



## ハイ アベイラビリティ コマンド

---

- clear diagnostic event-log (3 ページ)
- diagnostic monitor (4 ページ)
- diagnostic schedule module (6 ページ)
- diagnostic start (9 ページ)
- diagnostic stop (12 ページ)
- domain id (14 ページ)
- dual-active detection pagp (15 ページ)
- dual-active recovery-reload-disable (16 ページ)
- hw-module beacon switch (17 ページ)
- hw-module switch slot (18 ページ)
- hw-module switch usbflash (20 ページ)
- main-cpu (21 ページ)
- maintenance-template (22 ページ)
- mode sso (23 ページ)
- policy config-sync pre reload (24 ページ)
- redundancy (25 ページ)
- redundancy force-switchover (26 ページ)
- redundancy reload peer (27 ページ)
- redundancy reload shelf (29 ページ)
- reload (30 ページ)
- router routing protocol shutdown l2 (31 ページ)
- set platform software fed switch (32 ページ)
- set platform software nif-mgr switch (33 ページ)
- show diagnostic bootup (34 ページ)
- show diagnostic content (35 ページ)
- show diagnostic description (39 ページ)
- show diagnostic events (41 ページ)
- show diagnostic result (43 ページ)
- show diagnostic simulation failure (48 ページ)

- show diagnostic schedule (49 ページ)
- show hw-module switch subslot (50 ページ)
- show logging onboard switch (52 ページ)
- show platform software fed (56 ページ)
- show platform software nif-mgr switch (59 ページ)
- show redundancy (64 ページ)
- show redundancy config-sync (68 ページ)
- show stackwise-virtual (70 ページ)
- show tech-support stack (72 ページ)
- stackwise-virtual (74 ページ)
- stackwise-virtual dual-active-detection (75 ページ)
- stackwise-virtual link (76 ページ)
- standby console enable (77 ページ)
- start maintenance (78 ページ)
- stop maintenance (79 ページ)
- system mode maintenance (80 ページ)

## clear diagnostic event-log

特定のスイッチモジュールまたはイベントタイプの診断イベントログをクリアするには、特権 EXEC モードで **clear diagnostic event-log** コマンドを使用します。

```
clear diagnostic event-log [{event-type {error | info | warning} | switch {switch_num module
module_num | all [{event-type {error | info | warning}}]}]}
```

構文の説明	説明
<b>event-type error</b>	エラーイベントをクリアします。
<b>event-type info</b>	情報イベントをクリアします。
<b>event-type warning</b>	警告イベントをクリアします。
<b>switch num</b>	特定のスイッチのイベントをクリアします。
<b>module num</b>	特定のモジュールのイベントをクリアします。
<b>switch all</b>	すべてのスイッチのすべてのイベントログをクリアします。

コマンドモード 特権 EXEC (#)

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました

### 例

次に、エラーイベントログをクリアする例を示します。

```
Device# clear diagnostic event-log event-type error
```

次に、スイッチ 1 モジュール 1 のイベントログをクリアする例を示します。

```
Device# clear diagnostic event-log switch 1 module 1
```

次に、すべてのスイッチのエラーイベントログをクリアする例を示します。

```
Device# clear diagnostic event-log switch all
```

### 関連コマンド

コマンド	説明
<b>show diagnostic events</b>	診断イベントログを表示します。

# diagnostic monitor

ヘルスマモニタリング診断テストを設定するには、グローバルコンフィギュレーションモードで **diagnostic monitor** コマンドを使用します。テストをディセーブルにし、デフォルト設定に戻すには、このコマンドの **no** 形式を使用します。

**diagnostic monitor interval switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day* [**cardindex** *number*]

**diagnostic monitor switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} [**cardindex** *number*]

**diagnostic monitor threshold switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count* [**days** *number* | **hours** *number* | **milliseconds** *number* | **minutes** *number* | **runs** *number* | **seconds** *number*] **cardindex** *number*

**no diagnostic monitor interval switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} [**cardindex** *number*]

**no diagnostic monitor switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} [**cardindex** *number*]

**no diagnostic monitor threshold switch** *number* **module** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} { **failure count** [[*count* [**days** *number* | **hours** *number* | **milliseconds** *number* | **minutes** *number* | **runs** *number* | **seconds** *number*] | **cardindex** *number*] | **cardindex** *number*] }

## 構文の説明

<b>interval</b>	テストの間隔を設定します。
<b>switch</b> <i>number</i>	スイッチ番号（スタックメンバ番号）を指定します。スイッチがスタンドアロンスイッチの場合、スイッチ番号は1です。スイッチがスタック内にある場合、スタック内のスイッチメンバ番号に応じて1～9を指定できます。 このキーワードは、スタック対応スイッチでのみサポートされています。
<b>test</b>	実行するテストを指定します。
<i>name</i>	テストの名前。
<i>test-id</i>	テストのID番号。
<i>test-id-range</i>	テストのID番号の範囲。カンマおよびハイフンで区切られた整数で範囲を入力します（例：1,3-6はテストID 1、3、4、5および6）。
<b>all</b>	すべての診断テストを指定します。

<i>hh:mm:ss</i>	モニタリング間隔（時間、分、秒）。時間（0～24）、分（0～60）、秒（0～60）を入力します。
<i>milliseconds</i>	モニタリング間隔（ミリ秒（ms））。テスト時間をミリ秒（0～999）で入力します。
<i>day</i>	モニタリング間隔（日数）。テストの間隔を日数（0～20）で入力します。
<b>threshold</b>	障害しきい値を設定します。
<b>failure count</b> <i>count</i>	障害しきい値のカウントを設定します。
<b>cardindex</b> <i>number</i>	（任意）カードインデックス番号を指定します。

**コマンドデフォルト** モニタリングはディセーブルで、障害しきい値は設定されていません。

**コマンドモード** グローバル コンフィギュレーション（config）

コマンド履歴	リリース	変更内容
	Cisco IOS XE Gibraltar 16.11.1	このコマンドが導入されました。

**使用上のガイドライン** 診断モニタリングをイネーブルにする前に、障害しきい値およびテストの間隔を設定する必要があります。

**diagnostic monitor switch module test** コマンドを入力する際は、すべての接続ポートをディセーブルにしてネットワークトラフィックを隔離する必要があります。また、テスト中はテストパケットを送信しないでください。

例

次に、テスト 1 の障害しきい値カウントを 20 に設定する例を示します。

```
Device# configure terminal
Device(config)# diagnostic monitor threshold switch 2 test 1 failure count 20
```

次に、テスト 2 のモニタリング間隔を設定する例を示します。

```
Device# configure terminal
Device(config)# diagnostic monitor interval switch 2 test 2 12:30:00 750 5
```

関連コマンド

コマンド	説明
<b>show diagnostic content switch module</b>	オンライン診断テストの結果を表示します。

## diagnostic schedule module

特定のスイッチモジュールに対するテストベースの診断タスクをスケジューリングしたり、スーパーバイザエンジンのスイッチオーバーをスケジューリングしたりするには、グローバルコンフィギュレーションモードで **diagnostic schedule switch module** コマンドを使用します。スケジュールを削除するには、このコマンドの **no** 形式を使用します。

```
diagnostic schedule switch number module module-num test {test-id | {{complete |
minimal}} {dailyhh:mm | onmonth | weekly day-of-week }} | {{all | basic | non-disruptive | per-port
}} {dailyhh:mm | onmonth | port {interface-port-number | port-number-list | all {daily hh:mm | on month
| weekly day-of-week }} | weekly day-of-week }}
```

```
no diagnostic schedule switch number module module-num test {test-id | {{complete |
minimal}} {dailyhh:mm | onmonth | weekly day-of-week }} | {{all | basic | non-disruptive | per-port
}} {dailyhh:mm | onmonth | port {interface-port-number | port-number-list | all {daily hh:mm | on month
| weekly day-of-week }} | weekly day-of-week }}
```

### 構文の説明

<b>switch</b> <i>switch_num</i>	スイッチ番号を指定します。
<b>module</b> <i>module_num</i>	モジュール番号を指定します。
<b>test</b>	診断テストスイート属性を指定します。
<i>test-id</i>	実行するテストの ID 番号。 テスト ID のリストを表示するには、 <b>show diagnostic content</b> コマンドを使用します。
<b>all</b>	すべての診断テストを実行します。
<b>complete</b>	すべてのブートアップテストスイートを選択します。
<b>minimal</b>	最小限のブートアップテストスイートを選択します。
<b>non-disruptive</b>	中断を伴わないテストスイートを選択します。
<b>per-port</b>	ポート単位のテストスイートを選択します。 <b>per-port</b> は、スケジューリングされたスイッチオーバーを指定する場合はサポートされません。
<b>port</b>	(任意) テストのスケジュールを設定するポートを指定します。
<i>interface-port- number</i>	(任意) ポート番号です。範囲は 1 ~ 48 です。

<i>port-number-list</i>	(任意) ポート番号の範囲 (ハイフンで区切ります)。範囲は 1 ~ 48 です。
<b>all</b>	(任意) すべてのポートを指定します。
<b>on month</b>	テストベースの診断タスクのスケジュールを指定します。  January や February など、月の名前を大文字または小文字のいずれかで入力します。
<b>daily hh:mm</b>	テストベースの診断タスクの日次スケジュールを指定します。  2桁の数字 (24 時間表記) で時間および分を入力します。コロン (:) が必要です。
<b>weekly day-of-week</b>	テストベースの診断タスクの週次スケジュールを指定します。  Monday や Tuesday など、曜日を大文字または小文字のいずれかで入力します。

コマンドデフォルト

特定のスイッチモジュールに対するテストベースの診断タスクはスケジューリングされていません。

コマンドモード

グローバル コンフィギュレーション (config)

コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

使用上のガイドライン

アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへのスイッチオーバーをスケジューリングするには、**diagnostic schedule switch module test** コマンドを実行します。

**show diagnostic content switch module** コマンドを実行すると、テスト ID のリストが表示されます。**ScheduleSwitchover** フィールドにテスト ID が表示されます。

次のコマンドを使用すると、定期的なスイッチオーバー (毎日または毎週) または指定した時点での 1 回のスイッチオーバーを指定できます。

- **diagnostic schedule switch number module module\_num test test-id on mm**
- **diagnostic schedule switch number module module\_num test test-id daily hh:mm**
- **diagnostic schedule switch number module module\_num test test-id weekly day-of-week**



- (注) スタンバイ スーパーバイザ モジュールがシステムをスイッチ オーバーできない場合のシステムのダウンタイムを回避するため、スタンバイ スーパーバイザ モジュールからアクティブ スーパーバイザ モジュールへのスイッチオーバーをスイッチオーバーが発生してから 10 分後にスケジューリングすることを推奨します。

## 例

次に、特定のスイッチモジュールに対して特定の月の特定の日に診断テストを実行するようにスケジューリングする例を示します。

```
Device# configure terminal
Device(config)# diagnostic schedule switch 1 module 1 test 5 on may
```

次に、特定のスイッチモジュールに対して毎日特定の時間に診断テストを実行するようにスケジューリングする例を示します。

```
Device# configure terminal
Device(config)# diagnostic schedule switch 1 module 1 test 5 daily 12:25
```

次に、特定のスイッチモジュールに対して毎週特定の曜日に診断テストを実行するようにスケジューリングする例を示します。

```
Device# configure terminal
Device(config)# diagnostic schedule module 1 test 5 weekly friday
```

## 関連コマンド

コマンド	説明
<b>show diagnostic content</b>	すべてのテストおよびモジュールについて、テスト ID、テスト属性、サポート対象テストレベルなどのテスト情報を表示します。
<b>show diagnostic schedule</b>	現在スケジュールされている診断タスクを表示します。



## diagnostic start

指定した診断テストを実行するには、特権 EXEC モードで **diagnostic start** コマンドを使用します。

```
diagnostic start switch number module module_num test {test-id | minimal | complete | {{all | basic | non-disruptive | per-port }} {port{num | port_range | all}}}
```

構文の説明		
<b>switch</b> <i>switch_num</i>		スイッチ番号を指定します。
<b>module</b> <i>module_num</i>		モジュール番号を指定します。
<b>test</b>		実行するテストを指定します。
<i>test-id</i>		実行するテストの ID 番号を入力します。 カンマおよびハイフンで区切られた整数で <i>test-id-range</i> または <i>port_range</i> を入力します（例：1,3-6 はテスト ID 1、3、4、5、および 6）。
<b>minimal</b>		最小限のブートアップ診断テストを実行します。
<b>complete</b>		すべてのブートアップ診断テストを実行します。
<b>basic</b>		基本的なオンデマンド診断テストを実行します。
<b>per-port</b>		ポート単位のレベルテストを実行します。
<b>non-disruptive</b>		中断を伴わないヘルスマonitoringテストを実行します。
<b>all</b>		すべての診断テストを実行します。
<b>port</b> <i>num</i>		(任意) インターフェイスのポート番号を指定します。 範囲は 1 ~ 48 です。

### コマンドデフォルト

なし

### コマンドモード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 使用上のガイドライン

テスト ID のリストを表示するには、**show diagnostic content** コマンドを実行します。  
テストを停止する場合は、**diagnostic stop** コマンドを使用します。

## 例

次に、すべてのオンライン診断テストを実行する例を示します。

```
Device# diagnostic start switch 1 module 1 test all

Diagnostic[switch 1, module 1]: Running test(s) 2 may disrupt normal system operation
and requires reload
Do you want to continue? [no]: y
Device#
*Jul  5 03:04:49.081 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestGoldPktLoopback{ID=1} ...
*Jul  5 03:04:49.086 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestGoldPktLoopback{ID=1}
has completed successfully
*Jul  5 03:04:49.086 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestPhyLoopback{ID=2} ...
*Jul  5 03:04:49.092 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestPhyLoopback{ID=2} has
completed successfully
*Jul  5 03:04:49.092 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestThermal{ID=3} ...
*Jul  5 03:04:52.397 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestThermal{ID=3} has
completed successfully
*Jul  5 03:04:52.397 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestScratchRegister{ID=4} ...
*Jul  5 03:04:52.414 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestScratchRegister{ID=4}
has completed successfully
*Jul  5 03:04:52.414 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running TestPoe{ID=5}
...
*Jul  5 03:04:52.415 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestPoe{ID=5} has completed
successfully
*Jul  5 03:04:52.415 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestUnusedPortLoopback{ID=6} ...
*Jul  5 03:04:52.415 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestUnusedPortLoopback{ID=6}
has completed successfully
*Jul  5 03:04:52.415 PDT: %DIAG-6-TEST_RUNNING: switch 1, module 1: Running
TestPortTxMonitoring{ID=7} ...
*Jul  5 03:04:52.416 PDT: %DIAG-6-TEST_OK: switch 1, module 1: TestPortTxMonitoring{ID=7}
has completed successfull
```

## 関連コマンド

コマンド	説明
<b>diagnostic bootup level</b>	ブートアップ診断レベルを設定します。
<b>diagnostic event-log size</b>	診断イベントログのサイズをダイナミックに変更します。
<b>diagnostic monitor</b>	ヘルスマonitoring診断テストを設定します。
<b>diagnostic ondemand</b>	オンデマンド診断を設定します。
<b>diagnostic schedule</b>	特定のベイ、スロット、またはサブスロットの診断テストのスケジュールを設定します。
<b>diagnostic stop</b>	指定した診断テストを停止します。
<b>show diagnostic bootup</b>	設定されているブートアップ時の診断レベルを表示します。
<b>show diagnostic content module</b>	使用可能な診断テストを表示します。

コマンド	説明
<b>show diagnostic description</b>	診断テストの説明を表示します。
<b>show diagnostic events</b>	診断イベントログを表示します。
<b>show diagnostic ondemand settings</b>	オンデマンド診断の設定を表示します。
<b>show diagnostic result</b>	モジュールの診断テストの結果を表示します。
<b>show diagnostic schedule</b>	現在スケジュールされている診断タスクを表示します。
<b>show diagnostic status</b>	実行中の診断テストを表示します。

## diagnostic stop

テストを停止するには、特権 EXEC モードで **diagnostic stop** コマンドを使用します。

**diagnostic stop switch** *number* **module** *module\_num*

構文の説明	<b>switch</b> <i>switch_num</i>	スイッチ番号を指定します。
	<b>module</b> <i>module_num</i>	モジュール番号を指定します。

コマンド デフォルト なし

コマンド モード 特権 EXEC (#)

コマンド履歴

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

使用上のガイドライン テストを開始する場合は、**diagnostic start** コマンドを使用します。

例

次に、診断テストを停止する例を示します。

```
Device# diagnostic stop module 3
```

関連コマンド	コマンド	説明
	<b>diagnostic bootup level</b>	ブートアップ診断レベルを設定します。
	<b>diagnostic event-log size</b>	診断イベントログのサイズをダイナミックに変更します。
	<b>diagnostic monitor</b>	ヘルスマonitoring診断テストを設定します。
	<b>diagnostic ondemand</b>	オンデマンド診断を設定します。
	<b>diagnostic schedule</b>	特定のベイ、スロット、またはサブスロットの診断テストのスケジュールを設定します。
	<b>diagnostic start</b>	指定した診断テストを実行します。
	<b>show diagnostic bootup</b>	設定されているブートアップ時の診断レベルを表示します。
	<b>show diagnostic content module</b>	使用可能な診断テストを表示します。

コマンド	説明
<b>show diagnostic description</b>	診断テストの説明を表示します。
<b>show diagnostic events</b>	診断イベントログを表示します。
<b>show diagnostic ondemand settings</b>	オンデマンド診断の設定を表示します。
<b>show diagnostic result</b>	モジュールの診断テストの結果を表示します。
<b>show diagnostic schedule</b>	現在スケジュールされている診断タスクを表示します。
<b>show diagnostic status</b>	実行中の診断テストを表示します。

# domain id

スイッチで Cisco StackWise Virtual ドメイン ID を設定するには、StackWise Virtual コンフィギュレーションモードで **domain id** コマンドを使用します。無効にするには、このコマンドの **no** 形式を使用します。

**domain id**  
**no domain id**

構文の説明	domain	StackWise Virtual 設定を特定のドメインに関連付けます。
	<i>id</i>	ドメイン ID の値。範囲は 1～255 です。デフォルトは 1 です。

コマンド デフォルト      ドメイン ID が設定されていません。

コマンド モード      StackWise Virtual コンフィギュレーション (config-stackwise-virtual)

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

使用上のガイドライン      このコマンドはオプションです。ドメイン ID を設定する前に、**stackwise-virtual** コマンドを使用して StackWise Virtual を有効にする必要があります。

## 例

次に、Cisco StackWise Virtual を有効にして、ドメイン ID を設定する例を示します。

```
(config)# stackwise-virtual
(config-stackwise-virtual)#domain 2
```

## dual-active detection pagp

PAgP デュアルアクティブ検出を有効にするには、StackWise Virtual コンフィギュレーションモードで **dual-active detection pagp** コマンドを使用します。PAgP デュアルアクティブ検出をディセーブルにするには、このコマンドの **no** 形式を使用します。

**dual-active detection pagp**  
**no dual-active detection pagp**

構文の説明	<b>dual-active detection pagp</b>	pagp デュアルアクティブ検出を有効にします。
コマンドデフォルト	イネーブル	
コマンドモード	StackWise Virtual コンフィギュレーション (config-stackwise-virtual)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

例：

次に、channel-group で PAgP デュアルアクティブ検出の信頼モードを有効にする例を示します。

```
(config)# stackwise-virtual
(config-stackwise-virtual)#dual-active detection pagp
(config-stackwise-virtual)#dual-active detection pagp trust channel-group 1
```

## dual-active recovery-reload-disable

スイッチの自動リカバリによるリロードを無効にするには、StackWise Virtual コンフィギュレーションモードで **dual-active recovery-reload-disable** コマンドを使用します。自動リカバリによるリロードを無効にするには、このコマンドの **no** 形式を使用します。

**dual-active recovery-reload-disable**  
**no dual-active recovery-reload-disable**

構文の説明	<b>dual-active recovery-reload-disable</b>	自動リカバリによるリロードを無効にします。
コマンド デフォルト	イネーブル	
コマンド モード	StackWise Virtual コンフィギュレーション (config-stackwise-virtual)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Gibraltar 16.11.1 このコマンドが導入されました。	

例：

次に、スイッチの自動リカバリによるリロードを無効にする例を示します。

```
Device(config)# stackwise-virtual
Device(config-stackwise-virtual)#dual-active recovery-reload-disable
```



## hw-module beacon switch

Field Replaceable Unit (FRU) のブルービーコン LED を制御するには、特権 EXEC モードで **hw-module beacon switch** コマンドを使用します。

```
hw-module beacon switch {switch-number|active|standby}
{RP{active|standby}|fan-tray|power-supply power-supply slot number|slot slot number}
{off|on|status}
```

構文の説明		
	<i>switch-number</i>	アクセスするスイッチ。有効値は 1 と 2 です。
	<b>active</b>	スイッチのアクティブインスタンスを選択します。
	<b>standby</b>	スイッチのスタンバイインスタンスを選択します。
	<b>RP</b>	選択したスイッチのルートプロセッサを選択します。
	<b>fan-tray</b>	選択したスイッチのファンを選択します。
	<b>power-supply</b> <i>power-supply slot number</i>	電源のスロット番号を指定します。有効な値は 1 ～ 4 です。
	<b>slot</b> <i>slot-number</i>	スロット番号を指定します。有効な値は 1 ～ 4 です。
	<b>off</b>	選択したスイッチのルートプロセッサとスロットのビーコン LED をオフにし、ファンと電源をオフにします。
	<b>on</b>	選択したスイッチのルートプロセッサとスロットのビーコン LED をオンにし、ファンと電源をオフにします。
	<b>status</b>	選択したスイッチのルートプロセッサ、ファントレイ、電源スロット、およびスロットのビーコン LED ステータスを表示します。

コマンドデフォルト なし

コマンドモード 特権 EXEC (#)

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

## hw-module switch slot

スロットで使用可能なラインカードやスーパーバイザなどのコンポーネントを制御するには、グローバル コンフィギュレーション モードで **hw-module switch slot** コマンドを使用します。

**hw-module switch** *switch-number* **slot** *slot-number* { **logging onboard** [**counter** | **environment** | **message** | **poe** | **temperature** | **voltage**] | **shutdown** }

### 構文の説明

**switch-number** アクセスするスイッチ。有効値は 1 と 2 です。

**slot****slot-number** アクセスするスロット番号を指定します。有効な値は 1～4 です。

- 1 : ラインカードスロット 1
- 2 : スーパーバイザスロット 0
- 3 : スーパーバイザスロット 1
- 4 : ラインカードスロット 4

**logging onboard** オンボードロギングを有効にします。

**counter** (任意) オンボードカウンタロギングを設定します。

**environment** (任意) オンボード環境ロギングを設定します。

**message** (任意) オンボードメッセージロギングを設定します。

**poe** (任意) オンボード PoE ロギングを設定します。

**temperature** (任意) オンボード温度ロギングを設定します。

**voltage** (任意) オンボード電圧ロギングを設定します。

**shutdown** Field Replaceable Unit (FRU) をシャットダウンします。

コマンド デフォルト なし

コマンド モード グローバル コンフィギュレーション (config)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、スイッチ 1 スロット 1 に対してオンボードロギングを有効にする例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard
```

次に、スイッチ 1 スロット 1 に対してオンボードカウンタロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard counter
```

次に、スイッチ 1 スロット 1 に対してオンボード環境ロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard environment
```

次に、スイッチ 1 スロット 1 に対してオンボードメッセージロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard message
```

次に、スイッチ 1 スロット 1 に対してオンボード PoE ロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard poe
```

次に、スイッチ 1 スロット 1 に対してオンボード温度ロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard temperature
```

次に、スイッチ 1 スロット 1 に対してオンボード電圧ロギングを設定する例を示します。

```
Device# hw-module switch 1 slot 1 logging onboard voltage
```

次に、FRU をシャットダウンする例を示します。

```
Device# hw-module switch 1 slot 1 shutdown
```

## hw-module switch usbflash

USB SSD のマウントを解除するには、特権 EXEC モードで **hw-module switch** *switch-number* **usbflash** コマンドを使用します。

**hw-module switch** *switch-number* **usbflash unmount**

### 構文の説明

*switch number*    アクセスするスイッチ。有効値は1と2です。

**usbflash unmount**    USB SSD のマウントを解除します。

### コマンド デフォルト

なし

### コマンド モード

グローバル コンフィギュレーション (config)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、スイッチ 1 から USB SSD のマウントを解除する例を示します。

```
Device# hw-module switch 1 usbflash unmount
```

## main-cpu

冗長メイン コンフィギュレーション サブモードを開始し、スタンバイスイッチをイネーブルにするには、冗長コンフィギュレーション モードで **main-cpu** コマンドを使用します。

### main-cpu

#### 構文の説明

このコマンドには引数またはキーワードはありません。

#### コマンド デフォルト

なし

#### コマンド モード

冗長コンフィギュレーション (config-red)

#### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.2	このコマンドが Cisco Catalyst 9400 シリーズ スイッチに追加されました。

#### 使用上のガイドライン

冗長メイン コンフィギュレーション サブモードから、**standby console enable** コマンドを使用してスタンバイスイッチをイネーブルにします。

次に、冗長メイン コンフィギュレーション サブモードを開始し、スタンバイスイッチをイネーブルにする例を示します。

```
Device(config)# redundancy
Device(config-red)# main-cpu
Device(config-r-mc)# standby console enable
Device#
```

## maintenance-template

メンテナンステンプレートを作成するには、グローバル コンフィギュレーション モードで **maintenance-template** *template\_name* コマンドを使用します。テンプレートを削除するには、このコマンドの **no** 形式を使用します。

**maintenance-template** *template\_name*  
**no maintenance-template** *template\_name*

構文の説明	<b>maintenance-template</b>	特定の名前で GIR 用のテンプレートを作成します。
	<i>template_name</i>	メンテナンス テンプレートの名前。
コマンド デフォルト	ディセーブル	
コマンド モード	グローバル コンフィギュレーション (config)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.8.1a	このコマンドが導入されました。

例 :

次に、g1 という名前のメンテナンス テンプレートを設定する例を示します。

```
Device(config)# maintenance template g1
```

## mode sso

冗長モードをステートフルスイッチオーバー（SSO）に設定するには、冗長コンフィギュレーションモードで **mode sso** コマンドを使用します。

### mode sso

#### 構文の説明

このコマンドには引数またはキーワードはありません。

#### コマンドデフォルト

なし

#### コマンドモード

冗長コンフィギュレーション

#### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.2	このコマンドが導入されました。

#### 使用上のガイドライン

**mode sso** コマンドは、冗長コンフィギュレーションモードでのみ入力できます。

システムを SSO モードに設定する場合は、次の注意事項に従ってください。

- SSO モードをサポートするために、スーパーバイザモジュールでは同一の Cisco IOS イメージを使用する必要があります。Cisco IOS リリース間の相違のために、冗長機能が動作しない場合があります。
- モジュールの活性挿抜（OIR）を実行する場合、モジュールの状態が移行状態（Ready 以外の状態）である場合にだけ、ステートフルスイッチオーバーの間にスイッチはリセットし、ポートステートは再起動します。
- 転送情報ベース（FIB）テーブルはスイッチオーバー時に消去されます。ルーテッドトラフィックは、ルートテーブルが再コンバージェンスするまで中断されます。

次の例では、冗長モードを SSO に設定する方法を示します。

```
Device(config)# redundancy
Device(config-red)# mode sso
Device(config-red)#
```

## policy config-sync prc reload

Parser Return Code (PRC) の障害がコンフィギュレーションの同期中に発生した場合にスタンバイスイッチをリロードするには、冗長コンフィギュレーションモードで **policy config-sync reload** コマンドを使用します。Parser Return Code (PRC) の障害が発生した場合にスタンバイスイッチがリロードしないように指定するには、このコマンドの **no** 形式を使用します。

**policy config-sync {bulk | lbl} prc reload**  
**no policy config-sync {bulk | lbl} prc reload**

### 構文の説明

**bulk** バルク コンフィギュレーションモードを指定します。

**lbl** 1行ごと (lbl) のコンフィギュレーションモードを指定します。

### コマンド デフォルト

このコマンドは、デフォルトではイネーブルです。

### コマンド モード

冗長コンフィギュレーション (config-red)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.2	このコマンドが Cisco Catalyst 9400 シリーズ スイッチに追加されました。

次に、Parser Return Code (PRC) の障害がコンフィギュレーションの同期化中に発生した場合に、スタンバイスイッチがリロードされないように指定する例を示します。

```
Device(config-red)# no policy config-sync bulk prc reload
```



# redundancy

冗長コンフィギュレーションモードを開始するには、グローバルコンフィギュレーションモードで **redundancy** コマンドを使用します。

## redundancy

### 構文の説明

このコマンドには引数またはキーワードはありません。

### コマンドデフォルト

なし

### コマンドモード

グローバル コンフィギュレーション (config)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.1	このコマンドが Cisco Catalyst 9400 シリーズ スイッチに追加されました。

### 使用上のガイドライン

冗長コンフィギュレーションモードは、スタンバイスイッチをイネーブルにするために使用されるメイン CPU サブモードを開始するために使用されます。

メイン CPU サブモードを開始するには、冗長コンフィギュレーションモードで **main-cpu** コマンドを使用します。

スタンバイスイッチを有効にするには、メイン CPU サブモードから **standby console enable** コマンドを使用します。

冗長コンフィギュレーションモードを終了するには、**exit** コマンドを使用します。

次に、冗長コンフィギュレーションモードを開始する例を示します。

```
(config)# redundancy
(config-red)#
```

次の例では、メイン CPU サブモードを開始する方法を示します。

```
(config)# redundancy
(config-red)# main-cpu
(config-r-mc)#
```

### 関連コマンド

コマンド	説明
<b>show redundancy</b>	冗長ファシリティ情報を表示します。

# redundancy force-switchover

アクティブスイッチからスタンバイスイッチへのスイッチオーバーを強制的に実行するには、特権 EXEC モードで **redundancy force-switchover** コマンドを使用します。

## redundancy force-switchover

### 構文の説明

このコマンドには引数またはキーワードはありません。

### コマンド デフォルト

なし

### コマンド モード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.2	このコマンドが導入されました。

### 使用上のガイドライン

手動で冗長スイッチに切り替えるには、**redundancy force-switchover** コマンドを使用します。冗長スイッチはCisco IOS XE イメージを実行する新しいアクティブスイッチになり、モジュールはデフォルト設定にリセットされます。古いアクティブスイッチは新しいイメージで再起動します。

アクティブスイッチで **redundancy force-switchover** コマンドを使用すると、アクティブスイッチのスイッチポートがダウン状態になります。

部分リングスタック内のスイッチにこのコマンドを使用すると、次の警告メッセージが表示されます。

```
Device# redundancy force-switchover
```

```
Stack is in Half ring setup; Reloading a switch might cause stack split
This will reload the active unit and force switchover to standby[confirm]
```

次の例では、アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンに手動で切り替える方法を示します。

```
Device# redundancy force-switchover
Device#
```

## redundancy reload peer

スタンバイルートプロセッサ (RP) モジュールをリロードするには、特権 EXEC モードで **redundancy reload peer** コマンドを使用します。

### redundancy reload peer

#### 構文の説明

このコマンドには引数またはキーワードはありません。

#### コマンドデフォルト

デフォルトの動作や値はありません。

#### コマンドモード

特権 EXEC (#)

#### コマンド履歴

#### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.1	このコマンドが導入されました。

#### 使用上のガイドライン

アクティブおよびスタンバイ RP モジュールの 1 つまたは両方で障害、トレースバック、または機能や動作の不一致が生じた場合にスタンバイ RP モジュールをリセットするには、**redundancy reload peer** コマンドを使用します。

このコマンドは、スタンバイモジュールがリセットする際にスイッチオーバーが不要であることを前提とするアクティブ デバイスの動作には影響はありません。

#### 例

次に、スタンバイ RP モジュールを手動でリロードする例を示します。

```
Device# redundancy reload peer
Reload peer? [confirm] y
Preparing to reload peer
```



- (注) **Enter** または **y** を押すと、リロードが開始します。他の何らかのキーを押すと、リロードが中止され、制御がアクティブ RP モジュールに戻ります。

次に、スタンバイ RP モジュールがルータに装着されていない場合の出力例を示します。

```
Device# redundancy reload peer
System is running in SIMPLEX mode, reload anyway? [confirm] n
Peer reload not performed.
```

コマンド	説明
冗長性	同期パラメータを設定できるように、冗長性コンフィギュレーション モードを開始します。

コマンド	説明
<b>redundancy reload shelf</b>	両方の冗長 CPU スイッチモジュールをリロードします。
<b>show redundancy</b>	冗長ファシリティ情報を表示します。

## redundancy reload shelf

両方の冗長 CPU スイッチモジュールをリロードするには、特権 EXEC モードで **redundancy reload shelf** コマンドを使用します。

### redundancy reload shelf

#### 構文の説明

このコマンドには引数またはキーワードはありません。

#### コマンドデフォルト

デフォルトの動作や値はありません。

#### コマンドモード

特権 EXEC (#)

#### コマンド履歴

#### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.1	このコマンドが導入されました。

#### 使用上のガイドライン

このコマンドを実行すると、両方の CPU スイッチモジュールがリロードされます。

#### 例

次に、シェルフ全体をリロードする例を示します。

```
Device# redundancy reload shelf
Reload the entire shelf [confirm] y
Preparing to reload shelf
```

#### 関連コマンド

コマンド	説明
冗長性	同期パラメータを設定できるように、冗長性コンフィギュレーションモードを開始します。
<b>redundancy reload peer</b>	アクティブおよびスタンバイ RP モジュールの1つまたは両方で障害、トレースバック、または機能や動作の不一致が生じた場合にスタンバイ RP モジュールをリセットします。
<b>show redundancy</b>	冗長ファシリティ情報を表示します。

# reload

スタックメンバーをリロードし、設定変更を適用するには、特権 EXEC モードで **reload** コマンドを使用します。

**reload** [{ /noverify | /verify }] [{ at | cancel | in | pause | reason *reason* }]

構文の説明	
<b>/noverify</b>	(任意) リロードの前にファイルシグニチャを確認しないように指定します。
<b>/verify</b>	(任意) リロードの前にファイルシグニチャを確認します。
<b>at</b>	(任意) リロードを実行する時間を hh:mm 形式で指定します。
<b>cancel</b>	(任意) 保留中のリロードをキャンセルします。
<b>in</b>	(任意) リロードを実行する間隔を指定します。
<b>pause</b>	(任意) リロードを一時停止します。
<b>reason <i>reason</i></b>	(任意) システムをリロードする理由を指定します。

**コマンドデフォルト**      スタックメンバーをただちにリロードし、設定の変更を有効にします。

**コマンドモード**      特権 EXEC (#)

コマンド履歴	リリース	変更内容
	Cisco IOS XE Everest 16.6.1	このコマンドが導入されました。

**例**      次の例では、スイッチスタックをリロードする方法を示します。

```
Device# reload
System configuration has been modified. Save? [yes/no]: y
Proceed to reload the whole Stack? [confirm] y
```

## router routing protocol shutdown l2

メンテナンステンプレート内で隔離するインスタンスを作成するには、メンテナンス テンプレート コンフィギュレーション モードで **router routing\_protocol instance\_id | shutdown l2** コマンドを使用します。インスタンスを削除するには、このコマンドの **no** 形式を使用します。

```
{ router routing_protocol instance_id | shutdown l2 }
no{ router routing_protocol instance_id | shutdown l2 }
```

### 構文の説明

<b>router</b>	ルーティング プロトコルに関連付けられたインスタンスを構成します。
<i>routing_protocol</i>	テンプレート用に定義されているルーティング プロトコル。
<i>instance_id</i>	ルーティング プロトコルに関連付けられたインスタンス ID。
<b>shutdown l2</b>	レイヤ 2 インターフェイスをシャットダウンするインスタンスを構成します。

### コマンド デフォルト

ディセーブル

### コマンド モード

メンテナンス テンプレートの設定 (config-maintenance-temp)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.8.1a	このコマンドが導入されました。

例 :

次の例に、メンテナンス テンプレート temp1 でインスタンス ID が 1 である ISIS 用のインスタンスを作成する方法を示します。

```
Device(config)# maintenance template g1
Device(config-maintenance-temp1)# router isis 1
```

次の例に、メンテナンス テンプレート g1 でレイヤ 2 インターフェイスをシャットダウンするためのインスタンスを作成する方法を示します。

```
Device(config)# maintenance template g1
Device(config-maintenance-temp1)# shutdown l2
```

## set platform software fed switch

SVL ポート単位の packets キャッシュ数を設定するには、特権 EXEC モードまたはユーザ EXEC モードで **set platform software fed switch** コマンドを使用します。

**set platform software fed switch** {*switch-number* | **active** | **standby**} {**F0** | **F1 active**} **fss pak-cache** *count*

構文の説明	
<b>switch</b> { <i>switch-number</i>   <b>active</b>   <b>standby</b> }	スイッチに関する情報を指定します。次の選択肢があります。 <ul style="list-style-type: none"> <li>• <i>switch-number</i></li> <li>• <b>active</b> : アクティブなスイッチに関する情報を表示します。</li> <li>• <b>standby</b> : 存在する場合、スタンバイスイッチに関する情報を表示します。</li> </ul>
<b>F0</b>	Embedded Service Processor スロット 0 に関する情報を表示します。
<b>FP active</b>	アクティブな Embedded Service Processor に関する情報を表示します。
<b>pak-cache</b> <i>count</i>	パケットキャッシュ数を指定します。範囲は 10 ~ 600 です。デフォルトは 10 です。

コマンド デフォルト ポート単位の packets キャッシュ数のデフォルトは 10 です。

コマンド モード ユーザ EXEC (>)  
特権 EXEC (#)

コマンド履歴	リリース	変更内容
	Cisco IOS XE ジブラルタル 16.10.1	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、SVL ポート単位の packets キャッシュ数を設定する例を示します。

```
Device# set platform software fed switch active F1 active fss pak-cache 40
```



## set platform software nif-mgr switch

SVL ポート単位の packets キャッシュ数を設定するには、特権 EXEC モードまたはユーザ EXEC モードで **set platform software nif-mgr switch** コマンドを使用します。

**set platform software nif-mgr switch** {*switch-number* | **active** | **standby**} **R0 pak-cache** *count*

### 構文の説明

<b>switch</b> { <i>switch-number</i>   <b>active</b>   <b>standby</b> }	スイッチに関する情報を指定します。次の選択肢があります。 <ul style="list-style-type: none"> <li>• <i>switch-number</i></li> <li>• <b>active</b> : アクティブなスイッチに関する情報を表示します。</li> <li>• <b>standby</b> : 存在する場合、スタンバイスイッチに関する情報を表示します。</li> </ul>
<b>R0</b>	ルートプロセッサ (RP) スロット 0 に関する情報を指定します。
<b>pak-cache</b> <i>count</i>	パケットキャッシュ数を指定します。範囲は 10 ~ 600 です。デフォルトは 10 です。

### コマンドデフォルト

ポート単位の packets キャッシュ数のデフォルトは 10 です。

### コマンドモード

ユーザ EXEC (>)  
特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE ジブラルタル 16.10.1	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、SVL ポート単位の packets キャッシュ数を設定する例を示します。

```
Device# set platform software nif_mgr switch active R0 pak-cache 40
```

## show diagnostic bootup

スイッチの診断ブート情報を表示するには、特権 EXEC モードで **show diagnostic bootup** コマンドを使用します。

### show diagnostic bootup level

構文の説明	<b>level</b>	診断ブートレベル情報を表示します。
コマンドモード	特権 EXEC (#)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、**show diagnostic bootup level** コマンドの出力例を示します。

```
Device# show diagnostic bootup level
Current bootup diagnostic level: minimal
```

# show diagnostic content

スイッチの診断テストの内容を表示するには、特権 EXEC モードで **show diagnostic content** コマンドを使用します。

**show diagnostic content switch** {*switch-number* **module** {**1** | **2** | **4**} | **all** [**all**]}

## 構文の説明

<b>switch</b> <i>switch-number</i>	選択するスイッチを指定します。
<b>module</b>	スイッチのモジュールを選択します。
<b>1</b>	モジュール C9400-LC-48U の診断テストの内容を表示します。
<b>2</b>	モジュール C9400-SUP-1 の診断テストの内容を表示します。
<b>4</b>	モジュール C9400-LC-48T の診断テストの内容を表示します。
<b>switch all</b> [ <b>all</b> ]	<ul style="list-style-type: none"> <li>• <b>switch all</b> : すべてのスイッチを選択します。</li> <li>• (任意) <b>all</b> : すべてのスイッチのすべての診断テストの内容を表示します。</li> </ul>

## コマンドモード

特権 EXEC (#)

## コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

次に、**show diagnostic content switch all [all]** コマンドの出力例を示します。

```
Device# show diagnostic content switch all all

switch 1, module 1:

Diagnostics test suite attributes:
  M/C/* - Minimal bootup level test / Complete bootup level test / NA
  B/* - Basic ondemand test / NA
  P/V/* - Per port test / Per device test / NA
  D/N/* - Disruptive test / Non-disruptive test / NA
  S/* - Only applicable to standby unit / NA
  X/* - Not a health monitoring test / NA
  F/* - Fixed monitoring interval test / NA
  E/* - Always enabled monitoring test / NA
  A/I - Monitoring is active / Monitoring is inactive

Test Interval  Thre-
```

ID	Test Name	Attributes	day hh:mm:ss.ms	shold
1)	TestGoldPktLoopback	*BPN*X**I	not configured	n/a
2)	TestPhyLoopback	*BPD*X**I	not configured	n/a
3)	TestThermal	*B*N****A	000 00:01:30.00	1
4)	TestScratchRegister	*B*N****A	000 00:01:30.00	5
5)	TestPoe	*B*N*X**I	not configured	n/a
6)	TestUnusedPortLoopback	*BPN****I	not configured	1
7)	TestPortTxMonitoring	*BPN****A	000 00:01:15.00	1

switch 1, module 2:

Diagnostics test suite attributes:

M/C/\* - Minimal bootup level test / Complete bootup level test / NA  
 B/\* - Basic ondemand test / NA  
 P/V/\* - Per port test / Per device test / NA  
 D/N/\* - Disruptive test / Non-disruptive test / NA  
 S/\* - Only applicable to standby unit / NA  
 X/\* - Not a health monitoring test / NA  
 F/\* - Fixed monitoring interval test / NA  
 E/\* - Always enabled monitoring test / NA  
 A/I - Monitoring is active / Monitoring is inactive

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Thre- day shold
1)	TestGoldPktLoopback	*BPN*X**I	not configured	n/a
2)	TestFantray	*B*N****A	000 00:01:40.00	1
3)	TestPhyLoopback	*BPD*X**I	not configured	n/a
4)	TestThermal	*B*N****A	000 00:01:30.00	1
5)	TestScratchRegister	*B*N****A	000 00:01:30.00	5
6)	TestMemory	*B*D*X**I	not configured	n/a
7)	TestUnusedPortLoopback	*BPN****I	not configured	1
8)	TestPortTxMonitoring	*BPN****A	000 00:01:15.00	1

switch 1, module 4:

Diagnostics test suite attributes:

M/C/\* - Minimal bootup level test / Complete bootup level test / NA  
 B/\* - Basic ondemand test / NA  
 P/V/\* - Per port test / Per device test / NA  
 D/N/\* - Disruptive test / Non-disruptive test / NA  
 S/\* - Only applicable to standby unit / NA  
 X/\* - Not a health monitoring test / NA  
 F/\* - Fixed monitoring interval test / NA  
 E/\* - Always enabled monitoring test / NA  
 A/I - Monitoring is active / Monitoring is inactive

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Thre- day shold
1)	TestGoldPktLoopback	*BPN*X**I	not configured	n/a
2)	TestPhyLoopback	*BPD*X**I	not configured	n/a
3)	TestThermal	*B*N****A	000 00:01:30.00	1
4)	TestScratchRegister	*B*N****A	000 00:01:30.00	5
5)	TestUnusedPortLoopback	*BPN****I	not configured	1
6)	TestPortTxMonitoring	*BPN****A	000 00:01:15.00	1

switch 2, module 1:

Diagnostics test suite attributes:

M/C/\* - Minimal bootup level test / Complete bootup level test / NA  
 B/\* - Basic ondemand test / NA  
 P/V/\* - Per port test / Per device test / NA  
 D/N/\* - Disruptive test / Non-disruptive test / NA  
 S/\* - Only applicable to standby unit / NA  
 X/\* - Not a health monitoring test / NA  
 F/\* - Fixed monitoring interval test / NA  
 E/\* - Always enabled monitoring test / NA  
 A/I - Monitoring is active / Monitoring is inactive

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Three- day shold
1)	TestGoldPktLoopback	*BPN*X**I	not configured	n/a
2)	TestPhyLoopback	*BPD*X**I	not configured	n/a
3)	TestThermal	*B*N****A	000 00:01:30.00	1
4)	TestScratchRegister	*B*N****A	000 00:01:30.00	5
5)	TestPoe	*B*N*X**I	not configured	n/a
6)	TestUnusedPortLoopback	*BPN****I	not configured	1
7)	TestPortTxMonitoring	*BPN****A	000 00:01:15.00	1

switch 2, module 2:

Diagnostics test suite attributes:

M/C/\* - Minimal bootup level test / Complete bootup level test / NA  
 B/\* - Basic ondemand test / NA  
 P/V/\* - Per port test / Per device test / NA  
 D/N/\* - Disruptive test / Non-disruptive test / NA  
 S/\* - Only applicable to standby unit / NA  
 X/\* - Not a health monitoring test / NA  
 F/\* - Fixed monitoring interval test / NA  
 E/\* - Always enabled monitoring test / NA  
 A/I - Monitoring is active / Monitoring is inactive

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Three- day shold
1)	TestGoldPktLoopback	*BPN*X**I	not configured	n/a
2)	TestFantray	*B*N****A	000 00:01:40.00	1
3)	TestPhyLoopback	*BPD*X**I	not configured	n/a
4)	TestThermal	*B*N****A	000 00:01:30.00	1
5)	TestScratchRegister	*B*N****A	000 00:01:30.00	5
6)	TestMemory	*B*D*X**I	not configured	n/a
7)	TestUnusedPortLoopback	*BPN****I	not configured	1
8)	TestPortTxMonitoring	*BPN****A	000 00:01:15.00	1

switch 2, module 4:

Diagnostics test suite attributes:

M/C/\* - Minimal bootup level test / Complete bootup level test / NA  
 B/\* - Basic ondemand test / NA  
 P/V/\* - Per port test / Per device test / NA  
 D/N/\* - Disruptive test / Non-disruptive test / NA  
 S/\* - Only applicable to standby unit / NA  
 X/\* - Not a health monitoring test / NA  
 F/\* - Fixed monitoring interval test / NA  
 E/\* - Always enabled monitoring test / NA  
 A/I - Monitoring is active / Monitoring is inactive

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Three- day shold
----	-----------	------------	----------------------------------	---------------------

```
1) TestGoldPktLoopback -----> *BPN*X**I      not configured n/a
2) TestPhyLoopback -----> *BPD*X**I      not configured n/a
3) TestThermal -----> *B*N***A      000 00:01:30.00 1
4) TestScratchRegister -----> *B*N***A      000 00:01:30.00 5
5) TestUnusedPortLoopback -----> *BPN***I      not configured 1
6) TestPortTxMonitoring -----> *BPN***A      000 00:01:15.00 1
```

## show diagnostic description

スイッチの診断テストの説明を表示するには、特権 EXEC モードで **show diagnostic description** コマンドを使用します。

```
show diagnostic description switch {switch-number module {1 | 2 | 4} {test {test-id | all}}
| all test {test-list | test-id | all}}
```

構文の説明	switch <i>switch-number</i>	選択するスイッチを指定します。
	<b>switch all</b>	すべてのスイッチを選択します。
	<b>module</b>	スイッチのモジュールを選択します。
	<b>1</b>	モジュール C9400-LC-48U を選択します。
	<b>2</b>	モジュール C9400-SUP-1 を選択します。
	<b>4</b>	モジュール C9400-LC-48T を選択します。
	<b>test test-id</b>	指定したテスト ID またはテスト名について診断テストの説明を表示します。
	<b>test test-list</b>	指定した一連のテスト ID について診断テストの説明を表示します。
	<b>test all</b>	すべてのテスト ID について診断テストの説明を表示します。
コマンドモード	特権 EXEC (#)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、**show diagnostic description switch switch-number module 4 test all** コマンドの出力例を示します。

```
Device# show diagnostic description switch 1 module 4 test all

TestGoldPktLoopback :
The GOLD packet Loopback test verifies the MAC level loopback
functionality. In this test, a GOLD packet, for which doppler
provides the support in hardware, is sent. The packet loops back
at MAC level and is matched against the stored packet. It is a
non-disruptive test.

TestPhyLoopback :
The PHY Loopback test verifies the PHY level loopback
functionality. In this test, a packet is sent which loops back
```

at PHY level and is matched against the stored packet. It is a disruptive test and cannot be run as a health monitoring test.

TestThermal :

This test verifies the temperature reading from the sensor is below the yellow temperature threshold. It is a non-disruptive test and can be run as a health monitoring test.

TestScratchRegister :

The Scratch Register test monitors the health of application-specific integrated circuits (ASICs) by writing values into registers and reading back the values from these registers. It is a non-disruptive test and can be run as a health monitoring test.

TestUnusedPortLoopback :

This test verifies the PHY level loopback functionality for admin-down ports. In this test, a packet is sent which loops back at PHY level and is matched against the stored packet. It is a non-disruptive test and can be run as a health monitoring test.

TestPortTxMonitoring :

This test monitors the TX counters of a connected interface. This test verifies if the connected port is able to send the packets or not. It is a non-disruptive test and can be run as a health monitoring test.



## show diagnostic events

スイッチの診断イベントログを表示するには、特権 EXEC モードで **show diagnostic events** コマンドを使用します。

```
show diagnostic events switch {switch-number module {1 | 2 | 4} | all [event-type [error | info | warning]]}
```

構文の説明	<b>switch</b> <i>switch-number</i>	選択するスイッチを指定します。
	<b>switch all</b>	すべてのスイッチを選択します。
	<b>module</b>	スイッチのモジュールを選択します。
	<b>1</b>	C9400-LC-48U モジュールの診断イベントログを表示します。
	<b>2</b>	C9400-SUP-1 モジュールの診断イベントログを表示します。
	<b>4</b>	C9400-LC-48T モジュールの診断イベントログを表示します。
	<b>event-type</b>	(任意) 特定のイベントタイプのイベントログを表示します。有効な値は次のとおりです。 <ul style="list-style-type: none"> <li>• <b>error</b>: エラータイプのイベントログを表示します。</li> <li>• <b>info</b>: 情報タイプのイベントログを表示します。</li> <li>• <b>warning</b>: 警告タイプのイベントログを表示します。</li> </ul>
コマンドモード	特権 EXEC (#)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、**show diagnostic events switch switch-number module 2** コマンドの出力例を示します。

```
Device# show diagnostic events switch 1 module 2

Diagnostic events (storage for 500 events, 500 events recorded)
Number of events matching above criteria = 500
```

## show diagnostic events

Event Type (ET): I - Info, W - Warning, E - Error

Time Stamp	ET	[Card]	Event Message
07/08 13:54:05.110	E	[1-2]	TestThermal Failed
07/08 13:55:35.111	E	[1-2]	TestThermal Failed
07/08 13:57:05.111	E	[1-2]	TestThermal Failed
07/08 13:58:35.613	E	[1-2]	TestThermal Failed
07/08 14:00:05.614	E	[1-2]	TestThermal Failed
07/08 14:01:35.615	E	[1-2]	TestThermal Failed
07/08 14:03:05.616	E	[1-2]	TestThermal Failed
07/08 14:04:36.367	E	[1-2]	TestThermal Failed
07/08 14:06:06.368	E	[1-2]	TestThermal Failed
07/08 14:07:37.370	E	[1-2]	TestThermal Failed
07/08 14:09:07.371	E	[1-2]	TestThermal Failed
07/08 14:10:38.372	E	[1-2]	TestThermal Failed
07/08 14:12:10.873	E	[1-2]	TestThermal Failed
07/08 14:13:41.374	E	[1-2]	TestThermal Failed

<Output truncated>

## show diagnostic result

診断テストの結果の情報を表示するには、特権 EXEC モードで **show diagnostic result** コマンドを使用します。

```
show diagnostic result switch {switch-number module {1 | 2 | 4} [detail | failure [detail]
| test {test-id | all} [detail] | xml] | all [all [detail | failure [detail]]]}
```

構文の説明		
<b>switch</b> <i>switch-number</i>		選択するスイッチを指定します。
<b>module</b>		スイッチのモジュールを選択します。
<b>1</b>		モジュール C9400-LC-48U の診断テストの結果を表示します。
<b>2</b>		モジュール C9400-SUP-1 の診断テストの結果を表示します。
<b>4</b>		モジュール C9400-LC-48T の診断テストの結果を表示します。
<b>detail</b>		(任意) 詳細なテスト結果を表示します。
<b>failure</b>		(任意) 失敗したテストの結果を表示します。
<b>test</b> <i>test-id</i>		(任意) モジュールの選択したテスト ID またはテスト名か一連のテスト ID について診断テストの結果を表示します。
<b>test</b> <b>all</b>		(任意) モジュールのすべてのテストについて診断テストの結果を表示します。
<b>xml</b>		(任意) テスト結果を XML 形式で表示します。
<b>switch</b> <b>all</b> [ <b>all</b> ]		<ul style="list-style-type: none"> <li>• <b>switch all</b> : すべてのスイッチについて診断テストの結果を表示します。</li> <li>• (任意) <b>all</b> : すべてのスイッチのすべてのカードについて診断テストの結果を表示します。</li> </ul>
コマンドモード	特権 EXEC (#)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

## 例

次に、**show diagnostic result switch *switch-number* module 4 [failure [detail]]** コマンドの出力例を示します。

```
Device# show diagnostic result switch 1 module 4 failure detail

Current bootup diagnostic level: minimal

switch 1, module 4:   SerialNo : JAE204700PH

Overall Diagnostic Result for switch 1, module 4 : PASS
Diagnostic level at card bootup: minimal

Test results: (. = Pass, F = Fail, U = Untested)
```

次に、**show diagnostic result switch *switch-number* module 4 [detail]** コマンドの出力例を示します。

```
Device# show diagnostic result switch 1 module 4 detail

Current bootup diagnostic level: minimal

switch 1, module 4:   SerialNo : JAE204700PH

Overall Diagnostic Result for switch 1, module 4 : PASS
Diagnostic level at card bootup: minimal

Test results: (. = Pass, F = Fail, U = Untested)
```

---

1) TestGoldPktLoopback:

```
Port  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U

Port 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

---

2) TestPhyLoopback:

```
Port  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U

Port 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U
```

```

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time -----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

```

---

```
3) TestThermal -----> .
```

```

Error code -----> 0 (DIAG_SUCCESS)
Total run count -----> 1771
Last test testing type -----> Health Monitoring
Last test execution time -----> Jul 09 2018 03:06:53
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> Jul 09 2018 03:06:53
Total failure count -----> 0
Consecutive failure count ---> 0

```

---

```
4) TestScratchRegister -----> .
```

```

Error code -----> 0 (DIAG_SUCCESS)
Total run count -----> 1771
Last test testing type -----> Health Monitoring
Last test execution time -----> Jul 09 2018 03:06:53
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> Jul 09 2018 03:06:53
Total failure count -----> 0
Consecutive failure count ---> 0

```

---

```
5) TestUnusedPortLoopback:
```

```

Port  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U
Port 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
-----
      U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U

```

```

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time -----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

```

---

```
6) TestPortTxMonitoring:
```

```

Port  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
-----

```

```

. U U U U U U U U U U U U U U U U U U U U U U U U U U U U U
Port 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
-----
U U U U U U U U U U U U U U U U U U U U U U U U U U U U .

Error code -----> 0 (DIAG_SUCCESS)
Total run count -----> 2146
Last test testing type -----> Health Monitoring
Last test execution time ----> Jul 09 2018 03:07:08
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> Jul 09 2018 03:07:08
Total failure count -----> 0
Consecutive failure count ---> 0

```

次に、**show diagnostic result switch *switch-number* module 4 [test [*test-id*]]** コマンドの出力例を示します。

```

Device# show diagnostic result switch 1 module 4 test 3

Current bootup diagnostic level: minimal

Test results: (. = Pass, F = Fail, U = Untested)

3) TestThermal -----> .

Switch#show diagnostic result switch 1 module 4 test 3 detail ?
|   Output modifiers
<cr> <cr>

Switch#show diagnostic result switch 1 module 4 test 3 detail

Current bootup diagnostic level: minimal

Test results: (. = Pass, F = Fail, U = Untested)

-----

3) TestThermal -----> .

Error code -----> 0 (DIAG_SUCCESS)
Total run count -----> 1772
Last test testing type -----> Health Monitoring
Last test execution time ----> Jul 09 2018 03:08:23
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> Jul 09 2018 03:08:23
Total failure count -----> 0
Consecutive failure count ---> 0

```

次に、**show diagnostic result switch *switch-number* module 4 [xml]** コマンドの出力例を示します。

```

Device# show diagnostic result switch 1 module 4 xml

Current bootup diagnostic level: minimal

```

```
<?xml version="1.0" ?><diag>
<diag_results>
<diag_info>
This file report diag test results
</diag_info>
<diag_card_result>
<result overall_result="DIAG_PASS" new_failure="FALSE" diag_level="DIAG_LEVEL_MINIMAL"
/>
<card name="switch 1, module 4" index="3198" serial_no="JAE204700PH" >
<card_no>
9
</card_no>
<total_port>
48
</total_port>
<test name="TestGoldPktLoopback" >
<test_result>
<portmask>
00000000-00000000-00000000-00000000-00000000-00000000-11111111-11111111-11111111</portmask>
<per_port_result result="DIAG_RESULT_UNKNOWN" port="1" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="2" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="3" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="4" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="5" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="6" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="7" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="8" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="9" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="10" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="11" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="12" />
<per_port_result result="DIAG_RESULT_UNKNOWN" port="13" />

<Output truncated>
```

## show diagnostic simulation failure

スイッチ上のカードの診断障害シミュレーション情報を表示するには、特権 EXEC モードで **show diagnostic simulation failure** コマンドを使用します。

**show diagnostic simulation failure switch** {*switch-number* **module** {**1** | **2** | **4**} | **all** [**all**] }

### 構文の説明

<b>switch</b> <i>switch-number</i>	選択するスイッチを指定します。
<b>module</b>	スイッチのモジュールを選択します。
<b>1</b>	C9400-LC-48U モジュールの診断障害シミュレーション情報を表示します。
<b>2</b>	C9400-SUP-1 モジュールの診断障害シミュレーション情報を表示します。
<b>4</b>	C9400-LC-48T モジュールの診断障害シミュレーション情報を表示します。
<b>switch all</b> [ <b>all</b> ]	<ul style="list-style-type: none"> <li>• <b>switch all</b> : すべてのスイッチを選択します。</li> <li>• (任意) <b>all</b> : すべてのスイッチのすべての診断障害シミュレーション情報を表示します。</li> </ul>

### コマンドモード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、**show diagnostic simulation failure switch all** コマンドの出力例を示します。

```
Device# show diagnostic simulation failure switch all
There is no test failure simulation installed.
```



## show diagnostic schedule

スイッチのカードの診断スケジュール情報を表示するには、特権EXECモードで **show diagnostic schedule** コマンドを使用します。

**show diagnostic schedule switch** {*switch-number* **module** {**1** | **2** | **4**} | **all** [**all**]}

### 構文の説明

<b>switch</b> <i>switch-number</i>	選択するスイッチを指定します。
<b>module</b>	スイッチのモジュールを選択します。
<b>1</b>	C9400-LC-48U モジュールの診断スケジュール情報を表示します。
<b>2</b>	C9400-SUP-1 モジュールの診断スケジュール情報を表示します。
<b>4</b>	C9400-LC-48T モジュールの診断スケジュール情報を表示します。
<b>switch all</b> [ <b>all</b> ]	<ul style="list-style-type: none"> <li>• <b>switch all</b> : すべてのスイッチを選択します。</li> <li>• (任意) <b>all</b> : すべてのスイッチのすべての診断スケジュール情報を表示します。</li> </ul>

### コマンドモード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

### 例

次に、**show diagnostic schedule switch** *switch-number* **module 2** コマンドの出力例を示します。

```
Device# show diagnostic schedule switch 1 module 2
```

```
Current Time = 03:14:24 PDT Mon Jul 9 2018
```

```
Diagnostic for switch 1, module 2 is not scheduled.
```

## show hw-module switch subslot

システムおよびシャーシのロケーション情報でサポートされているすべてのモジュールの情報を表示するには、特権 EXEC モードで **show hw-module switch switch-number subslot** コマンドを使用します。この機能をディセーブルにするには、このコマンドの **no** 形式を使用します。

### show hw-module switch switch-number subslot

```
{slot/subslot|all{attribute|entity|oir|sensors[limits]|subblock|tech-support}}
```

### no show hw-module switch switch-number subslot

```
{slot/subslot|all{attribute|entity|oir|sensors[limits]|subblock|tech-support}}
```

#### 構文の説明

<i>switch number</i>	アクセスするスイッチを指定します。有効な値は 1 と 2 です。
<b>subslot</b> <i>slot/subslot</i>	モジュールのスロットまたはサブスロット番号を指定します。  slot の有効な値は 1 ~ 4 です。 subslot の有効な値は 0 です。
<b>all</b>	サブスロットレベルのサポートされているすべてのモジュールを選択します。
<b>attribute</b>	モジュールの属性情報を表示します。
<b>entity</b>	エンティティ MIB の詳細を表示します。  (注) 実稼働での使用を目的としたものではありません。
<b>oir</b>	活性挿抜 (OIR) のサマリーを表示します。
<b>sensors</b>	環境センサーのサマリーを表示します。
<b>limits</b>	センサーの制限を表示します。
<b>subblock</b>	サブブロックの詳細を表示します。  (注) 実稼働での使用を目的としたものではありません。
<b>tech-support</b>	テクニカルサポートに使用するサブスロット情報を表示します。

コマンド デフォルト なし

コマンドモード	特権 EXEC (#)
---------	-------------

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

## 例

次に、スイッチ 1 のサブスロットレベルのすべてのモジュールについて、モジュールの属性情報を取得する例を示します。

```
Device# show hw-module switch 1 subslot all attribute
```

次に、スイッチ 1 のサブスロットレベルのすべてのモジュールについて、モジュールの OIR 情報を取得する例を示します。

```
Device# show hw-module switch 1 subslot all oir
```

次に、スイッチ 1 のサブスロットレベルのすべてのモジュールについて、環境センサーのサマリーを取得する例を示します。

```
Device# show hw-module switch 1 subslot all sensors
```

次に、スイッチ 1 のサブスロットレベルのすべてのモジュールについて、センサーの制限の情報を取得する例を示します。

```
Device# show hw-module switch 1 subslot all sensors limit
```

次に、スイッチ 1 のサブスロットレベルのすべてのモジュールについて、テクニカルサポートに使用するサブスロット情報を取得する例を示します。

```
Device# show hw-module switch 1 subslot all tech-support
```

## show logging onboard switch

スイッチのオンボード障害ロギング（OBFL）情報を表示するには、特権EXECモードで **show logging onboard switch** コマンドを使用します。

```
show logging onboard switch {switch-number | active | standby} {RP {standby | active}
| slot {1 | 4 | F0 | F1 | R0 | R1}} {{clilog | counter | environment | message
| poe | temperature | uptimevo | voltage} [continuous | detail | summary] [start
hh:mm:ss day month year] [end hh:mm:ss day month year]} | state | status}
```

構文の説明		
	<i>switch-number</i>	OBFL 情報を表示するスイッチ。
	<b>active</b>	アクティブスイッチに関する OBFL 情報を表示します。
	<b>standby</b>	スタンバイスイッチに関する OBFL 情報を表示します。
	<b>RP</b>	ルートプロセッサ（RP）を指定します。
	<b>slot</b>	スロット情報を指定します。
	<b>clilog</b>	スタンドアロンスイッチまたは指定したスタックメンバで入力された OBFL コマンドを表示します。
	<b>counter</b>	スタンドアロンスイッチまたは指定したスタックメンバのカウンタを表示します。
	<b>environment</b>	スタンドアロンスイッチまたは指定したスタックメンバの固有デバイス識別子（UDI）情報を表示します。接続中のすべての FRU デバイスの製品 ID（PID）、バージョン ID（VID）、シリアル番号も表示します。
	<b>message</b>	スタンドアロンスイッチまたは指定したスタックメンバによって生成されたハードウェア関連のシステムメッセージを表示します。
	<b>poe</b>	スタンドアロンスイッチまたは指定したスタックメンバの Power over Ethernet（PoE）ポートの消費電力を表示します。
	<b>state</b>	スタンドアロンスイッチまたは指定したスタックメンバの状態を表示します。
	<b>status</b>	スタンドアロンスイッチまたは指定したスタックメンバのステータスを表示します。
	<b>temperature</b>	スタンドアロンスイッチまたは指定したスタックメンバの温度を表示します。

<b>uptime</b>	スタンドアロンスイッチまたは指定したスタックメンバの起動時刻、スタンドアロンスイッチまたは指定したスタックメンバの再起動の理由、およびスタンドアロンスイッチまたは指定したスタックメンバの最後の再起動からの稼働時間を表示します。
<b>voltage</b>	スタンドアロンスイッチまたは指定したスイッチスタックメンバのシステム電圧を表示します。
<b>continuous</b>	(任意) 連続ファイルのデータを表示します。
<b>detail</b>	(任意) 連続データおよびサマリーデータの両方を表示します。
<b>summary</b>	(任意) サマリーファイルのデータを表示します。
<b>start hh:mm:ss day month year</b>	(任意) 指定した日時からのデータを表示します。24時間表記の2桁の数値で時刻を入力します。13:32:45のように、必ずコロン(:)を使用してください。dayの範囲は1～31です。monthは大文字または小文字で入力します。January または august など、月の名前をすべて入力することも、jan または Aug のように月の名前の最初の3文字を入力することもできます。year は、2008 のように4桁の数字で入力します。範囲は1970～2099です。
<b>end hh:mm:ss day month year</b>	(任意) 指定した日時までのデータを表示します。24時間表記の2桁の数値で時刻を入力します。13:32:45のように、必ずコロン(:)を使用してください。dayの範囲は1～31です。monthは大文字または小文字で入力します。January または august など、月の名前をすべて入力することも、jan または Aug のように月の名前の最初の3文字を入力することもできます。year は、2008 のように4桁の数字で入力します。範囲は1970～2099です。

コマンドモード 特権 EXEC (#)

コマンド履歴

リリース	変更内容
Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

使用上のガイドライン

OBFL がイネーブルの場合、スイッチはすべてのデータが格納される連続ファイルに OBFL データを記録します。連続ファイルは循環式です。連続ファイルがいっぱいになると、スイッチはサマリーファイル(別名、履歴ファイル)にデータをまとめます。サマリーファイルを作成すると、連続ファイルのスペースが解放されるので、スイッチは新しいデータを書き込めます。

特定の時間内にだけ収集されたデータを表示するには、**start** キーワードと **end** キーワードを使用します。

## 例

次に、**show logging onboard switch 1 RP active message** コマンドの出力例を示します。

```
Device# show logging onboard switch 1 RP active message

-----
ERROR MESSAGE SUMMARY INFORMATION
-----
MM/DD/YYYY HH:MM:SS Facility-Sev-Name | Count | Persistence Flag
-----
07/06/2018 00:45:23 %IOSXE-2-DIAGNOSTICS_FAILED : >254 LAST Diagnostics Thermal failed
07/06/2018 00:19:57 %IOSXE-2-DIAGNOSTICS_PASSED : >254 LAST Diagnostics Fantray passed
07/07/2018 11:36:10 %IOSXE-2-TRANSCEIVER_INSERTED : >254 LAST Transceiver module
inserted in TenGigabitEthernet1/2/0/5
05/03/2018 05:49:57 %IOSXE-2-TRANSCEIVER_REMOVED : 82 : LAST : Transceiver module
removed from TenGigabitEthernet1/2/0/7
07/07/2018 08:20:36 %IOSXE-2-SPA_REMOVED : >254 LAST SPA removed from subslot 14/0
07/06/2018 01:50:33 %IOSXE-2-SPA_INSERTED : >254 LAST SPA inserted in subslot 11/0
-----
```

次に、**show logging onboard switch 1 slot 4 status** コマンドの出力例を示します。

```
Device# show logging onboard switch 1 slot 4 status
```

```
-----
OBFL Application Status
-----
Application Uptime:
    Path: /obfl0/
    Cli enable status: enabled
Application Message:
    Path: /obfl0/
    Cli enable status: enabled
Application Voltage:
    Path: /obfl0/
    Cli enable status: enabled
Application Temperature:
    Path: /obfl0/
    Cli enable status: enabled
Application POE:
    Path: /obfl0/
    Cli enable status: enabled
Application Environment:
    Path: /obfl0/
    Cli enable status: enabled
Application Counter:
    Path: /obfl0/
    Cli enable status: enabled
Application Clilog:
    Path: /obfl0/
    Cli enable status: enabled
```

次に、**show logging onboard switch 1 slot 4 state** コマンドの出力例を示します。

```
Device# show logging onboard switch 1 slot 4 state
```

```
GREEN
```

## 関連コマンド

コマンド	説明
<b>clear logging onboard</b>	フラッシュメモリから OBFL データを削除します。
<b>hw-module logging onboard</b>	OBFL をイネーブルにします。

## show platform software fed

FED と Network Interface Manager (NIF Mgr) のソフトウェアプロセス間におけるポート単位の SDP/LMP 制御パケット交換履歴を表示するには、特権 EXEC モードで **show platform software fed** コマンドを使用します。

**show platform software fed** {**active** | **standby**} **fss** {**counters** | **interface-counters interface** {*interface-type interface-number*} | **lmp-packets interface** {*interface-type interface-number*} | **sdp-packets**

### 構文の説明

**active** | **standby** スイッチに関する情報を表示します。次の選択肢があります。

- **active** : アクティブなスイッチに関する情報を表示します。
- **standby** : 存在する場合、スタンバイスイッチに関する情報を表示します。

(注) このキーワードはサポートされていません。

**fss** 前面スタック構成 (FSS) に関する情報を指定します。

**counters** SDP、LMP、OOB1/2、EMP、および LOOPBACK タイプの TX パケットと RX パケットの数を表示します。

**interface-counters** すべてのインターフェイスについて、TX パケットと RX パケットの数を表示します。特定の SVL インターフェイスについての情報を表示するように出力をフィルタ処理するには、**interface-counters interface** {*interface-type interface-number*} コマンドを使用します。

**lmp-packets** すべての SVL インターフェイスについて、FED と NIF Manager の間でやり取りされた LMP パケットトランザクションの詳細を表示します。特定の SVL インターフェイスについての情報を表示するように出力をフィルタ処理するには、**lmp-packets interface** {*interface-type interface-number*} コマンドを使用します。

**sdp-packets** すべての SVL インターフェイスについて、FED と NIF Manager の間で送信された SDP パケットの詳細を表示します。

コマンド デフォルト なし

コマンド モード 特権 EXEC (#)

コマンド履歴 リリース 変更内容

Cisco IOS XE ジブラルタル 16.10.1 このコマンドが導入されました。



**使用上のガイドライン** デフォルトでは、**show platform software fed active fss sdp-packets** コマンドの出力にパケットキャッシュ数が 10 と表示されます。ポート単位のパケットキャッシュ数は、**set platform software fed switch** コマンドを使用して最大 600 に設定できます。

### 例

次に、**show platform software fed active fss lmp-packets interface interface-type interface-number** コマンドの出力例を示します。

```
Device# show platform software fed active fss lmp-packets interface TwentyFiveGigE2/4/0/1
```

```
Interface: TwentyFiveGigE2/4/0/1 IFID:0x1d
```

```
FED FSS LMP packets max 10:
```

```
FED --> Nif Mgr
```

Timestamp	Local LPN	Peer LPN	Seq Num
Tue Sep 18 12:45:13 2018	11	11	4329
Tue Sep 18 12:45:14 2018	11	11	4330

次に、**show platform software fed active fss sdp-packets** コマンドの出力例を示します。

```
Device# show platform software fed active fss sdp-packets
```

```
FED FSS SDP packets max 10:
```

```
FED-> Nif Mgr
```

Timestamp	Src Mac	Dst Mac.	Seq Num
Thu Oct 4 05:54:04 2018	e4aa:5d54:8aa8	ffff:ffff:ffff	262
Thu Oct 4 05:54:08 2018	e4aa:5d54:8aa8	ffff:ffff:ffff	263
Thu Oct 4 05:54:12 2018	e4aa:5d54:8aa8	ffff:ffff:ffff	264

次に、**show platform software fed active fss counters** コマンドの出力例を示します。

```
Device# show platform software fed active fss counters
```

```
FSS Packet Counters
```

SDP		LMP	
TX	RX	TX	RX
1493	1494	4988	4988
OOB1		OOB2	
TX	RX	TX	RX
22	8	134858	133833
EMP		LOOPBACK	
TX	RX		
0	0	71	

次に、**show platform software fed active fss interface-counters interface interface-type interface-number** コマンドの出力例を示します。

## show platform software fed

```

Device# show platform software fed active fss interface-counters TwentyFiveGigE2/4/0/1

Interface TwentyFiveGigE2/4/0/1 IFID: 0x1d Counters
          LMP
          TX | RX
          -----
          6391      6389

```

## 関連コマンド

コマンド	説明
<b>set platform software fed switch</b>	SVL インターフェイスのポート単位の packets キャッシュ数を設定します。

## show platform software nif-mgr switch

Network Interface Manager (NIF Mgr) ソフトウェアプロセスと StackWise Virtual リンク (SVL) インターフェイスの間における制御パケット交換履歴を表示するには、特権 EXEC モードで **show platform software nif-mgr switch** コマンドを使用します。

```
show platform software nif-mgr switch {switch-number | active | standby}
R0counters {slotslot-number } {port port-number } packets {slotslot-number } {port port-number }
switch-info
```

### 構文の説明

<b>switch</b> {switch-number   active   standby}	スイッチに関する情報を表示します。次の選択肢があります。 <ul style="list-style-type: none"> <li>• <b>switch-number</b>。</li> <li>• <b>active</b> : アクティブなスイッチに関する情報を表示します。</li> <li>• <b>standby</b> : 存在する場合、スタンバイスイッチに関する情報を表示します。</li> </ul> <p>(注) このキーワードはサポートされていません。</p>
<b>R0</b>	ルートプロセッサ (RP) スロット 0 に関する情報を表示します。
<b>counters</b>	LMP および SDP タイプの TX パケットと RX パケットの数を表示します。
<b>slot slot-number</b>	デバイスのスーパーバイザモジュールのスロット番号を指定します。
<b>port port-number</b>	モジュールのポート番号を指定します。
<b>packets</b>	LMP および SDP タイプの TX パケットと RX パケットの詳細を表示します。
<b>switch-info</b>	NIF Manager の運用データベースに関する情報を表示します。

### コマンド デフォルト

なし

### コマンド モード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE ジブラルタル 16.10.1	このコマンドが導入されました。

### 使用上のガイドライン

**show platform software nif-mgr switch active R0 counters** コマンドの出力には、送信された LMP および SDP パケットのカウンタが表示されます。

**show platform software nif-mgr switch active R0 switch-info** コマンドの出力には、SVL リンクの詳細と各リンクにおけるプロトコルのフラップ数が表示されます。

- LMP から FED
- SDP から FED
- FED から LMP
- FED から SDP
- Stack Manager から SDP
- SDP から Stack Manager

**show platform software nif-mgr switch active R0 packets** コマンドの出力には、送信された LMP および SDP パケットのタイムスタンプの詳細が表示されます。

- FED からの最後の 10 個の LMP フレームのタイムスタンプ
- FED への最後の 10 個の LMP フレームのタイムスタンプ
- Stack Manager からの最後の 10 個の SDP フレームのタイムスタンプ
- Stack Manager への最後の 10 個の SDP フレームのタイムスタンプ

デフォルトでは、ブートアップ時の SVL ポート単位のパケットキャッシュ数は 10 です。ポート単位のパケットキャッシュ数を設定するには、**set platform software nif-mgr switch** コマンドを使用します。

## 例

次に、**show platform software nif-mgr switch active R0 switch-info** コマンドの出力例を示します。

```
Device# show platform software nif-mgr switch active R0 switch-info
NIF Manager Local Switch Info

Stackwise Virtual Enabled:      Yes
Stackwise Virtual Domain:      1
System Model:                   C9400-SUP-1
Stack MAC:                      40ce:2499:a9d0

Stackwise Virtual Link State
-----
Local  Stack  Link  Protocol  Protocol
Port  Port  State State      Flaps
-----
3      1      Up    Ready     1
4      1      Down  Disconnected 0
```

次に、**show platform software nif-mgr switch active R0 counters** コマンドの出力例を示します。

```
Device# show platform software nif-mgr switch active R0 counters
NIF Manager Counters

Packet  From    To      From      To
Type    FED     FED     Stk Mgr   Stk Mgr
```

```

-----
                Success Fail      Success Fail
-----
LMP      680      680      0      0      0      0
SDP      0      0      226      0      230      0
-----

```

次に、**show platform software nif-mgr switch active R0 counters** {slot slot-number} {port port-number} コマンドの出力例を示します。

```
Device# show platform software nif-mgr switch active R0 counters slot 3 port 3
NIF Manager LPN Counters
```

```

Packet Stack Port LPN From To
Type Link Index FED FED
-----
LMP 1 1 3 713 713
-----

```

次に、**show platform software nif-mgr switch active R0 packets** コマンドの出力例を示します。

```
Device# show platform software nif-mgr switch active R0 packets
NIF manager packets max 10:
```

```
Stack Link : 1
```

```
LMP
```

```
-----
FED->
```

```
Nif Mgr
```

```

Timestamp                                Local Peer Seq
                                           LPN LPN Num
-----
Wed Jun 20 02:20:49 2018      3    3   1050
Wed Jun 20 02:20:50 2018      3    3   1051
Wed Jun 20 02:20:41 2018      3    3   1042
Wed Jun 20 02:20:42 2018      3    3   1043
Wed Jun 20 02:20:43 2018      3    3   1044
Wed Jun 20 02:20:44 2018      3    3   1045
Wed Jun 20 02:20:45 2018      3    3   1046
Wed Jun 20 02:20:46 2018      3    3   1047
Wed Jun 20 02:20:47 2018      3    3   1048
Wed Jun 20 02:20:48 2018      3    3   1049
-----

```

```
Nif Mgr->
```

```
FED
```

```

Timestamp                                Local Peer Seq
                                           LPN LPN Num
-----
Wed Jun 20 02:20:49 2018      3    3   1050
Wed Jun 20 02:20:50 2018      3    3   1051
Wed Jun 20 02:20:41 2018      3    3   1042
Wed Jun 20 02:20:42 2018      3    3   1043
Wed Jun 20 02:20:43 2018      3    3   1044
Wed Jun 20 02:20:44 2018      3    3   1045
Wed Jun 20 02:20:45 2018      3    3   1046
Wed Jun 20 02:20:46 2018      3    3   1047
Wed Jun 20 02:20:47 2018      3    3   1048
Wed Jun 20 02:20:48 2018      3    3   1049
-----

```

```
SDP
```

```
-----
Nif Mgr->
```

```
Stack Mgr
```

```

Timestamp                                Src Mac      Dst Mac      Seq Num
-----

```

## show platform software nif-mgr switch

```

-----
Wed Jun 20 02:20:40 2018      40ce:2499:aa90 ffff:ffff:ffff 320
Wed Jun 20 02:20:44 2018      40ce:2499:aa90 ffff:ffff:ffff 321
Wed Jun 20 02:20:48 2018      40ce:2499:aa90 ffff:ffff:ffff 322
Wed Jun 20 02:20:12 2018      40ce:2499:aa90 ffff:ffff:ffff 313
Wed Jun 20 02:20:16 2018      40ce:2499:aa90 ffff:ffff:ffff 314
Wed Jun 20 02:20:20 2018      40ce:2499:aa90 ffff:ffff:ffff 315
Wed Jun 20 02:20:24 2018      40ce:2499:aa90 ffff:ffff:ffff 316
Wed Jun 20 02:20:28 2018      40ce:2499:aa90 ffff:ffff:ffff 317
Wed Jun 20 02:20:32 2018      40ce:2499:aa90 ffff:ffff:ffff 318
Wed Jun 20 02:20:36 2018      40ce:2499:aa90 ffff:ffff:ffff 319

```

Stack Mgr-&gt;

Nif Mgr

```

Timestamp                               Src Mac      Dst Mac      Seq Num
-----
Wed Jun 20 02:20:17 2018      40ce:2499:a9d0 ffff:ffff:ffff 310
Wed Jun 20 02:20:21 2018      40ce:2499:a9d0 ffff:ffff:ffff 311
Wed Jun 20 02:20:25 2018      40ce:2499:a9d0 ffff:ffff:ffff 312
Wed Jun 20 02:20:29 2018      40ce:2499:a9d0 ffff:ffff:ffff 313
Wed Jun 20 02:20:33 2018      40ce:2499:a9d0 ffff:ffff:ffff 314
Wed Jun 20 02:20:37 2018      40ce:2499:a9d0 ffff:ffff:ffff 315
Wed Jun 20 02:20:41 2018      40ce:2499:a9d0 ffff:ffff:ffff 316
Wed Jun 20 02:20:45 2018      40ce:2499:a9d0 ffff:ffff:ffff 317
Wed Jun 20 02:20:49 2018      40ce:2499:a9d0 ffff:ffff:ffff 318
Wed Jun 20 02:20:13 2018      40ce:2499:a9d0 ffff:ffff:ffff 309

```

次に、**show platform software nif-mgr switch active R0 packets {slot slot-number} {port port-number}** コマンドの出力例を示します。

```
Device# show platform software nif-mgr switch active R0 packets slot 3 port 3
NIF Manager LPN Packets max 10:
```

LMP

FED-&gt;

Nif Mgr

```

Timestamp                               Stack Link  Local LPN  Peer LPN  Seq Num
-----
Wed Jun 20 02:21:29 2018      1        3        3        1090
Wed Jun 20 02:21:20 2018      1        3        3        1081
Wed Jun 20 02:21:21 2018      1        3        3        1082
Wed Jun 20 02:21:22 2018      1        3        3        1083
Wed Jun 20 02:21:23 2018      1        3        3        1084
Wed Jun 20 02:21:24 2018      1        3        3        1085
Wed Jun 20 02:21:25 2018      1        3        3        1086
Wed Jun 20 02:21:26 2018      1        3        3        1087
Wed Jun 20 02:21:27 2018      1        3        3        1088
Wed Jun 20 02:21:28 2018      1        3        3        1089

```

Nif Mgr-&gt;

FED

```

Timestamp                               Stack Link  Local LPN  Peer LPN  Seq Num
-----
Wed Jun 20 02:21:29 2018      1        3        3        1090
Wed Jun 20 02:21:20 2018      1        3        3        1081
Wed Jun 20 02:21:21 2018      1        3        3        1082
Wed Jun 20 02:21:22 2018      1        3        3        1083
Wed Jun 20 02:21:23 2018      1        3        3        1084
Wed Jun 20 02:21:24 2018      1        3        3        1085

```

```
Wed Jun 20 02:21:25 2018      1      3      3      1086
Wed Jun 20 02:21:26 2018      1      3      3      1087
Wed Jun 20 02:21:27 2018      1      3      3      1088
Wed Jun 20 02:21:28 2018      1      3      3      1089
```

## 関連コマンド

コマンド	説明
<b>set platform software nif-mgr switch</b>	SVLインターフェイスのポート単位のパケットキャッシュ数を設定します。

## show redundancy

冗長ファシリティ情報を表示するには、特権 EXEC モードで **show redundancy** コマンドを使用します。

```
show redundancy [{clients|config-sync|counters|history [{reload|reverse}]|slaves[slave-name]
{clients|counters}|states|switchover history [domain default]]
```

### 構文の説明

<b>clients</b>	(任意) 冗長ファシリティクライアントに関する情報を表示します。
<b>config-sync</b>	(任意) コンフィギュレーション同期の失敗または無視された Mismatched Command List (MCL) を表示します。
<b>counters</b>	(任意) 冗長ファシリティカウンタに関する情報を表示します。
<b>history</b>	(任意) 冗長ファシリティの過去のステータスのログおよび関連情報を表示します。
<b>history reload</b>	(任意) 冗長ファシリティの過去のリロード情報を表示します。
<b>history reverse</b>	(任意) 冗長ファシリティの過去のステータスおよび関連情報のログを逆順で表示します。
<b>slaves</b>	(任意) 冗長ファシリティのすべてのスタンバイスイッチを表示します。
<i>slave-name</i>	(任意) 特定の情報を表示する冗長スタンバイスイッチの名前。指定スタンバイスイッチのすべてのクライアントまたはカウンタを表示するには、追加でキーワードを入力します。
<b>clients</b>	指定セカンダリスイッチのすべての冗長ファシリティクライアントを表示します。
<b>counters</b>	指定スタンバイスイッチのすべてのカウンタが表示されます。
<b>states</b>	(任意) 冗長ファシリティの状態 (ディセーブル、初期化、スタンバイ、アクティブなど) に関する情報を表示します。
<b>switchover history</b>	(任意) 冗長ファシリティのスイッチオーバー履歴に関する情報を表示します。
<b>domain default</b>	(任意) スイッチオーバー履歴を表示するドメインとしてデフォルトドメインを表示します。

コマンドデフォルト なし

コマンドモード 特権 EXEC (#)



## コマンド履歴

## リリース

## 変更内容

Cisco IOS XE Everest 16.6.1 このコマンドが Cisco Catalyst 9400 シリーズ スイッチに追加されました。

次の例では、冗長ファシリティに関する情報を表示する方法を示します。

```
Device# show redundancy
```

```
Redundant System Information :
```

```
-----
Available system uptime = 6 days, 5 hours, 28 minutes
Switchovers system experienced = 0
Standby failures = 0
Last switchover reason = none
```

```
Hardware Mode = Duplex
Configured Redundancy Mode = sso
Operating Redundancy Mode = sso
Maintenance Mode = Disabled
Communications = Up
```

```
Current Processor Information :
```

```
-----
Active Location = slot 5
Current Software state = ACTIVE
Uptime in current state = 6 days, 5 hours, 28 minutes
Image Version = Cisco IOS Software, Catalyst L3 Switch Software
(CAT9K_IOSXE), Experimental Version 16.x.x [S2C-build-v16x_throttle-4064-/
nobackup/mcpre/BLD-BLD_V16x_THROTTLE_LATEST 102]
Copyright (c) 1986-201x by Cisco Systems, Inc.
Compiled Mon 07-Oct-xx 03:57 by mcpre
BOOT = bootflash:packages.conf;
Configuration register = 0x102
```

```
Peer Processor Information :
```

```
-----
Standby Location = slot 6
Current Software state = STANDBY HOT
Uptime in current state = 6 days, 5 hours, 25 minutes
Image Version = Cisco IOS Software, Catalyst L3 Switch Software
(CAT9K_IOSXE), Experimental Version 16.x.x [S2C-build-v16x_throttle-4064-/
nobackup/mcpre/BLD-BLD_V16x_THROTTLE_LATEST_20191007_000645 102]
Copyright (c) 1986-201x by Cisco Systems, Inc.
Compiled Mon 07-Oct-xx 03:57 by mcpre
BOOT = bootflash:packages.conf;
CONFIG_FILE =
Configuration register = 0x102
```

```
Device#
```

次の例では、冗長ファシリティクライアント情報を表示する方法を示します。

```
Device# show redundancy clients
```

```
Group ID = 1
clientID = 29          clientSeq = 60      Redundancy Mode RF
clientID = 139        clientSeq = 62      IfIndex
clientID = 25         clientSeq = 71      CHKPT RF
clientID = 10001      clientSeq = 85      QEMU Platform RF
clientID = 77         clientSeq = 87      Event Manager
clientID = 1340      clientSeq = 104     RP Platform RF
```

```

clientID = 1501      clientSeq = 105      CWAN HA
clientID = 78        clientSeq = 109      TSPTUN HA
clientID = 305       clientSeq = 110      Multicast ISSU Consolidation RF
clientID = 304       clientSeq = 111      IP multicast RF Client
clientID = 22        clientSeq = 112      Network RF Client
clientID = 88        clientSeq = 113      HSRP
clientID = 114       clientSeq = 114      GLBP
clientID = 225       clientSeq = 115      VRRP
clientID = 4700      clientSeq = 118      COND_DEBUG RF
clientID = 1341      clientSeq = 119      IOSXE DPIDX
clientID = 1505      clientSeq = 120      IOSXE SPA TSM
clientID = 75        clientSeq = 130      Tableid HA
clientID = 501       clientSeq = 137      LAN-Switch VTP VLAN

```

<output truncated>

出力には、次の情報が表示されます。

- clientID には、クライアントの ID 番号が表示されます。
- clientSeq には、クライアントの通知シーケンス番号が表示されます。
- 現在の冗長ファシリティの状態。

次の例では、冗長ファシリティカウンタ情報を表示する方法を示します。

Device# **show redundancy counters**

```

Redundancy Facility OMs
  comm link up = 0
  comm link down = 0

  invalid client tx = 0
  null tx by client = 0
  tx failures = 0
  tx msg length invalid = 0

  client not rxing msgs = 0
  rx peer msg routing errors = 0
  null peer msg rx = 0
  errored peer msg rx = 0

  buffers tx = 135884
  tx buffers unavailable = 0
  buffers rx = 135109
  buffer release errors = 0

  duplicate client registers = 0
  failed to register client = 0
  Invalid client syncs = 0

```

Device#

次の例では、冗長ファシリティ履歴情報を表示する方法を示します。

Device# **show redundancy history**

```

00:00:04 client added: Redundancy Mode RF(29) seq=60
00:00:04 client added: IfIndex(139) seq=62
00:00:04 client added: CHKPT RF(25) seq=71
00:00:04 client added: QEMU Platform RF(10001) seq=85
00:00:04 client added: Event Manager(77) seq=87
00:00:04 client added: RP Platform RF(1340) seq=104

```

```

00:00:04 client added: CWAN HA(1501) seq=105
00:00:04 client added: Network RF Client(22) seq=112
00:00:04 client added: IOSXE SPA TSM(1505) seq=120
00:00:04 client added: LAN-Switch VTP VLAN(501) seq=137
00:00:04 client added: XDR RRP RF Client(71) seq=139
00:00:04 client added: CEF RRP RF Client(24) seq=140
00:00:04 client added: MFIB RRP RF Client(306) seq=150
00:00:04 client added: RFS RF(520) seq=163
00:00:04 client added: klib(33014) seq=167
00:00:04 client added: Config Sync RF client(5) seq=168
00:00:04 client added: NGWC FEC Rf client(10007) seq=173
00:00:04 client added: LAN-Switch Port Manager(502) seq=190
00:00:04 client added: Access Tunnel(530) seq=192
00:00:04 client added: Mac address Table Manager(519) seq=193
00:00:04 client added: DHCP(100) seq=238
00:00:04 client added: DHCPD(101) seq=239
00:00:04 client added: SNMP RF Client(34) seq=251
00:00:04 client added: CWAN APS HA RF Client(1502) seq=252
00:00:04 client added: History RF Client(35) seq=261

```

<output truncated>

次の例では、冗長ファシリティスタンバイスイッチに関する情報を表示する方法を示します。

Device# **show redundancy slaves**

```

Group ID = 1
Slave/Process ID = 6107 Slave Name = [installer]
Slave/Process ID = 6109 Slave Name = [eicored]
Slave/Process ID = 6128 Slave Name = [snmp_subagent]
Slave/Process ID = 8897 Slave Name = [wcm]
Slave/Process ID = 8898 Slave Name = [table_mgr]
Slave/Process ID = 8901 Slave Name = [iosd]

```

Device#

次の例では、冗長ファシリティの状態に関する情報を表示する方法を示します。

Device# **show redundancy states**

```

my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 5

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
Redundancy State = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up

client count = 115
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0

```

Device#

## show redundancy config-sync

コンフィギュレーション同期障害情報または無視された Mismatched Command List (MCL) (存在する場合) を表示するには、EXEC モードで **show redundancy config-sync** コマンドを使用します。

**show redundancy config-sync {failures {bem|mcl|prc}|ignored failures mcl}**

構文の説明	failures	MCL エントリまたはベスト エフォート方式 (BEM) /パーサー リターンコード (PRC) の障害を表示します。
	<b>bem</b>	BEM 障害コマンドリストを表示し、スタンバイスイッチを強制的にリブートします。
	<b>mcl</b>	スイッチの実行コンフィギュレーションに存在するがスタンバイスイッチのイメージでサポートされていないコマンドを表示し、スタンバイスイッチを強制的にリブートします。
	<b>prc</b>	PRC 障害コマンドリストを表示し、スタンバイスイッチを強制的にリブートします。
	<b>ignored failures mcl</b>	無視された MCL 障害を表示します。

コマンド デフォルト なし

コマンド モード ユーザ EXEC  
特権 EXEC

コマンド履歴	リリース	変更内容
	Cisco IOS XE Everest 16.6.2	このコマンドが導入されました。

**使用上のガイドライン** 2つのバージョンの Cisco IOS イメージが含まれている場合は、それぞれのイメージによってサポートされるコマンドセットが異なる可能性があります。このような不一致コマンドのいずれかがアクティブスイッチで実行された場合、スタンバイスイッチでそのコマンドを認識できない可能性があり、これにより設定の不一致状態が発生します。バルク同期中にスタンバイスイッチでコマンドの構文チェックが失敗すると、コマンドはMCLに移動し、スタンバイスイッチはリセットされます。すべての不一致コマンドを表示するには、**show redundancy config-sync failures mcl** コマンドを使用します。

MCL を消去するには、次の手順を実行します。

1. アクティブスイッチの実行コンフィギュレーションから、不一致コマンドをすべて削除します。

2. **redundancy config-sync validate mismatched-commands** コマンドを使用して、修正した実行コンフィギュレーションに基づいて MCL を再確認します。
3. スタンバイスイッチをリロードします。

または、次の手順を実行して MCL を無視することもできます。

1. **redundancy config-sync ignore mismatched-commands** コマンドを入力します。
2. スタンバイスイッチをリロードします。システムは SSO モードに遷移します。



(注) 不一致コマンドを無視する場合、アクティブスイッチとスタンバイスイッチの同期してのコンフィギュレーションは存在したままです。

3. 無視された MCL は、**show redundancy config-sync ignored mcl** コマンドを使用して確認できます。

各コマンドでは、そのコマンドを実装するアクション機能において戻りコードが設定されます。この戻りコードは、コマンドが正常に実行されたかどうかを示します。アクティブスイッチは、コマンドの実行後に PRC を維持します。スタンバイスイッチはコマンドを実行し、アクティブスイッチに PRC を返します。これら2つの PRC が一致しないと、PRC 障害が発生します。バルク同期または1行ごとの (LBL) 同期中にスタンバイスイッチで PRC エラーが生じた場合、スタンバイスイッチはリセットされます。すべての PRC 障害を表示するには、**show redundancy config-sync failures prc** コマンドを使用します。

ベストエフォート方式 (BEM) エラーを表示するには、**show redundancy config-sync failures bem** コマンドを使用します。

次に、BEM 障害を表示する例を示します。

```
Device> show redundancy config-sync failures bem
BEM Failed Command List
-----

The list is Empty
```

次に、MCL 障害を表示する例を示します。

```
Device> show redundancy config-sync failures mcl
Mismatched Command List
-----

The list is Empty
```

次に、PRC 障害を表示する例を示します。

```
Device# show redundancy config-sync failures prc
PRC Failed Command List
-----

The list is Empty
```

## show stackwise-virtual

Cisco StackWise Virtual の設定情報を表示するには、**show stackwise-virtual** コマンドを使用します。

```
show stackwise-virtual { [switch [switch number <1-2>] {link | bandwidth | neighbors |
dual-active-detection} }
```

構文の説明	switch number	(任意) スタック内の特定のスイッチの情報を表示します。
	link	Stackwise Virtual リンク情報を表示します。
	bandwidth	Stackwise Virtual の帯域幅の可用性を表示します。
	neighbors	Stackwise Virtual のネイバーを表示します。
	dual-active-detection	Stackwise Virtual のデュアルアクティブ検出情報を表示します。

コマンド デフォルト なし

コマンド モード 特権 EXEC (#)

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

例 :

次に、**show stackwise-virtual** コマンドの出力例を示します。

```
# show stackwise-virtual

Stackwise Virtual: <Enabled/Disabled>
Domain Number:    <Domain Number>
Switch    Stackwise Virtual Link    Ports
-----
1          1          Tengigabitethernet1/0/4
           2          Tengigabitethernet1/0/5
2          1          Tengigabitethernet2/0/4
           2          Tengigabitethernet2/0/5
```

次に、**show stackwise-virtual link** コマンドの出力例を示します。

```
# show stackwise-virtual link

Stackwise Virtual Link (SVL) Information:
-----
Flags:
-----
Link Status
-----
U-Up D-Down
Protocol Status
-----
S-Suspended P-Pending E-Error T-Timeout R-Ready
-----
Switch   SVL     Ports                               Link-Status   Protocol-Status
-----   -
1         1       FortyGigabitEthernet1/1/1         U              R
2         1       FortyGigabitEthernet2/1/1         U              R
```

次に、**show stackwise-virtual bandwidth** コマンドの出力例を示します。

```
# show stackwise-virtual bandwidth

Switch   Bandwidth

1         160
2         160
```

次に、**show stackwise-virtual neighbors** コマンドの出力例を示します。

```
#show stackwise-virtual neighbors

Switch Number           Local Interface           Remote Interface

1                       Tengigabitethernet1/0/1  Tengigabitethernet2/0/1
                        Tengigabitethernet1/0/2  Tengigabitethernet2/0/2
2                       Tengigabitethernet2/0/1  Tengigabitethernet1/0/1
                        Tengigabitethernet2/0/2  Tengigabitethernet2/0/2
```

次に、**show stackwise-virtual dual-active-detection** コマンドの出力例を示します。

```
#show stackwise-virtual dual-active-detection

Stackwise Virtual Dual-Active-Detection (DAD) Configuration:
Switch Number           Dual-Active-Detection Interface

1                       Tengigabitethernet1/0/10
                        Tengigabitethernet1/0/11
2                       Tengigabitethernet2/0/12
                        Tengigabitethernet2/0/13

Stackwise Virtual Dual-Active-Detection (DAD) Configuration After Reboot:
Switch Number           Dual-Active-Detection Interface

1                       Tengigabitethernet1/0/10
                        Tengigabitethernet1/0/11
2                       Tengigabitethernet2/0/12
                        Tengigabitethernet2/0/13
```

# show tech-support stack

テクニカルサポートに使用するスイッチスタック関連のすべての情報を表示するには、特権 EXEC モードで **show tech-support stack** コマンドを使用します。

## show tech-support stack

### コマンドモード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
Cisco IOS XE Gibraltar 16.10.1	このコマンドが導入されました。
Cisco IOS XE Gibraltar 16.12.1	このコマンドの出力が拡張され、より多くのスタック関連情報が含まれるようになりました。

### 使用上のガイドライン

**show tech-support stack** コマンドは、スタック構成の状態のスナップショットをキャプチャし、問題のデバッグに役立つ情報を提供します。このコマンドは、スタック構成に関する問題（スタックケーブルの問題、サイレントリロード、スイッチが待受開始状態にならない、スタックのクラッシュなど）が発生した場合に使用します。

**show tech-support stack** コマンドの出力は非常に長くなります。この出力を効率よく処理するには、ローカルの書き込み可能なストレージ、またはリモートファイルシステムで、この出力をファイルにリダイレクトします（たとえば、**show tech-support stack | redirect flash:filename**）。

**show tech stack** コマンドの出力には次のコマンドの出力が表示されます。

次のコマンドは、待受開始状態のスタック構成のスイッチでのみ使用できます。

- **show platform software stack-mgr switch**
- **show platform software sif switch**
- **show platform hardware fed switch**
- **dir crashinfo:**
- **dir flash:/core**

次のコマンドは、待受開始状態のスタック非対応のスイッチでのみ使用できます。

- **show redundancy switchover history**
- **show platform software fed switch active**
- **show platform software fed switch standby**
- **show stackwise-virtual bandwidth**
- **show stackwise-virtual dual-active-detection**
- **show stackwise-virtual link**



- **show stackwise-virtual neighbors**
- **dir crashinfo:**
- **dir flash:/core**

## stackwise-virtual

スイッチの Cisco StackWise Virtual を有効にするには、グローバル コンフィギュレーション モードで **stackwise-virtual** コマンドを使用します。Cisco StackWise Virtual を無効にするには、このコマンドの **no** 形式を使用します。

**stackwise-virtual**  
**no stackwise-virtual**

構文の説明	<b>stackwise-virtual</b>	Cisco StackWise Virtual を有効にします。
コマンド デフォルト	ディセーブル	
コマンド モード	グローバル コンフィギュレーション (config)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

**使用上のガイドライン** Cisco StackWise Virtual を無効にしたら、スイッチをリロードしてスタック解除する必要があります。

### 例

次に、Cisco StackWise Virtual を有効にする例を示します。

```
(config)# stackwise-virtual
```

## stackwise-virtual dual-active-detection

インターフェイスをデュアルアクティブ検出リンクとして設定するには、インターフェイスコンフィギュレーションモードで **stackwise-virtual dual-active-detection** コマンドを使用します。インターフェイスの関連付けを解除するには、このコマンドの **no** 形式を使用します。

**stackwise-virtual dual-active-detection**  
**no stackwise-virtual dual-active-detection**

構文の説明	<b>stackwise-virtual dual-active-detection</b>	指定されたインターフェイスの Cisco StackWise Virtual デュアルアクティブ検出を有効にします。
コマンド デフォルト	ディセーブル	
コマンド モード	インターフェイス コンフィギュレーション (config-if)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

例：

次に、10 ギガビットイーサネット インターフェイスをデュアル アクティブ検出リンクとして設定する例を示します。

```
Device(config)# interface TenGigabitEthernet1/0/2
(config-if)#stackwise-virtual dual-active-detection
```

## stackwise-virtual link

インターフェイスを設定済みの StackWise Virtual リンクと関連付けるには、インターフェイス コンフィギュレーションモードで **stackwise-virtual link** コマンドを使用します。インターフェイスの関連付けを解除するには、このコマンドの **no** 形式を使用します。

**stackwise-virtual link** *link-value*  
**no stackwise-virtual link** *link-value*

構文の説明	<b>stackwise-virtual link</b>	StackWise Virtual リンクにインターフェイスを関連付けます。
	<i>link value</i>	Cisco StackWise Virtual に対して設定されているドメイン ID。
コマンド デフォルト	ディセーブル	
コマンド モード	インターフェイス コンフィギュレーション (config-if)。	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.9.1	このコマンドが導入されました。

例：

次に、40 ギガビット イーサネット インターフェイスと設定済みの Stackwise Virtual Link (SVL) を関連付ける例を示します。

```
(config)# interface FortyGigabitEthernet1/1/1
(config-if)#stackwise-virtual link 1
```

## standby console enable

スタンバイ コンソール スイッチへのアクセスをイネーブルにするには、冗長メイン コンフィギュレーション サブモードで **standby console enable** コマンドを使用します。スタンバイ コンソール スイッチへのアクセスをディセーブルにするには、このコマンドの **no** 形式を使用しません。

**standby console enable**  
**no standby console enable**

### 構文の説明

このコマンドには引数またはキーワードはありません。

### コマンド デフォルト

スタンバイ コンソール スイッチへのアクセスはディセーブルです。

### コマンド モード

冗長メイン コンフィギュレーション サブモード

### コマンド履歴

リリース	変更内容
Cisco IOS XE Everest 16.6.1	このコマンドが導入されました。

### 使用上のガイドライン

このコマンドは、スタンバイ コンソールに関する特定のデータを収集し、確認するために使用されます。コマンドは、主にシスコのテクニカルサポート担当がスイッチのトラブルシューティングを行うのに役立ちます。

次に、冗長メインコンフィギュレーションサブモードを開始し、スタンバイ コンソール スイッチへのアクセスをイネーブルにする例を示します。

```
Device(config)# redundancy  
Device(config-red)# main-cpu  
Device(config-r-mc)# standby console enable  
Device(config-r-mc)#
```

## start maintenance

システムをメンテナンスモードにするには、特権 EXEC モードで **start maintenance** コマンドを使用します。

### start maintenance

構文の説明	<b>start maintenance</b>	システムをメンテナンスモードにします。
コマンド デフォルト	ディセーブル	
コマンド モード	特権 EXEC	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.8.1a	このコマンドが導入されました。

例：

次に、メンテナンスモードを開始する例を示します。

```
Device# start maintenance
```

## stop maintenance

システムをメンテナンスモードから解除するには、特権 EXEC モードで **stop maintenance** コマンドを使用します。

### stop maintenance

---

コマンド デフォルト      ディセーブル

---

コマンド モード          特権 EXEC

---

コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.8.1a	このコマンドが導入されました。

---

例 :

次に、メンテナンス モードを停止する例を示します。

```
Device# stop maintenance
```

## system mode maintenance

システムモードメンテナンスコンフィギュレーションモードを開始するには、グローバルコンフィギュレーションモードで **system mode maintenance** コマンドを使用します。

### system mode maintenance

構文の説明	<b>system mode maintenance</b>	メンテナンス コンフィギュレーションモードを開始します。
コマンド デフォルト	ディセーブル	
コマンド モード	グローバル コンフィギュレーション (config)	
コマンド履歴	リリース	変更内容
	Cisco IOS XE Fuji 16.8.1a	このコマンドが導入されました。

例：

次に、メンテナンス コンフィギュレーション モードを開始する例を示します。

```
Device(config)# system mode maintenance
Device(config-maintenance)#
```