



## 診断と有用性

---

この章は次のトピックで構成されています。

- [診断と有用性について \(1 ページ\)](#)
- [コマンドの表示 \(1 ページ\)](#)
- [デバッグ ログ \(3 ページ\)](#)
- [診断の提案 \(6 ページ\)](#)
- [マニュアルの変更履歴 \(8 ページ\)](#)

### 診断と有用性について

Cisco NX-OSは、Netconf、Restconf、gNMI/gNOI、テレメトリなどの幅広いプロトコルインターフェイスを介して、モデル駆動型プログラマビリティ (MDP) をサポートします。実際、これらのインターフェイスは、共通の基盤となる YANG および DME/ CLI インフラストラクチャを中心に動作します。ユーザーは、一般的に使用される一群のユーティリティにより、動作を診断できます。

### コマンドの表示

このセクションでは、一般的に使用される `show` コマンドについて説明します。スイッチの実行状態を確認するために使用できます。

## コマンドの表示

表 1:表示コマンド: 診断と有用性

項目	コマンド	使用法
netconf	<b>show netconf nxsd़k event-history { msgs   cli   internal libnxsd़k   all }</b>	詳細イベント履歴の表示
	<b>show netconf nxsd़k event-history component { agent   mtx }</b>	イベント履歴の表示
	<b>show netconf internal mtx</b>	MTX インフラ情報を表示します。
	<b>show running-config netconf</b>	netconf 構成の表示
	<b>show netconf nxsd़k event-history {events   errors}</b>	イベント履歴の表示
restconf	<b>show running-config   grep restconf</b>	restconf 構成の表示
	<b>show restconf nxsd़k event-history {events   errors}</b>	イベント履歴の表示
	<b>show restconf nxsd़k event-history { msgs   cli   internal libnxsd़k   all }</b>	詳細イベント履歴の表示
	<b>show restconf nxsd़k event-history component { agent   mtx }</b>	イベント履歴の表示
	<b>show restconf internal mtx</b>	MTX インフラ情報を表示します。
	<b>show tech-support restconf</b>	restconf テクニカルサポートの収集

項目	コマンド	使用法
grpc	<b>show running-config grpc</b>	grpc 構成の表示
	<b>show grpc nx sdk event-history {events   errors}</b>	イベント履歴の表示
	<b>show tech-support grpc</b>	grpc テクニカルサポートの収集
	<b>show grpc nx sdk event-history {msgs   cli   internal libnx sdk   all}</b>	詳細イベント履歴の表示
	<b>show grpc nx sdk event-history component {agent   mtx   telemetry}</b>	イベント履歴の表示
gnmi	<b>show grpc gnmi service statistics</b>	grpc サーバの状態の確認
	<b>show grpc gnmi rpc [all] {summary   detail}</b>	現在/過去の gNMI サブスクリプションのリスト表示
	<b>show grpc gnmi transactions</b>	List gNMI Get/Set
gnoi	<b>show grpc gnoi service statistics</b>	grpc サーバの状態の確認
openconfig	<b>show running-config openconfig</b>	openconfig 構成の表示
	<b>show openconfig nx sdk event-history {events   errors}</b>	イベント履歴の表示
dme	<b>show system internal dme transaction history</b>	DME トランザクションの確認
	<b>show tech-support dme</b>	DME テクニカルサポートの収集

## デバッグ ログ

このセクションでは、デバッグログを有効にして収集する方法について説明します。

## プログラマビリティ エージェント ログ

Netconf、Restconf、および gRPC エージェントの場合、次の方法でログを収集できます。

- **show** コマンド

これは、エージェントイベントを表示/確認する簡単な方法です。これらのコマンドは、エージェントがクライアント接続とどのように対話するかを確認するのに役立ちます。

```
show netconf nxsdk event-history { events | errors }
show netconf nxsdk event-history component { agent events | mtx events }
show restconf nxsdk event-history { events | errors }
show restconf nxsdk event-history component { agent events | mtx events }
show grpc nxsdk event-history { events | errors }
show grpc nxsdk event-history component { agent events | mtx events | telemetry
events }
```

#### • ログファイル

より長い履歴、またはエージェントを無効にした後のログを確認する場合は、**/volatile** ディレクトリに保存されているログファイルを確認します。ユーザーには、スイッチの bash シェルにアクセスする権限が必要です。

```
/volatile/netconf-internal-log
      grpc-internal-log
      restconf-internal-log
```

## YANG インフラログ

YANG インフラのデバッグ/トレース レベルのログとファイルのロギングは、デフォルトで無効になっています。ユーザーはファイル ロギングを有効にできます。ファイル ロギングが有効になった後、YANG インフラ ログは、**/volatile** ディレクトリに保存されます。

ユーザーには、スイッチの bash シェルにアクセスする権限が必要です。

```
debug netconf mtx file
debug restconf mtx file
debug grpc mtx file

/volatile/mtx-internal.netconf.log
      mtx-internal.grpc.log
      mtx-internal.restconf.log
```

## DME ログ

DME インフラログは、**/nxos/dme\_logs** ディレクトリに保存されます。ユーザーには、スイッチの bash シェルにアクセスする権限が必要です。 「<https://developer.cisco.com/site/cisco-nexus-nx-api-references/>」 を参照してください。

```
/nxos/dme_logs/svc_ifc_policyelem.<pid>.log
```

## XOS ログ

tmtrace.bin コマンドによって公開される XOS ログ。

ユーザーには、スイッチの bash シェルにアクセスする権限が必要です。

```
tmtrace.bin -d cmi-errors
tmtrace.bin -d cmi-events
tmtrace.bin -d cmi-debug
tmtrace.bin -d cmi-msg
```

## CLI を使用したログ構成の・更

CLI を使用して、プロセスを再起動せずに上記のロギング構成をダイナミックに変更することができます。これらのコマンドは、エージェント EXEC コマンドごとであり、構成ファイルへの変更ではないため、現在の操作に影響を与えることなく変更できます。

### 手順の概要

1. [no] debug grpc mtx enable-all
2. [no] debug grpc mtx level <level>
3. [no] debug grpc mtx item <item>

### 手順の詳細

#### 手順

	コマンドまたはアクション	目的
ステップ1	[no] debug grpc mtx enable-all	これは、すべてのログを有効にする便利なcliです。
ステップ2	[no] debug grpc mtx level <level>  例： switch# debug grpc mtx level info	ロギングレベルを切り替えます：エラー（error）、警告（warning）、情報（info）、デバッグ（debug）から選択できます。 デフォルトのレベルは <b>info</b> です。
ステップ3	[no] debug grpc mtx item <item>  例： switch# debug grpc mtx item MTX-EvtMgr	特定の項目のロギングを切り替えます。これは自由形式の文字列で、 <b>show grpc internal mtx debug</b> を使用して使用可能な項目を表示します。

#### 例

次の show cli は、現在のロギング構成を表示します。

```
show grpc internal mtx debug

Example:
switch# show grpc internal mtx debug
Log enabled : 1
File enabled : 0
All active : 0
Log Level : Info
Log items :
* : 0
DB : 0
DtxUserFunc : 0
INST-ADAPTER : 0
INST-ADAPTER-GNMI : 0
INST-ADAPTER-GNOI : 0
INST-ADAPTER-GRPC : 1
INST-ADAPTER-NC : 1
```

INST-ADAPTER-RC	:	1
INST-ADAPTER-TM	:	0
INST-MTX-API	:	1
LIBUTILS	:	1
MTX-API	:	1
MTX-ActionMgr	:	0
MTX-Coder	:	0
MTX-Dy-EvtMgr	:	0
MTX-EvtMgr	:	0
MTX-RbacMgr	:	0
MTXEXPR	:	0
MTXItem	:	0
MTXNetConfMessage	:	0
MTXOperation	:	0
MTXRestConfMessage	:	0
MTXgNMIMessage	:	0
Model-*	:	1
Model-Cisco-NX-OS-device	:	1
Model-openconfig-bgp	:	0
PVSH	:	0
RPC	:	0
SYSTEM	:	1
TM-ADPT	:	0
TM-ADPT-JSON	:	0

## 診断の提案

このセクションでは、発生しやすい問題をトリアージするためのいくつかの手順を示します。

### 接続の問題

ユーザーのプログラミングクライアントがスイッチに接続できない場合は、次の点を確認します。

- ・実行構成をチェックして、機能が有効になっているかどうかを確認します。
- ・個々のエージェントの `show` コマンドをチェックして、サーバが実行されていることを確認します。
- ・IP/ポートをチェックして、接続がファイアウォールなどによって制限されていないことを確認します。
- ・クライアントが正しいユーザー名/パスワードを送信することを確認します。
- ・証明書ベースの認証を使用する場合は、トラストポイントがスイッチに正しく構成されていること、およびクライアント証明書が一致し、期限切れになっていないことを確認します。

### ネイティブデバイス YANG

読み取り/書き込み操作に関連するネイティブ openconfig YANG に問題がある場合は、次の点を確認してください。

- ・「書き込み」操作の場合は、DME トランザクションを確認して障害の詳細を確認します。
- ・同等の DME REST 要求を送信し、同じ問題があるかどうかを確認します。

### OpenConfig Yang

ネイティブ openconfig YANG の読み取り/書き込みに問題がある場合は、次のことを確認します。

- ・**feature openconfig** が有効になっているかどうかを確認します。
- ・公開されている YANG との相違点をチェックして、サポートステータスを確認します。
- ・**write** 操作の場合は、DME トランザクションを確認して障害の詳細を確認します。

### テレメトリ

テレメトリは、「feature telemetry」構成を介して YANG およびその他のデータソースを収集するため使用されます。テレメトリは、「feature grpc」を介した gNMI 登録にも使用されます。トラブルシューティングの手順は、使用シナリオによって異なります。

#### デバッグ ログ

デバッグログは、次の方法で表示できます。

- **show telemetry internal event history { errors | events }**
- **show grpc nxsdk event-history { events | errors }**

#### データ/イベント収集の問題 :

show コマンドで、失敗した、またはスキップされた収集を確認します。

- **show telemetry data collector detail**
- **show telemetry event collector {errors | stats}**
- **show grpc internal gnmi subscription statistics**

#### 収集時間またはサイズの問題 :

次の show コマンドを使用して、収集のサイズと時間を確認します。

- **show telemetry control database**
- **show grpc internal gnmi rpc subscription-data**

#### トランスポートの問題 :

次の show コマンドを使用して、トランスポートの問題を確認します。トランスポートの問題は、**feature telemetry** シナリオにのみ影響することに注意してください。

- **show telemetry transport <num> stats | errors**

# マニュアルの変更履歴

表 2: 改訂履歴: 診断と有用性

リリース	説明
10.4 (2)	ロギング パラメータを変更するための新しい CLI コマンド
10.5 (3)	Logging 2.0 のサポート。新しいコンポーネント event-history コマンドを追加します。feature dynamic event-history コマンドのサポート

## 翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。