35 port: fu_priority_select: - setting fd[3] for select call -
setting fd[5] for select call - setting fd[6] for select call
Apr 10 15:30:35 port: fu_priority_select_select_queue: round credit(16)
Apr 10 15:30:35 port: curr_q - FU_PSEL_Q_CAT_FD, usr_q_info(32), fd(5),
priority(3), credit(2), empty
Apr 10 15:30:35 port: fu_priority_select: returning FU_PSEL_Q_CAT_MTS queue, fd(3),
usr_q_info(8)
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug port-channel

ポートチャネルのデバッグをイネーブルにするには、EXEC モードで **debug port-channel** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug port-channel {all | error | event | ha | trace | warning}
```

```
no debug port-channel {all | error | event | ha | trace | warning}
```

シンタックスの説明	
all	すべてのポートチャネルのデバッグ オプションをイネーブルにします。
demux	ポートチャネル メッセージに対するデバッグをイネーブルにします。
deque	ポートチャネル メッセージ デキューに対するデバッグをイネーブルにします。
error	ポートチャネルのエラー状態に対するデバッグをイネーブルにします。
event	ポートチャネル Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	ポートチャネルのハイ アベイラビリティに対するデバッグをイネーブルにします。
trace	ポートチャネル トレースに対するデバッグをイネーブルにします。
warning	ポートチャネル警告に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug port-channel all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug port-channel all
2005 Mar 10 03:03:26 port_channel: fu_fsm_execute_all: match_msg_id(0),
log_already_open(0)
2005 Mar 10 03:03:26 port_channel: fu_fsm_execute_all: null fsm_event_list
2005 Mar 10 03:03:26 port_channel: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 7005958) dropped
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show port-channel	既存のポートチャネルの設定情報を表示します。

debug port-resources

ポートリソースモジュールのデバッグをイネーブルにするには、EXECモードで **debug port-resources** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug port-channel {all | demux | deque | error | event | ha | mts | trace | warning}
```

```
no debug port-channel {all | demux | deque | error | event | ha | mts | trace | warning}
```

シンタックスの説明

all	すべてのポートリソースデバッグオプションをイネーブルにします。
demux	ポートリソースメッセージのデバッグをイネーブルにします。
deque	ポートリソースメッセージデキューのデバッグをイネーブルにします。
error	ポートリソースのエラー状態に対するデバッグをイネーブルにします。
event	ポートリソース Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	ポートリソースのハイアベイラビリティに対するデバッグをイネーブルにします。
mts	ポートリソースメッセージ MTS イベントに対するデバッグをイネーブルにします。
trace	ポートリソーストレースに対するデバッグをイネーブルにします。
warning	ポートリソース警告に対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXECモード

コマンド履歴

リリース	変更内容
3.0(1)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug port-resources demux** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug port-resources demux vsan 2
2006 Jan 19 22:10:59.244892 port-resources: fu_priority_select: - setting fd[5]
  for select call
2006 Jan 19 22:10:59.244985 port-resources: fu_priority_select_select_queue: rou
nd credit(12)
2006 Jan 19 22:10:59.245018 port-resources:      curr_q - FU_PSEL_Q_CAT_CQ, usr_q
_info(2), priority(7), credit(6), empty
2006 Jan 19 22:10:59.245051 port-resources: fu_priority_select: returning FU_PSE
L_Q_CAT_MTS queue, fd(5), usr_q_info(1)
2006 Jan 19 22:10:59.245168 port-resources: prm_get_data_from_queue(664): dequeued mts
msg (128136), MTS_OPC_DEBUG_WRAP_MSG
2006 Jan 19 22:10:59.245205 port-resources: fu_fsm_engine: line[2205]
2006 Jan 19 22:10:59.245248 port-resources: prm_demux: ev[0]
ips-hac2# 2006 Jan 19 22:10:59.246440 port-resources: fu_fsm_execute_all: match_
msg_id(0), log_already_open(0)
2006 Jan 19 22:10:59.246507 port-resources: fu_fsm_execute_all: null fsm_event_list
2006 Jan 19 22:10:59.246578 port-resources: fu_fsm_engine_post_event_processing:
mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 128136) dropped
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show port-resources module	ジェネレーション 2 モジュールのポート リソースに関する情報を表示します。

debug qos

Quality of Service (QoS; サービス品質) に対するデバッグをイネーブルにするには、EXEC モードで **debug qos** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug qos {all [interface fc slot/port] | detail | errors supervisor | flow | trace}
```

```
no debug qos {all [interface fc slot/port] | detail | errors supervisor | flow | trace}
```

シンタックスの説明

all	すべての QoS デバッグ オプションをイネーブルにします。
interface fc slot/port	指定のインターフェイスにデバッグを制限します。
detail	すべての QoS デバッグ出力をイネーブルにします。
errors supervisor	スーパーバイザ QoS のエラー状態に対するデバッグをイネーブルにします。
flow	フローレベルの QoS デバッグ オプションをイネーブルにします。
trace	QoS トレースに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug qos all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug qos all
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show qos	現在の QoS 設定 (プライオリティの高いフレームの数など) を表示します。

debug radius

ブート変数のデバッグをイネーブルにするには、EXEC モードで **debug radius** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug radius {aaa-request | aaa-request-lowlevel | all | config | config-lowlevel | server-monitor |
server-monitor-errors}
```

```
no debug radius {aaa-request | aaa-request-lowlevel | all | config | config-lowlevel | server-monitor |
server-monitor-errors}
```

シンタックスの説明

aaa-request	Remote Authentication Dial-In User Service (RADIUS) AAA 要求に対するデバッグをイネーブルにします。
aaa-request-lowlevel	RADIUS AAA 要求に対する低レベルのデバッグをイネーブルにします。
all	すべてのデバッグ フラグをイネーブルにします。
config	RADIUS 設定に対するデバッグをイネーブルにします。
config-lowlevel	RADIUS 設定の低レベル デバッグをイネーブルにします。
server-monitor	RADIUS サーバ モニタリングをイネーブルにします。
server-monitor-errors	RADIUS サーバ モニタ エラーをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(1)	このコマンドが導入されました。
3.0(1)	server-monitor オプションおよび server-monitor-errors オプションが追加されました。

使用上のガイドライン

なし

例

次に、**debug radius config-lowlevel** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug radius config-lowlevel
Nov 20 06:36:42 radius: radius_new_debug_conf_open: entering...
Nov 20 06:36:42 radius: radius_new_conf_close: entering...
Nov 20 06:36:42 radius: radius_new_conf_close: returning 0
Nov 20 06:36:42 radius: radius_new_enable_info_config: entering for Radius Daemon
debug
Nov 20 06:36:42 radius: radius_new_debug_conf_open: entering...
Nov 20 06:36:42 radius: radius_new_debug_conf_open: exiting
Nov 20 06:36:42 radius: radius_new_enable_info_config: SET_REQ for Radius Daemon debug
with 1
Nov 20 06:36:42 radius: radius_new_enable_info_config: SET_REQ done for Radius Daemon
debug with 1
Nov 20 06:36:42 radius: radius_new_enable_info_config: got back the return value of
configuration operation:success
Nov 20 06:36:42 radius: radius_new_debug_conf_close: entering...
Nov 20 06:36:42 radius: radius_new_debug_conf_close: returning 0
Nov 20 06:36:42 radius: radius_new_enable_info_config: exiting for Radius Daemon debug
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show radius	RADIUS Cisco Fabric Services (CFS) 配信ステータスおよびその他の詳細を表示します。

debug rd-reg

読み取り登録機能を使用しているデバイス リストに対するデバッグをイネーブルにするには、EXEC モードで **debug rd-reg** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug rd-reg [device-name | register address]
```

シンタックスの説明	パラメータ	説明
	<i>device-name</i>	必要なデバイスにデバイス名を指定します。
	<i>register address</i>	必要なデバイスの登録アドレスを指定します。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug rd-reg abc** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug rd-reg abc
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug rdl errors

Remote Digital Loopback (RDL) エラーに対するデバッグをイネーブルにするには、EXEC モードで **debug rdl errors** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

debug rdl errors

シンタックスの説明 このコマンドには、引数またはキーワードはありません。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug rdl errors** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug rdl errors
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug rib

Routing Information Base (RIB) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug rib** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug rib {all | detail | error | event | liod_error | liod_event | liod_trace | trace}
```

シンタックスの説明

all	すべての RIB 機能に対するデバッグをイネーブルにします。
detail	すべての RIB 機能に対する詳細デバッグをイネーブルにします。
error	RIB エラーに対するデバッグをイネーブルにします。
event	RIB イベントに対するデバッグをイネーブルにします。
liod_error	無損失順次配信 (LIOD) エラーに対するデバッグをイネーブルにします。
liod_event	LIOD エラーに対するデバッグをイネーブルにします。
liod_trace	LIOD トレース イベントに対するデバッグをイネーブルにします。
trace	トレース イベントに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。
3.0(1)	liod_error 、 liod_event 、および liod_trace の各オプションが追加されました。

使用上のガイドライン

RIB 操作が無視されるか、またはサポートされていない場合、**debug rib all** コマンドを実行して詳細情報を確認してください。

例

次に、**debug rib error** コマンドの例を示します。

```
switch# debug rib error
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug rlr

Registered Link Incident Report (RLIR) をイネーブルにするには、EXEC モードで **debug rlr** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug rlr {all | errors | events | mts-errors | mts-events}
```

```
no debug rlr {all | errors | events | mts-errors | mts-events}
```

シンタックスの説明	
all	すべての RLIR 機能に対するデバッグをイネーブルにします。
errors	RLIR のエラー状態に対するデバッグをイネーブルにします。
events	RLIR イベントに対するデバッグをイネーブルにします。
mts-errors	Messaging and Transaction Service (MTS) のエラー状態に対するデバッグをイネーブルにします。
mts-events	MTS イベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug rlr all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug rlr all
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show rlr	RILR、Link Incident Record Registration (LIPR)、および Distribute Registered Link Incident Record (DRLIR) フレームの情報を表示します。

debug rscn

Registered State Change Notification (RSCN) 機能に対するデバッグをイネーブルにするには、EXEC モードで **debug rscn** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug rscn {all | errors | events | mts-errors | mts-events} [vsan vsan-id]
```

```
no debug rscn {all | errors | events | mts-errors | mts-events} [vsan vsan-id]
```

シンタックスの説明

all	すべての RSCN 機能に対するデバッグをイネーブルにします。
errors	RSCN エラーに対するデバッグをイネーブルにします。
events	RSCN イベントに対するデバッグをイネーブルにします。
mts-errors	RSCN MTS エラーに対するデバッグをイネーブルにします。
mts-events	RSCN MTS イベントに対するデバッグをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug rscn errors** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug rscn errors
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show rscn	RSCN 情報を表示します。

debug san-ext-tuner

Storage Area Network (SAN; ストレージエリア ネットワーク) 拡張チューナーのデバッグをイネーブルにするには、EXEC モードで **debug san-ext-tuner** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug isns {all | demux | deque | error | event | ha | trace [detail] | warning}
```

```
no debug isns {all | bypass ficon_mgr | demux | deque | error | event | ha | trace [detail] | warning}
```

シンタックスの説明

all	すべての SAN 拡張チューナーのデバッグをイネーブルにします。
demux	SAN 拡張チューナー メッセージ デマルチプレクサに対するデバッグをイネーブルにします。
deque	SAN 拡張チューナー メッセージ デキューに対するデバッグをイネーブルにします。
error	SAN 拡張チューナーのエラー状態に対するデバッグをイネーブルにします。
event	SAN 拡張チューナー イベントに対するデバッグをイネーブルにします。
ha	SAN 拡張チューナー ハイ アベイラビリティに対するデバッグをイネーブルにします。
trace	SAN 拡張チューナー トレースに対するデバッグをイネーブルにします。
detail	SAN 拡張チューナー トレースに対する詳細デバッグをイネーブルにします。
warning	SAN 拡張チューナー警告に対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
2.0(x)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug san-ext-tuner error** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug san-ext-tuner error
```

関連コマンド

コマンド	説明
isns-server enable	iSNS サーバをイネーブルにします。
no debug all	すべてのデバッグをディセーブルにします。
show isns	iSNS 情報を表示します。
show san-ext-tuner	SAN 拡張チューナー情報を表示します。

debug scsi-flow

SCSI フローのデバッグをイネーブルにするには、**debug scsi-flow** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug scsi-flow {all | demux vsan vsan-id | deque | error | event vsan vsan-id | ha | trace {detail vsan vsan-id | vsan vsan-id} | warning vsan vsan-id}
```

```
no debug scsi-flow {all | demux vsan vsan-id | deque | error | event vsan vsan-id | ha | trace {detail vsan vsan-id | vsan vsan-id} | warning vsan vsan-id}
```

シンタックスの説明	all	すべての SCSI フローに対するすべてのデバッグ フラグをイネーブルにします。
	demux	SCSI フロー デマルチプレクサ機能に対するデバッグをイネーブルにします。
	deque	SCSI フロー デキュー イベントに対するデバッグをイネーブルにします。
	error	SCSI フロー エラーに対するデバッグをイネーブルにします。
	event	SCSI フロー イベントに対するデバッグをイネーブルにします。
	ha	SCSI フロー ハイ アベイラビリティ イベントに対するデバッグをイネーブルにします。
	trace	SCSI フロー トレースに対するデバッグをイネーブルにします。
	detail	SCSI フロー詳細トレースのデバッグをイネーブルにします。
	warning	SCSI フロー警告メッセージに対するデバッグをイネーブルにします。
	vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。有効範囲は 1 ~ 4093 です。

デフォルト なし

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	2.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、すべての SCSI フローに対するすべてのデバッグ フラグをイネーブルにする例を示します。

```
switch# debug scsi-flow all
2004 Nov 29 17:24:49 sfm: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
2004 Nov 29 17:24:49 sfm: fu_fsm_execute_all: null fsm_event_list
2004 Nov 29 17:24:49 sfm: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 536440) dropped
switch#
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show scsi-flow	SCSI フロー情報を表示します。

debug scsi-target

SCSI ターゲットに対するデバッグをイネーブルにするには、EXEC モードで **debug scsi-target** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug scsi-target {error | flow}
```

```
no debug scsi-target {error | flow}
```

シンタックスの説明	error	SCSI ターゲット デーモンのエラー状態に対するデバッグをイネーブルにします。
	flow	SCSI ターゲット フローに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.1(1)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug scsi-target flow** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug scsi-target flow
Apr 28 21:11:52 vhbad: vhba_mts_handler: sdwrap_dispatch: retval:0
Apr 28 21:11:54 vhbad: vhbad_handle_timeout: timer:1 context:(nil)
Apr 28 21:12:06 vhbad: vhba_mts_handler: sysmgr_dispatch: retval:-1
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show scsi-target	既存の SCSI ターゲットの設定情報を表示します。

debug sdv

SAN デバイス仮想化のデバッグをイネーブルにするには、EXEC モードで **debug sdv** コマンドを使用します。

```
debug sdv {all | all-sdv | ddas {errors | events} | ddas-config {errors | events | packets} | discovery
           {errors vsan vsan-id | events vsan vsan-id} | distribution {errors vsan vsan-id | events vsan
           vsan-id} | errors vsan vsan-id | fu {ha | transition} | mgmt {errors | events} | ns {errors | events |
           packets} | rewrite {errors | events | packets} | trace vsan vsan-id | virtual-domain {errors vsan
           vsan-id | events vsan vsan-id} | zone-activation {errors | events | packets}}
```

シンタックスの説明

all	すべての SDV デバッグを設定します。
all-sdv	SDV デバッグのすべてのフィルタを設定します。
ddas	DDAS デバッグをイネーブルにします。
errors	エラーに対するデバッグをイネーブルにします。
events	イベントに対するデバッグをイネーブルにします。
ddas-config	DDAS-CFG デバッグをイネーブルにします。
packets	パケットに対するデバッグをイネーブルにします。
discovery	Disc デバッグをイネーブルにします。
vsan vsan-id	VSAN（仮想 SAN）数を指定します。有効範囲は 1 ～ 4093 です。
distribution	Dist デバッグをイネーブルにします。
errors	Error デバッグをイネーブルにします。
fu	FU デバッグをイネーブルにします。
ha	FU HA デバッグをイネーブルにします。
mgmt	Config FSM デバッグをイネーブルにします。
ns	NS デバッグをイネーブルにします。
rewrite	Rewrite デバッグをイネーブルにします。
trace	Trace デバッグをイネーブルにします。
virtual-domain	Virtual Domain デバッグをイネーブルにします。
zone-activation	ZS-ACTV デバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
3.1(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例 次に、**debug sdv all** コマンドが実行された場合に表示される、システムの実出力結果の例を示します。

```
switch# debug sdv all
2007 Jan 26 22:17:25.232055 sdv: fu_fsm_execute_all: match_msg_id(0),
log_already_open(0)
2007 Jan 26 22:17:25.232151 sdv: fu_fsm_execute_all: null fsm_event_list
2007 Jan 26 22:17:25.232233 sdv: fu_fsm_engine_post_event_processing: mts msg
MTS_OPC_DEBUG_WRAP_MSG(msg_id 83409) dropped
```

関連コマンド

コマンド	説明
sdv enable	SAN デバイス バーチャライゼーションをイネーブルまたはディセーブルにします。
show sdv statistics	SAN デバイス バーチャライゼーションの統計情報を表示します。

debug security

セキュリティおよびアカウンティング機能に対するデバッグをイネーブルにするには、EXEC モードで **debug security** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug security {all | events | mts | radius}
```

```
no debug security {all | events | mts | radius}
```

シンタックスの説明	
all	すべてのセキュリティ機能に対するデバッグをイネーブルにします。
events	セキュリティ イベントに対するデバッグをイネーブルにします。
mts	セキュリティ Messaging and Transaction Service (MTS) パケットに対するデバッグをイネーブルにします。
radius	Remote Authentication Dial-In User Service (RADIUS) イベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンド モード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug security radius** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug security radius
Mar  5 00:51:13 securityd: RADIUS is enabled, hence it will be tried first for CHAP
authentication
Mar  5 00:51:13 securityd: reading RADIUS configuration
Mar  5 00:51:13 securityd: opening radius configuration for group:default
Mar  5 00:51:13 securityd: opened the configuration successfully
Mar  5 00:51:13 securityd: GET request for RADIUS global config
Mar  5 00:51:13 securityd: got back the return value of global radius configuration
operation:success
Mar  5 00:51:13 securityd: closing RADIUS pss configuration
Mar  5 00:51:13 securityd: opening radius configuration for group:default
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug sensor

センサー マネージャに対するデバッグをイネーブルにするには、EXEC モードで **debug sensor** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug sensor {demux | deque | error | info | init}
```

```
no debug sensor {demux | deque | error | info | init}
```

シンタックスの説明	説明
demux	センサー デマルチプレクサ機能に対するデバッグをイネーブルにします。
deque	センサー デキュー イベントに対するデバッグをイネーブルにします。
error	センサー エラーに対するデバッグをイネーブルにします。
info	センサー情報に対するデバッグをイネーブルにします。
init	センサーの初期化に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン このコマンドは、センサー マネージャのイベントおよび情報をデバッグします。

例 次に、**debug sensor info** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug sensor info
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show environment temperature	現在の温度しきい値設定および状態を表示します。

debug sme

Cisco SME 機能のデバッグをイネーブルにするには、**debug sme** コマンドを使用します。デバッグコマンドをディセーブルにするには、コマンドの **no** 形式を使用します。

```
debug sme {all | demux vsan vsan id | deque | error | event vsan vsan id | ha vsan vsan id | trace vsan vsan id | trace-detail vsan vsan id | warning vsan vsan id}
```

```
no debug sme {all | demux vsan vsan id | deque | error | event vsan vsan id | ha vsan vsan id | trace vsan vsan id | trace-detail vsan vsan id | warning vsan vsan id}
```

シンタックスの説明

all	すべての Cisco SME 機能のデバッグをイネーブルにします。
demux	Cisco SME メッセージ デマルチプレクサに対するデバッグをイネーブルにします。
vsan vsan id	指定 VSAN (仮想 SAN) ID へのデバッグを制限します。有効範囲は 1 ~ 4094 です。
deque	Cisco SME メッセージ デキューに対するデバッグをイネーブルにします。
error	Cisco SME エラーに対するデバッグをイネーブルにします。
event	Cisco SME の Finite State Machine (FSM) およびイベントに対するデバッグをイネーブルにします。
ha	Cisco SME のハイ アベイラビリティ (HA) に対するデバッグをイネーブルにします。
trace	Cisco SME トレースに対するデバッグをイネーブルにします。
trace-detail	Cisco SME トレース詳細に対するデバッグをイネーブルにします。
warning	Cisco SME 警告に対するデバッグをイネーブルにします。

デフォルト

なし

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
3.2(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug sme all** コマンドによるシステムの出力結果の例を示します。

```
switch# debug sme all
2007 Sep 23 15:44:44.490796 sme:fu_priority_select:- setting fd[5] for select
call
2007 Sep 23 15:44:44.490886 sme: fu_priority_select_select_queue: round credit(8)
)
2007 Sep 23 15:44:44.490918 sme: curr_q - FU_PSEL_Q_CAT_CQ, usr_q_info(2), p
riority(7), credit(4), empty
2007 Sep 23 15:44:44.490952 sme: fu_priority_select: returning FU_PSEL_Q_CAT_MTS
queue, fd(5), usr_q_info(1)
2007 Sep 23 15:44:44.491059 sme: sme_get_data_from_queue(1031): dequeued mts msg
(34916564), MTS_OPC_DEBUG_WRAP_MSG
2007 Sep 23 15:44:44.491096 sme: fu_fsm_engine: line[2253]
2007 Sep 23 15:44:44.492596 sme: fu_fsm_execute_all: match_msg_id(0), log_alread
y_open(0)
```

関連コマンド

コマンド	説明
<code>no debug all</code>	すべてのデバッグをディセーブルにします。
<code>show sme</code>	Cisco SME の情報をすべて表示します。

debug snmp

SNMP（簡易ネットワーク管理プロトコル）マネージャに対するデバッグをイネーブルにするには、EXEC モードで `debug snmp` コマンドを使用します。`debug` コマンドをディセーブルにするには、このコマンドの `no` 形式を使用するか、`no debug all` コマンドを使用してすべてのデバッグを終了させます。

```
debug snmp {
  all |
  errors |
  mts {pkt {both | rx [node range | opcode range | sap range] | tx} |
  pkthdr {both | rx [numpkt range] | tx}} |
  pkt-dump | trace {trace-entryexit | trace-stub}}

no debug snmp {
  all |
  errors |
  mts {pkt {both | rx [node range | opcode range | sap range] | tx} |
  pkthdr {both | rx [numpkt range] | tx}} |
  pkt-dump | trace {trace-entryexit | trace-stub}}
```

シンタックスの説明

<code>all</code>	すべての SNMP 出力に対するデバッグをイネーブルにします。
<code>errors</code>	SNMP のエラー出力に対するデバッグをイネーブルにします。
<code>mts</code>	SNMP パケットおよびヘッダーに対するデバッグをイネーブルにします。
<code>pkt-dump</code>	デバッグ出力のパケット ダンプをイネーブルにします。
<code>trace</code>	トレース レベルのデバッグ出力をイネーブルにします。
<code>pkt</code>	パケットに対するデバッグを指定します。
<code>pkthdr</code>	ヘッダーに対するデバッグを指定します。
<code>both</code>	送受信の両方向に対するデバッグを指定します。
<code>tx</code>	送信方向に対するデバッグを指定します。
<code>rx</code>	受信方向に対するデバッグを指定します。
<code>node</code>	受信方向のパケットのノードを指定します。
<code>opcode</code>	受信方向のパケットの <code>opcode</code> を指定します。
<code>sap</code>	受信方向のパケットの <code>sap</code> を指定します。
<code>numpkt</code>	必要なパケット数を指定します。
<code>range</code>	1 ~ 4095 までの整数の範囲を指定します。
<code>trace-entryexit</code>	トレース レベルのエントリまたは終了デバッグ出力を指定します。
<code>trace-stub</code>	トレース レベルのスタブ デバッグ出力を指定します。

デフォルト

ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug snmp trace** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug snmp trace
Apr 29 16:03:34 snmpd[1177]: SDWRAP message Successfully processed
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show snmp	SNMP ステータスおよび設定情報を表示します。
	snmp-server	SNMP サーバ情報、スイッチの場所、およびスイッチ名を設定する例を示します。
	snmp-server enable traps	SNMP サーバ通知（通知およびトラップ）をイネーブルにします。

debug span

Switched Port Analyzer (SPAN; スイッチド ポート アナライザ) のデバッグをイネーブルにするには、EXEC モードで **debug span** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug span {all | buffer-size bytes | error | event | trace | warning}
```

```
no debug span {all | error | event | trace | warning}
```

シンタックスの説明

all	すべての SPAN 機能に対するデバッグをイネーブルにします。
buffer-size bytes	SPAN のイベント ログ バッファ サイズを設定します。有効範囲は 4096 ~ 131072 です。
error	SPAN エラーに対するデバッグをイネーブルにします。
event	SPAN イベントに対するデバッグをイネーブルにします。
ha	SPAN HA (ハイ アベイラビリティ) に対するデバッグをイネーブルにします。
lib	SPAN ライブラリに対するデバッグをイネーブルにします。
trace	SPAN トレースに対するデバッグをイネーブルにします。
warning	SPAN の警告メッセージに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug span all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug span all
Apr 29 16:06:44 span: span_demux: msg consumed by sdwrap_process msg
Apr 29 16:06:44 span: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr 29 16:06:44 span: fu_fsm_execute_all: null_fsm_event_list
Apr 29 16:06:44 span: fu_fsm_engine: mts msg MTS_OPC_DEBUG_WRAP_MSG(msg_id 2548887)
dropped
Apr 29 16:06:48 span: fu_priority_select: - setting fd[3] for select call
Apr 29 16:06:48 span: fu_priority_select_select_queue: round credit(12)
Apr 29 16:06:48 span: curr_q - FU_PSEL_Q_CAT_CQ, usr_q_info(4), priority(7),
credit(6), empty
Apr 29 16:06:48 span: fu_priority_select: returning FU_PSEL_Q_CAT_MTS queue, fd(3),
usr_q_info(2)
Apr 29 16:06:48 span: span_get_data_from_mts_q dequeued mts msg (26e525),
MTS_OPC_DEBUG_WRAP_MSG
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show span session	SPAN セッションの特定の情報を表示します。

debug system health

システムヘルスモニタリングデバッグをイネーブルにするには、EXECモードで **debug system health** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug system health {all | asic-counters | battery-charger | bootflash | cache-disk | cfr | eobc | error |
event | external-loopback | failure-analysis | fc2 | free-disk | ha | inband | loopback | mgmt | misc
| mts | nvram | plog | pss | serdes | special | trace | xipc}
```

```
no debug system health {all | asic-counters | battery-charger | bootflash | cache-disk | cfr | eobc | error
| event | external-loopback | failure-analysis | fc2 | free-disk | ha | inband | loopback | mgmt | misc
| mts | nvram | plog | pss | serdes | special | trace | xipc}
```

シンタックスの説明

all	すべてのオンラインヘルスフラグのデバッグをイネーブルにします。
asic-counters	システムヘルスASIC統計情報のデバッグをイネーブルにします。
battery-charger	システムヘルスバッテリー充電テストのデバッグをイネーブルにします。
bootflash	システムヘルスブートフラッシュテストのデバッグをイネーブルにします。
cache-disk	システムヘルスキャッシュディスクテストのデバッグをイネーブルにします。
cfr	システムヘルスコンパクトヘルステストのデバッグをイネーブルにします。
eobc	システムヘルスEthernet Out-of-Band Channel (EOBC) テストのデバッグをイネーブルにします。
error	システムヘルスのエラー状態のデバッグをイネーブルにします。
event	システムヘルスイベントのデバッグをイネーブルにします。
external-loopback	システムヘルス外部ループバックテストのデバッグをイネーブルにします。
failure-analysis	システムヘルス障害分析のデバッグをイネーブルにします。
fc2	システムヘルスFC2フレームのデバッグをイネーブルにします。
free-disk	システムヘルス空きディスクのデバッグをイネーブルにします。
ha	ヘルスモニタリングハイアベイラビリティフラグに対するデバッグをイネーブルにします。
inband	システムヘルス帯域内テストのデバッグをイネーブルにします。
loopback	システムヘルスループバックテストのデバッグをイネーブルにします。
mgmt	システムヘルス管理ポートのポートテストのデバッグをイネーブルにします。
misc	システムヘルスmiscのデバッグをイネーブルにします。
mts	システムヘルスMessaging and Transaction Service (MTS) のデバッグをイネーブルにします。
nvram	システムヘルスNVRAM(不揮発性RAM)のデバッグをイネーブルにします。
plog	システムヘルス永続ロギングのデバッグをイネーブルにします。
pss	システムヘルスPSSのデバッグをイネーブルにします。
serdes	システムヘルスSerializing/Deserializing Method (SerDes) テストのデバッグをイネーブルにします。
special	システムヘルススペシャルのデバッグをイネーブルにします。
trace	ヘルスモニタリングトレースフラグのデバッグをイネーブルにします。
xipc	システムヘルスXIPCのデバッグをイネーブルにします。

デフォルト

ディセーブル

■ debug system health

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。
	3.0(1)	free-disk 、 nvr am、および plog の各オプションが追加されました。

使用上のガイドライン なし

例 次に、**debug system health** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug system health all
2005 Mar 10 01:49:28 SystemHealth: ohms_snake_fd_activity: Module 1 Snake Frame came.
2005 Mar 10 01:49:28 SystemHealth: ohms_snake_fd_activity: Module 8 waiting for Snake
Frame to come.
2005 Mar 10 01:49:28 SystemHealth: ohms_dequeue: select timeout 0 998000
2005 Mar 10 01:49:28 SystemHealth: fu_priority_select: - setting fd[4] for select
call - setting fd[20] for select call - setting fd[22] for select call - setting
fd[28] for select call - setting fd[29] for select call - setting fd[30] for select
call
2005 Mar 10 01:49:28 SystemHealth: fu_priority_select_select_queue: round credit(14)
2005 Mar 10 01:49:28 SystemHealth: curr_q - FU_PSEL_Q_CAT_FD, usr_q_info(466240),
fd(29), priority(6), credit(3), empty
2005 Mar 10 01:49:28 SystemHealth: fu_priority_select: returning FU_PSEL_Q_CAT_CQ
queue, usr_q_info(1)
2005 Mar 10 01:49:28 SystemHealth: ohms_dequeue: Select woken up
2005 Mar 10 01:49:28 SystemHealth: ohms_dequeue: Process event type 0x1
2005 Mar 10 01:49:28 SystemHealth: ohms_dequeue: Processing timer type
2005 Mar 10 01:49:28 SystemHealth: fu_fsm_engine: line[2139]
2005 Mar 10 01:49:28 SystemHealth: fu_fsm_handle_sysmgr_msg: Not mts event
2005 Mar 10 01:49:28 SystemHealth: ohms_timer_event_handler: called.
2005 Mar 10 01:49:28 SystemHealth: fu_fsm_execute_all: match_msg_id(0),
log_already_open(0)
.
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show system health	設定した Online Health Management System (OHMS) 情報を表示します。

debug tacacs+

ブート変数のデバッグをイネーブルにするには、EXEC モードで **debug tacacs+** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug tacacs+ {aaa-request | aaa-request-lowlevel | all | config | config-lowlevel | server-monitor |
server-monitor-errors}
```

```
no debug tacacs+ {aaa-request | aaa-request-lowlevel | all | config | config-lowlevel | server-monitor
| server-monitor-errors}
```

シンタックスの説明

aaa-request	TACACS+ AAA 要求に対するデバッグをイネーブルにします。
aaa-request-lowlevel	TACACS+ AAA 要求に対する低レベルのデバッグをイネーブルにします。
all	すべてのデバッグ フラグをイネーブルにします。
config	Terminal Access Controller Access Control System Plus (TACACS+) 設定に対するデバッグをイネーブルにします。
config-lowlevel	TACACS+ 設定の低レベル デバッグをイネーブルにします。
server-monitor	TACACS+ サーバ モニタリングをイネーブルにします。
server-monitor-errors	TACACS+ サーバ モニタ エラーをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.3(1)	このコマンドが導入されました。
3.0(1)	server-monitor オプションおよび server-monitor-errors オプションが追加されました。

使用上のガイドライン

なし

例

次に、**debug tacacs+ config-lowlevel** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug tacacs+ config-lowlevel
Nov 20 06:39:44 tacacs: tacacs_debug_conf_open: entering...
172.22.94.252# Nov 20 06:39:44 tacacs: tacacs_debug_conf_open: exiting
Nov 20 06:39:44 tacacs: tacacs_conf_close: entering...
Nov 20 06:39:44 tacacs: tacacs_conf_close: returning 0
Nov 20 06:39:44 tacacs: tacacs_enable_info_config: entering for TACACS+ Daemon debug
Nov 20 06:39:44 tacacs: tacacs_debug_conf_open: entering...
Nov 20 06:39:44 tacacs: tacacs_debug_conf_open: exiting
Nov 20 06:39:44 tacacs: tacacs_enable_info_config: SET_REQ for TACACS+ Daemon debug
with 1
Nov 20 06:39:44 tacacs: tacacs_enable_info_config: SET_REQ done for TACACS+ Daemon
debug with 1
Nov 20 06:39:44 tacacs: tacacs_enable_info_config: got back the return value of
configuration operation:success
Nov 20 06:39:44 tacacs: tacacs_debug_conf_close: entering...
Nov 20 06:39:44 tacacs: tacacs_debug_conf_close: returning 0
Nov 20 06:39:44 tacacs: tacacs_enable_info_config: exiting for TACACS+ Daemon debug
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show tacacs+	TACACS+ Cisco Fabric Services (CFS) 配信ステータスおよびその他の詳細を表示します。

debug tcap

例外ログに対するデバッグをイネーブルにするには、EXEC モードで **debug tcap** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug tcap {demux | deque | error | info | init}
```

```
no debug tcap {demux | deque | error | info | init}
```

シンタックスの説明	オプション	説明
	demux	端末キャプチャ デマルチプレクサ機能に対するデバッグをイネーブルにします。
	deque	端末キャプチャ デキュー イベントに対するデバッグをイネーブルにします。
	error	端末キャプチャ エラーに対するデバッグをイネーブルにします。
	info	端末キャプチャ情報に対するデバッグをイネーブルにします。
	init	端末キャプチャの初期化に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン このコマンドは、端末キャプチャ ユーティリティのイベントおよび情報をデバッグします。

例 次に、**debug tcap demux** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug tcap demux
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug tlport

Transitive Loop (TL) ポート インターフェイスのデバッグをイネーブルにするには、EXEC モードで **debug tlport** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug tlport {all | errors | events {fc2 {terminal | transit} | mts | pss}} [interface fc slot/port]
```

```
no debug tlport {all | errors | events {fc2 {terminal | transit} | mts | pss}} [interface fc slot/port]
```

シンタックスの説明

all	すべての TL ポート機能に対するデバッグをイネーブルにします。
errors	TL ポートのエラー状態に対するデバッグをイネーブルにします。
events	TL ポート モニタリング イベントに対するデバッグをイネーブルにします。
fc2	TL ポート モニタリング FC 2 イベントに対するデバッグをイネーブルにします。
terminal	TL ポート モニタリング FC 2 終了イベントを指定します。
transit	TL ポート モニタリング FC 2 送信イベントを指定します。
mts	TL ポート モニタリング Messaging and Transaction Service (MTS) パケットに対するデバッグをイネーブルにします。
pss	TL ポート モニタリング PSS パケットに対するデバッグをイネーブルにします。
interface fc slot/port	指定のインターフェイスにデバッグを制限します。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug tlport events pss** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug tlport events pss
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show tlport	設定した TL ポート情報を表示します。

debug ttyd

TTYD デバッグをイネーブルにするには、EXEC モードで **debug ttyd** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug ttyd {all | errors | events}
```

```
no debug ttyd {all | errors | events}
```

シンタックスの説明		
all		すべての TTYD 機能に対するデバッグをイネーブルにします。
errors		TTYD のエラー状態に対するデバッグをイネーブルにします。
events		TTYD イベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug ttyd events** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug ttyd events
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug vni

Virtual Network Interface (VNI) に対するデバッグをイネーブルにするには、EXEC モードで **debug vni** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug vni {all | errors | events | info | pss}
```

```
no debug vni {all | errors | events | info | pss}
```

シンタックスの説明

all	すべての VNI 機能に対するデバッグをイネーブルにします。
errors	VNI のエラー状態に対するデバッグをイネーブルにします。
events	VNI イベントに対するデバッグをイネーブルにします。
info	VNI イベントに対するデバッグをイネーブルにします。
pss	VNI PSS パケットに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug vni all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug vni all
Apr 29 17:00:59 vni: Received MTS message
Apr 29 17:00:59 vni: message not processed by system mgr library , so process it
normal way
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug vrrp

Virtual Router Redundancy Protocol (VRRP) に対するデバッグをイネーブルにするには、EXEC モードで **debug vrrp** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug vrrp {configuration | engine} {all | error | event | info}
```

```
no debug vrrp {configuration | engine} {all | error | event | info}
```

シンタックスの説明	
configuration	VRRP 設定に対するデバッグをイネーブルにします。
engine	VRRP エンジンに対するデバッグをイネーブルにします。
all	すべての VRRP 機能に対するデバッグをイネーブルにします。
error	VRRP のエラー状態に対するデバッグをイネーブルにします。
event	VRRP イベントに対するデバッグをイネーブルにします。
info	VRRP イベントに対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug vrrp engine all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug vrrp engine all
Apr 29 17:35:58 vrrp_eng: fu_priority_select: - setting fd[7] for select call -
setting fd[11] for select call - setting fd[12] for select call - setting fd [13]
for select call - setting fd[15] for select call
Apr 29 17:35:58 vrrp_eng: fu_priority_select_select_queue: round credit(6)
Apr 29 17:35:58 vrrp_eng: curr_q - FU_PSEL_Q_CAT_FD, usr_q_info(6), fd(15),
priority(2), credit(1), empty
Apr 29 17:35:58 vrrp_eng: fu_priority_select: returning FU_PSEL_Q_CAT_FD queue, fd(7),
usr_q_info(3)
Apr 29 17:35:58 vrrp_eng: heartbeat sent
Apr 29 17:35:58 vrrp_eng: message not processed by system mgr library , so process it
normal way
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show vrrp	VRRP 設定情報を表示します。

debug vsan

VSAN (仮想 SAN) のデバッグをイネーブルにするには、EXEC モードで **debug vsan** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug vsan {all | global | ha | info | membership | mts}
```

```
no debug vsan {all | global | ha | info | membership | mts}
```

シンタックスの説明

all	VSAN 機能のすべてのデバッグ フラグをイネーブルにします。
global	VSAN グローバル パラメータ データベースのイベントに対するデバッグをイネーブルにします。
ha	VSAN のハイ アベイラビリティ関連のイベントに対するデバッグをイネーブルにします。
info	VSAN 情報データベースのイベントに対するデバッグをイネーブルにします。
membership	VSAN メンバシップ データベースのイベントに対するデバッグをイネーブルにします。
mts	Messaging and Transaction Service (MTS) の Tx/Rx パケットのデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug vsan all** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug vsan all
2005 Mar 10 01:44:35 vsan: Calling handling function
2005 Mar 10 01:44:35 vsan: querying trunking membership(readonly) for
interface:16859136
2005 Mar 10 01:44:35 vsan: Replying to trunking membership query for interface:fc1/21
with VSAN bitmap:1-4093
2005 Mar 10 01:44:35 vsan: got back reply_code:0
2005 Mar 10 01:44:35 vsan: Returned from handling function
2005 Mar 10 01:44:35 vsan: Freeing notifications
2005 Mar 10 01:44:35 vsan: Src: 0x00000601/15 Dst: 0x00000601/27 ID: 0x0067CEA1
Size: 520 [RSP] Opc: 116 (MTS_OPC_VSAN_GET_PORT_TRUNKING_MEMBERSHIP) RR: 0x0067CEA0
HA_SEQNO: 0x00000000 TS: 0x24E717EAC7CE2 REJ:0 SYNC:1
2005 Mar 10 01:44:35 vsan: 00 00 00 00 00 00 02 00 7F FF FF FF FF FF FF
2005 Mar 10 01:44:35 vsan: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
2005 Mar 10 01:44:35 vsan: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
2005 Mar 10 01:44:35 vsan: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
2005 Mar 10 01:44:35 vsan: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
.
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。
	show vsan	設定した VSAN の情報を表示します。

debug wr-reg

書き込み登録機能を使用しているデバイス リストに対するデバッグをイネーブルにするには、EXEC モードで **debug wr-reg** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug wr-reg [device-name | register-address]
```

```
no debug wr-reg [device-name | register-address]
```

シンタックスの説明	device-name	register-address
	必要なデバイスにデバイス名を指定します。	必要なデバイスの登録アドレスを指定します。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug wr-reg** コマンドが実行された場合に表示される、システムの実行結果の例を示します。

```
switch# debug wr-reg
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug wwn

World Wide Name (WWN) マネージャに対するデバッグをイネーブルにするには、EXEC モードで **debug wwn** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug wwn {all | detail | errors | flow | trace}
```

```
no debug wwn {all | detail | errors | flow | trace}
```

シンタックスの説明

all	すべての WWN デバッグ オプションをイネーブルにします。
detail	すべての WWN 出力をイネーブルにします。
error	WWN のエラー状態に対するデバッグをイネーブルにします。
flow	フローレベルの WWN デバッグ オプションをイネーブルにします。
trace	WWN トレースに対するデバッグをイネーブルにします。

デフォルト

ディセーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug wwn all** コマンドが実行された場合に表示される、システムの出力量の結果の例を示します。

```
switch# debug wwn all
Apr 29 19:24:17 wwn: 53601-wwnm_sdwrap_dispatch:77|SDWRAP message Successfully
processed
Apr 29 19:24:17 wwn: Src: 0x00000601/5206 Dst: 0x00000601/46 ID: 0x002C7DE4 Size:
252 [REQ] OpC: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x002C7DE4 HA_SEQNO: 0x00000000 TS:
0x55D49A130243 REJ:0
Apr 29 19:24:17 wwn: 2F 64 65 76 2F 70 74 73 2F 30 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 29 19:24:17 wwn: 53601-wwnm_unmask_sigalrm:1261|TRACE:
FILE=_manager/wwnm/wwnm_utilities.c
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。
show wwn	WWN 設定のステータスを表示します。

debug xbar

Crossbar Debugging (XBAR) をイネーブルにするには、EXEC モードで **debug xbar** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug xbar {all | demux | deque | error [module slot] | fsm [module slot] | ha [module slot] |
init | main}
```

シンタックスの説明

all	すべての XBAR デバッグ オプションをイネーブルにします。
demux	XBAR デマルチプレクサ機能に対するデバッグをイネーブルにします。
deque	XBAR デキュー イベントに対するデバッグをイネーブルにします。
error	XBAR エラーに対するデバッグをイネーブルにします。
fsm	XBAR FSM に対するデバッグをイネーブルにします。
ha	XBAR のハイ アベイラビリティ情報に対するデバッグをイネーブルにします。
init	XBAR の初期化に対するデバッグをイネーブルにします。
main	主要な機能に対する XBAR デバッグをイネーブルにします。
module slot	デバッグするモジュールのスロット番号を指定します。

デフォルト

イネーブル

コマンドモード

EXEC モード

コマンド履歴

リリース	変更内容
1.0(2)	このコマンドが導入されました。

使用上のガイドライン

なし

例

次に、**debug xbar all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug xbar all
Apr 29 19:48:34 xbar: its a sdwrap msg, fsm utils dropping the mts msg
Apr 29 19:48:34 xbar: fu_fsm_engine: (Error) SYSERR_FU_xx: 0x10, err_num (16) in demux
Apr 29 19:48:34 xbar: fu_fsm_execute_all: match_msg_id(0), log_already_open(0)
Apr 29 19:48:34 xbar: fu_fsm_execute_all: null fsm_event_list
...
```

関連コマンド

コマンド	説明
no debug all	すべてのデバッグをディセーブルにします。

debug xbar_driver

Crossbar Driver (XBAR Driver) をイネーブルにするには、EXEC モードで **debug xbar_driver** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug xbar {error | flow | trace}
```

シンタックスの説明	error	XBAR Driver エラーに対するデバッグをイネーブルにします。
	flow	XBAR Driver フローに対するデバッグをイネーブルにします。
	trace	XBAR Driver トレースに対するデバッグをイネーブルにします。

デフォルト イネーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	3.0(1)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug xbar_driver** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug xbar_driver error
switch# 2006 Jan 23 22:02:41.770329 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:03:41.780356 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:04:41.780356 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:05:41.780357 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:06:41.780356 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:07:41.780359 xbar_driver: sc_stats_timer_hdlr called
2006 Jan 23 22:08:41.790341 xbar_driver: sc_stats_timer_hdlr called...
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug xbc

Crossbar Client Debugging (XBC) をイネーブルにするには、EXEC モードで **debug xbc** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug xbc {demux | deque | init | main}
```

```
no debug xbc {demux | deque | init | main}
```

シンタックスの説明	説明
demux	クロスバー デマルチプレクサ機能に対するデバッグをイネーブルにします。
deque	クロスバー デキュー イベントに対するデバッグをイネーブルにします。
init	クロスバーの初期化に対するデバッグをイネーブルにします。
main	クロスバーの主要な機能に対するデバッグをイネーブルにします。

デフォルト ディセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン このコマンドは、クロスバー クライアントのイベントおよび情報をデバッグします。

例 次に、**debug xbc init** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug xbc init
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをディセーブルにします。

debug zone

ゾーンに対するデバッグをイネーブルにするには、EXEC モードで **debug zone** コマンドを使用します。**debug** コマンドをディセーブルにするには、このコマンドの **no** 形式を使用するか、**no debug all** コマンドを使用してすべてのデバッグを終了させます。

```
debug zone {all |
  change {errors | events | packets} |
  database {detail | errors | events} |
  gs errors {errors | events | packets} |
  lun-zoning {errors | events | packets} |
  merge {errors | events | packets} |
  mts notifications |
  pss {errors | events} ||
  read-only-zoning {errors | events | packets} |
  tcam errors {errors | events | packets} |
  transit {errors | events}} [vsan vsan-id]

no debug zone {all |
  change {errors | events | packets} |
  database {detail | errors | events} |
  gs errors {errors | events | packets} |
  lun-zoning {errors | events | packets} |
  merge {errors | events | packets} |
  mts notifications |
  pss {errors | events} ||
  read-only-zoning {errors | events | packets} |
  tcam errors {errors | events | packets} |
  transit {errors | events}} [vsan vsan-id]
```

シンタックスの説明

all	すべてのゾーン サーバ デバッグ オプションをイネーブルにします。
vsan vsan-id	指定の VSAN (仮想 SAN) にデバッグを制限します。
change	変更プロトコル メッセージに対するデバッグをイネーブルにします。
database	ゾーン データベース メッセージに対するデバッグをイネーブルにします。
errors	ゾーン エラーに対するデバッグをイネーブルにします。
events	ゾーン イベントに対するデバッグをイネーブルにします。
packets	ゾーン パケットに対するデバッグをイネーブルにします。
database	データベース メッセージに対するデバッグをイネーブルにします。
gs	Ground Start (GS) プロトコル メッセージに対するデバッグをイネーブルにします。
lun-zoning	Logical Unit Number (LUN) ゴーニング メッセージに対するデバッグをイネーブルにします。
merge	結合プロトコル メッセージに対するデバッグをイネーブルにします。
mts notification	Messaging and Transaction Service (MTS) 通知メッセージに対するデバッグをイネーブルにします。
pss	PSS デバッグ メッセージに対するデバッグをイネーブルにします。
read-only-zoning	読み取り専用 ゴーニング メッセージに対するデバッグをイネーブルにします。
tcam	Ternary Content Addressable Memory (TCAM) メッセージに対するデバッグをイネーブルにします。
transit	送信フレーム メッセージに対するデバッグをイネーブルにします。

■ debug zone

デフォルト デイセーブル

コマンドモード EXEC モード

コマンド履歴	リリース	変更内容
	1.0(2)	このコマンドが導入されました。

使用上のガイドライン なし

例 次に、**debug zone all** コマンドが実行された場合に表示される、システムの出力結果の例を示します。

```
switch# debug zone all
2005 Mar 10 01:46:36 zone: Src: 0x00000601/18 Dst: 0x00000601/94 ID: 0x0067D5CD
Size: 276 [REQ] Opc: 182 (MTS_OPC_DEBUG_WRAP_MSG) RR: 0x0067D5CD HA_SEQNO: 0x00000000
TS: 0x24E95060E0EF4 REJ:0 SYNC:0
2005 Mar 10 01:46:36 zone: 01 00 00 00 E8 03 00 00 00 00 00 00 00 00 00 00
2005 Mar 10 01:46:36 zone: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2005 Mar 10 01:46:36 zone: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2005 Mar 10 01:46:36 zone: FF FF FF FF 2F 64 65 76 2F 70 74 73 2F 30 00 00
2005 Mar 10 01:46:36 zone: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2005 Mar 10 01:46:36 zone: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
.
```

関連コマンド	コマンド	説明
	no debug all	すべてのデバッグをデイセーブルにします。
	show zone	ゾーン情報を表示します。