



Cisco IOS XR インターフェイスおよびハードウェア コンポーネント コンフィギュレーション ガイド

Cisco IOS XR Interface and Hardware Component Configuration Guide

Cisco IOS XR ソフトウェア リリース 3.8

【注意】シスコ製品をご使用になる前に、安全上の注意
(www.cisco.com/jp/go/safety_warning/)をご確認ください。

本書は、米国シスコシステムズ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。

また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えられていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコシステムズおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコシステムズおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコシステムズまたはその供給者に知らされていても、それらに対する責任は一切負わないものとします。

CCDE, CCENT, Cisco Eos, Cisco HealthPresence, the Cisco logo, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, Cisco TelePresence, Cisco WebEx, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn and Cisco Store are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0812R)

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco IOS XR インターフェイスおよびハードウェア コンポーネント コンフィギュレーション ガイド
© 2009 Cisco Systems, Inc.
All rights reserved.

Copyright © 2009–2010, シスコシステムズ合同会社.
All rights reserved.



CONTENTS

はじめに	HC-xvii
マニュアルの変更履歴	HC-xvii
マニュアルの入手方法およびテクニカル サポート	HC-xvii
Cisco IOS XR ソフトウェアでの ATM インターフェイスの設定	HC-1
この章の構成	HC-2
ATM の実装の前提条件	HC-2
ATM に関する情報	HC-3
仮想回線 (VC) クラスのマッピング	HC-4
VP トンネル	HC-4
ATM インターフェイスでの F5 OAM	HC-5
ATM インターフェイスでの ILMI	HC-5
ATM インターフェイスでのレイヤ 2 VPN	HC-6
Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポートアダプタでの Circuit-Emulation-over-Packet (CEoP)	HC-7
ATM インターフェイスの始動および設定方法	HC-8
ATM インターフェイスの始動	HC-8
オプションの ATM インターフェイス パラメータの設定	HC-11
PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法	HC-13
PVC を持つポイントツーポイント ATM サブインターフェイスの作成	HC-13
オプションのポイントツーポイント ATM PVC パラメータの設定	HC-15
VP トンネルの作成および設定方法	HC-18
ATM インターフェイスでの VP トンネルの作成および設定	HC-19
VP トンネルでの PVC を持つサブインターフェイスの作成および設定	HC-22
レイヤ 2 接続回路 (AC) の設定方法	HC-25
レイヤ 2 ポート モード AC の作成	HC-25
レイヤ 2 ポート モード AC での オプション パラメータの設定	HC-27
PVC を持つ ATM レイヤ 2 サブインターフェイスの作成	HC-29
オプションの ATM レイヤ 2 PVC パラメータの設定	HC-30
PVP を持つ ATM レイヤ 2 サブインターフェイスの作成	HC-33
オプションの ATM レイヤ 2 PVP パラメータの設定	HC-34
VC クラスの作成および設定方法	HC-37
VC クラスの作成および設定	HC-37

ポイントツーポイント ATM メイン インターフェイスへの VC クラスの付加	HC-40
ポイントツーポイント ATM サブインターフェイスへの VC クラスの付加	HC-41
ATM サブインターフェイスの PVC への VC クラスの付加	HC-42
ATM インターフェイスでの ILMI の設定方法	HC-44
ATM インターフェイスでの ILMI のイネーブル化	HC-44
ATM インターフェイスでの ILMI のディセーブル化	HC-46
チャネライズド ATM の設定方法	HC-48
仮想パス (VP) トンネルを持つクリア チャネル ATM の設定方法	HC-51
Cisco IOS XR ソフトウェアでの ATM の設定例	HC-53
ATM インターフェイスの始動と設定 : 例	HC-54
ポイントツーポイント ATM サブインターフェイスの設定 : 例	HC-54
VP トンネル設定 : 例	HC-56
レイヤ 2 AC の作成および設定 : 例	HC-56
VC クラスの作成および設定 : 例	HC-57
チャネライズド ATM の設定 : 例	HC-58
仮想パス (VP) トンネルを持つクリア チャネル ATM の設定 : 例	HC-58
その他の参考資料	HC-59
関連資料	HC-59
規格	HC-59
MIB	HC-59
RFC	HC-59
シスコのテクニカル サポート	HC-60
Cisco IOS XR での双方向フォワーディング検出の設定	HC-61
この章の構成	HC-62
BFD の実装の前提条件	HC-62
BFD に関する情報	HC-63
IPv4 での BFD	HC-63
IPv6 での BFD	HC-65
バンドル VLAN での BFD	HC-65
BFD のパケット形式	HC-66
制約事項	HC-66
BFD の設定	HC-66
BFD カウンタのクリアおよびディセーブル化	HC-77
Cisco IOS XR での 双方向フォワーディング検出 (BFD) の設定例	HC-78
双方向フォワーディング検出 (BFD) : 例	HC-78
スタティック ルートでの BFD : 例	HC-79
BFD エコー モードのディセーブル化 : 例	HC-80
関連情報	HC-80

その他の参考資料	HC-80	
関連資料	HC-80	
規格	HC-81	
RFC	HC-81	
MIB	HC-81	
シスコのテクニカル サポート	HC-81	
Cisco IOS XR ソフトウェアでのチャネライズド SONET の設定	HC-83	
この章の構成	HC-83	
チャネライズド SONET 設定の前提条件	HC-83	
チャネライズド SONET の設定に関する情報	HC-84	
チャネライズド SONET の概要	HC-84	
チャネライズド SDH の概要	HC-89	
チャネライズド SONET/SDH のデフォルト設定値	HC-91	
チャネライズド SONET の設定方法	HC-91	
SONET T3 チャンネルおよび VT1.5 がマッピングされた T1 チャンネルの設定	HC-92	
Packet over Sonet チャンネルの設定	HC-96	
クリア チャンネル T3 の設定	HC-99	
チャネライズド SONET 自動保護スイッチング (APS) の設定	HC-103	
SDH AU-3 の設定	HC-106	
SDH AU-4 の設定	HC-109	
チャネライズド SONET の設定例	HC-114	
チャネライズド SONET T3 から T1 への設定 : 例	HC-114	
チャネライズド Packet over SONET の設定 : 例	HC-115	
クリア チャンネル T3 の設定 : 例	HC-115	
チャネライズド SONET APS の設定 : 例	HC-116	
チャネライズド SDH AU-3 の設定 : 例	HC-116	
チャネライズド SDH AU-4 の設定 : 例	HC-116	
関連情報	HC-117	
その他の参考資料	HC-117	
関連資料	HC-118	
規格	HC-118	
MIB	HC-118	
RFC	HC-118	
シスコのテクニカル サポート	HC-119	
Cisco IOS XR ソフトウェアでの高密度波長分割多重コントローラの設定	HC-121	
この章の構成	HC-121	
DWDM コントローラ インターフェイスを設定するための前提条件	HC-122	
DWDM コントローラに関する情報	HC-122	

DWDM コントローラの設定方法	HC-123
光パラメータの設定	HC-123
G.709 パラメータの設定	HC-125
例	HC-128
DWDM コントローラでパフォーマンス モニタリングを実行する方法	HC-128
DWDM コントローラのパフォーマンス モニタリングの設定	HC-129
その他の参考資料	HC-134
関連資料	HC-134
規格	HC-135
MIB	HC-135
RFC	HC-135
シスコのテクニカル サポート	HC-135
Cisco IOS XR の診断の実行	HC-137
この章の構成	HC-138
診断を実行するための前提条件	HC-138
診断の実行に関する制約事項	HC-138
使用できるオンライン診断テスト	HC-139
その他の参考資料	HC-141
関連資料	HC-141
規格	HC-141
MIB	HC-141
RFC	HC-141
シスコのテクニカル サポート	HC-142
Cisco IOS XR ソフトウェアでのイーサネット インターフェイスの設定	HC-143
この章の構成	HC-144
イーサネット インターフェイスの前提条件	HC-144
イーサネット インターフェイスの設定に関する情報	HC-145
イーサネット テクノロジーの概要	HC-146
ギガビット イーサネットおよび 10 ギガビット イーサネットのデフォルト設定値	HC-146
ファスト イーサネットのデフォルト設定値	HC-146
イーサネット インターフェイスでのレイヤ 2 VPN	HC-147
ギガビット イーサネット プロトコル規格の概要	HC-148
MAC アドレス	HC-149
MAC アカウンティング	HC-149
イーサネット MTU	HC-149
イーサネット インターフェイスでのフロー制御	HC-150
802.1Q VLAN	HC-150

VRRP	HC-150	
HSRP	HC-151	
ファスト イーサネット インターフェイスでのデュプレックス モード		HC-151
ファスト イーサネット インターフェイスの速度	HC-151	
イーサネット インターフェイスでのリンクのオートネゴシエーション		HC-152
イーサネット インターフェイスでのキャリア遅延	HC-153	
イーサネット インターフェイスの設定方法	HC-153	
ギガビット イーサネット インターフェイスまたは 10 ギガビット イーサネット インターフェイスの設定	HC-153	
ファスト イーサネット インターフェイスの設定	HC-157	
イーサネット インターフェイスでの MAC アカウンティングの設定		HC-159
イーサネット ポートでの接続回路の設定	HC-161	
イーサネット インターフェイスの設定例	HC-163	
イーサネット インターフェイスの設定 : 例	HC-164	
ファスト イーサネット インターフェイスの設定 : 例	HC-164	
MAC アカウンティングの設定 : 例	HC-165	
レイヤ 2 VPN AC : 例	HC-165	
関連情報	HC-165	
その他の参考資料	HC-166	
関連資料	HC-166	
規格	HC-166	
MIB	HC-166	
RFC	HC-166	
シスコのテクニカル サポート	HC-167	
Cisco IOS XR ソフトウェアでのフレームリレーの設定		HC-169
この章の構成	HC-169	
フレームリレー設定の前提条件	HC-170	
フレームリレー インターフェイスに関する情報	HC-170	
フレームリレーのカプセル化	HC-170	
マルチリンク フレームリレー	HC-173	
エンドツーエンド フラグメンテーション (FRF.12)	HC-177	
フレームリレーの設定	HC-177	
インターフェイスでのデフォルト フレームリレー設定の変更	HC-178	
フレームリレーのカプセル化を設定したインターフェイスでの LMI のディセーブル	HC-180	
マルチリンク フレームリレー バンドル インターフェイスの設定	HC-182	
チャネライズド フレームリレー シリアル インターフェイスでの FRF.12 エンドツーエンド フラグメンテーションの設定	HC-188	
フレームリレーの設定例	HC-192	

オプションのフレームリレー パラメータ : 例	HC-193
マルチリンク フレームリレー : 例	HC-195
エンドツーエンド フラグメンテーション : 例	HC-196

その他の参考資料 HC-196

関連資料	HC-196
------	--------

規格	HC-196
----	--------

MIB	HC-197
-----	--------

RFC	HC-197
-----	--------

シスコのテクニカル サポート	HC-197
----------------	--------

Cisco IOS XR ソフトウェアでのリンク バンドルの設定 HC-199

この章の構成	HC-200
--------	--------

リンク バンドルを設定するための前提条件	HC-200
----------------------	--------

Cisco CRS-1 ルータでリンク バンドルを設定するための前提条件	HC-200
--------------------------------------	--------

Cisco XR 12000 シリーズ ルータでリンク バンドルを設定するための前提条件	HC-201
--	--------

リンク バンドルを設定する際の制約事項	HC-202
---------------------	--------

リンク バンドルの設定に関する情報	HC-202
-------------------	--------

リンク バンドルの概要	HC-202
-------------	--------

リンク バンドルの特性	HC-203
-------------	--------

LACP を通じたリンク集約	HC-205
----------------	--------

Cisco CRS-1 ルータでの QoS とリンク バンドル	HC-206
---------------------------------	--------

イーサネット リンク バンドル上の VLAN	HC-206
------------------------	--------

リンク バンドルの設定の概要	HC-207
----------------	--------

RP スイッチオーバー時のノンストップ フォワーディング	HC-207
------------------------------	--------

リンク スイッチオーバー	HC-207
--------------	--------

リンク バンドルの設定方法	HC-208
---------------	--------

イーサネット リンク バンドルの設定	HC-208
--------------------	--------

VLAN バンドルの設定	HC-213
--------------	--------

POS リンク バンドルの設定	HC-220
-----------------	--------

リンク バンドルの設定例	HC-225
--------------	--------

その他の参考資料 HC-226

関連資料	HC-226
------	--------

規格	HC-226
----	--------

MIB	HC-226
-----	--------

RFC	HC-226
-----	--------

シスコのテクニカル サポート	HC-227
----------------	--------

Cisco IOS XR ソフトウェアでの仮想のループバック インターフェイスおよびマルチ インターフェイスの設定 HC-229

この章の構成	HC-230
仮想インターフェイス設定の前提条件	HC-230
仮想インターフェイスの設定に関する情報	HC-230
仮想ループバック インターフェイスの概要	HC-230
マルチ インターフェイスの概要	HC-231
仮想管理インターフェイスの概要	HC-231
アクティブ/スタンバイ RP および仮想インターフェイスの設定	HC-231
仮想インターフェイスの設定方法	HC-232
仮想ループバック インターフェイスの設定	HC-232
マルチ インターフェイスの設定	HC-233
IPv4 仮想インターフェイスの設定	HC-235
仮想インターフェイスの設定例	HC-236
ループバック インターフェイスの設定 : 例	HC-236
マルチ インターフェイスの設定 : 例	HC-237
仮想 IPv4 インターフェイスの設定 : 例	HC-237
その他の参考資料	HC-237
関連資料	HC-238
規格	HC-238
MIB	HC-238
RFC	HC-238
シスコのテクニカル サポート	HC-239

Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイスの高度な設定と変更 HC-241

この章の構成	HC-242
管理イーサネット インターフェイス設定の前提条件	HC-242
管理イーサネット インターフェイスの設定に関する情報	HC-243
デフォルト インターフェイス設定	HC-243
高度な管理イーサネット インターフェイス設定の実行方法	HC-243
管理イーサネット インターフェイスの設定	HC-243
管理イーサネット インターフェイスのデュプレックス モードの設定	HC-245
管理イーサネット インターフェイスの速度の設定	HC-247
管理イーサネット インターフェイスの MAC アドレスの変更	HC-248
管理イーサネット インターフェイス設定の確認	HC-249
管理イーサネット インターフェイスの設定例	HC-250
管理イーサネット インターフェイスの設定 : 例	HC-250
その他の参考資料	HC-251

関連資料	HC-251
規格	HC-251
MIB	HC-251
RFC	HC-252
シスコのテクニカル サポート	HC-252

Cisco IOS XR ソフトウェアでの NetFlow の設定 HC-253

この章の構成	HC-254
NetFlow を設定するための前提条件	HC-254
NetFlow を設定するための制約事項	HC-255
NetFlow の設定に関する情報	HC-255
NetFlow の概要	HC-255
モニタ マップの概要	HC-256
サンプラ マップの概要	HC-256
エクスポート マップの概要	HC-257
NetFlow コンフィギュレーション サブモード	HC-257
IPv4 と IPv6 をサポートする MPLS フロー モニタ	HC-260
宛先ベースの NetFlow アカウンティング	HC-261

Cisco IOS XR ソフトウェアでの NetFlow の設定方法 HC-262

NetFlow の設定例	HC-279
サンプラ マップ : 例	HC-279
エクスポート マップ : 例	HC-279
フロー モニタ マップ : 例	HC-280
IPv4 と IPv6 をサポートする MPLS フロー モニタ : 例	HC-280
宛先ベースの NetFlow アカウンティング : 例	HC-281

その他の参考資料	HC-282
関連資料	HC-282
規格	HC-282
MIB	HC-282
RFC	HC-283
シスコのテクニカル サポート	HC-283

Cisco IOS XR ソフトウェアでの POS インターフェイスの設定 HC-285

この章の構成	HC-287
POS インターフェイスを設定するための前提事項	HC-287
POS インターフェイスの設定に関する情報	HC-288
POS インターフェイスのデフォルト設定	HC-288
Cisco HDLC カプセル化	HC-289
PPP カプセル化	HC-289
キープアライブ タイマー	HC-290

フレームリレーのカプセル化	HC-291	
POS インターフェイスの設定方法	HC-293	
POS インターフェイスの始動	HC-294	
オプションの POS インターフェイス パラメータの設定	HC-296	
PVC を持つポイントツーポイント POS サブインターフェイスの作成	HC-299	
オプションの PVC パラメータの設定	HC-301	
POS インターフェイスでのキープアライブ インターバルの変更	HC-304	
レイヤ 2 接続回路 (AC) の設定方法	HC-306	
PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成	HC-306	
オプションのレイヤ 2 PVC パラメータの設定	HC-309	
オプションのレイヤ 2 サブインターフェイス パラメータの設定	HC-311	
POS インターフェイスの設定例	HC-312	
POS インターフェイスの始動と Cisco HDLC カプセル化の設定 : 例	HC-312	
POS インターフェイスでのフレームリレー カプセル化の設定 : 例	HC-313	
POS インターフェイスでの PPP カプセル化の設定 : 例	HC-314	
その他の参考資料	HC-315	
関連資料	HC-315	
規格	HC-315	
MIB	HC-316	
RFC	HC-316	
シスコのテクニカル サポート	HC-316	
Cisco IOS XR ソフトウェアでの PPP の設定	HC-317	
この章の構成	HC-317	
PPP 認証設定の前提条件	HC-318	
PPP 認証について	HC-318	
PAP 認証	HC-319	
CHAP 認証	HC-319	
MS-CHAP 認証	HC-320	
PPP 認証の設定方法	HC-320	
PAP、CHAP、MS-CHAP 認証のイネーブル化	HC-320	
PAP 認証パスワードの設定	HC-323	
CHAP 認証パスワードの設定	HC-325	
MS-CHAP 認証パスワードの設定	HC-327	
デフォルトの PPP 設定の変更方法	HC-329	
認証プロトコルをディセーブルにする方法	HC-332	
インターフェイスでの PAP 認証のディセーブル化	HC-332	
インターフェイスでの CHAP 認証のディセーブル化	HC-334	
インターフェイスでの MS-CHAP 認証のディセーブル化	HC-336	

マルチリンク PPP について	HC-337
サポートされるカード	HC-338
機能の概要	HC-338
制限事項	HC-338
マルチリンク PPP の設定方法	HC-339
コントローラの設定	HC-339
インターフェイスの設定	HC-342
MLPPP オプション機能の設定	HC-344
PPP の設定例	HC-346
POS インターフェイスでの PPP カプセル化の設定 : 例	HC-346
シリアル インターフェイスでの PPP カプセル化の設定 : 例	HC-347
MLPPP の設定 : 例	HC-347
マルチリンク PPP 設定の確認	HC-348
その他の参考資料	HC-350
関連資料	HC-350
規格	HC-350
MIB	HC-350
RFC	HC-351
シスコのテクニカル サポート	HC-351

Cisco IOS XR ソフトウェアでの物理インターフェイスのプリコンフィギュレーション HC-353

この章の構成	HC-354
物理インターフェイスのプリコンフィギュレーションを行うための前提条件	HC-354
物理インターフェイスのプリコンフィギュレーションに関する情報	HC-355
物理インターフェイスのプリコンフィギュレーションの概要	HC-355
インターフェイスのプリコンフィギュレーションを行う利点	HC-356
インターフェイス プリコンフィギュレーション コマンドの使用方法	HC-356
アクティブ/スタンバイ RP および仮想インターフェイスの設定	HC-356
物理インターフェイスのプリコンフィギュレーションを行う方法	HC-357
物理インターフェイスのプリコンフィギュレーション例	HC-358
インターフェイスのプリコンフィギュレーション : 例	HC-358
関連情報	HC-359
その他の参考資料	HC-359
関連資料	HC-359
規格	HC-360
MIB	HC-360
RFC	HC-360
シスコのテクニカル サポート	HC-360

Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定	HC-361
この章の構成	HC-362
シリアル インターフェイスを設定するための前提事項	HC-362
シリアル インターフェイスに関する情報	HC-363
概要 : クリア チャネル SPA 上のシリアル インターフェイスの設定	HC-364
概要 : チャネライズド SPA 上のシリアル インターフェイスの設定	HC-365
シリアル インターフェイスの概念	HC-366
Cisco HDLC カプセル化	HC-366
PPP カプセル化	HC-366
キープアライブ タイマー	HC-368
フレームリレーのカプセル化	HC-369
フレームリレー上の Layer 2 Tunnel Protocol Version 3 ベースのレイヤ 2 VPN	HC-370
シリアル インターフェイス コンフィギュレーションのデフォルト設定	HC-371
シリアル インターフェイスの表記方法	HC-371
シリアル インターフェイスの設定方法	HC-372
シリアル インターフェイスの始動	HC-372
オプションのシリアル インターフェイス パラメータの設定	HC-375
PVC を持つポイントツーポイント シリアル サブインターフェイスの作成	HC-378
オプションの PVC パラメータの設定	HC-381
シリアル インターフェイスでのキープアライブ インターバルの変更	HC-383
レイヤ 2 接続回路 (AC) の設定方法	HC-385
PVC を持つシリアル レイヤ 2 サブインターフェイスの作成	HC-385
オプションのシリアル レイヤ 2 PVC パラメータの設定	HC-387
シリアル インターフェイスの設定例	HC-390
シリアル インターフェイスの始動と Cisco HDLC カプセル化の設定 : 例	HC-390
シリアル インターフェイスでのフレームリレー カプセル化の設定 : 例	HC-391
シリアル インターフェイスでの PPP カプセル化の設定 : 例	HC-392
その他の参考資料	HC-393
関連資料	HC-393
規格	HC-393
MIB	HC-393
RFC	HC-394
シスコのテクニカル サポート	HC-394
Cisco IOS XR ソフトウェアでの SRP インターフェイスの設定	HC-395
この章の構成	HC-395
SRP インターフェイスを設定するための前提条件	HC-396
SRP インターフェイスの設定に関する情報	HC-396

SRP インターフェイスの設定方法	HC-397
PortPLIM ポートでの SRP のイネーブル化	HC-398
OC-48/STM-16 SPA ポートでの SRP のイネーブル化	HC-400
OC-192/STM-64 SPA ポートでの SRP のイネーブル化	HC-403
基本的な SRP 設定の作成	HC-405
インテリジェント保護スイッチング (IPS) の設定	HC-407
SRP によるモジュラ サービス品質コマンドライン インターフェイス (MQC) の設定	HC-410
リングへのノードの追加	HC-414
SRP インターフェイスの設定例	HC-420
SRP のイネーブル化 : 例	HC-420
基本的な SRP の設定 : 例	HC-420
SRP によるモジュラ QoS の設定 : 例	HC-421
その他の参考資料	HC-421
関連資料	HC-421
規格	HC-422
MIB	HC-422
RFC	HC-422
シスコのテクニカル サポート	HC-422
Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定	HC-423
この章の構成	HC-424
クリア チャネル SONET コントローラを設定するための前提条件	HC-424
SONET コントローラの設定に関する情報	HC-425
SONET コントローラの概要	HC-425
SONET コントローラのデフォルト設定値	HC-426
SONET APS	HC-427
クリア チャネル SONET コントローラの設定方法	HC-428
クリア チャネル SONET コントローラの設定	HC-428
SONET APS の設定	HC-432
Fast Reroute がトリガーされないように hold-off タイマーを設定する	HC-437
SONET コントローラの設定例	HC-438
SONET コントローラの設定 : 例	HC-439
SONET APS グループの設定 : 例	HC-439
関連情報	HC-440
その他の参考資料	HC-440
関連資料	HC-440
規格	HC-440
MIB	HC-440

RFC HC-441

シスコのテクニカル サポート HC-441

Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定 HC-443

この章の構成 HC-444

T3/E3 コントローラ設定の前提条件 HC-444

T3/E3 コントローラおよびシリアル インターフェイスに関する情報 HC-444

T3 および E3 コントローラのデフォルト設定値 HC-445

T1 および E1 コントローラのデフォルト設定値 HC-446

クリア チャネル T3/E3 コントローラおよびチャネル化された T1/E1 コントローラの設定方法 HC-447

カード タイプの設定 HC-448

クリア チャネル E3 コントローラの設定 HC-450

デフォルトの E3 コントローラ設定の変更 HC-451

クリア チャネル T3 コントローラの設定 HC-454

チャネル化された T3 コントローラの設定 HC-455

デフォルトの T3 コントローラ設定の変更 HC-457

T1 コントローラの設定 HC-460

E1 コントローラの設定 HC-463

BERT の設定 HC-467

クリア チャネル T3/E3 およびチャネル化した T3 コントローラの設定 : 例 HC-472

クリア チャネル T3 コントローラの設定 : 例 HC-472

T3 コントローラでのチャネル化した T1 コントローラの設定 : 例 HC-473

T3 コントローラでの BERT の設定 : 例 HC-474

その他の参考資料 HC-475

関連資料 HC-475

規格 HC-475

MIB HC-476

RFC HC-476

シスコのテクニカル サポート HC-476

Cisco IOS XR ソフトウェアでのトンネル インターフェイスの設定 HC-477

この章の構成 HC-478

トンネル インターフェイスを設定するための前提事項 HC-478

トンネル インターフェイスの設定に関する情報 HC-478

トンネル インターフェイスの概要 HC-478

仮想インターフェイスの命名規則 HC-479

Tunnel-IPSec の概要 HC-479

Tunnel-IPSec の命名規則 HC-480

クリプト プロファイル セット	HC-480
トンネル インターフェイスの設定方法	HC-480
Tunnel-IPSec インターフェイスの設定	HC-480
トンネル インターフェイスの設定例	HC-482
Tunnel-IPSec : 例	HC-482
関連情報	HC-483
その他の参考資料	HC-483
関連資料	HC-483
規格	HC-484
MIB	HC-484
RFC	HC-484
シスコのテクニカル サポート	HC-484

Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定 HC-485

この章の構成	HC-486
802.1Q VLAN インターフェイス設定の前提条件	HC-486
802.1Q VLAN インターフェイスの設定に関する情報	HC-486
802.1Q VLAN の概要	HC-487
802.1Q タグ付きフレーム	HC-487
サブインターフェイス	HC-487
サブインターフェイス MTU	HC-487
ネイティブ VLAN	HC-488
イーサネット バンドルでの VLAN サブインターフェイス	HC-488
VLAN インターフェイスでのレイヤ 2 VPN	HC-488
802.1Q VLAN インターフェイスの設定方法	HC-489
802.1Q VLAN サブインターフェイスの設定	HC-489
ネイティブ VLAN の設定	HC-492
VLAN での接続回路の設定	HC-494
802.1Q VLAN サブインターフェイスの削除	HC-497
VLAN インターフェイスの設定例	HC-499
VLAN サブインターフェイス : 例	HC-499
その他の参考資料	HC-501
関連資料	HC-501
規格	HC-501
MIB	HC-501
RFC	HC-501
シスコのテクニカル サポート	HC-502

Index



はじめに

『Cisco IOS XR インターフェイスおよびハードウェア コンポーネント コンフィギュレーション ガイド』では、ルータ インターフェイスおよびハードウェア設定に関連する情報と手順について説明します。

ここで説明する内容は、次のとおりです。

- [マニュアルの変更履歴](#)
- [マニュアルの入手方法およびテクニカル サポート](#)

マニュアルの変更履歴

表 1 に、初版以降このマニュアルに加えられた技術的な変更内容を示します。

表 1 マニュアルの変更履歴

リビジョン	日付	変更点
OL-17356-01-J	2009 年 3 月	本マニュアルの初版

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手方法、テクニカル サポート、その他の有用な情報について、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

『*What's New in Cisco Product Documentation*』は RSS フィードとして購読できます。また、リーダーアプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。



Cisco IOS XR ソフトウェアでの ATM インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアを使用して Cisco XR 12000 シリーズ ルータに Asynchronous Transfer Mode (ATM; 非同期転送モード) を設定する手順について説明します。ATM は、Wide Area Network (WAN) で広く使用されているセルスイッチングおよび多重化のテクノロジーです。ATM プロトコル標準は、さまざまな低速および高速のネットワーク メディアを使用して、ポイントツーポイント、ポイントツーマルチポイント、ブロードキャスト サービスの接続を可能にします。2 つの ATM Permanent Virtual Circuit (PVC; 相手先固定接続) 間の接続は、ATM シグナリング メカニズムを使用して確立されます。次に示す ATM フォーラム規格により各種 ATM シグナリング規格が定義されています。

- User-Network Interface (UNI; ユーザ網インターフェイス) バージョン 3.0、バージョン 3.1、バージョン 4.0
- International Telecommunication Union (ITU; 国際電気通信連合)
- Internet Engineering Task Force (IETF; インターネット技術タスク フォース)

Cisco IOS XR での双方向フォワーディング検出 (BFD) 設定機能の履歴

リリース	変更点
リリース 3.4.0	Cisco XR 12000 シリーズ ルータの次のハードウェアに、この機能が追加されました。 <ul style="list-style-type: none">• Cisco XR 12000 シリーズ 4 ポート OC-3c/STM-1c ATM ISE ラインカード、マルチモード• Cisco XR 12000 シリーズ 4 ポート OC-3c/STM-1c ATM ISE ラインカード、シングルモード• Cisco XR 12000 シリーズ 4 ポート OC-12/STM-4 ATM マルチモード ISE ラインカード、SC コネクタ搭載• Cisco XR 12000 シリーズ 4 ポート OC-12/STM-4 ATM シングルモード、Intermediate-Reach (IR; 中距離) ISE ラインカード、SC コネクタ搭載
リリース 3.4.1	Layer 2 Virtual Private Network (L2VPN; レイヤ 2 バーチャルプライベートネットワーク) 機能が、初めて Cisco XR 12000 シリーズ ルータの ATM インターフェイスでサポートされました。
リリース 3.5.0	Operation、Administration、Maintenance (OAM) の設定が、初めて L2VPN ATM インターフェイスでサポートされました。
リリース 3.6.0	変更ありません。

リリース 3.7.0	次の新しい情報が追加されました。 <ul style="list-style-type: none"> • Cisco XR 12000 シリーズ ルータの OC3/OC-12-ATM-V2 Shared Port Adapter (SPA; 共有ポート アダプタ) での Circuit-Emulation-over-Packet • チャネライズド ATM • Virtual Path (VP; 仮想パス) トンネルのあるクリア チャネル ATM
リリース 3.8.0	次のカードのサポートが追加されました。 <ul style="list-style-type: none"> • 1 ポート OC3c/STM1 SFP 光ファイバ ATM-v2 SPA • 1 ポート OC12c/STM4 SFP 光ファイバ ATM-v2 SPA • 3 ポート OC3c/STM1 SFP 光ファイバ ATM-v2 SPA 次の項で、表示出力例が更新されました。 <ul style="list-style-type: none"> • 「クリア チャネル SONET コントローラの設定方法」 • 「Fast Reroute がトリガーされないようにするための hold-off タイマーの設定」

この章の構成

- 「ATM の実装の前提条件」 (P.2)
- 「ATM に関する情報」 (P.3)
- 「ATM インターフェイスの始動および設定方法」 (P.8)
- 「PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法」 (P.13)
- 「VP トンネルの作成および設定方法」 (P.18)
- 「レイヤ 2 接続回路 (AC) の設定方法」 (P.25)
- 「VC クラスの作成および設定方法」 (P.37)
- 「ATM インターフェイスでの ILMI の設定方法」 (P.44)
- 「チャネライズド ATM の設定方法」 (P.48)
- 「仮想パス (VP) トンネルを持つクリア チャネル ATM の設定方法」 (P.51)
- 「Cisco IOS XR ソフトウェアでの ATM の設定例」 (P.53)
- 「その他の参考資料」 (P.59)

ATM の実装の前提条件

次に、ATM を実装するための前提条件を示します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

- Cisco IOS XR ソフトウェアが稼動している Cisco XR 12000 シリーズ ルータであること。

ATM に関する情報

ネットワーク ノードは、データを 53 バイトの ATM セルとして構成して転送するために、ATM 接続を使用します。ユーザ情報（音声、ビデオ、データなど）は、接続の一端で ATM セルにセグメント化され、接続の他端で再構成されます。ATM Adaptation Layer (AAL; ATM アダプテーション レイヤ) は、ユーザ情報の ATM セルへの変換を定義します。AAL1 および AAL2 はアイソクロナストラフィック（音声やビデオなど）を処理し、ATM ノードに Circuit Emulation Service (CES; 回線エミュレーション サービス) ATM インターフェイス カードが搭載されているか、または Voice over AAL2 機能を装備している場合のみ、ATM ノードに関連します。AAL3/4 および AAL5 は、データ通信のサポート、つまり、データ パケットのセグメント化と再構成を行います。

1 つの ATM ネットワークには、スイッチとルータという 2 種類のデバイスがあります。通常、ATM スイッチはレイヤ 2 でパケット スイッチングを行うのに対し、ATM ルータはレイヤ 3 アドレス (IPv4 ネットワーク アドレス、IPv6 ネットワーク アドレス、Multiprotocol Label Switching (MPLS; マルチプロトコル ラベル スイッチング) ラベルなど) を使用してパケット スイッチングを行います。

ATM は、次のラインカードでサポートされます。

- 1 ポート OC3c/STM1 SFP 光ファイバ ATM-v2 SPA
- 1 ポート OC12c/STM4 SFP 光ファイバ ATM-v2 SPA
- 3 ポート OC3c/STM1 SFP 光ファイバ ATM-v2 SPA
- 4 ポート OC-3c/STM-1c ATM ISE
- 4 ポート OC-12/STM-4 ATM ISE

Cisco IOS XR ソフトウェア ATM インターフェイスには、次の動作モードがあります。

- ポイントツーポイント
- レイヤ 2 ポート モード



(注)

単一の ATM インターフェイスで、ポイントツーポイントと L2VPN のサブインターフェイスを同時にサポートすることができます。

Cisco IOS XR ソフトウェアでは、ATM インターフェイスの設定は階層的であり、次に示す要素で構成されます。

1. 物理インターフェイスである ATM メイン インターフェイス。ATM メイン インターフェイスは、ポイントツーポイント サブインターフェイス、VP トンネル、Interim Local Management Interface (ILMI; 暫定ローカル管理インターフェイス) インターフェイスを持つように設定するか、またはレイヤ 2 ポート モードの Attachment Circuit (AC; 接続回路) またはレイヤ 2 サブインターフェイス AC として設定できます。
2. ATM メイン インターフェイス下に設定される ATM サブインターフェイス。ATM サブインターフェイスは、PVC または Permanent Virtual Path (PVP; 相手先固定パス) を ATM サブインターフェイス下に設定しないとトラフィックをアクティブに伝送しません。

3. ATM サブインターフェイス下に設定される PVC。PVC は、各サブインターフェイスに 1 つ設定できます。PVC は、ポイントツーポイントおよびレイヤ 2 のサブインターフェイス下でサポートされます。
4. レイヤ 2 ATM サブインターフェイス下に設定される相手先固定パス (PVP)。PVP は、各サブインターフェイスに 1 つ設定できます。

仮想回線 (VC) クラスのマッピング

Virtual Circuit (VC; 仮想回線) クラスでは、メイン インターフェイス、サブインターフェイス、PVC にマッピングされる VC パラメータを設定することができます。VC クラスがない場合は、各 ATM メイン インターフェイス、サブインターフェイス、PVC、およびルータ上で多数の手動設定を行う必要があります。この設定には時間がかかり、エラーが発生する可能性が高くなります。VC クラスの作成後は、その VC クラスを必要な数の ATM インターフェイス、サブインターフェイス、PVC に適用することができます。

VC クラスには、次のタイプの設定データが含まれます。

- VC のための ATM カプセル化
- OAM 管理
- トラフィック シェーピング

設定の優先順位は、次のリストで示すように階層的であり、PVC での設定が最優先、ATM メイン インターフェイスに付加された VC クラスでの設定が最低順位となります。

1. PVC での設定
2. PVC に付加された VC クラスでの設定
3. サブインターフェイスでの設定
4. サブインターフェイスに付加された VC クラスでの設定
5. ATM メイン インターフェイスでの設定
6. ATM メイン インターフェイスに付加された VC クラスでの設定

たとえば、ある PVC で Unspecified Bit Rate (UBR; 未指定ビット レート) トラフィック シェーピングが設定されており、その PVC が、Constant Bit Rate (CBR; 固定ビット レート) トラフィック シェーピングが設定されているクラス マップに付加されている場合、その PVC は UBR トラフィック シェーピングを保持します。



(注) VC クラスは、レイヤ 2 ポート モード AC およびレイヤ 2 PVP には適用されません。レイヤ 2 VPN 設定では、VC クラスは PVC のみに適用されます。

VP トンネル

ATM インターフェイスは、VP トンネルをサポートします。VP トンネルは、一般的に、PVC のバンドルへのシェーピングや、F4 Operation、Administration、Maintenance (OAM) の管理に使用されます。VP トンネルが ATM メイン インターフェイス下に設定されている場合は、サブインターフェイスと PVC を VP トンネルに追加できます。VP トンネルとその下に設定されている PVC は、同じ Virtual Path Identifier (VPI; 仮想パス識別子) を共有します。VP トンネルが停止すると、その VP トンネル下に設定されているすべての PVC も停止します。

デフォルトでは、各 VP トンネルについて、2 つの F4 OAM 接続が自動的に開きます。任意の VP トンネルの F4 OAM パケットをディセーブルにするには、ATM VP トンネル コンフィギュレーション モードで **f4oam disable** コマンドを使用します。

ATM インターフェイスでの F5 OAM

F5 Operation、Administration、Maintenance (OAM) 機能は、PVC での障害管理およびパフォーマンス管理機能を実行します。PVC で F5 OAM 機能がイネーブルになっていない場合、ネットワーク接続が切断され、サービスが中断されても、その PVC はエンド デバイス上でアップ状態のままとなります。その結果、その接続を指しているルーティング エントリがルーティング テーブルに残るため、パケットは失われます。F5 OAM 機能は、パス上に中断がある場合に、そのような障害を検出して、PVC を停止させます。

PVC で F5 OAM 機能をイネーブルにするには、**oam-pvc manage** コマンドを使用します。PVC で OAM がイネーブルになると、PVC では F5 ループバック セルを生成することができ、PVC での Continuity Check (CC; 継続性チェック) 管理の設定を行えます。PVC での継続性チェックを設定するには、**oam ais-rdi** コマンドおよび **oam retry** コマンドを使用します。

ATM インターフェイスで現在受信している OAM セルおよび今後受信する OAM セルをすべて廃棄するには、インターフェイス コンフィギュレーション モードで **atm oam flush** コマンドを使用します。



(注) **oam ais-rdi** コマンドと **oam retry** コマンドは、**oam-pvc manage** コマンドを使用して PVC での OAM 管理がイネーブルになっている場合のみ有効になります。

ATM インターフェイスでの ILMI

ATM インターフェイスでの物理レイヤ、ATM レイヤ、仮想パス、仮想回線のパラメータの設定とキャプチャについて、ATM フォーラムにより ILMI プロトコルが定義されています。2 つの ATM インターフェイスが ILMI プロトコルを実行すると、これらのインターフェイスは物理的接続を通じて ILMI パケットを交換します。これらのパケットは、最大 484 オクテットの SNMP メッセージで構成されます。ATM インターフェイスは、これらのメッセージを ATM Adaptation Layer 5 (AAL5; ATM アダプテーション レイヤ 5) トレーラーにカプセル化し、パケットをセルにセグメント化して、セルの送信をスケジューリングします。

ILMI 用に設定されているエンド デバイスと通信する ATM インターフェイスで ILMI をイネーブルにする必要があります。ILMI をイネーブルにするには、インターフェイス コンフィギュレーション モードで **pvc vpi/vci ilmi** コマンドを使用して、ATM メイン インターフェイス直下に ILMI のカプセル化を備えた PVC を作成します。

PVC は、ILMI メッセージの伝送に ILMI のカプセル化を使用します。ATM メイン インターフェイスで ILMI PVC を作成するには、インターフェイス コンフィギュレーション モードで **pvc vpi/vci ilmi** コマンドを使用します。



(注) エンド デバイスとルータを接続する PVC の両端では、同じ VPI 値と Virtual Channel Identifier (VCI; 仮想チャネル識別子) 値を使用する必要があります。



(注) ILMI コンフィギュレーション コマンドは、ATM メイン インターフェイスで ILMI PVC が作成されてからのみ使用可能になります。ILMI の設定が、ATM メイン インターフェイスで有効になります。



(注) ILMI の設定は、レイヤ 2 ポート モード AC ではサポートされません。

ATM インターフェイスでのレイヤ 2 VPN

Layer 2 VPN (L2VPN; レイヤ 2 VPN) 機能を使用すると、異なるタイプのレイヤ 2 Attachment Circuit (AC; 接続回路) と疑似接続での接続が可能となり、ユーザはさまざまなタイプのエンドツーエンド サービスを実装することができます。

Cisco IOS XR ソフトウェアは、2 つの ATM AC が結合されているポイントツーポイント、エンドツーエンドのサービスをサポートします。

スイッチングには、次の 2 つの方法があります。

- AC-to-PW : Provider Edge (PE; プロバイダー エッジ) に到達したトラフィックは Pseudowire (PW; 疑似接続) を介してトンネリングされます。また、それとは反対に PW を介して到達したトラフィックは AC を介して送信されます。これが最も一般的なシナリオです。
- ローカル スイッチング: トラフィックが 1 つの AC に到達すると、疑似接続を通過せずに、ただちに別の AC に送信されます。

ATM インターフェイスに L2VPN を設定する場合は、次の事項に注意する必要があります。

- Cisco IOS XR ソフトウェアは、ラインカードごとに最大 2,000 個の AC をサポートします。
- ATM-over-MPLS では、次の 2 種類のセル カプセル化をサポートします。
 - AAL5 CPCS モード: セグメント化されていない ATM セルが MPLS バックボーンを介して転送されます。
 - ATM セル (AAL0) モード: セルがセグメント化されてから、再構成またはパッキングされます。AAL0 は、ATM メインポート、PVC、PVP でサポートされます。AAL0 モードを使用する利点は、帯域幅の効率性を最大化するラベルを ATM セルのグループが共有できることです。



(注) AAL5 モードは、PVC でのみサポートされます。

AC および疑似接続の情報を表示するには、次のコマンドを使用します。

- **show interfaces**
- **show l2vpn xconnect**
- **show atm pvp**
- **show atm pvc**



(注) L2VPN ネットワークの設定の詳細については、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「Implementing MPLS Layer 2 VPNs」モジュールを参照してください。

AAL0 モードのカプセル化による L2VPN AC でのセル パッキング

セル パッキングは、AAL0 モードのカプセル化が設定されている L2VPN ATM インターフェイスでサポートされます。セル パッキングは、ATM 規格で定義されている遅延変動に関連しています。ユーザは、疑似接続によって処理できるセル数を指定し、セル パッキングと組み合わせて使用する Maximum Cell Packing Timeout (MCPT; 最大セル パッキング タイムアウト) タイマーを設定することができます。

cell-packing コマンドを使用して、次に示すタスクを実行できます。

- 単一のパケットで送信できるセルの最大数を設定
- 3 つある MCPT タイマーのいずれかをレイヤ 2 ポート モード AC、PVC、PVP に個別に付加

3 つの MCPT タイマーは、**atm mcpt-timer** コマンドを使用して、ATM メイン インターフェイス下で定義されます。このコマンドでは、単一パケットでのセルパッキングを実行してそのパケットが送信されるまでの最大待機時間をマイクロ秒単位で指定することができます。パッキング可能なセルの最大数に達する前に、関連付けられている MCPT タイマーが終了した場合、そのパケットはそれまでにパッキングされたセル数で送信されます。

さまざまな ATM トラフィック クラスに対応するために、3 つの MCPT タイマーには、それぞれ低、中、高レベルの値を設定することをお勧めします。一般的に、低遅延の固定ビットレート (CBR) トラフィックでは MCPT タイマーに低い値を使用し、高遅延の未指定ビットレート (UBR) トラフィックでは MCPT タイマーで高い値が必要となります。Variable Bit Rate real-time (VBR-rt; 可変ビットレート - リアルタイム) および Variable Bit Rate non-real-time (VBR-nrt; 可変ビットレート - 非リアルタイム) のトラフィックでは、通常、MCPT タイマーに中程度の値を使用します。

Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポートアダプタでの Circuit-Emulation-over-Packet (CEoP)

Circuit-Emulation-over-Packet (CEoP) SPA は、サービス プロバイダーや企業において、データと回線の両サービスを効率的に提供できる単一パケット ネットワークへの移行を可能にする低速 ATM SPA です。

Circuit-Emulation-over-Packet (CEoP) は、物理的接続を仮想的に模倣したものです。CEoP は、ATM 変換および ATM データのルータ エンジンへのトランスペアレントな転送を行います。CEoP は最初にセル同期を行ってから、Segmentation And Reassembly (SAR; セグメント化と再構成)、ルータ エンジンへのセルの送信の順に実行します。

Circuit-Emulation-over-Packet (CEoP) は、パケットスイッチド ネットワークと Time-Division Multiplexed (TDM; 時分割多重) ネットワークの両方を運用しており、スケーラビリティと効率性を確保するために、データ サービスを TDM ネットワークからパケット ネットワークへ移行することを希望しているサービス プロバイダーにおいて役立ちます。

シスコでは、イーサネット、IP、フレームリレーのようなレイヤ 2 およびレイヤ 3 プロトコルの転送が可能なルーティングおよびスイッチングのソリューションを提供しています。ほとんどのアプリケーションやサービスは、パケットベース ネットワークに移行していますが、音声やレガシー アプリケーションを含む一部のアプリケーションやサービスでは、依然として転送に回線や専用線を必要とします。CEoP SPA は、パケットベース ネットワークで回線を転送することにより、Circuit-Emulation-over-Packet (CEoP) を実行します。CEoP SPA は、サービス プロバイダーにおいて、データ サービスと回線サービスの両方を効率的に提供できる単一パケット ネットワークへの移行に役立ちます。

Circuit-Emulation-over-Packet (CEoP) は、Cisco XR 12000 シリーズ ルータの次のカードでサポートされます。

- Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタ (SPA-2CHT3-CE-ATM)

カード上で Circuit-Emulation-over-Packet (CEoP) をイネーブルにするための新規設定や固有の設定、Command-Line Interface (CLI; コマンドライン インターフェイス) は必要ありません。

CEoP でサポートされる機能

CEoP は次の機能をサポートします。

- T3 の T1 へのチャンネル化
- クリア チャンネル T3
- ATM カプセル化のみ
- パケット転送
- QoS
- スケーラビリティ：各 CEoP SPA に 2K L3 インターフェイス装備
- スケーラビリティ：各 CEoP SPA に 2K L2 接続装備
- On-Line Insertion and Removal (OIR; ホットスワップ)
- Quack 認証
- 環境モニタリング
- FPD

CEoP でサポートされない機能

CEoP は次の機能をサポートしません。

- E3 でのチャンネル化
- Inverse Multiplexing (IMA; 逆多重化) および Circuit Emulation (CEM; 回線エミュレーション)

ATM インターフェイスの始動および設定方法

ATM インターフェイスの設定作業について、次の手順で説明します。

- [「ATM インターフェイスの始動」\(P.8\)](#)
- [「オプションの ATM インターフェイス パラメータの設定」\(P.11\)](#)

ATM インターフェイスの始動

ここでは、ATM インターフェイスの始動に使用するコマンドについて説明します。

前提条件

Cisco IOS XR ソフトウェアを実行する Cisco XR 12000 シリーズ ルータに、次のいずれかのラインカードが設置されている必要があります。

- 4 ポート OC12
- 4 ポート OC3

制約事項

アクティブにするインターフェイスについて、ATM 接続の両端での設定が一致している必要があります。

手順の概要

1. **configure**
2. **interface atm interface-path-id**
3. **no shutdown**
4. **end**
または
commit
5. **exit**
6. **exit**
7. 接続の他端でインターフェイスを始動するために、ステップ 1 ~ 6 を繰り返します。
8. **show interfaces atm interface-path-id brief**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id 例： RP/0/0/CPU0:router (config)# interface atm 0/6/0/1	ATM インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	no shutdown 例： RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます。

ATM インターフェイスの始動および設定方法

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router (config-if)# <code>end</code> または RP/0/0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>exit</code></p> <p>例: RP/0/0/CPU0:router (config-if)# <code>exit</code></p>	<p>インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。</p>
<p>ステップ 6 <code>exit</code></p> <p>例: RP/0/0/CPU0:router (config)# <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。</p>
<p>ステップ 7 接続の他端でインターフェイスを始動するために、ステップ 1 ~ 6 を繰り返します。</p>	<p>接続を始動します。</p> <p>(注) ATM 接続の両端で設定が一致している必要があります。</p>
<p>ステップ 8 <code>show interfaces atm interface-path-id brief</code></p> <p>例: RP/0/0/CPU0:router# <code>show interfaces atm 0/6/0/1 brief</code></p>	<p>(任意) インターフェイスがアクティブであり、適切に設定されていることを確認します。</p> <p>ATM インターフェイスが適切に始動されていると、show interfaces atm コマンドの出力結果で、そのインターフェイスの [Intf State] フィールドに [up] と表示されます。</p>

次に行う作業

- 始動した ATM インターフェイスのデフォルト設定を変更するには、「オプションの ATM インターフェイス パラメータの設定」(P.11) を参照してください。
- 始動した ATM インターフェイスでポイントツーポイント サブインターフェイスを設定するには、「PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法」(P.13) を参照してください。

- 始動した ATM インターフェイスで VP トンネルを作成するには、「[VP トンネルの作成および設定方法](#)」(P.18) を参照してください。
- インターフェイスをレイヤ 2 ポスト モード AC として使用するには、「[レイヤ 2 接続回路 \(AC\) の設定方法](#)」(P.25) を参照してください。
- 始動した ATM インターフェイスに VC クラスを付加するには、「[VC クラスの作成および設定方法](#)」(P.37) を参照してください。
- 始動した ATM インターフェイスで ILMI をイネーブルにするには、「[ATM インターフェイスでの ILMI の設定方法](#)」(P.44) を参照してください。

オプションの ATM インターフェイス パラメータの設定

ここでは、ATM インターフェイスでのデフォルト設定の変更を使用できるコマンドについて説明します。

前提条件

ATM インターフェイスのデフォルト設定を変更する前に、ATM インターフェイスを始動して、「[ATM インターフェイスの始動](#)」(P.8) で説明するように shutdown 設定を削除することをお勧めします。

制約事項

アクティブにするインターフェイスについて、ATM 接続の両端での設定が一致している必要があります。

手順の概要

1. **configure**
2. **interface atm *interface-path-id***
3. **atm maxvpi-bits 12**
4. **atm oam flush**
5. **atm mcpt-timers *timer-1 timer-2 timer-3***
6. **end**
または
commit
7. **exit**
8. **exit**
9. **show atm interface atm [*interface-path-id*]**
10. **show interfaces atm *interface-path-id* brief**

ATM インターフェイスの始動および設定方法

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface atm interface-path-id</code> 例: RP/0/0/CPU0:router (config)# <code>interface atm 0/6/0/1</code>	ATM インターフェイス コンフィギュレーション モードを開始します。
ステップ 3 <code>atm maxvpi-bits 12</code> 例: RP/0/0/CPU0:router (config-if)# <code>atm maxvpi-bits 12</code>	(任意) 12 ビット VPI Network-to-Network Interface (NNI) セル形式のサポートをイネーブルにします。
ステップ 4 <code>atm oam flush</code> 例: RP/0/0/CPU0:router (config-if)# <code>atm oam flush</code>	(任意) ATM インターフェイスで現在受信している OAM セルおよび今後受信する OAM セルをすべて廃棄します。
ステップ 5 <code>atm mcpt-timers timer-1 timer-2 timer-3</code> 例: RP/0/0/CPU0:router (config-if)# <code>atm mcpt-timers 50 100 200</code>	(任意) インターフェイスごとに 3 つあるそれぞれの MCPT タイマーで、セルパッキングの最大タイムアウト値をマイクロ秒単位で指定します。 (注) 各タイマーのデフォルト値は 50 マイクロ秒です。 (注) <code>atm mcpt-timers</code> コマンドは、レイヤ 2 ATM AC のみに適用されます。
ステップ 6 <code>end</code> または commit 例: RP/0/0/CPU0:router (config-if)# <code>end</code> または RP/0/0/CPU0:router (config-if)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - <code>yes</code> と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - <code>no</code> と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - <code>cancel</code> と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、<code>commit</code> コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 7	<code>exit</code> 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 8	<code>exit</code> 例: RP/0/0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ 9	<code>show atm interface atm [interface-path-id]</code> 例: RP/0/0/CPU0:router# show atm interface atm 0/6/0/1	(任意) 指定した ATM インターフェイスの ATM 固有データを表示します。
ステップ 10	<code>show interfaces atm interface-path-id</code> 例: RP/0/0/CPU0:router# show interfaces atm 0/6/0/1	(任意) 指定した ATM インターフェイスの一般情報を表示します。

次に行う作業

- 始動した ATM インターフェイスでポイントツーポイント サブインターフェイスを設定するには、「[PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法](#)」(P.13) を参照してください。
- 始動した ATM インターフェイスで VP トンネルを作成するには、「[VP トンネルの作成および設定方法](#)」(P.18) を参照してください。
- インターフェイスをレイヤ 2 ATM AC として使用するには、「[レイヤ 2 接続回路 \(AC\) の設定方法](#)」(P.25) を参照してください。
- 始動した ATM インターフェイスに VC クラスを付加するには、「[VC クラスの作成および設定方法](#)」(P.37) を参照してください。
- 始動した ATM インターフェイスで ILMI をイネーブルにするには、「[ATM インターフェイスでの ILMI の設定方法](#)」(P.44) を参照してください。

PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法

PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定作業について、次の手順で説明します。

- 「[PVC を持つポイントツーポイント ATM サブインターフェイスの作成](#)」(P.13)
- 「[オプションのポイントツーポイント ATM PVC パラメータの設定](#)」(P.15)

PVC を持つポイントツーポイント ATM サブインターフェイスの作成

ここに記載する手順では、ポイントツーポイント ATM サブインターフェイスを作成し、その ATM サブインターフェイスに相手先固定接続 (PVC) を設定します。

■ PVC を持つポイントツーポイント ATM サブインターフェイスの作成および設定方法

前提条件

ATM インターフェイスに ATM サブインターフェイスを作成する前に、「[ATM インターフェイスの始動](#)」(P.8) で説明するように ATM インターフェイスを始動する必要があります。

制約事項

各ポイントツーポイント ATM サブインターフェイスで設定できる PVC は 1 つだけです。

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface point-to-point**
3. **ipv4 address ipv4_address/prefix**
4. **pvc vpi/vci**
5. **end**
または
commit
6. 接続の他端で ATM サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 5 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface point-to-point 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1.10	ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ipv4 address ipv4_address/prefix 例: RP/0/0/CPU0:router (config-subif)# ipv4 address 10.46.8.6/24	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 4	pvc vpi/vci 例: RP/0/0/CPU0:router (config-subif)# pvc 5/10	(任意) ATM 相手先固定接続 (PVC) を作成し、ATM PVC コンフィギュレーション サブモードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

コマンドまたはアクション	目的
<p>ステップ 5 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router (config-subif)# <code>end</code> または RP/0/0/CPU0:router (config-subif)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6 接続の他端で ATM サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1～5 を繰り返します。</p>	<p>ATM 接続を始動します。</p> <p>(注) サブインターフェイス接続の両端で設定が一致している必要があります。</p>

次に行う作業

- オプションの PVC パラメータを設定するには、「[オプションのポイントツーポイント ATM PVC パラメータの設定](#)」(P.15) を参照してください。
- PVC サブモードでレイヤ 3 サービス ポリシー (Multiprotocol Label Switching (MPLS; マルチプロトコル ラベル スイッチング) や Quality Of Service (QoS; サービス品質) など) を PVC に付加するには、適切な Cisco IOS XR ソフトウェア コンフィギュレーション ガイドを参照してください。
- VC クラスを設定して、ATM サブインターフェイスまたは PVC に適用するには、「[VC クラスの作成および設定](#)」(P.37) を参照してください。

オプションのポイントツーポイント ATM PVC パラメータの設定

ここでは、ATM PVC でのデフォルト設定の変更には使用できるコマンドについて説明します。

前提条件

デフォルト PVC 設定を変更する前に、「[PVC を持つポイントツーポイント ATM サブインターフェイスの作成](#)」(P.13) で説明するように ATM サブインターフェイスで PVC を作成する必要があります。

制約事項

PVC の両端での設定が、アクティブにする接続に合っている必要があります。

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface point-to-point**
3. **pvc vpi/vci**
4. **encapsulation {aal5mux ipv4 | aal5nlpid | aal5snap}**
5. **oam-pvc manage [frequency] [disable] [keep-vc-up [seg-aisrdis-failure]]**
6. **oam ais-rdi [down-count [up-count]]**
7. **oam retry**
8. **shape [cbr peak_output_rate |ubr peak_output_rate |vbr-nrt peak_output_rate sustained_output_rate burst_size|vbr-rt peak_output_rate sustained_output_rate burst_size]**
9. **service-policy [input | output] policy_name**
10. **end**
または
commit
11. 接続の他端で PVC を設定するために、ステップ 1 ~ 10 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface point-to-point 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1.10 point-to-point	ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc vpi/vci 例: RP/0/0/CPU0:router (config-subif)# pvc 5/10	PVC に対するサブインターフェイス コンフィギュレーション モードを開始します。
ステップ 4	encapsulation {aal5mux ipv4 aal5nlpid aal5snap} 例: RP/0/0/CPU0:router (config-atm-vc)# encapsulation aal5snap	PVC の ATM アダプテーション レイヤ (AAL) およびカプセル化タイプを設定します。 (注) VC クラスのデフォルト カプセル化タイプは AAL5/SNAP です。

コマンドまたはアクション	目的
<p>ステップ 5 <code>oam-pvc manage [frequency] [disable]</code> <code>[keep-vc-up [seg-aisrdi-failure]]</code></p> <p>例: RP/0/0/CPU0:router (config-atm-vc)# oam-pvc manage 200 keep-vc-up</p>	<p>ATM OAM F5 ループバック セル生成をイネーブルにして、ATM 相手先固定接続 (PVC) の継続性チェック (CC) 管理を設定します。</p> <ul style="list-style-type: none"> 指定した PVC での OAM 管理をディセーブルにするには、disable キーワードを含めます。 CC セルが接続の障害を検出した場合に PVC をアップ状態のまま維持するように指定するには、keep-vc-up キーワードを含めます。 セグメント Alarm Indication Signal/Remote Defect Indication (AIS/RDI; アラーム表示信号/リモート障害表示) セルが受信された場合に、エンド CC 障害やループバック障害が原因で VC が停止しないように指定するには、seg-aisrdi-failure キーワードを含めます。
<p>ステップ 6 <code>oam ais-rdi [down-count [up-count]]</code></p> <p>例: RP/0/0/CPU0:router (config-atm-vc)# oam ais-rdi 25 5</p>	<p>関連付けられている PVC で指定数の OAM Alarm Indication Signal/Remote Defect Indication (AIS/RDI; アラーム表示信号/リモート障害表示) セルが受信された後に PVC が停止するように PVC を設定します。</p>
<p>ステップ 7 <code>oam retry [up-count [down-count</code> <code>[retry-frequency]]]</code></p> <p>例: RP/0/0/CPU0:router (config-atm-vc)# oam retry 5 10 5</p>	<p>PVC の OAM 管理に関連するパラメータを設定します。指定した期間内に OAM AIS/RDI セルが受信されない場合、PVC は始動されます。</p>
<p>ステップ 8 <code>shape [cbr peak_output_rate ubr</code> <code>peak_output_rate vbr-nrt peak_output_rate</code> <code>sustained_output_rate burst_size vbr-rt</code> <code>peak_output_rate sustained_output_rate</code> <code>burst_size]</code></p> <p>例: RP/0/0/CPU0:router (config-atm-vc)# shape vbr-nrt 100000 100000 8000</p>	<p>PVC の ATM トラフィック シェーピングを設定します。ATM トラフィック シェーピングを設定する前に、必要な帯域幅を概算する必要があります。</p> <ul style="list-style-type: none"> peak_output_rate : トラフィックで常時使用可能な最大セル レートを設定します。 Sustained_output_rate : ビット レートの持続出力レート。 burst size : ビット レートのバーストセル サイズ。有効値の範囲は 1 ~ 8,192 です。
<p>ステップ 9 <code>service-policy [input output] policy_name</code></p> <p>例: RP/0/0/CPU0:router (config-atm-vc)# service-policy input policyA</p>	<p>入力または出力 PVC に QoS ポリシーを付加します。policy_name は、PVC に付加するサービス ポリシー名に置き換えてください。</p> <p>(注) サービス ポリシーの作成および設定の詳細については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。</p>

コマンドまたはアクション	目的
<p>ステップ 10</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router (config-subif)# end または RP/0/0/CPU0:router (config-subif)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <p>設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。</p>
<p>ステップ 11</p> <p>接続の他端で PVC を設定するために、ステップ 1～10 を繰り返します。</p>	<p>接続を始動します。</p> <p>(注) 接続の両端で設定が一致している必要があります。</p>

次に行う作業

- PVC サブモードでレイヤ 3 サービス ポリシー (MPLS や QoS など) を PVC に付加するには、適切な Cisco IOS XR ソフトウェア コンフィギュレーション ガイドを参照してください。
- VC クラスを設定して、ATM サブインターフェイスまたは PVC に適用するには、「[VC クラスの作成および設定](#)」(P.37) を参照してください。

VP トンネルの作成および設定方法

ATM VP トンネルの作成および設定作業について、次の手順で説明します。

- 「[ATM インターフェイスでの VP トンネルの作成および設定](#)」(P.19)
- 「[VP トンネルでの PVC を持つサブインターフェイスの作成および設定](#)」(P.22)



(注)

VP トンネルは、ポイントツーポイント ATM インターフェイス固有のものであり、ATM AC では設定できません。

ATM インターフェイスでの VP トンネルの作成および設定

ここに記載する手順では、ポイントツーポイント ATM メイン インターフェイスで VP トンネルを作成します。VP トンネルの作成と設定は、4 段階の手順で行います。

1. 「[ATM インターフェイスの始動](#)」(P.8) で説明するように、ATM インターフェイスを始動します。
2. 「[ATM インターフェイスでの VP トンネルの作成および設定](#)」(P.19) で説明するように、ATM インターフェイスで VP トンネルを作成し、設定します。
3. 「[VP トンネルでの PVC を持つサブインターフェイスの作成および設定](#)」(P.22) で説明するように、VP トンネルで PVC を持つサブインターフェイスを作成します。
4. 「[VP トンネルでの PVC を持つサブインターフェイスの作成および設定](#)」(P.22) で説明するように、VP トンネルの設定を確認するために、VP トンネルを介した接続の他端に対して PING を実行します。

ここに記載する手順では、ATM メイン インターフェイスで VP トンネルを作成します。

前提条件

ATM メイン インターフェイスで VP トンネルを作成する前に、「[ATM インターフェイスの始動](#)」(P.8) で説明するように ATM インターフェイスを始動する必要があります。

制約事項

- VP トンネルは、「[VP トンネルでの PVC を持つサブインターフェイスの作成および設定](#)」(P.22) で説明するように PVC が VP トンネルと同じ VPI 値で作成されるまで、実際にアクティブにはなりません。
- VP トンネルが停止すると、その VP トンネル下に設定されているすべての VC も停止します。
- 次のカードは、VPI 値が 0 の VP トンネルをサポートしません。
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、マルチモード
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、シングルモード
 - 4 ポート OC-12/STM-4 ATM マルチモード ISE ラインカード、SC コネクタ搭載
 - シリーズ 4 ポート OC-12/STM-4 ATM シングルモード、Intermediate-Reach (IR; 中距離) ISE ラインカード、SC コネクタ搭載

手順の概要

1. **configure**
2. **interface atm *interface-path-id***
3. **vp-tunnel *vpi***
4. **f4oam disable**
5. **shape [*cbr peak_output_rate* | **vbr-nrt** *peak_output_rate sustained_output_rate burst_size* | **vbr-rt** *peak_output_rate sustained_output_rate burst_size*]**
6. **end**
または
commit
7. **exit**

■ VP トンネルの作成および設定方法

8. `exit`

9. 接続の他端で VP トンネルを始動するために、ステップ 1～8 を繰り返します。

10. `show atm vp-tunnel interface atm [interface-path-id]`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface atm interface-path-id</code> 例: RP/0/0/CPU0:router (config)# <code>interface atm 0/6/0/1</code>	ATM インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>vp-tunnel vpi</code> 例: RP/0/0/CPU0:router (config)# <code>vp-tunnel 10</code>	ATM インターフェイスで VP トンネルを設定します。
ステップ 4	<code>f4oam disable</code> 例: RP/0/0/CPU0:router (config-atm-vp-tunnel)# <code>f4oam disable</code>	(任意) OAM パケットの送信をディセーブルにします。
ステップ 5	<code>shape [cbr peak_output_rate vbr-nrt peak_output_rate sustained_output_rate burst_size vbr-rt peak_output_rate sustained_output_rate burst_size]</code> 例: RP/0/0/CPU0:router (config-if)# <code>shape</code>	<p>PVC の ATM トラフィック シェーピングを設定します。</p> <p>ATM トラフィック シェーピングを設定する前に、必要な帯域幅を概算する必要があります。</p> <ul style="list-style-type: none"> • <code>peak_output_rate</code> : トラフィックで常時使用可能な最大セル レートを設定します。 • <code>Sustained_output_rate</code> : ビット レートの持続出力レート。 • <code>burst size</code> : ビット レートのバーストセルサイズ。有効値の範囲は 1～8,192 です。 <p>(注) VP トンネルでトラフィック シェーピングを設定した後は、その VP トンネル下に設定されている PVC では直接トラフィック シェーピングを設定できません。VP トンネル下に設定されている PVC で <code>shape</code> コマンドを使用しても、コマンドは拒否されます。</p>

	コマンドまたはアクション	目的
ステップ 6	<pre>end</pre> または <pre>commit</pre> 例: <pre>RP/0/0/CPU0:router (config-if)# end</pre> または <pre>RP/0/0/CPU0:router(config-if)# commit</pre>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</pre> [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 7	<pre>exit</pre> 例: <pre>RP/0/0/CPU0:router (config-if)# exit</pre>	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 8	<pre>exit</pre> 例: <pre>RP/0/0/CPU0:router (config)# exit</pre>	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ 9	接続の他端で VP トンネルを始動するために、ステップ 1 ~ 8 を繰り返します。	VP トンネルを始動します。
ステップ 10	<pre>show atm vp-tunnel interface atm</pre> [interface-path-id] 例: <pre>RP/0/0/CPU0:router (config)# show atm</pre> <pre>vp-tunnel interface atm 0/6/0/1</pre>	ルータ全体または特定の ATM インターフェイスの VP トンネル情報を表示します。

次に行う作業

レイヤ 3 サービス ポリシー (MPLS や QoS など) を VP トンネルまたはその PVC に付加するには、適切な Cisco IOS XR ソフトウェア コンフィギュレーション ガイドを参照してください。

VP トンネルでの PVC を持つサブインターフェイスの作成および設定

ここに記載する手順では、PVC を持つサブインターフェイスを VP トンネルで作成し、設定します。



(注)

VP トンネルは、PVC が VP トンネルと同じ VPI 値で作成されるまで、実際にアクティブにはなりません。

前提条件

ATM VP トンネルで PVC を持つサブインターフェイスを作成する前に、「[ATM インターフェイスでの VP トンネルの作成および設定](#)」(P.19) で説明するように ATM メイン インターフェイスで VP トンネルを作成する必要があります。

制約事項

- PVC とそのホスト VP トンネルの接続がアクティブになるには、PVC とそのホスト VP トンネルが同じ VPI 値を共有している必要があります。
- 次のカードは、VPI 値が 0 の VP トンネルをサポートしません。
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、マルチモード
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、シングルモード
 - 4 ポート OC-12/STM-4 ATM マルチモード ISE ラインカード、SC コネクタ搭載
 - シリーズ 4 ポート OC-12/STM-4 ATM シングルモード中距離 ISE ラインカード、SC コネクタ搭載

手順の概要

1. **configure**
2. **interface atm *interface-path-id.subinterface* point-to-point**
3. **ipv4 address *ipv4_address/prefix***
4. **pvc *vpi/vci***
5. **end**
または
commit
6. 接続の他端で ATM サブインターフェイスおよび PVC を始動するために、ステップ 1 ~ 5 を繰り返します。
7. **ping atm interface atm *interface-path-id.subinterface* *vpi/vci***
8. **show atm vp-tunnel [*interface atm interface-path-id*]**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface atm interface-path-id.subinterface point-to-point</code> 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1.10 point-to-point	新しいサブインターフェイスを作成して、そのサブインターフェイスに対する ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>ipv4 address ipv4_address/prefix</code> 例: RP/0/0/CPU0:router (config-subif)# ipv4 address 10.46.8.6/24	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 4	<code>pvc vpi/vci</code> 例: RP/0/0/CPU0:router (config-subif)# pvc 5/10	サブインターフェイスで ATM 相手先固定接続 (PVC) を作成し、「 ATM インターフェイスでの VP トンネルの作成および設定 」(P.19) で作成した VP トンネルにその ATM PVC を付加します。 <i>vpi</i> は、PVC を作成している VP トンネルの VPI に置き換えてください。 (注) PVC の VPI と VP トンネルの VCI は一致している必要があります。一致していない場合、接続はアクティブになりません。 (注) VP トンネルは、その下に PVC を作成しないと使用可能になりません。

VP トンネルの作成および設定方法

コマンドまたはアクション	目的
<p>ステップ 5</p> <pre>end または commit</pre> <p>例: RP/0/0/CPU0:router (config-subif)# end または RP/0/0/CPU0:router(config-subif)# commit </p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6</p> <p>VP トンネルの他端でサブインターフェイスと PVC を始動するために、ステップ 1 ~ 5 を繰り返します。</p>	<p>サブインターフェイスと PVC を始動します。</p>
<p>ステップ 7</p> <pre>ping atm interface atm interface-path-id.subinterface vpi/vci</pre> <p>例: RP/0/0/CPU0:router # ping atm interface atm 0/2/0/0.10 10/100 </p>	<p>ステップ 1 ~ 6 で設定した VP トンネルを介した 2 つの ATM 接続エンドポイント間の接続を確認します。</p> <ul style="list-style-type: none"> • <i>interface-path-id.subinterface</i> は、接続確認を行う VP トンネルに設定されている ATM サブインターフェイスに置き換えてください。これは、ステップ 2 で設定した <i>interface-path-id.subinterface</i> と同じです。 • <i>vci</i> は、接続確認を行う VP トンネルに設定されている PVC の VCI に置き換えてください。これは、ステップ 4 で設定した <i>vci</i> と同じです。 • <i>vpi</i> は、接続確認を行う VP トンネルに設定されている PVC の VPI に置き換えてください。これは、ステップ 4 で設定した <i>vpi</i> と同じです。
<p>ステップ 8</p> <pre>show atm vp-tunnel [interface atm interface-path-id]</pre> <p>例: RP/0/0/CPU0:router (config)# show atm vp-tunnel interface atm 0/6/0/1 </p>	<p>ルータ全体または特定の ATM インターフェイスの VP トンネル情報を表示します。</p>

次に行う作業

- VP トンネルで ATM サブインターフェイスと PVC を作成および設定するには、「[VP トンネルでの PVC を持つサブインターフェイスの作成および設定](#)」(P.22) を参照してください。
- VC クラスを設定して、ATM インターフェイスに適用するには、「[VC クラスの作成および設定](#)」(P.37) を参照してください。

レイヤ 2 接続回路 (AC) の設定方法

レイヤ 2 Attachment Circuit (AC; 接続回路) の設定作業について、次の手順で説明します。

- [レイヤ 2 ポート モード AC の作成](#)
- [レイヤ 2 ポート モード AC での オプション パラメータの設定](#)
- [PVC を持つ ATM レイヤ 2 サブインターフェイスの作成](#)
- [オプションの ATM レイヤ 2 PVC パラメータの設定](#)
- [PVP を持つ ATM レイヤ 2 サブインターフェイスの作成](#)
- [オプションの ATM レイヤ 2 PVP パラメータの設定](#)



(注)

レイヤ 2 スイッチングのためのインターフェイスの設定後は、**ipv4 address** などのルーティング コマンドは使用できません。インターフェイスにルーティング コマンドを設定すると、**l2transport** コマンドが拒否されます。

レイヤ 2 ポート モード AC の作成

ここに記載する手順では、レイヤ 2 ポート モード AC を作成します。

前提条件

レイヤ 2 ポート モード AC を作成する前に、「[ATM インターフェイスの始動](#)」(P.8) で説明するように ATM メイン インターフェイスを始動する必要があります。

制約事項

ILMI の設定は、レイヤ 2 ポート モード AC ではサポートされません。

制約事項

レイヤ 2 ポート モード AC を作成する前に、そのポートにサブインターフェイスなどの既存の設定がないことを確認する必要があります。既存の設定がある場合は、削除する必要があります。

手順の概要

1. **configure**
2. **interface atm interface-path-id**

■ レイヤ 2 接続回路 (AC) の設定方法

3. **l2transport**
4. **end**
または
commit
5. 接続の他端で ATM AC を始動するために、ステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id 例： RP/0/0/CPU0:router (config)# interface atm 0/6/0/1	ATM インターフェイスに対するインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	l2transport 例： RP/0/0/CPU0:router (config-if)# l2transport	ATM レイヤ 2 転送コンフィギュレーション モードを開始して、この ATM インターフェイスでレイヤ 2 ポート モードをイネーブルにします。
ステップ 4	end または commit 例： RP/0/0/CPU0:router (config-if-l2)# end または RP/0/0/CPU0:router(config-if-l2)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 5	接続の他端でレイヤ 2 ポートモード AC を始動するために、ステップ 1 ~ 4 を繰り返します。	レイヤ 2 ポートモード AC を始動します。 (注) 接続の両端で設定が一致している必要があります。

次に行う作業

- 作成したレイヤ 2 ポート モード AC でポイントツーポイント疑似接続 XConnect を設定するには、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「Implementing MPLS Layer 2 VPNs」モジュールを参照してください。
- ATM AC でオプションのレイヤ 2 VPN パラメータを設定するには、「レイヤ 2 ポート モード AC での オプション パラメータの設定」(P.27) を参照してください。

レイヤ 2 ポート モード AC での オプション パラメータの設定

ここに記載する手順では、レイヤ 2 ポート モード AC でオプションのレイヤ 2 VPN 転送パラメータを設定します。

前提条件

レイヤ 2 ポート モード AC でレイヤ 2 VPN パラメータを設定する前に、「レイヤ 2 ポート モード AC の作成」(P.25) で説明するようにレイヤ 2 ポート モード AC を作成する必要があります。

手順の概要

- configure**
- interface atm interface-path-id**
- atm mcpt-timers timer-1 timer-2 timer-3**
- l2transport**
- cell-packing cells timer**
- end**
または
commit
- 接続の他端でレイヤ 2 ポート モード AC を設定するために、ステップ 1 ～ 6 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1	ATM インターフェイスに対するインターフェイス コンフィギュレーション モードを開始します。

レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
ステップ 3 <code>atm mcpt-timers timer-1 timer-2 timer-3</code> 例: RP/0/0/CPU0:router (config-if)# <code>atm mcpt-timers 50 100 200</code>	インターフェイスごとに 3 つあるそれぞれの MCPT タイマーで、セル パッキングの最大タイムアウト値をマイクロ秒単位で指定します。 (注) 各タイマーのデフォルト値は 50 マイクロ秒です。
ステップ 4 <code>l2transport</code> 例: RP/0/0/CPU0:router (config-if)# <code>l2transport</code>	ATM レイヤ 2 転送コンフィギュレーション モードを開始します。
ステップ 5 <code>cell-packing cells timer</code> 例: RP/0/0/CPU0:router (config-if-l2)# <code>cell-packing 6 1</code>	パケット単位で許容されるセルの最大数を設定し、セル パッキングに使用する Maximum Cell Packing Timeout (MCPT; 最大セル パッキング タイムアウト) タイマーを指定します。 <ul style="list-style-type: none"> <code>cells</code> は、パケットあたりに使用するセルの最大数に置き換えてください。有効値の範囲は 2 ~ 86 です。 <code>timer</code> は、セル パッキングに使用する適切な MCPT タイマーを示す番号に置き換えてください。使用できる番号は、1、2、3 のいずれかです。1 つのメイン インターフェイスにつき、最大 3 種類の MCPT 値を設定することができます。
ステップ 6 <code>end</code> または commit 例: RP/0/0/CPU0:router (config-if-l2)# <code>end</code> または RP/0/0/CPU0:router (config-if-l2)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 接続の他端で AC を設定するために、ステップ 1 ~ 6 を繰り返します。	レイヤ 2 ポート モード AC を始動します。 (注) 接続の両端で設定が一致している必要があります。

PVC を持つ ATM レイヤ 2 サブインターフェイスの作成

ここに記載する手順では、PVC を持つレイヤ 2 サブインターフェイスを作成します。

前提条件

ATM インターフェイスでサブインターフェイスを作成する前に、「[ATM インターフェイスの始動 \(P.8\)](#)」で説明するように ATM インターフェイスを始動する必要があります。

制約事項

各 ATM サブインターフェイスで設定できる PVC は 1 つだけです。

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface l2transport**
3. **pvc vpi/vci**
4. **end**
または
commit
5. AC の他端で ATM サブインターフェイスおよび関連付けられている PVC を始動するためにステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config)# interface atm 0/6/0/1.10 l2transport	サブインターフェイスを作成して、そのサブインターフェイスに対する ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc vpi/vci 例: RP/0/0/CPU0:router(config-if)# pvc 5/20	ATM 相手先固定接続 (PVC) を作成して、ATM レイヤ 2 転送 PVC コンフィギュレーション モードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

■ レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
<p>ステップ 4</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router (config-atm-l2transport-pvc) # end または RP/0/0/CPU0:router (config-atm-l2transport-pvc) # commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5</p> <p>AC の他端で ATM サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 4 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) AC の両端で設定が一致している必要があります。</p>

次に行う作業

- オプションの PVC パラメータを設定するには、「[オプションの ATM レイヤ 2 PVC パラメータの設定](#)」(P.30) を参照してください。
- VC クラスを設定して、PVC に適用するには、「[ATM サブインターフェイスの PVC への VC クラスの付加](#)」(P.42) を参照してください。
- 作成した AC でポイントツーポイント疑似接続 XConnect を設定するには、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「[Implementing MPLS Layer 2 VPNs](#)」モジュールを参照してください。

オプションの ATM レイヤ 2 PVC パラメータの設定

ここでは、ATM レイヤ 2 PVC でのデフォルト設定の変更可以使用なコマンドについて説明します。

前提条件

PVC のデフォルト設定を変更する前に、「[PVC を持つ ATM レイヤ 2 サブインターフェイスの作成](#)」(P.29) で説明するようにレイヤ 2 ATM サブインターフェイスで PVC を作成する必要があります。

制約事項

PVC の両端での設定が、アクティブにする接続に合っている必要があります。

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface l2transport**
3. **pvc vpi/vci**
4. **encapsulation {aal0 | aal5}**
5. **cell-packing cells timer**
6. **shape [cbr peak_output_rate |ubr peak_output_rate | vbr-nrt peak_output_rate sustained_output_rate burst_size|vbr-rt peak_output_rate sustained_output_rate burst_size]**
7. **end**
または
commit
8. AC の他端で PVC を設定するために、ステップ 1～7 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config-if)# interface atm 0/6/0/1.10 l2transport	レイヤ 2 ATM サブインターフェイスに対する ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc vpi/vci 例: RP/0/0/CPU0:router(config-atm-l2transport-pvc)# pvc 5/20	指定した PVC に対する ATM レイヤ 2 転送 PVC コンフィギュレーション モードを開始します。
ステップ 4	encapsulation {aal0 aal5} 例: RP/0/0/CPU0:router(config-atm-l2transport-pvc)# encapsulation aal5	PVC の ATM アダプテーション レイヤ (AAL) およびカプセル化タイプを設定します。 (注) PVC のデフォルトカプセル化タイプは AAL5 です。

レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
<p>ステップ 5 <code>cell-packing cells timer</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvc)# cell-packing 5 2</p>	<p>パケット単位で許容されるセルの最大数を設定し、セルパッキングに使用する最大セルパッキングタイムアウト (MCPT) タイマーを指定します。</p> <ul style="list-style-type: none"> • <i>cells</i> は、パケットあたりに使用するセルの最大数に置き換えてください。有効値の範囲は 2 ~ 86 です。 • <i>timer</i> は、セルパッキングに使用する適切な MCPT タイマーを示す番号に置き換えてください。使用できる番号は、1、2、3 のいずれかです。1 つのメインインターフェイスにつき、最大 3 種類の MCPT 値を設定することができます。
<p>ステップ 6 <code>shape [cbr peak_output_rate ubr peak_output_rate vbr-nrt peak_output_rate sustained_output_rate burst_size vbr-rt peak_output_rate sustained_output_rate burst_size]</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvc)# shape vbr-nrt 100000 100000 8000</p>	<p>PVC の ATM トラフィックシェーピングを設定します。ATM トラフィックシェーピングを設定する前に、必要な帯域幅を概算する必要があります。</p> <ul style="list-style-type: none"> • <i>peak_output_rate</i> : トラフィックで常時使用可能な最大セルレートを設定します。 • <i>Sustained_output_rate</i> : ビットレートの持続出力レート。 • <i>burst size</i> : ビットレートのバーストセルサイズ。有効値の範囲は 1 ~ 8,192 です。
<p>ステップ 7 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvc)# end または RP/0/0/CPU0:router(config-atm-l2transport-pvc)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <p>設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。</p>
<p>ステップ 8 AC の他端で PVC を設定するために、ステップ 1 ~ 7 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) 接続の両端で設定が一致している必要があります。</p>

次に行う作業

- 作成した AC で疑似接続 XConnect を設定するには、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「Implementing MPLS Layer 2 VPNs」モジュールを参照してください。
- VC クラスを設定して、PVC に適用するには、「ATM サブインターフェイスの PVC への VC クラスの付加」(P.42) を参照してください。

PVP を持つ ATM レイヤ 2 サブインターフェイスの作成

ここに記載する手順では、ATM レイヤ 2 サブインターフェイスと、その ATM サブインターフェイスで相手先固定パス (PVP) を作成します。

前提条件

ATM インターフェイスで PVP を持つサブインターフェイスを作成する前に、「ATM インターフェイスの始動」(P.8) で説明するように ATM インターフェイスを始動する必要があります。

制約事項

- 各 L2VPN ATM AC で設定できる PVP は 1 つだけです。
- F4 OAM エミュレーションは、レイヤ 2 PVP ではサポートされません。

手順の概要

- configure**
- interface atm interface-path-id.subinterface l2transport**
- pvp vpi**
- end**
または
commit
- AC の他端で ATM サブインターフェイスおよび関連付けられている PVP を始動するためにステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config)# interface atm 0/6/0/1.10 l2transport	ATM サブインターフェイスを作成して、そのインターフェイスに対する ATM レイヤ 2 転送コンフィギュレーション モードを開始します。

■ レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
ステップ 3 <code>pvp vpi</code> 例: RP/0/0/CPU0:router(config-if)# pvp 100	(任意) ATM PVP を作成して、ATM PVP コンフィギュレーション サブモードを開始します。 (注) 各サブインターフェイスに設定できる PVP は 1 つだけです。
ステップ 4 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-atm-l2transport-pvp)# end または RP/0/0/CPU0:router(config-atm-l2transport-pvp)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 5 AC の他端で ATM サブインターフェイスおよび関連付けられている PVP を始動するために、ステップ 1 ~ 4 を繰り返します。	ATM AC を始動します。 (注) AC 接続の両端で設定が一致している必要があります。

次に行う作業

- オプションの PVP パラメータを設定するには、「[オプションの ATM レイヤ 2 PVP パラメータの設定](#)」(P.34) を参照してください。
- 作成した AC でポイントツーポイント疑似接続 XConnect を設定するには、『*Cisco IOS XR Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs*」モジュールを参照してください。

オプションの ATM レイヤ 2 PVP パラメータの設定

ここでは、ATM レイヤ 2 PVP でのデフォルト設定の変更を使用できるコマンドについて説明します。

前提条件

PVP のデフォルト設定を変更する前に、「[PVP を持つ ATM レイヤ 2 サブインターフェイスの作成](#)」(P.33) で説明するように ATM サブインターフェイスで PVP を作成する必要があります。

制約事項

- 次のカードは、VPI 値が 0 の VP トンネルをサポートしません。
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、マルチモード
 - 4 ポート OC-3c/STM-1c ATM ISE ラインカード、シングルモード
 - 4 ポート OC-12/STM-4 ATM マルチモード ISE ラインカード、SC コネクタ搭載
 - シリーズ 4 ポート OC-12/STM-4 ATM シングルモード、中距離 ISE ラインカード、SC コネクタ搭載

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface l2transport**
3. **pvp vpi**
4. **cell-packing cells timer**
5. **shape [cbr peak_output_rate | ubr peak_output_rate | vbr-nrt peak_output_rate sustained_output_rate burst_size | vbr-rt peak_output_rate sustained_output_rate burst_size]**
6. **end**
または
commit
7. 接続の他端で PVP を設定するために、ステップ 1 ~ 6 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config)# interface atm 0/6/0/1.10 l2transport	ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvp vpi 例: RP/0/0/CPU0:router(config-if)# pvp 10	PVP に対するサブインターフェイス コンフィギュレーション モードを開始します。

レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
<p>ステップ 4 <code>cell-packing cells timer</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvp)# cell-packing 5 2</p>	<p>パケット単位で許容されるセルの最大数を設定し、セルパッキングに使用する最大セルパッキングタイムアウト (MCPT) タイマーを指定します。</p> <ul style="list-style-type: none"> • <i>cells</i> は、パケットあたりに使用するセルの最大数に置き換えてください。有効値の範囲は 2 ~ 86 です。 • <i>timer</i> は、セルパッキングに使用する適切な MCPT タイマーを示す番号に置き換えてください。使用できる番号は、1、2、3 のいずれかです。1 つのメインインターフェイスにつき、最大 3 種類の MCPT 値を設定することができます。
<p>ステップ 5 <code>shape [cbr peak_output_rate ubr peak_output_rate vbr-nrt peak_output_rate sustained_output_rate burst_size vbr-rt peak_output_rate sustained_output_rate burst_size]</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvp)# shape vbr-nrt 100000 100000 8000</p>	<p>PVC の ATM トラフィックシェーピングを設定します。ATM トラフィックシェーピングを設定する前に、必要な帯域幅を概算する必要があります。</p> <ul style="list-style-type: none"> • <i>peak_output_rate</i> : トラフィックで常時使用可能な最大セルレートを設定します。 • <i>Sustained_output_rate</i> : ビットレートの持続出力レート。 • <i>burst size</i> : ビットレートのバーストセルサイズ。有効値の範囲は 1 ~ 8,192 です。
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-atm-l2transport-pvp)# end または RP/0/0/CPU0:router(config-atm-l2transport-pvp)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <p>設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。</p>
<p>ステップ 7 AC の他端で PVP を設定するために、ステップ 1 ~ 6 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) AC 接続の両端で設定が一致している必要があります。</p>

次に行う作業

- 作成した AC でポイントツーポイント疑似接続 XConnect を設定するには、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「Implementing MPLS Layer 2 VPNs」モジュールを参照してください。

VC クラスの作成および設定方法

ATM VC クラスの作成および設定作業について、次の手順で説明します。

- 「VC クラスの作成および設定」(P.37)
- 「ポイントツーポイント ATM メイン インターフェイスへの VC クラスの付加」(P.40)
- 「ポイントツーポイント ATM サブインターフェイスへの VC クラスの付加」(P.41)
- 「ATM サブインターフェイスの PVC への VC クラスの付加」(P.42)

VC クラスの作成および設定

ここでは、仮想回線 (VC) クラスの作成と、ATM メイン インターフェイス、サブインターフェイス、相手先固定接続 (PVC) への VC クラスの付加に必要なタスクとコマンドについて説明します。

制約事項

レイヤ 2 VPN AC 設定では、VC クラスは PVC のみに適用可能です。VC クラスは、レイヤ 2 ポートモード インターフェイスや PVP ではサポートされません。

手順の概要

- configure**
- vc-class atm name**
- encapsulation {aal5mux ipv4 | aal5nlpid | aal5snap}**
- oam ais-rdi [down-count [up-count]]**
- oam retry [up-count [down-count [retry-frequency]]]**
- oam-pvc manage seconds**
- shape [cbr peak_output_rate |ubr peak_output_rate |vbr-nrt peak_output_rate sustained_output_rate burst_size|vbr-rt peak_output_rate sustained_output_rate burst_size]**
- end**
または
commit

VC クラスの作成および設定方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>vc-class atm name</code> 例： RP/0/0/CPU0:router (config)# vc-class atm class1	ATM インターフェイスの VC クラスを作成して、VC クラス コンフィギュレーション モードを開始します。
ステップ 3	<code>encapsulation {aal5mux ipv4 aal5nlpid aal5snap}</code> 例： RP/0/0/CPU0:router (config-vc-class-atm)# encapsulation aal5snap	ATM VC クラスの ATM アダプテーション レイヤ (AAL) およびカプセル化タイプを設定します。 (注) VC クラスのデフォルト カプセル化タイプは AAL5/SNAP です。 (注) VC クラスでは、 encapsulation コマンドはレイヤ 3 ポイントツーポイント設定のみに適用されます。
ステップ 4	<code>oam ais-rdi [down-count [up-count]]</code> 例： RP/0/0/CPU0:router (config-vc-class-atm)# oam ais-rdi 25 5	関連付けられている PVC で指定数の OAM アラーム表示信号/リモート障害表示 (AIS/RDI) セルが受信された後に VC クラスが停止するように VC クラスを設定します。 (注) VC クラスでは、 oam ais-rdi コマンドはレイヤ 3 ポイントツーポイント設定のみに適用されます。
ステップ 5	<code>oam retry [up-count [down-count [retry-frequency]]]</code> 例： RP/0/0/CPU0:router (config-vc-class-atm)# oam retry 5 10 5	OAM 管理に関連するパラメータを設定します。 (注) VC クラスでは、 oam retry コマンドはレイヤ 3 ポイントツーポイント設定のみに適用されます。
ステップ 6	<code>oam-pvc manage seconds</code> 例： RP/0/0/CPU0:router (config-vc-class-atm)# oam-pvc manage 300	ATM OAM F5 ループバック周波数を設定します。 (注) VC クラスでは、 oam-pvc manage コマンドはレイヤ 3 ポイントツーポイント設定のみに適用されます。

コマンドまたはアクション	目的
<p>ステップ 7 <code>shape [cbr peak_output_rate ubr peak_output_rate vbr-nrt peak_output_rate sustained_output_rate burst_size vbr-rt peak_output_rate sustained_output_rate burst_size]</code></p> <p>例: RP/0/0/CPU0:router (config-vc-class-atm)# shape vbr-nrt 100000 100000 8000</p>	<p>PVC の ATM トラフィック シェーピングを設定します。</p> <p>ATM トラフィック シェーピングを設定する前に、必要な帯域幅を概算する必要があります。</p> <ul style="list-style-type: none"> • <i>peak_output_rate</i> : トラフィックで常時使用可能な最大セル レートを設定します。 • <i>Sustained_output_rate</i> : ビット レートの持続出力レート。 • <i>burst size</i> : ビット レートのバーストセル サイズ。有効値の範囲は 1 ~ 8,192 です。
<p>ステップ 8 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router (config-if)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

次に行う作業

VC クラスを ATM メイン インターフェイス、サブインターフェイス、または PVC に付加します。

- VC クラスを ATM メイン インターフェイスに付加するには、「ポイントツーポイント ATM メイン インターフェイスへの VC クラスの付加」(P.-40) を参照してください。
- VC クラスを ATM サブインターフェイスに付加するには、「ポイントツーポイント ATM サブインターフェイスへの VC クラスの付加」(P.-41) を参照してください。
- VC クラスを ATM PVC に付加するには、「ATM サブインターフェイスの PVC への VC クラスの付加」(P.-42) を参照してください。

ポイントツーポイント ATM メイン インターフェイスへの VC クラスの付加

ここでは、ポイントツーポイント ATM メイン インターフェイスへの VC クラスの付加に必要なタスクとコマンドについて説明します。

制約事項

VC クラスは、レイヤ 2 ポート モード AC には適用されません。レイヤ 2 VPN 設定では、VC クラスは PVC のみに適用されます。

手順の概要

1. **configure**
2. **interface atm *interface-path-id* point-to-point**
3. **class-int *vc-class-name***
4. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm <i>interface-path-id</i> point-to-point 例： RP/0/0/CPU0:router (config)# interface atm 0/6/0/1 point-to-point	ATM インターフェイス コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 3 <code>class-int vc-class-name</code> 例: RP/0/0/CPU0:router (config-if)# class-int classA	VC クラスを ATM メイン インターフェイスに付加します。 <code>vc-class-name</code> 引数は、「 VC クラスの作成および設定 (P.37) 」で設定した VC クラスの名前に置き換えてください。
ステップ 4 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

ポイントツーポイント ATM サブインターフェイスへの VC クラスの付加

ここでは、ATM サブインターフェイスへの VC クラスの付加に必要なタスクとコマンドについて説明します。

手順の概要

1. **configure**
2. **interface atm interface-path-id.subinterface point-to-point**
3. **class-int vc-class-name**
4. **end**
または
commit

VC クラスの作成および設定方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface atm interface-path-id.subinterface point-to-point</code> 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1.10 point-to-point	ATM サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>class-int vc-class-name</code> 例: RP/0/0/CPU0:router (config-subif)# class-int classA	VC クラスを ATM サブインターフェイスに割り当てます。 <i>vc-class-name</i> 引数は、「VC クラスの作成および設定」(P.-37) で設定した VC クラスの名前に置き換えてください。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router (config-subif)# end または RP/0/0/CPU0:router(config-subif)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

ATM サブインターフェイスの PVC への VC クラスの付加

ここでは、ATM サブインターフェイスの PVC への VC クラスの付加に必要なタスクとコマンドについて説明します。



(注)

VC クラスは、ポイントツーポイントおよびレイヤ 2 PVC でサポートされます。

手順の概要

1. **configure**
2. **interface atm interface-path-id[.subinterface] [point-to-point | l2transport]**
3. **pvc vpi/vci**
4. **class vc vc-class-name**
5. **end**
または
commit

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface atm interface-path-id.subinterface [point-to-point l2transport]</code> 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1.10	サブインターフェイス コンフィギュレーション モードを開始して、既存の ATM サブインターフェイス がない場合には作成します。 VC クラスをポイントツーポイント サブインターフェイス に付加する場合は、 point-to-point キーワードを使用します。VC クラスをレイヤ 2 転送サブインターフェイス に付加する場合は、 l2transport キーワードを使用します。 (注) ATM サブインターフェイスの作成および設定の詳細については、「 PVC を持つポイントツーポイント ATM サブインターフェイスの作成 」(P.13) を参照してください。
ステップ 3 <code>pvc vpi/vci</code> 例: RP/0/0/CPU0:router (config-if)# pvc 5/50	ATM PVC コンフィギュレーション モードを開始して、既存の PVC がない場合には作成します。 (注) ATM サブインターフェイスでの PVC の作成および設定の詳細については、「 PVC を持つポイントツーポイント ATM サブインターフェイスの作成 」(P.13) を参照してください。

ATM インターフェイスでの ILMI の設定方法

コマンドまたはアクション	目的
ステップ 4 <code>class-vc vc-class-name</code> 例: RP/0/0/CPU0:router (config-atm-vc)# class-vc classA	VC クラスを ATM PVC に割り当てます。 <i>vc-class-name</i> 引数は、PVC に付加する VC クラスの名前に置き換えてください。
ステップ 5 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router (config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

ATM インターフェイスでの ILMI の設定方法

ATM インターフェイスでの ILMI 管理のための設定作業について、次の手順で説明します。

- 「[ATM インターフェイスでの ILMI のイネーブル化](#)」(P.44)
- 「[ATM インターフェイスでの ILMI のディセーブル化](#)」(P.46)

ATM インターフェイスでの ILMI のイネーブル化

ここでは、ILMI 用 ATM インターフェイス の設定に使用できるコマンドについて説明します。



(注)

ILMI では、PVC は ATM メイン インターフェイスで直接設定されます。ILMI に使用される ATM インターフェイスでは、サブインターフェイスの設定は必要ありません。

前提条件

「[ATM インターフェイスの始動](#)」(P.8) で説明するように、ATM インターフェイスを始動し、shutdown 設定を削除する必要があります。

制約事項

- アクティブにするインターフェイスについて、ATM ILMI 接続の両端での設定が一致している必要があります。
- ILMI の設定は、レイヤ 2 ポート モード AC ではサポートされません。

手順の概要

1. **configure**
2. **interface atm interface-path-id**
3. **atm address-registration**
4. **atm ilmi-keepalive [act-poll-freq frequency] [retries count] [inact-poll-freq frequency]**
5. **pvc vpi/vci ilmi**
6. **end**
または
commit
7. **exit**
8. **exit**
9. **show atm ilmi-status [atm interface-path-id]**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1	ATM インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	atm address-registration 例: RP/0/0/CPU0:router (config-if)# atm address-registration	(任意) Interim Local Management Interface (ILMI; 暫定ローカル管理インターフェイス) とのアドレス登録およびコールバック機能を実行するために、ルータをイネーブルにします。
ステップ 4	atm ilmi-keepalive [act-poll-freq frequency] [retries count] [inact-poll-freq frequency] 例: RP/0/0/CPU0:router (config-if)# atm ilmi-keepalive	(任意) ATM インターフェイスで ILMI キープアライブをイネーブルにします。
ステップ 5	pvc vpi/vci ilmi 例: RP/0/0/CPU0:router (config-if)# pvc 5/30 ilmi	ILMI カプセル化との ATM 相手先固定接続 (PVC) を作成します。

ATM インターフェイスでの ILMI の設定方法

コマンドまたはアクション	目的
ステップ 6 end または commit 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 7 exit 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 8 exit 例: RP/0/0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ 9 show atm ilmi-status [atm interface-path-id] 例: RP/0/0/CPU0:router (config)# show atm ilmi-status atm 0/6/0/1	(任意) 指定のインターフェイスの ILMI 設定を確認します。

ATM インターフェイスでの ILMI のディセーブル化

ここでは、ATM インターフェイスでの ILMI のディセーブル化に使用できるコマンドについて説明します。

手順の概要

1. **configure**
2. **interface atm interface-path-id**
3. **atm ilmi-config disable**

4. **end**
または
commit
5. **exit**
6. **exit**
7. **show atm ilmi-status [atm interface-path-id]**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface atm interface-path-id 例: RP/0/0/CPU0:router (config)# interface atm 0/6/0/1	ATM インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	atm ilmi-config disable 例: RP/0/0/CPU0:router (config-if)# atm ilmi-config disable	(任意) ATM インターフェイスで ILMI をディセーブルに します。 ATM インターフェイスで ILMI を再びイネーブルにするに は、このコマンドの no atm ilmi-config disable 形式を使用 します。
ステップ 4	end または commit 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	<code>exit</code> 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 6	<code>exit</code> 例: RP/0/0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ 7	<code>show atm ilmi-status [atm interface-path-id]</code> 例: RP/0/0/CPU0:router (config)# show atm ilmi-status atm 0/6/0/1	(任意) 指定のインターフェイスの ILMI 設定を確認します。

チャネライズド ATM の設定方法

ここでは、1 つの T3 パスを、ATM トラフィックを伝送する複数の T1 チャネルに設定する手順について説明します。

前提条件

リリース 3.7.0 のチャネライズド ATM には、次の前提条件が適用されます。

- ルータに Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタがインストールされている必要があります。
- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

リリース 3.7.0 のチャネライズド ATM には、次の制約事項が適用されます。

- チャネライズド ATM は、Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタでのみサポートされます。
- ATM T3 パスは、T1 ATM チャネルまたは Virtual Path (VP; 仮想パス) トンネルのみにチャネル化できます。
- DS0 はサポートされません。

手順の概要

1. `configure`
2. `hw-module subslot subslot-id cardtype {t3 | e3}`
3. `controller t3 interface-path-id`
4. `interface atm interface-path-id`
5. `mode mode`

6. **controller t1 interface-path-id**
7. **mode mode**
8. **interface atm interface-path-id.subinterface point-to-point**
9. **pvc vpi/vci**
10. **ipv4 address ipv4_address/prefix**
11. **end**
または
commit

詳細手順

コマンドまたはアクション	目的
ステップ 1 configure 例 : RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 hw-module subslot subslot-id cardtype {t3 e3} 例 : RP/0/0/CPU0:router(config)# hw-module subslot 0/1/0 cardtype t3	SPA のカード タイプを設定します。 <ul style="list-style-type: none"> • t3 : B3ZS コーディングを使用するネットワークでの 44,210 Kbps の T3 接続を指定します。これがデフォルトの設定です。 • e3 : 主に欧州で使用されているデータ転送レート 34,010 Kbps の広域デジタル転送方式を指定します。
ステップ 3 controller t3 interface-path-id 例 : RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	T3 コントローラを作成して、T3 コントローラ コンフィギュレーション モードを開始します。T3 コントローラの interface-path-id の ID を rack/slot/module/port 表記で指定します。
ステップ 4 mode mode 例 : RP/0/0/CPU0:router(config-t3)# mode t1	インターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • atm : ATM を伝送するクリア チャネル • e1 : 21 個の E1 にチャンネル化 • serial : hdlc に類似するペイロードを伝送するクリア チャネル • t1 : 28 個の T1 にチャンネル化
ステップ 5 controller t1 interface-path-id 例 : RP/0/0/CPU0:router(config-t3)# controller t1 0/1/0/0	T1 コントローラを作成して、T1 コントローラ コンフィギュレーション サブモードを開始します。T1 コントローラの interface-path-id を rack/slot/module/port 表記で指定します。
ステップ 6 mode mode 例 : RP/0/0/CPU0:router(config-t1)# mode atm	インターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • atm : ATM を伝送するクリア チャネル • e1 : 21 個の E1 にチャンネル化 • serial : hdlc に類似するペイロードを伝送するクリア チャネル • t1 : 28 個の T1 にチャンネル化

コマンドまたはアクション	目的
ステップ 7 <code>interface atm interface-path-id</code> 例: RP/0/0/CPU0:router(config-t1)# interface atm 0/1/0/0	ATM インターフェイスを作成して、ATM インターフェイス コンフィギュレーション モードを開始します。ATM インターフェイスを <i>rack/slot/module/port</i> 表記で指定します。
ステップ 8 <code>interface atm interface-path-id.subinterface point-to-point</code> 例: RP/0/0/CPU0:router(config-if)# interface atm 0/1/0/1.1 point-to-point	ポイントツーポイント リンクの一方のエンドポイントとして ATM サブインターフェイスを作成して、ATM サブインターフェイス コンフィギュレーション モードを開始します。ATM インターフェイスを <i>rack/slot/module/port.subinterface</i> 表記で指定します。
ステップ 9 <code>pvc vpi/vci</code> 例: RP/0/0/CPU0:router(config-subif)# pvc 10/100	ATM 相手先固定接続 (PVC) を作成して、ATM PVC コンフィギュレーション サブモードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。
ステップ 10 <code>ipv4 address ipv4_address/prefix</code> 例: RP/0/0/CPU0:router(config-atm-vc)# ipv4 address 10.212.4.22 255.255.255.0	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 11 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

仮想パス (VP) トンネルを持つクリア チャネル ATM の設定方法

ここでは、複数の Virtual Path (VP; 仮想パス) トンネルを持つ T3 ATM パスの設定手順について説明します。

前提条件

リリース 3.7.0 のチャネライズド ATM には、次の前提条件が適用されます。

- ルータに Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタがインストールされている必要があります。
- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

リリース 3.7.0 のチャネライズド ATM には、次の制約事項が適用されます。

- チャネライズド ATM は、Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタでのみサポートされます。
- ATM T3 パスは、T1 ATM チャネルまたは仮想パス (VP) トンネルのみにチャネル化できます。
- DS0 はサポートされません。

手順の概要

1. **configure**
2. **hw-module subslot *subslot-id* cardtype {t3 | e3}**
3. **controller t3 *interface-path-id***
4. **mode *mode***
5. **interface atm *interface-path-id***
6. **vp-tunnel *vpi***
7. **interface atm *interface-path-id.subinterface* point-to-point**
8. **pvc *vpi/vci***
9. **ipv4 address *ipv4_address/prefix***
10. **interface atm *interface-path-id.subinterface* point-to-point**
11. **pvc *vpi/vci***
12. **ipv4 address *ipv4_address/prefix***
13. **end**
または
commit

■ 仮想パス (VP) トンネルを持つクリア チャネル ATM の設定方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>hw-module subslot subslot-id cardtype</code> {t3 e3} 例: RP/0/0/CPU0:router(config)# hw-module subslot 0/1/0 cardtype t3	SPA のカード タイプを設定します。 <ul style="list-style-type: none"> • t3 : B3ZS コーディングを使用するネットワークでの 44,210 Kbps の T3 接続を指定します。これがデフォルトの設定です。 • e3 : 主に欧州で使用されているデータ転送レート 34,010 Kbps の広域デジタル転送方式を指定します。
ステップ 3	<code>controller t3 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	T3 コントローラを作成して、T3 コントローラ コンフィギュレーション モードを開始します。T3 コントローラの <i>interface-path-id</i> を <i>rack/slot/module/port</i> 表記で指定します。
ステップ 4	<code>mode mode</code> 例: RP/0/0/CPU0:router(config-t3)# mode t1	インターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • atm : ATM を伝送するクリア チャネル • e1 : 21 個の E1 にチャンネル化 • serial : hdlc に類似するペイロードを伝送するクリア チャネル • t1 : 28 個の T1 にチャンネル化
ステップ 5	<code>interface atm interface-path-id</code> 例: RP/0/0/CPU0:router(config-t1)# interface atm 0/1/0/0	ATM インターフェイスを作成して、ATM インターフェイス コンフィギュレーション モードを開始します。ATM インターフェイスを <i>rack/slot/module/port</i> 表記で指定します。
ステップ 6	<code>vp-tunnel vpi</code> 例: RP/0/0/CPU0:router (config)# vp-tunnel 10	ATM インターフェイスで VP トンネルを設定します。
ステップ 7	<code>interface atm</code> <i>interface-path-id.subinterface</i> <code>point-to-point</code> 例: RP/0/0/CPU0:router(config-if)# interface atm 0/1/0/1.1 point-to-point	ポイントツーポイント リンクの一方のエンドポイントとして ATM サブインターフェイスを作成して、ATM サブインターフェイス コンフィギュレーション モードを開始します。ATM インターフェイスを <i>rack/slot/module/port.subinterface</i> 表記で指定します。
ステップ 8	<code>pvc vpi/vci</code> 例: RP/0/0/CPU0:router(config-subif)# pvc 10/100	ATM 相手先固定接続 (PVC) を作成して、ATM PVC コンフィギュレーション サブモードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

コマンドまたはアクション	目的
ステップ 9 <code>ipv4 address ipv4_address/prefix</code> 例: RP/0/0/CPU0:router(config-atm-vc)#ipv4 address 10.212.8.22 255.255.255.0	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 10 <code>interface atm</code> <code>interface-path-id.subinterface</code> <code>point-to-point</code> 例: RP/0/0/CPU0:router(config-if)# interface atm 0/1/0/1.2 point-to-point	ポイントツーポイント リンクの一方のエンドポイントとして ATM サブインターフェイスを作成して、ATM サブインターフェイス コンフィギュレーション モードを開始します。ATM インターフェイスを <code>rack/slot/module/port.subinterface</code> 表記で指定します。
ステップ 11 <code>pvc vpi/vci</code> 例: RP/0/0/CPU0:router(config-subif)# pvc 10/200	ATM PVC を作成して、ATM PVC コンフィギュレーション サブモードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。
ステップ 12 <code>ipv4 address ipv4_address/prefix</code> 例: RP/0/0/CPU0:router(config-atm-vc)#ipv4 address 10.212.12.22 255.255.255.0	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 13 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

Cisco IOS XR ソフトウェアでの ATM の設定例

ここでは、次の設定例について説明します。

- 「[ATM インターフェイスの始動と設定：例](#)」(P.54)
- 「[ポイントツーポイント ATM サブインターフェイスの設定：例](#)」(P.54)

- 「VP トンネル設定 : 例」 (P.56)
- 「レイヤ 2 AC の作成および設定 : 例」 (P.56)
- 「VC クラスの作成および設定 : 例」 (P.57)
- 「チャネライズド ATM の設定 : 例」 (P.58)
- 「仮想パス (VP) トンネルを持つクリア チャネル ATM の設定 : 例」 (P.58)

ATM インターフェイスの始動と設定 : 例

次に、ATM インターフェイスを始動して設定する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface atm 0/6/0/0
RP/0/0/CPU0:router(config-if)# atm address-registration
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# commit
```

ポイントツーポイント ATM サブインターフェイスの設定 : 例

次に、ATM メイン インターフェイスでポイントツーポイント ATM サブインターフェイスを設定する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router (config)# interface atm 0/2/0/2.1 point-to-point
RP/0/0/CPU0:router (config-if)# ipv4 address 10.46.8.6/24
RP/0/0/CPU0:router (config-if)# pvc 0/200
RP/0/0/CPU0:router (config-atm-vc)# commit
RP/0/0/CPU0:router (config-atm-vc)# exit
RP/0/0/CPU0:router (config-if)# exit
RP/0/0/CPU0:router (config)# exit
```

```
RP/0/0/CPU0:router # show interfaces atm 0/2/0/2.1
```

```
ATM0/2/0/2.1 is up, line protocol is up
Hardware is ATM network sub-interface(s)
Description: Connect to P4_C12810 ATM 1/2.1
Internet address is 10.46.8.6/24
MTU 4470 bytes, BW 155000 Kbit
    reliability Unknown, txload Unknown, rxload Unknown
Encapsulation AAL5/SNAP, controller loopback not set,
Last clearing of "show interface" counters Unknown
Datarate information unavailable.
Interface counters unavailable.
```

```
RP/0/0/CPU0:router # show atm interface atm 0/2/0/3
```

```
Interface                : ATM0/2/0/3
AAL Enabled              : AAL5
Max-VP                   : 254
Max-VC                   : 2046
Configured L2 PVPs      : 0
Configured L2 PVCs      : 0
Configured L3 VP-Tunnels : 0
Configured L3 PVCs      : 1
L2 PVPs in Down State   : 0
L2 PVCs in Down State   : 0
L3 VP-Tunnels in Down State : 0
L3 PVCs in Down State   : 0
```



```
Cell packing count                : 0

Received Side Statistics:
  Received Cells                   : 0
  Received Bytes                   : 0
  Received AAL Packets             : 0

Receive Side Cells Dropped:
  Unrecognized VPI/VCI            : 0

Receive Side AAL5 Packets Dropped:
  Unavailable SAR Buffer           : 0
  Non-Resource Exhaustion         : 0
  Reassembly Timeout              : 0
  Zero Length                     : 0
  Unavailable Host Buffer          : 0
  Packet size exceeds MPS         : 0
  AAL5 Trailer Length Errors      : 0

Transmit Side Statistics:
  Transmitted Cells               : 1899716067
  Transmitted Bytes               : 0
  Transmitted AAL Packets         : 0

Transmit Side Cells Dropped:
  Unrecognized VPI/VCI           : 0

Transmit Side AAL5 Packets Dropped:
  Unavailable SAR Buffer           : 0
  Non-Resource Exhaustion         : 0
  WRED Threshold                  : 0
  WRED Random                     : 0

RP/0/0/CPU0:router # show atm pvc 10/100

Detailed display of VC(s) with VPI/VCI = 10/100

ATM0/2/0/3.100: VPI: 10 VCI: 100
UBR, PeakRate: 622000 Kbps
AAL5-LLC/SNAP
OAM frequency: 10 second(s), OAM retry frequency: 1 second(s),
OAM up retry count: 3, OAM down retry count: 5,
OAM Keep-vc-up: False, OAM AIS-RDI failure: None,
OAM AIS-RDI down count: 1, OAM AIS-RDI up time: 3 second(s),
OAM Loopback status: No loopback enabled,
OAM VC state: Loopback Not verified,
VC is not managed by OAM,

OAM cells received: 0,
F5 InEndLoop: 0, F5 InSegLoop: 0,
F5 InEndAIS: 0, F5 InSegAIS: 0,
F5 InEndRDI: 0, F5 InSegRDI: 0,
OAM cells sent: 0,
F5 OutEndLoop: 0, F5 OutSegLoop: 0,
F5 OutEndAIS: 0, F5 OutSegAIS: 0,
F5 OutEndRDI: 0, F5 OutSegRDI: 0,
OAM cells drops: 0

InPkts: 0                      OutPkts: 0
InBytes: 0                      OutBytes: 0
WRED pkt drop: 0
Non WRED pkt drop: 0

Internal state: READY
```

```
Status: UP
```

VP トンネル設定 : 例

次に、ATM メイン インターフェイスで VP トンネルの一方のエンドポイントを設定する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface atm 0/6/0/0
RP/0/0/CPU0:router(config-if)# vp-tunnel 10
RP/0/0/CPU0:router(config-atm-vp-tunnel)# shape cbr 150000
RP/0/0/CPU0:router(config-atm-vp-tunnel)# f4oam disable
RP/0/0/CPU0:router(config-atm-vp-tunnel)# commit
RP/0/0/CPU0:router(config-atm-vp-tunnel)# exit
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config)# exit
RP/0/0/CPU0:router# show atm vp-tunnel
```

Interface	VPI	SC	Data VCs	Peak Kbps	Avg/Min Kbps	Burst Cells	Status
ATM0/2/0/3	30	UBR	2	155000	N/A	N/A	UP

次に、VP トンネルの一方のエンドポイントで ATM サブインターフェイスと PVC を作成して設定し、その VP トンネルの接続を確認する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface atm 0/6/0/0.16 point-to-point
RP/0/0/CPU0:router(config-subif)# pvc 10/100
RP/0/0/CPU0:router (config-atm-vc)# commit
RP/0/0/CPU0:router (config-subif)# exit
RP/0/0/CPU0:router (config)# exit
RP/0/0/CPU0:router # ping atm interface atm 0/6/0/0.16 10/100
```

Sending 5, 53-byte end-to-end OAM echos, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

レイヤ 2 AC の作成および設定 : 例

次に、レイヤ 2 ポート モード AC の一方のエンドポイントを作成して設定する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface atm 0/6/0/1
RP/0/0/CPU0:router (config-if)# l2transport
RP/0/0/CPU0:router (config-if-l2)# cell-packing 6 1
RP/0/0/CPU0:router(config-if-l2)# commit
```

次に、PVC を持つレイヤ 2 サブインターフェイスで AC を作成して設定する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface atm 0/1/0/0.230 l2transport
RP/0/0/CPU0:router(config-if)# pvc 15/230
RP/0/0/CPU0:router(config-atm-l2transport-pvc)# encapsulation aal0
RP/0/0/CPU0:router(config-atm-l2transport-pvc)# cell-packing 5 2
RP/0/0/CPU0:router(config-atm-l2transport-pvc)# shape cbr 622000
RP/0/0/CPU0:router(config-atm-l2transport-pvc)# commit
RP/0/0/CPU0:router(config-atm-l2transport-pvc)#
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config)# exit
RP/0/0/CPU0:router# show atm pvc
```

Interface	VPI	VCI	Type	Encaps	SC	Peak Kbps	Avg/Min Kbps	Burst Cells	Sts
ATM0/1/0/0.230	15	230	PVC	AAL0	UBR	622000	N/A	N/A	UP
ATM0/1/0/3.19	17	19	PVC	SNAP	UBR	622000	N/A	N/A	UP

次に、PVP を持つ ATM サブインターフェイスで AC を作成して設定する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface atm 0/6/0/1.10 l2transport
RP/0/0/CPU0:router(config-if)# pvp 100
RP/0/0/CPU0:router(config-atm-l2transport-pvp)# cell-packing 5 2
RP/0/0/CPU0:router(config-atm-l2transport-pvp)# shape ubr 155000
RP/0/0/CPU0:router(config-atm-l2transport-pvp)# commit
```

```
RP/0/0/CPU0:router# show atm pvp interface atm 0/6/0/1
```

Interface	VPI	SC	Peak Kbps	Avg/Min Kbps	Burst Cells	Sts
ATM0/6/0/1.10	100	UBR	155000	N/A	N/A	UP

VC クラスの作成および設定 : 例

次に、VC クラスを設定する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# vc-class atm atm-class-1
RP/0/0/CPU0:router(config-vc-class-atm)# encapsulation aal5snap
RP/0/0/CPU0:router(config-vc-class-atm)# oam ais-rdi 25 5
RP/0/0/CPU0:router(config-vc-class-atm)# oam retry 5 10 5
RP/0/0/CPU0:router(config-vc-class-atm)# oam-pvc manage 300
RP/0/0/CPU0:router(config-vc-class-atm)# shape cbr 100000
RP/0/0/CPU0:router(config-vc-class-atm)# commit
```

次に、VC クラスを ATM メイン インターフェイスに付加する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface ATM0/2/0/0.1 point-to-point
RP/0/0/CPU0:router (config-if)# class-int atm-class-1
RP/0/0/CPU0:router (config-if)# commit
```

次に、VC クラスを ATM サブインターフェイスに付加する例を示します。

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface ATM0/2/0/0.1 point-to-point
RP/0/0/CPU0:router(config-if)# pvc 10/100
RP/0/0/CPU0:router (config-atm-vc)# class-vc atm-class-1
RP/0/0/CPU0:router (config-atm-vc)# commit
```

次に、特定の ATM VC クラスに関する情報を表示する例を示します。

```
RP/0/0/CPU0:router # show atm vc-class atm-class-1
ATM vc-class atm-class-1
```

```
encapsulation      - aal5snap
shape               - cbr 100000
oam ais-rdi        - not configured
oam retry          - not configured
oam-pvc            - manage 300
```

次に、特定の PVC に関連付けられている Virtual Circuit (VC; 仮想回線) パラメータについての設定情報を表示する例を示します。

```
RP/0/0/CPU0:router # show atm class-link 10/100

Detailed display of VC(s) with VPI/VCI = 10/100

Class link for VC 10/100
ATM0/2/0/0.1: VPI: 10 VCI: 100
shape : cbr 100000 (VC-class configured on VC)
encapsulation : aal5snap (VC-class configured on VC)
oam-pvc : manage 300 (VC-class configured on VC)
oam retry : 3 5 1 (Default value)
oam ais-rdi : 1 3 (Default value)
```

チャネライズド ATM の設定 : 例

次に、1つの T3 パスを、ATM トラフィックを伝送する複数の T1 チャンネルに設定する例を示します。

```
RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# hw-module subslot 0/4/0 cardtype t3
RP/0/0/CPU0:router(config)# controller T3 0/4/0/0
RP/0/0/CPU0:router(config-t3)# mode t1
RP/0/0/CPU0:router(config-t3)# controller T1 0/4/0/0/1
RP/0/0/CPU0:router(config-t1)# mode atm
RP/0/0/CPU0:router(config-t1)# interface ATM 0/4/0/0/1
RP/0/0/CPU0:router(config-if)# interface ATM 0/4/0/0/1.1 point-to-point
RP/0/0/CPU0:router(config-subif)# pvc 10/100
RP/0/0/CPU0:router(config-atm-vc)# ipv4 address 10.212.4.22 255.255.255.0
RP/0/0/CPU0:router(config-subif)# commit
```

仮想パス (VP) トンネルを持つクリア チャンネル ATM の設定 : 例

次に、複数の仮想パス (VP) トンネルを持つ T3 ATM パスを設定する例を示します。

```
RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# hw-module subslot 0/4/0 cardtype t3
RP/0/0/CPU0:router(config)# controller T3 0/4/0/1
RP/0/0/CPU0:router(config-t3)# mode atm
RP/0/0/CPU0:router(config-t3)# interface ATM 0/4/0/1
RP/0/0/CPU0:router(config-if)# vp-tunnel 10
RP/0/0/CPU0:router(config-atm-vp-tunnel)# interface ATM 0/4/0/1.1 point-to-
RP/0/0/CPU0:router(config-subif)# pvc 10/100
RP/0/0/CPU0:router(config-atm-vc)# ipv4 address 10.212.8.22 255.255.255.0
RP/0/0/CPU0:router(config-subif)# interface ATM 0/4/0/1.2 point-to-point
RP/0/0/CPU0:router(config-subif)# pvc 10/200
RP/0/0/CPU0:router(config-atm-vc)# ipv4 address 10.212.12.22 255.255.255.0
RP/0/0/CPU0:router(config-subif)# commit
```

その他の参考資料

ここでは、Cisco IOS XR ソフトウェアでの ATM の実装に関する参考資料について説明します。

関連資料

内容	参照先
ATM コマンド: コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト、使用上のガイドライン、例	『Cisco IOS XR Interface and Hardware Command Reference』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC 1483	『Multiprotocol Encapsulation over ATM Adaptation Layer 5』
RFC 1577	『Classical IP and ARP over ATM』
RFC 2225	『Classical IP and ARP over ATM』
RFC 2255	『The LDAP URL Format』
RFC 2684	『Multiprotocol Encapsulation over ATM Adaptation Layer 5』
RFC 4385	『Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN』
RFC 4717	『Encapsulation Methods for Transport of Asynchronous Transfer Mode (ATM) over MPLS Networks』
RFC 4816	『Pseudowire Emulation Edge-to-Edge (PWE3) Asynchronous Transfer Mode (ATM) Transparent Cell Transport Service』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR での双方向フォワーディング検出の設定

Bidirectional Forwarding Detection (BFD; 双方向フォワーディング検出) では、隣接する転送エンジン間のパスにおける障害を低オーバーヘッド、短期間で検出できます。BFD では、あらゆるメディアおよびあらゆるプロトコル レイヤでの障害検出に単一のメカニズムを使用でき、広範な検出時間とオーバーヘッドに対応できます。障害の迅速な検出が可能のため、リンクやネイバの障害発生時にもただちに障害に対応することができます。

Cisco IOS XR での双方向フォワーディング検出 (BFD) 設定機能の履歴

リリース	変更点
リリース 3.2	<p>この機能が Cisco CRS-1 キャリア ルーティング システム に追加され、次の機能がサポートされました。</p> <ul style="list-style-type: none">• 物理 Packet-over-SONET/SDH (POS) およびギガビット イーサネット番号付きリンク、VLAN での IPv4 非同期モードおよびエコノミーモード• BFD IPv4 単一ホップ• ラインカードでの配信• BFD バージョン 0 およびバージョン 1
リリース 3.3.0	<ul style="list-style-type: none">• BFD の次の機能へのサポートが追加されました。<ul style="list-style-type: none">– バンドル VLAN 経由のスタティック ルートを使用する BFD– トラフィック損失とネットワークのチャーンを最小限に抑えながらノード CPU の再起動を可能にする Minimum Disruption Restart (MDR; 最小限の中断による再起動)– イーサネット インターフェイスで BFD を使用した Fast Reroute/Traffic Engineering (FRR/TE; 高速再ルーティング/トラフィック エンジニアリング)• clear bfd counters packet コマンドおよび show bfd counters packet コマンドをサポートするために、設定手順が追加されました。
リリース 3.3.1	BFD サポートが Cisco XR 12000 シリーズ ルータに追加されました。

リリース 3.3.2	<ul style="list-style-type: none"> BFD が Unicast Reverse Path Forwarding (uRPF; ユニキャスト RPF) と組み合わせて使用されているルータまたはインターフェイスでのエコー モードをユーザがディセーブルにできるように、echo disable コマンドが追加されました。 ユーザがエコー モードをディセーブルにできる新しい BFD コンフィギュレーション モードが追加されました。ユーザが新しい BFD コンフィギュレーション モードを開始できるようにする bfd コマンドが追加されました。
リリース 3.4.0	バンドル VLAN での BFD で、Open Shortest Path First (OSPF) および Intermediate System-to-Intermediate System (IS-IS) がサポートされました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	IPv6 の BFD が Cisco CRS-1 ルータに追加されました。
リリース 3.8.0	BFD での OSPFv3 のサポートが Cisco CRS-1 ルータに追加されました。BFD MIB サポートが Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータに追加されました。

この章の構成

- 「BFD の実装の前提条件」(P.62)
- 「BFD に関する情報」(P.63)
- 「Cisco IOS XR での 双方向フォワーディング検出 (BFD) の設定例」(P.78)
- 「関連情報」(P.80)
- 「その他の参考資料」(P.80)

BFD の実装の前提条件

次に、BFD を実装するための前提条件を示します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- Cisco IOS XR ソフトウェアを稼働している Cisco CRS-1 ルータまたは Cisco XR 12000 シリーズ ルータであること。
- マルチプロトコル ラベル スイッチング (MPLS) で BFD をイネーブルにする場合は、MPLS パッケージを含んだインストール済みの複合 PIE ファイル、または複合パッケージ イメージが必要です。Border Gateway Protocol (BGP; ボーダー ゲートウェイ プロトコル)、Intermediate System-to-Intermediate System (IS-IS)、スタティック、Open Shortest Path First (OSPF) の場合は、インストール済みの Cisco IOS XR IP Unicast Routing Core Bundle イメージが必要です。

- IS-IS または OSPF を使用している場合、ルータで Interior Gateway Protocol (IGP; 内部ゲートウェイ プロトコル) がアクティブになっていること。
- Cisco CRS-1 ルータでは、BFD をサポートする各ラインカードが次のタスクを実行できる必要があります。
 - エコー パケットを 15 ms ごとに送信 (通常の状態)。
 - 制御パケットを 15 ms ごとに送信 (負荷のある状態)。
 - 6700 pps を超える User Datagram Protocol (UDP; ユーザ データグラム プロトコル) を送受信。これにより、15 ms 間隔で 100 セッション (または 150 ms 間隔で 1024 セッション) を維持します。
- Cisco XR 12000 シリーズ ルータ プラットフォームでは、BFD をサポートする各ラインカードが次のタスクを実行できる必要があります。
 - エコー パケットを 50 ms ごとに送信 (通常の状態)。
 - 制御パケットを 250 ms ごとに送信 (負荷のある状態)。
 - 12000 シリーズ プラットフォームでは、1000 pps を超える User Datagram Protocol (UDP; ユーザ データグラム プロトコル) を送受信。これにより、50 ms 間隔で 50 セッションを維持します。
- ネイバの BFD をイネーブルにするには、その近接ルータが BFD をサポートしている必要があります。
- BFD セッションを設定する前に、グローバル コンフィギュレーション モードで **router-id** コマンドを使用してローカル ルータ ID を設定することをお勧めします。ローカル ルータ ID を設定しない場合、BFD エコー モードでの IP パケットの送信元アドレスは、出力インターフェイスの IP アドレスとなります。

BFD に関する情報

Cisco IOS XR ソフトウェアは、IPv4 と IPv6 の両方について双方向フォワーディング検出 (BFD) をサポートします。

IPv4 での BFD

IPv4 単一ホップ接続の BFD において、Cisco IOS XR ソフトウェアは、番号付けされた物理 Packet-over-SONET/SDH (POS) およびギガビット イーサネット リンクでの非同期モードとエコーモードの両方を次のようにサポートします。

- エコー モードは、BFD 制御パケットを使用してセッションが確立された後にのみ開始されます。BFD エコー パケットは、送信元および宛先ポート 3785 を使用して、UDP/IPv4 で転送されます。IP パケットの送信元アドレスはローカル ルータ ID、宛先アドレスは ローカル インターフェイス アドレスとなります。



(注) ローカル ルータ ID 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 **router-id** コマンドを使用します。BFD セッションを設定する前にローカル ルータ ID を設定していない場合、IP パケットの送信元アドレスは出力インターフェイスの IP アドレスとなります。

- BFD 非同期パケットは、送信元ポート 49152 および宛先ポート 3784 を使用して、UDP および IPv4 で転送されます。非同期モードの場合、IP パケットの送信元アドレスはローカル インターフェイス アドレス、宛先アドレスはリモート インターフェイス アドレスとなります。



(注)

エコモードは、VLAN バンドルではサポートされません。



注意

特定のインターフェイスで BFD をユニキャスト RPF (uRPF) と併用している場合は、**echo disable** コマンドを使用して、そのインターフェイスでのエコモードをディセーブルにする必要があります。そうしないと、エコパケットは拒否されます。エコモードのディセーブル化は、ルータ全体、または個別のインターフェイスに対して行えます。

IPv4 インターフェイスでの IPv4 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv4 verify unicast source reachable-via** コマンドを使用します。IPv6 インターフェイスでの loose 方式 IPv6 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv6 verify unicast source reachable-via any** コマンドを使用します。

Cisco IOS XR ソフトウェアで BFD を設定する際には、次の事項に注意する必要があります。

- BFD は固定長の hello プロトコルで、接続の各終端で転送パスを通じてパケットを定期的に転送します。Cisco IOS XR ソフトウェアは、BFD の適応型検出時間をサポートします。
- BFD は、次のアプリケーションと併用することができます。
 - BGP
 - IS-IS
 - OSPF
 - MPLS Traffic-Engineering (MPLS-TE; MPLS トラフィック エンジニアリング)
 - Static
 - Protocol Independent Multicast (PIM)
- BFD は、次のインターフェイス タイプでの接続でサポートされます。
 - Packet-over-SONET/SDH (POS)
 - Gigabit Ethernet (GigE; ギガビット イーサネット)
 - Ten Gigabit Ethernet (TenGigE; 10 ギガビット イーサネット)
 - Virtual LAN (VLAN; 仮想 LAN)
- Cisco IOS XR ソフトウェアは、BFD バージョン 0 およびバージョン 1 をサポートします。BFD セッションは、ネイバに応じていずれかのバージョンを使用して確立されます。BFD バージョン 1 はデフォルトバージョンであり、セッション確立において最初に試行されます。
- BFD は、IPv4 で直接接続されている外部 BGP ピアでサポートされます。

ユーザは、ルータで次の作業を行えます。

- アプリケーション コンフィギュレーション スペースでの BFD パラメータ (適切な間隔および検出係数) の設定
- BFD 動作ステータス (ステート、カウンタ、追跡など) の表示
- BFD カウンタのクリア

IPv6 での BFD

IPv6 での双方向フォワーディング検出 (BFD) では、IPv6 アドレスを使用するインターフェイスでの稼働中の接続の確認をサポートします。

稼働中の接続の確認は、IPv4 と IPv6 の両方のインターフェイスについて、同じサービスとプロセスによって実行されます。

- ルート プロセッサ上の BFD サーバ
- ラインカード上の BFD エージェント

ただし、BFD サーバと BFD エージェントは個別のデータベースを使用します。

- IPv4 アドレスを保存するためのセッションデータベース 1 つ
- IPv6 アドレスを保存するためのセッションデータベース 1 つ

同一ラインカード上で、IPv4 と IPv6 の両方の BFD セッションを同時に実行することができます。

次に記載する制約事項を除き、IPv4 での BFD でサポートされる機能および設定と同じものが、IPv6 での BFD でもサポートされます。

Cisco IOS XR ソフトウェア リリース 3.8.0 での IPv6 の BFD には、次の制約事項が適用されます。

- IPv6 での BFD は、バンドル VLAN インターフェイスではサポートされません。
- IPv6 での BFD は、エコー モードではサポートされません。
- IPv6 での BFD は、エコー モードではなく非同期 モードでのみサポートされます。

バンドル VLAN での BFD

BFD は、スタティック ルーティング、IS-IS、OSPF を使用するバンドル VLAN でサポートされます。バンドル VLAN インターフェイスで BFD セッションを実行すると、VLAN バンドルがアップ状態である限り BFD セッションはアクティブな状態となります。

VLAN バンドルがアクティブであれば、次に示すイベントによって BFD セッションが失敗することはありません。

- コンポーネント リンクの失敗
- 1 つまたは複数のコンポーネント リンクをホストするラインカードのホットスワップ (OIR)
- バンドルへのコンポーネント リンクの追加 (設定による)
- バンドルからのコンポーネント リンクの削除 (設定による)
- コンポーネント リンクのシャットダウン
- RP スイッチオーバー



(注) VLAN バンドル設定の詳細については、「[Cisco IOS XR ソフトウェアでのリンク バンドルの設定](#)」モジュールを参照してください。

バンドル VLAN で BFD を設定する場合には、次の事項に注意する必要があります。

- RP スイッチオーバーの場合、設定されているネクストホップは Routing Information Base (RIB) に登録されます。
- BFD 再起動の場合、スタティック ルートは RIB に残ります。BFD セッションは、BFD の再起動時に再確立されます。



(注)

スタティック BFD セッションは、ネクストホップがルータに直接接続されているアドレス プレフィックスを持つピアでサポートされます。

BFD のパケット形式

BFD ペイロード制御パケットは、宛先ポート 3784 および送信元ポート 49152 を使用して、UDP パケットにカプセル化されます。イーサネットのような共有型メディアでも、BFD 制御パケットは常にユニキャスト パケットとして BFD ピアに送信されます。

エコー パケットも、宛先ポート 3785 および送信元ポート 3785 を使用して、UDP パケットにカプセル化されます。

制約事項

BFD には、次の制約事項が適用されます。

- 特定のインターフェイスで BFD を uRPF と併用している場合は、**echo disable** コマンドを使用して、そのインターフェイスでのエコー モードをディセーブルにする必要があります。そうしないと、エコー パケットは拒否されます。エコー モードのディセーブル化は、ルータ全体、または個別のインターフェイスに対して行えます。
- バンドル VLAN での BFD では、スタティック、OSPF、IS-IS の各アプリケーションのみがサポートされます。
- IPv6 での BFD は、バンドル VLAN インターフェイスではサポートされません。
- IPv6 での BFD は、エコー モードではサポートされません。
- IPv6 での BFD は、エコー モードではなく非同期 モードでのみサポートされます。
- IPv6 スタティック ルートでの BFD は、クライアントのみによりサポートされます。

BFD の設定

次に、BGP での BFD を設定する手順について説明します。BFD は、ネイバ単位またはインターフェイス単位でイネーブルにすることができます。ネイバ単位で BFD をイネーブルにするには、「[ネイバでの BFD のイネーブル化](#)」(P.66) に記載されている手順を使用します。インターフェイス単位で BFD をイネーブルにするには、「[特定インターフェイスでの OSPF の BFD のイネーブル化](#)」(P.68) に記載されている手順を使用します。



(注)

BFD を使用する FRR/TE は、POS インターフェイスおよびイーサネット インターフェイスでサポートされます。

ネイバでの BFD のイネーブル化

次に、近接ルータで BGP での BFD を設定する手順について説明します。



(注)

BFD 近接ルータの設定は、BGP でのみサポートされます。

手順の概要

1. **configure**
2. **router bgp** *autonomous-system-number*
3. **bfd minimum-interval** *milliseconds*
4. **bfd multiplier** *multiplier*
5. **neighbor** *ip-address*
6. **remote-as** *autonomous-system-number*
7. **bfd fast-detect**
8. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	router bgp <i>autonomous-system-number</i> 例: RP/0/RP0/CPU0:router(config)# router bgp 120	BGP コンフィギュレーション モードを開始します。このモードでは、BGP ルーティング プロセスの設定を行えます。現在のルータの <i>autonomous-system-number</i> を取得するには、EXEC モードで show bgp コマンドを使用します。
ステップ 3	bfd minimum-interval <i>milliseconds</i> 例: RP/0/RP0/CPU0:router(config-bgp)# bfd minimum-interval 6500	BFD の最小間隔を設定します。有効値の範囲は 15 ~ 30000 ms です。 この例では、BFD の最小間隔を 6500 ms に設定しています。
ステップ 4	bfd multiplier <i>multiplier</i> 例: RP/0/RP0/CPU0:router(config-bgp)# bfd multiplier 7	BFD 係数を設定します。 この例では、BFD 係数を 7 に設定しています。
ステップ 5	neighbor <i>ip-address</i> 例: RP/0/RP0/CPU0:router(config-bgp)# neighbor 172.168.40.24	BGP ルーティングのためにルータを ネイバ コンフィギュレーション モードにして、ネイバの IP アドレスを BGP ピアとして設定します。 この例では、IP アドレス 172.168.40.24 を BGP ピアとして設定しています。
ステップ 6	remote-as <i>autonomous-system-number</i> 例: RP/0/RP0/CPU0:router(config-bgp-nbr)# remote-as 2002	ネイバを作成し、そのネイバをリモート自律システムに割り当てます。 この例では、設定されるリモート自律システムは 2002 です。

BFD の設定

コマンドまたはアクション	目的
ステップ 7 <code>bfd fast-detect</code> 例: RP/0/RP0/CPU0:router (config-bgp-nbr) # bfd fast-detect	ローカル ネットワーキング装置と、ステップ 5 で IP アドレスを BGP ピアとして設定したネイバ間での BFD をイネーブルにします。 ステップ 5 の例では、IP アドレス 172.168.40.24 が BGP ピアとして設定されています。この例では、ローカル ネットワーキング装置とネイバ 172.168.40.24 間での BFD がイネーブルになります。
ステップ 8 <code>end</code> または commit 例: RP/0/RP0/CPU0:router (config-bgp-nbr) # end または RP/0/RP0/CPU0:router (config-bgp-nbr) # commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

特定インターフェイスでの OSPF の BFD のイネーブル化

次に、Open Shortest Path First (OSPF) での BFD を特定のインターフェイスで設定する手順について説明します。この方法の手順は、コマンド モードが異なる点を除き、IS-IS および MPLS-TE での BFD を設定する手順と共通です。



(注)

インターフェイス単位での BFD の設定は、OSPF、OSPFv3、IS-IS、MPLS-TE でのみサポートされます。OSPFv3 インターフェイスでの BFD の設定の詳細については、「[特定インターフェイスでの OSPFv3 の BFD のイネーブル化](#)」(P.70) を参照してください。

手順の概要

1. `configure`
2. `router ospf process-name`
3. `bfd minimum-interval milliseconds`
4. `bfd multiplier multiplier`
5. `area area-id`

6. `interface type interface-path-id`
7. `bfd fast-detect`
8. `end`
または
`commit`
9. `show run router ospf`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>router ospf process-name</code> 例: RP/0/RP0/CPU0:router(config)# <code>router ospf 0</code>	OSPF コンフィギュレーション モードを開始します。このモードでは、OSPF ルーティング プロセスの設定を行えます。 現在のルータの <code>process-name</code> を取得するには、EXEC モードで <code>show ospf</code> コマンドを使用します。 (注) IS-IS または MPLS-TE での BFD を設定するには、対応するコンフィギュレーション モードを開始します。たとえば、MPLS-TE の場合は、MPLS-TE コンフィギュレーション モードを開始します。
ステップ 3	<code>bfd minimum-interval milliseconds</code> 例: RP/0/RP0/CPU0:router(config-ospf)# <code>bfd minimum-interval 6500</code>	BFD の最小間隔を設定します。有効値の範囲は 15 ~ 30000 ms です。 この例では、BFD の最小間隔を 6500 ms に設定しています。
ステップ 4	<code>bfd multiplier multiplier</code> 例: RP/0/RP0/CPU0:router(config-ospf)# <code>bfd multiplier 7</code>	BFD 係数を設定します。 この例では、BFD 係数を 7 に設定しています。
ステップ 5	<code>area area-id</code> 例: RP/0/RP0/CPU0:router(config-ospf)# <code>area 0</code>	Open Shortest Path First (OSPF) 領域を設定します。 <code>area-id</code> は、OSPF 領域の ID に置き換えてください。
ステップ 6	<code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config-ospf-ar)# <code>interface gigabitEthernet 0/3/0/1</code>	インターフェイス コンフィギュレーション モードを開始して、インターフェイス名と <code>rack/slot/module/port</code> 表記を指定します。 <ul style="list-style-type: none"> この例では、モジュラー サービス カード スロット 3 にあるギガビット イーサネット インターフェイスを示しています。
ステップ 7	<code>bfd fast-detect</code> 例: RP/0/RP0/CPU0:router(config-ospf-ar-if)# <code>bfd fast-detect</code>	隣接する転送エンジン間のパスで障害を検出するために、BFD をイネーブルにします。

BFD の設定

コマンドまたはアクション	目的
<p>ステップ 8</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router (config-ospf-ar-if)# end または RP/0/RP0/CPU0:router (config-ospf-ar-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 9</p> <pre>show run router ospf</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router (config-ospf-ar-if)# show run router ospf</pre>	<p>適切なインターフェイスで BFD がイネーブルになっていることを確認します。</p>

特定インターフェイスでの OSPFv3 の BFD のイネーブル化

次に、OSPFv3 での BFD を特定のインターフェイスで設定する手順について説明します。この方法の手順は、コマンドモードが異なる点を除き、IS-IS および MPLS-TE での BFD を設定する手順と共通です。



(注)

インターフェイス単位での BFD の設定は、OSPF、OSPFv3、IS-IS、MPLS-TE でのみサポートされます。OSPF インターフェイスでの BFD の設定の詳細については、「[特定インターフェイスでの OSPF の BFD のイネーブル化](#)」(P.68) を参照してください。

手順の概要

1. **configure**
2. **router ospfv3 process-name**
3. **bfd minimum-interval milliseconds**
4. **bfd multiplier multiplier**
5. **area area-id**
6. **interface type interface-path-id**
7. **bfd fast-detect**

8. `end`
または
`commit`
9. `show run router ospfv3`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>router ospfv3 process-name</code> 例: RP/0/RP0/CPU0:router(config)# <code>router ospfv3 0</code>	OSPFv3 コンフィギュレーション モードを開始します。このモードでは、OSPFv3 ルーティング プロセスの設定を行います。 現在のルータの <code>process name</code> を取得するには、EXEC モードで <code>show ospfv3</code> コマンドを使用します。 (注) IS-IS または MPLS-TE での BFD を設定するには、対応するコンフィギュレーション モードを開始します。たとえば、MPLS-TE の場合は、MPLS-TE コンフィギュレーション モードを開始します。
ステップ 3	<code>bfd minimum-interval milliseconds</code> 例: RP/0/RP0/CPU0:router(config-ospfv3)# <code>bfd minimum-interval 6500</code>	BFD の最小間隔を設定します。有効値の範囲は 15 ~ 30000 ms です。 この例では、BFD の最小間隔を 6500 ms に設定しています。
ステップ 4	<code>bfd multiplier multiplier</code> 例: RP/0/RP0/CPU0:router(config-ospfv3)# <code>bfd multiplier 7</code>	BFD 係数を設定します。 この例では、BFD 係数を 7 に設定しています。
ステップ 5	<code>area area-id</code> 例: RP/0/RP0/CPU0:router(config-ospfv3)# <code>area 0</code>	OSPFv3 領域を設定します。 <code>area-id</code> は、OSPFv3 領域の ID に置き換えてください。
ステップ 6	<code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config-ospfv3-ar)# <code>interface gigabitEthernet 0/1/5/0</code>	インターフェイス コンフィギュレーション モードを開始して、インターフェイス名と <code>rack/slot/module/port</code> 表記を指定します。 • この例では、モジュラー サービス カード スロット 1 にあるギガビットイーサネットインターフェイスを示しています。
ステップ 7	<code>bfd fast-detect</code> 例: RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# <code>bfd fast-detect</code>	隣接する転送エンジン間のパスで障害を検出するために、BFD をイネーブルにします。

BFD の設定

コマンドまたはアクション	目的
<p>ステップ 8 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router (config-ospfv3-ar-if)# <code>end</code> または RP/0/RP0/CPU0:router (config-ospfv3-ar-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 9 <code>show run router ospfv3</code></p> <p>例: RP/0/RP0/CPU0:router (config-ospfv3-ar-if)# <code>show run router ospfv3</code></p>	<p>適切なインターフェイスで BFD がイネーブルになっていることを確認します。</p>

スタティック ルートでの BFD のイネーブル化

次に、スタティック ルートでの BFD をイネーブルにする手順について説明します。



(注) バンドル VLAN セッションは、間隔 250 ms、係数 3 の場合のみに制限されます。これよりも強力なパラメータは使用できません。

手順の概要

1. `configure`
2. `router static`
3. `address-family ipv4 unicast address nexthop bfd fast-detect [minimum interval interval] [multiplier multiplier]`
4. `vrf vrf-name`
5. `address-family ipv4 unicast address nexthop bfd fast-detect [minimum interval interval] [multiplier multiplier]`
6. `end`
または
`commit`

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>router static</code> 例: RP/0/RP0/CPU0:router(config)# router static	スタティック ルート コンフィギュレーション モードを開始します。このモードでは、スタティック ルーティングの設定を行えます。
ステップ 3 <code>address-family ipv4 unicast address nexthop bfd fast-detect [minimum-interval interval] [multiplier multiplier]</code> 例: RP/0/RP0/CPU0:router(config-static)# address-family ipv4 unicast 0.0.0.0/0 2.6.0.1 bfd fast-detect minimum-interval 1000 multiplier 5	指定の IPv4 ユニキャスト宛先アドレス プレフィクスおよびフォワーディング ネクストホップ アドレスで BFD 高速検出をイネーブルにします。 (注) ネクストホップが確実に同じ hello 間隔で割り当てられるようにするには、オプションの minimum-interval キーワードおよび引数を含めます。 <i>interval</i> 引数は、間隔を ms 単位で指定する数字に置き換えてください。有効値の範囲は 10 ~ 10,000 です。 (注) ネクストホップが確実に同じ検出係数で割り当てられるようにするには、オプションの multiplier キーワード引数を含めます。 <i>multiplier</i> 引数は、検出係数を指定する数字に置き換えてください。有効値の範囲は 1 ~ 10 です。 (注) バンドル VLAN セッションは、間隔 250 ms、係数 3 の場合のみに制限されます。これよりも強力なパラメータは使用できません。
ステップ 4 <code>vrf vrf-name</code> 例: RP/0/RP0/CPU0:router(config-static)# vrf vrf1	VPN Routing and Forwarding (VRF; VPN ルーティングおよび転送) インスタンスを指定して、その VRF に対するスタティック ルート コンフィギュレーション モードを開始します。

BFD の設定

コマンドまたはアクション	目的
ステップ 5 <code>address-family ipv4 unicast address nexthop bfd fast-detect</code> 例: RP/0/RP0/CPU0:router (config-static-vrf)# address-family ipv4 unicast 0.0.0.0/0 2.6.0.2	指定の IPv4 ユニキャスト宛先アドレス プレフィックスおよびフォワーディング ネクストホップアドレスで BFD 高速検出をイネーブルにします。
ステップ 6 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router (config-static-vrf)# end または RP/0/RP0/CPU0:router (config-static-vrf)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、システムが変更をコミットするように求めるプロンプトを表示します。 Uncommitted changes found.Commit them? <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、設定変更のコミットは行われず、同じコマンド モードが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

ルータでのエコー モードのディセーブル化

次に、uRPF と組み合わせて BFD を使用しているルータでエコー モードをディセーブルにする手順について説明します。



(注)

IPv4 インターフェイスでの IPv4 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv4 verify unicast source reachable-via** コマンドを使用します。IPv6 インターフェイスでの loose 方式 IPv6 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv6 verify unicast source reachable-via any** コマンドを使用します。

手順の概要

1. **configure**
2. **bfd**
3. **echo disable**
4. **end**
 または
commit

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>bfd</code> 例: RP/0/RP0/CPU0:router(config)# <code>bfd</code>	BFD コンフィギュレーション モードを開始します。
ステップ 3 <code>echo disable</code> 例: RP/0/RP0/CPU0:router(config-bfd)# <code>echo disable</code>	ルータでエコー モードをディセーブルにします。
ステップ 4 <code>end</code> または commit 例: RP/0/RP0/CPU0:router (config-bfd)# <code>end</code> または RP/0/RP0/CPU0:router(config-bfd)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

個別のインターフェイスまたはバンドルでのエコー モードのディセーブル化

次に、uRPF と組み合わせて BFD を使用しているインターフェイスまたはバンドルでエコー モードをディセーブルにする手順について説明します。



(注)

IPv4 インターフェイスでの IPv4 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv4 verify unicast source reachable-via** コマンドを使用します。IPv6 インターフェイスでの loose 方式 IPv6 uRPF チェックをイネーブル/ディセーブルにするには、インターフェイス コンフィギュレーション モードで **[no] ipv6 verify unicast source reachable-via any** コマンドを使用します。

手順の概要

1. **configure**
2. **bfd**
3. **interface** *type interface-path-id*
4. **echo disable**
5. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	bfd 例： RP/0/RP0/CPU0:router (config)# bfd	BFD コンフィギュレーション モードを開始します。
ステップ 3	interface <i>type interface-path-id</i> 例： RP/0/RP0/CPU0:router (config-bfd)# interface gigabitEthernet 0/1/5/0	特定のインターフェイスまたはバンドルに対して BFD インターフェイス コンフィギュレーション モードを開始します。BFD インターフェイス コンフィギュレーション モードでは、個別のインターフェイスまたはバンドルでエコー モードをディセーブルにすることができます。

コマンドまたはアクション	目的
ステップ 4 <code>echo disable</code> 例: RP/0/RP0/CPU0:router(config-bfd-if)# echo disable	ルータでエコー モードをディセーブルにします。
ステップ 5 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router (config-bfd-if)# end または RP/0/RP0/CPU0:router(config-bfd-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

BFD カウンタのクリアおよびディセーブル化

次に、BFD パケット カウンタの表示およびクリアの手順について説明します。特定ノードまたは特定インターフェイスでホストされている BFD セッションのパケット カウンタをクリアすることができます。

手順の概要

1. `show bfd counters packet [interface type interface-path-id] location node-id`
2. `clear bfd counters packet [interface type interface-path-id] location node-id`
3. `show bfd counters packet [interface type interface-path-id] location node-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<pre>show bfd counters [ipv4 ipv6 all] packet [interface type interface-path-id] location node-id</pre> <p>例: RP/0/RP0/CPU0:router# show bfd counters all packet location 0/3/cpu0</p>	IPv4 パケット、IPv6 パケット、またはすべてのパケットの BFD カウンタを表示します。
ステップ 2	<pre>clear bfd counters [ipv4 ipv6 all] packet [interface type interface-path-id] location node-id</pre> <p>例: RP/0/RP0/CPU0:router# clear bfd counters all packet interface POS 0/5/0/1 location 0/5/cpu0</p>	IPv4 パケット、IPv6 パケット、またはすべてのパケットの BFD カウンタをクリアします。
ステップ 3	<pre>show bfd counters [ipv4 ipv6 all] packet [interface type interface-path-id] location node-id</pre> <p>例: RP/0/RP0/CPU0:router# show bfd counters all packet location 0/3/cpu0</p>	IPv4 パケット、IPv6 パケット、またはすべてのパケットの BFD カウンタがクリアされていることを確認します。

Cisco IOS XR での双方向フォワーディング検出 (BFD) の設定例

ここでは、次の設定例について説明します。

- 「双方向フォワーディング検出 (BFD) : 例」 (P.78)
- 「スタティック ルートでの BFD : 例」 (P.79)
- 「BFD エコー モードのディセーブル化 : 例」 (P.80)

双方向フォワーディング検出 (BFD) : 例

次に、自律システム 65000 とネイバ 192.168.70.24 間での BFD を設定する例を示します。

```
configure
router bgp 65000
  bfd multiplier 2
  bfd minimum-interval 20
  neighbor 192.168.70.24
    remote-as 2
    bfd fast-detect
```

次に、ギガビット イーサネット インターフェイスで OSPF での BFD をイネーブルにする例を示します。

```
configure
router ospf 0
  area 0
    interface gigabitEthernet 0/3/0/1
```



```
bfd fast-detect
commit
```

```
RP/0/0/CPU0:Dec 2 07:06:48.508 : config[65685]: %MGBL-LIBTARCFG-6-COMMIT : Configuration
committed by user 'xxx'. Use 'show configuration commit changes 1000001134' to view the
changes.
```

```
end
```

```
RP/0/0/CPU0:Dec 2 07:06:48.848 : config[65685]: %MGBL-SYS-5-CONFIG_I : Configured from
console by lab
```

```
show run router ospf
```

```
router ospf 0
area 0
interface GigabitEthernet0/3/0/1
bfd fast-detect
!
```

次に、ギガビットイーサネットインターフェイスで OSPFv3 での BFD をイネーブルにする例を示します。

```
configure
router ospfv3 0
  bfd minimum-interval 6500
  bfd multiplier 7
  area 0
    interface gigabitethernet 0/1/5/0
      bfd fast-detect
    commit
  show run router ospfv3
```

```
router ospfv3 0
bfd multiplier 7
bfd minimum-interval 6500
area 0
interface GigabitEthernet0/1/5/0
  bfd fast-detect
!
!
!
```

スタティック ルートでの BFD : 例

次に、IPv4 スタティック ルートでの BFD をイネーブルにする例を示します。この例では、BFD セッションは、ネクストホップ 3.3.3.3 が到達可能になると、このネクストホップで確立されます。

```
configure
router static
  address-family ipv4 unicast 2.2.2.0/24 3.3.3.3 bfd fast-detect
end
```

次に、IPv6 スタティック ルートでの BFD をイネーブルにする例を示します。この例では、BFD セッションは、ネクストホップ 2001:0DB8:D987:398:AE3:B39:333:783 が到達可能になると、このネクストホップで確立されます。

```
configure
router static
  address-family ipv6 unicast 2001:0DB8:C18:2:1::F/64
2001:0DB8:D987:398:AE3:B39:333:783 bfd fast-detect minimum-interval 150 multiplier 4
```

```

end
exit
exit
show run router static address-family ipv6 unicast

```

BFD エコー モードのディセーブル化 : 例

次に、ルータでエコー モードをディセーブルにする例を示します。

```

configure
bfd
echo disable

```

次に、インターフェイスでエコー モードをディセーブルにする例を示します。

```

configure
bfd
interface pos 0/1/0/0
echo disable

```

関連情報

BFD は、複数プラットフォームでサポートされます。これらのコマンドの詳細については、関連するコマンドリファレンス マニュアルを参照してください。

- 『*BGP Commands on Cisco IOS XR Software*』
- 『*IS-IS Commands on Cisco IOS XR Software*』
- 『*OSPF Commands on Cisco IOS XR Software*』
- 『*Static Routing Commands on Cisco IOS XR Software*』
- 『*MPLS Traffic Engineering Commands on Cisco IOS XR Software*』

その他の参考資料

ここでは、Cisco IOS XR ソフトウェアでの BFD の実装に関する参考資料について説明します。

関連資料

内容	参照先
BFD コマンド : コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト、使用上のガイドライン、例	『 <i>Cisco IOS XR Interface and Hardware Command Reference</i> 』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

RFC

RFC	タイトル
draft-ietf-bfd-base-06	『 <i>Bidirectional Forwarding Detection</i> 』 2007 年 3 月
draft-ietf-bfd-v4v6-1hop-06	『 <i>BFDD for IPv4 and IPv6 (Single Hop)</i> 』 2007 年 3 月

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。 Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのチャネライズド SONET の設定

ここでは、チャネライズド Synchronous Optical Network (SONET; 同期光ファイバ ネットワーク) の設定について説明します。

Cisco IOS XR ソフトウェアでのチャネライズド SONET 設定機能の履歴

リリース	変更点
リリース 3.5.0	Cisco XR 12000 シリーズ ルータにこの機能が追加されました。
リリース 3.6.0	変更ありません。
リリース 3.7.0	チャネライズド ATM およびクリア チャネル ATM の設定手順と例が追加されました。
リリース 3.8.0	変更ありません。

この章の構成

- [「チャネライズド SONET 設定の前提条件」 \(P.83\)](#)
- [「チャネライズド SONET の設定に関する情報」 \(P.84\)](#)
- [「チャネライズド SONET の設定方法」 \(P.91\)](#)
- [「チャネライズド SONET の設定例」 \(P.114\)](#)

チャネライズド SONET 設定の前提条件

チャネライズド SONET を設定する前に、次に示す作業が実施されており、条件を満たしていることを確認する必要があります。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- シャーシに、次の共有ポート アダプタ (SPA) のうち少なくとも 1 つが設置されている必要があります。
 - Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS3/OC-48 共有ポート アダプタ
 - Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタ
- 汎用表記 *rack/slot/module/port* を使用して SONET コントローラ名と *interface-path-id* を適用/指定する方法を理解する必要があります。SONET コントローラ名と *interface-path-id* は、**controller sonet** コマンドで必要となります。

チャネライズド SONET の設定に関する情報

チャネライズド SONET を設定するには、次の概念を理解する必要があります。

- 「[チャネライズド SONET の概要](#)」 (P.84)
- 「[チャネライズド SDH の概要](#)」 (P.89)
- 「[チャネライズド SONET/SDH のデフォルト設定値](#)」 (P.91)

チャネライズド SONET の概要

同期光ファイバ ネットワーク (SONET) は、光ファイバでのデジタル テレコミュニケーション サービス伝送において使用される American National Standards Institute (ANSI; 米国規格協会) の規格形式です。

Synchronous Digital Hierarchy (SDH; 同期デジタル ハイアラキ) は、SONET の国際版に相当します。

チャネライズド SONET では、多重化 T3/E3 および Virtual Tributary Group (VTG; 仮想トリビュタリグループ) チャネルで SONET フレームを転送することができます。

チャネライズド SONET は、次の共有ポート アダプタ (SPA) でサポートされます。

- Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ
- Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ
- Cisco 1 ポート チャネライズド DS3/OC-48 共有ポート アダプタ

チャネライズド SDH は、次の共有ポート アダプタ (SPA) でのみサポートされます。

- Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ

チャネライズド T3/E3 ATM は、次の共有ポート アダプタ (SPA) でのみサポートされます。

- Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポート アダプタ

SONET は、Synchronous Transport Signal (STS; 同期転送信号) フレーム構成を使用します。STS は、Optical Carrier 1 (OC1; オプティカル キャリア 1) の電気版に相当します。

SDH は、Synchronous Transport Mode (STM; 同期転送モード) フレーム構成を使用します。1 つの STM1 は、3 つのオプティカル キャリア 1 (OC1) に相当します。

チャネライズド SONET インターフェイスは、複数の STS ストリームを複合したものであり、固有のペイロード ポインタを持つ独立したフレームとして維持されます。フレームは、転送される前に多重化されます。

回線がチャネル化されると、パスと呼ばれるより小さい帯域幅のチャネルに論理的に分割されます。これらのパスが SONET ペイロードを伝送します。全パスの帯域幅の合計は回線の帯域幅を超過できません。

回線がチャネル化されない場合、この回線はクリア チャネルと呼ばれ、回線の全帯域幅がブロードバンド サービスを伝送する単一のチャネル専用となります。

STS ストリームは、次のタイプのチャネルにチャネル化することができます。

- T3/E3
- VT1.5 がマッピングされた T1
- Packet over SONET/SDH (POS) (OC12 および OC48 のみ)

T3/E3 チャネルは、さらに T1 にチャネル化でき、T1 はタイム スロット (DS0) にチャネル化できます。ただし、DS0 をサポートしない ATM Cisco 2 ポート チャネライズド T3/E3 ATM および回線エミュレーション共有ポートアダプタの場合は除きます。

SONET 回線のチャネル化は、次の 2 つの主要なプロセスで構成されます。

- コントローラの設定
- インターフェイスのチャネライズドパスへの設定

最初に、STS パスのモードを設定することによりコントローラを設定します。モードは、T3、VT1.5 がマッピングされた T1、または POS に設定できます。



(注)

POS は、Cisco 1 ポート チャネライズド DS0/OC-12 SPA の STS-3c パスと STS-12c パス、および Cisco 1 ポート チャネライズド DS3/OC-48 SPA の STS-3c、STS-12c、STS-48c の各パスでのみサポートされます。

モードが指定されると、各コントローラが作成され、残りの設定がそのコントローラに適用されます。たとえば、T3 モードでは T3 コントローラが作成されます。T3 コントローラは、シリアル チャネルに対して設定するか、または T1 を伝送するためにさらにチャネル化できます。これらの T1 は、シリアル インターフェイスに対して設定できます。

Cisco 1 ポート チャネライズド STM-1/OC-3 SPA では、デフォルト設定は SONET カードのインストール時に設定済みの次のパスで構成されます。

- STS 1
- STS 2
- STS 3

各 STS パスは、個別に T3、E3、VTG などに設定できます。

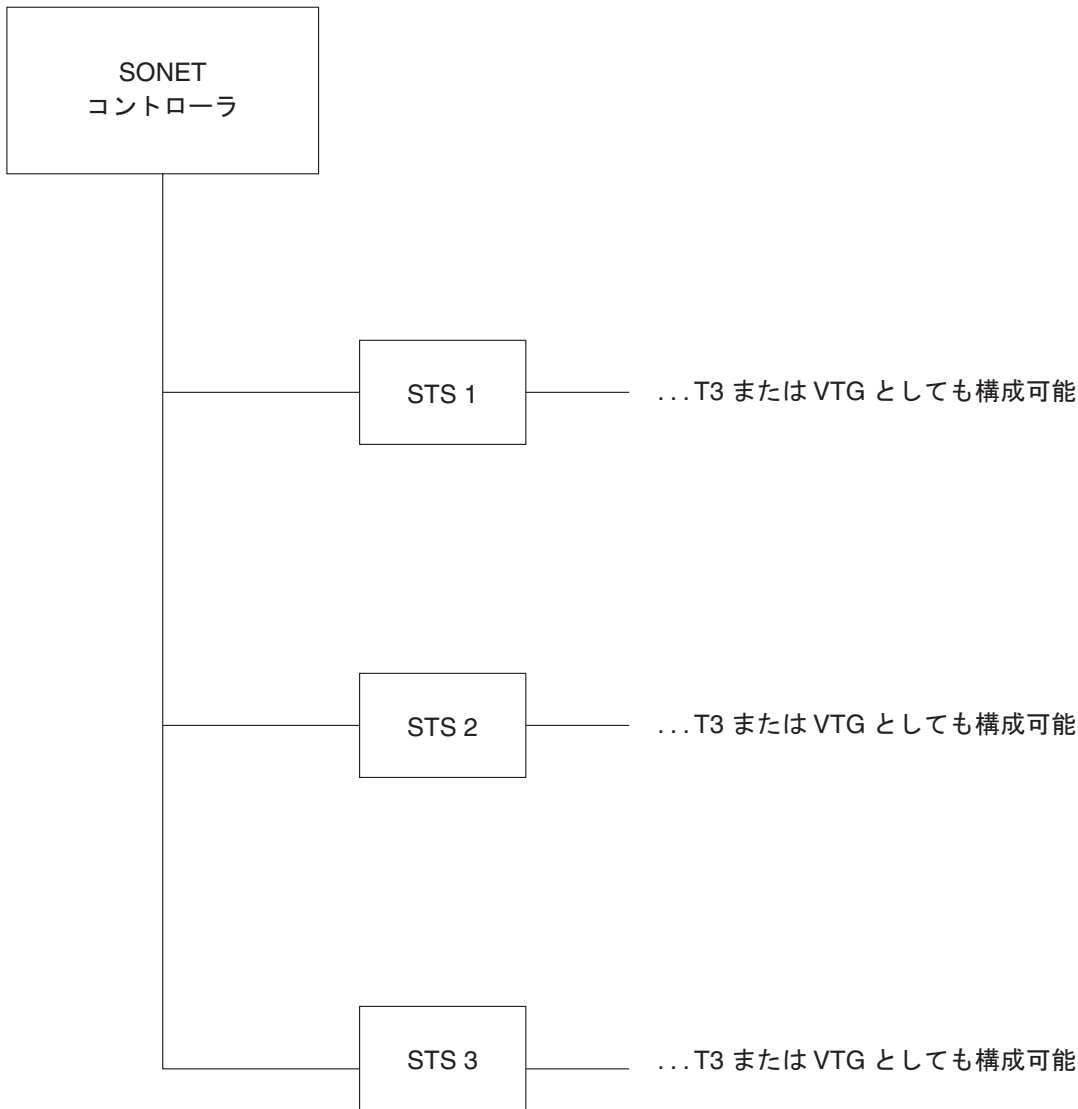
図 1 は、カードのインストール時に設定済みの SONET コントローラのデフォルト設定を示しています。

図 2 は、SONET コントローラの設定の組み合わせを示しています。

図 3 は、設定可能な T3 パスを示しています。

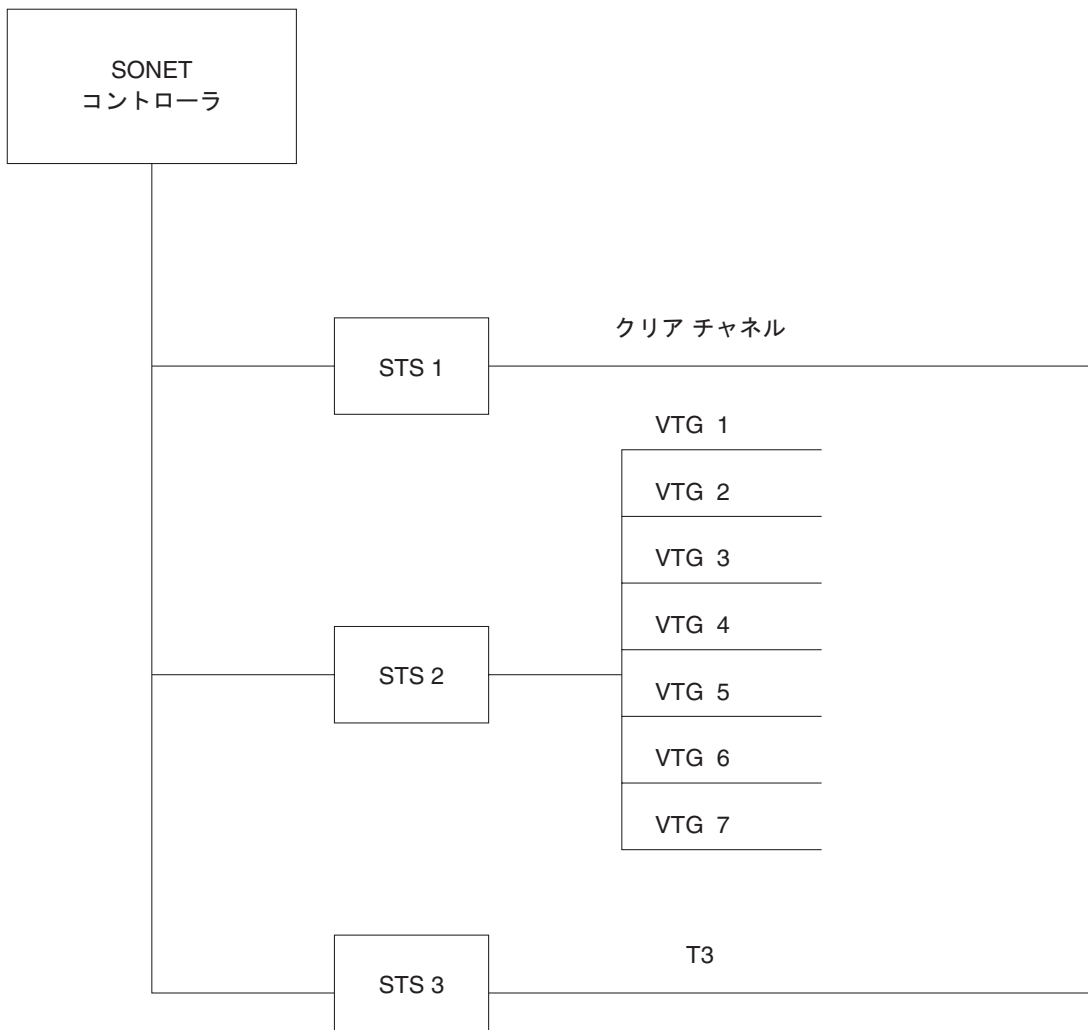
図 4 は、設定可能な VTG パスを示しています。

図 1 SONET コントローラのデフォルト設定



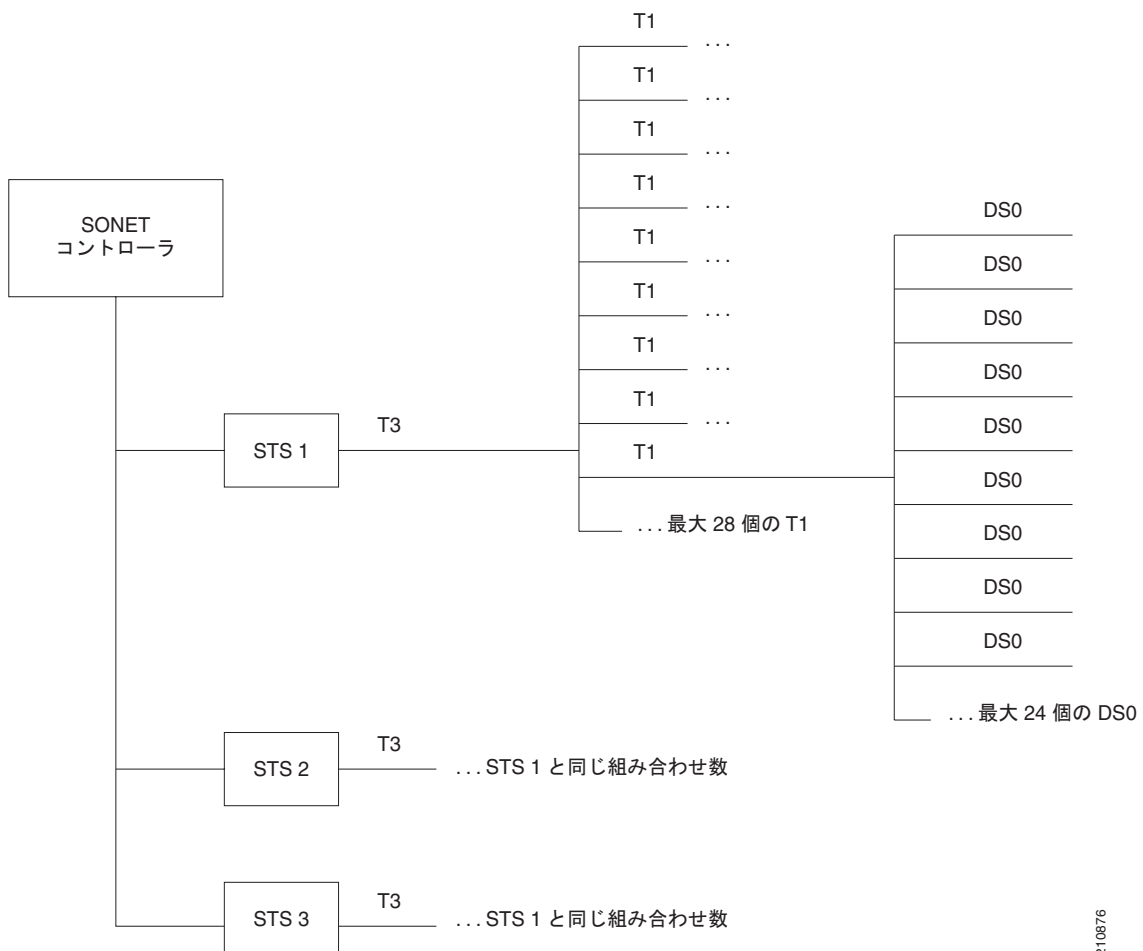
210870

図 2 SONET コントローラの設定の組み合わせ



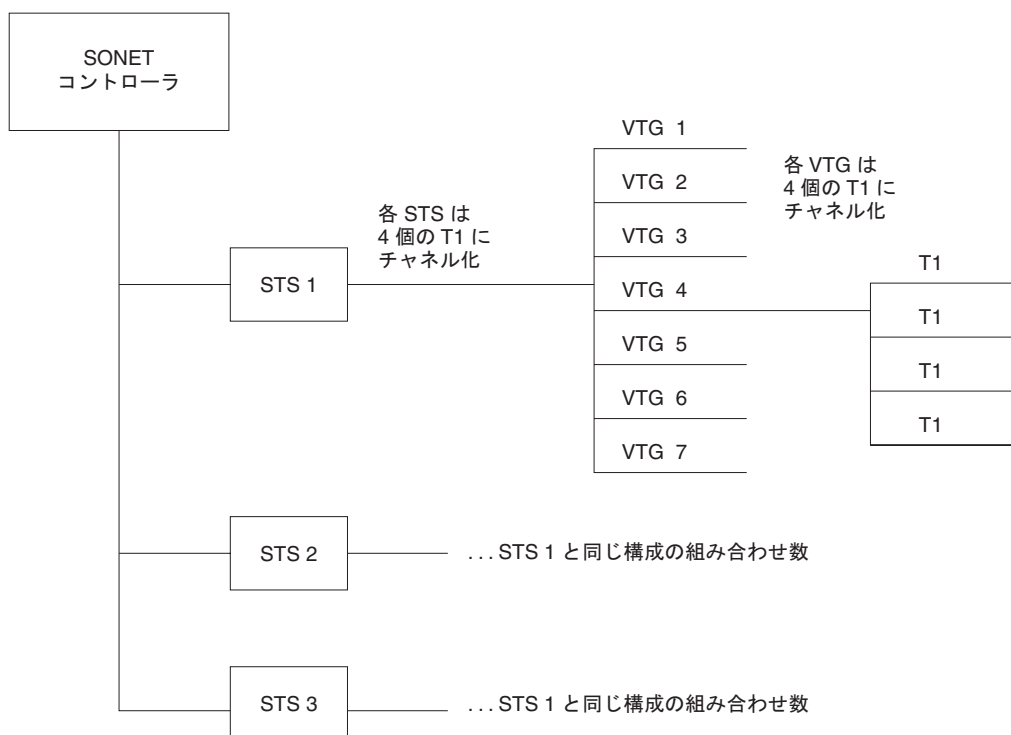
210873

図 3 SONET T3 チャネライズドパス



210876

図 4 SONET VTG チャネライズドパス



210877

チャネライズド SDH の概要

同期デジタルハイアラキー (SDH) は、SONET の国際版に相当します。

チャネライズド SDH は、次の共有ポートアダプタ (SPA) でのみサポートされます。

- Cisco I ポート チャネライズド STM-1/OC-3 共有ポートアダプタ

同期転送モジュール (STM) 信号は、SONET の STS の同期デジタルハイアラキー (SDH) 版に相当しますが、各帯域幅で番号は異なります。ここでは、STM という用語はパス幅と光回線レートの両方を表します。STM 信号内のパスは、Administrative Unit (AU; 管理ユニット) と呼ばれます。

SONET と SDH 間での基本的な用語の違いの概要を次に示します。

- SONET の STS は、SDH の管理ユニット (AU) に相当
- SONET の Virtual Tributary (VT; 仮想トリビュタリ) は、SDH の Tributary Unit (TU; トリビュタリユニット) に相当
- SDH の基本ビルディングブロックは STM1 (STS-3 に相当) および STM-0 (STS-1 に相当)

管理ユニット (AU) は、より上位のパスレイヤと多重化セクションレイヤ間の適合を可能にする情報構造です。AU は、情報ペイロード (より上位の仮想コンテナ) と管理ユニットポインタで構成されません。管理ユニットポインタは、ペイロードフレーム開始のオフセットを多重化セクションフレーム開始と相対的に示します。

AU は、トリビュタリユニット (TU) およびトリビュタリユニットグループ (TUG) にチャネライズすることができます。

Administrative Unit 4 (AU-4; 管理ユニット 4) は、3 つの STM-1 または 1 つの STM-3 で構成されません。

Administrative Unit 3 (AU-3; 管理ユニット 3) は、1 つの STM-1 で構成されます。

Administrative Unit Group (AUG; 管理ユニットグループ) は、STM ペイロードにおいて固定の定義された位置を占める 1 つまたは複数の管理ユニットで構成されます。

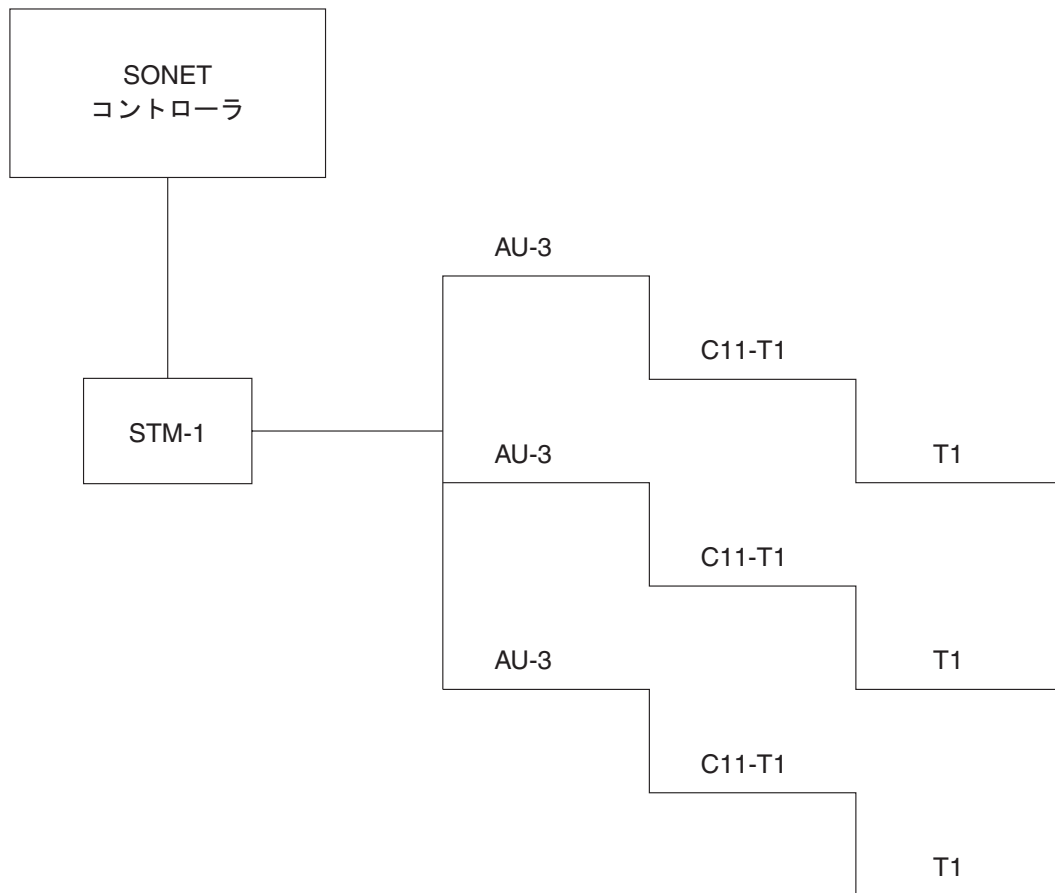
表 2 SONET/SDH 用語対照表

SONET 用語	SDH 用語
SONET	SDH
STS-3c	AU-4
STS-1	AU-3
VT	TU
SPE	VC
セクション	リジェネレータ セクション
回線	多重化セクション
パス	パス

図 5 は、設定可能な SDH AU3 パスを示しています。

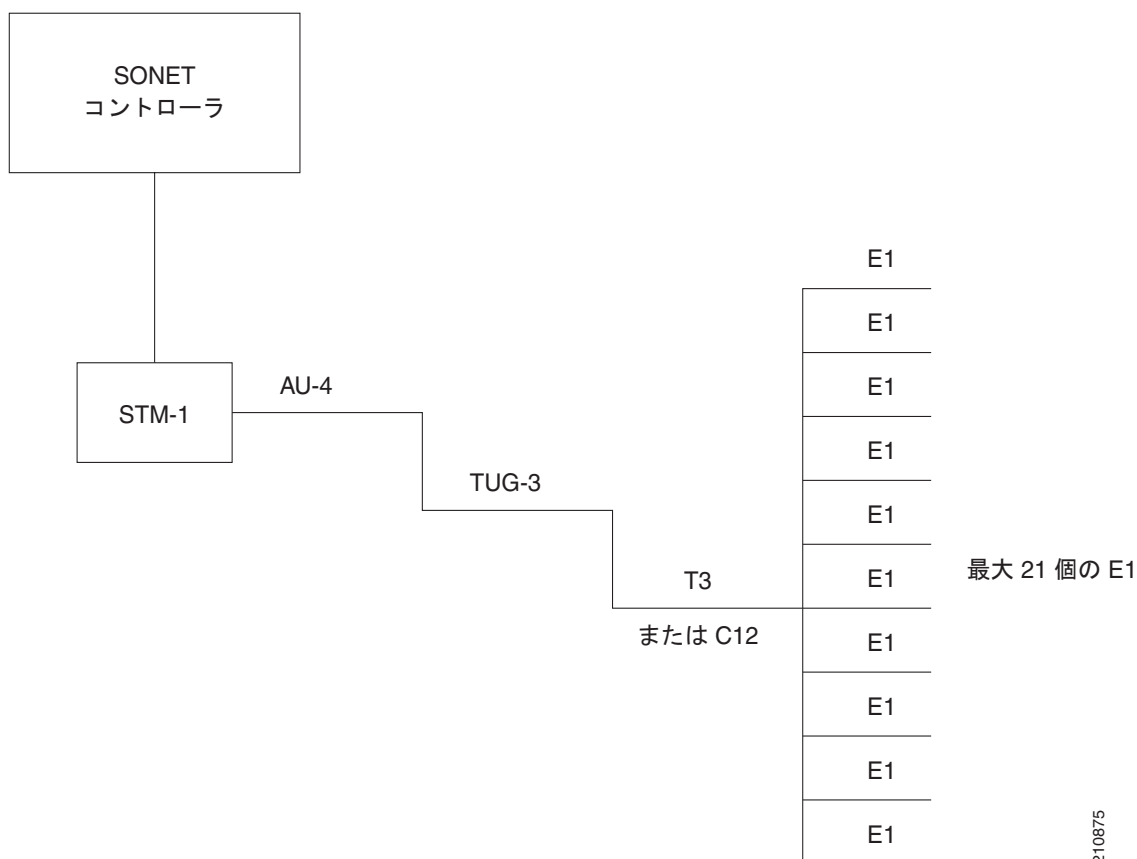
図 6 は、設定可能な SDH AU4 パスを示しています。

図 5 SDH AU3 パス



210874

図 6 SDH AU4 パス



チャネライズド SONET/SDH のデフォルト設定値

表 3 に、チャネライズド SONET/SDH に存在するデフォルト設定パラメータを示します。

表 3 SONET/SDH コントローラのデフォルト設定値

パラメータ	デフォルト値	コンフィギュレーションファイルのエントリ
clock source	line	clock source {internal line}
SONET framing	sonet	framing {sdh sonet}

チャネライズド SONET の設定方法

ここでは、次の手順について説明します。

- 「[SONET T3 チャンネルおよび VT1.5 がマッピングされた T1 チャンネルの設定](#)」 (P.92)
- 「[Packet over Sonet チャンネルの設定](#)」 (P.96)
- 「[クリア チャンネル T3 の設定](#)」 (P.99)
- 「[チャネライズド SONET 自動保護スイッチング \(APS\) の設定](#)」 (P.103)

- 「SDH AU-3 の設定」 (P.106)
- 「SDH AU-4 の設定」 (P.109)

SONET T3 チャンネルおよび VT1.5 がマッピングされた T1 チャンネルの設定

ここでは、SONET 回線を T3 チャンネルおよび VT がマッピングされた T1 チャンネルに設定する手順について説明します。

前提条件

- 「Cisco IOS XR ソフトウェアでのクリア チャンネル SONET コントローラの設定」モジュールの「クリア チャンネル SONET コントローラの設定方法」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

Cisco IOS XR リリース 3.5.0 には、次の制約事項が適用されます。

- STS パスの T3 へのチャンネル化は、次の共有ポート アダプタで行えます。
 - Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS3/OC-48 共有ポート アダプタ
- STS パスの VTG がマッピングされた T1 へのチャンネル化は、次の共有ポート アダプタで行えます。
 - Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ
- T3 パスの T1 または E1 へのチャンネル化は、次の共有ポート アダプタで行えます。
 - Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ
 - Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ (このリリースでは E1 のサポートなし)
- T1 は、次の共有ポート アダプタではサポートされません。
 - Cisco 1 ポート チャネライズド DS3/OC-48 共有ポート アダプタ

手順の概要

1. **configure**
2. **controller sonet interface-path-id**
3. **clock source {internal | line}**
4. **framing {sdh | sonet}**
5. **sts number**
6. **mode mode**
7. **width number**
8. **root**
9. **controller controllerName instance**

10. `mode mode`
11. `root`
12. `controller t1 interface-path-id`
13. `channel-group number`
14. `timeslots num1:num2:num3:num4`
または
`timeslots range1-range2`
15. `show configuration`
16. `root`
17. `interface serial interface-path-id`
18. `encapsulation frame-relay | hdlc | ppp`
19. `ipv4 ip-address mask`
20. `no shutdown`
21. `end`
または
`commit`
22. `show`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller sonet interface-path-id</code> 例： RP/0/0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション サブモードを開始して、SONET コントローラ名と <code>interface-path-id</code> を <code>rack/slot/module/port</code> 表記で指定します。
ステップ 3	<code>clock source {internal line}</code> 例： RP/0/0/CPU0:router(config-sonet)# clock source internal	SONET ポート転送クロック ソースを設定します。ここで、 internal キーワードは内部クロック、 line キーワードは回線から回収されたクロックを設定します。 <ul style="list-style-type: none">• ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。2 つのルータがバックツーバックまたは光ファイバで接続されており、クロッキングが得られない場合は、internal キーワードを使用します。• デフォルト キーワードは line です。 (注) Spatial Reuse Protocol (SRP; スペース再利用プロトコル) インターフェイスでは、内部クロッキングが必要です。
ステップ 4	<code>framing {sdh sonet}</code> 例： RP/0/0/CPU0:router(config-sonet)# framing sonet	(任意) 同期デジタル ハイアラキー (SDH) フレーム構成の場合は sdh キーワード、SONET フレーム構成の場合は sonet キーワードを使用して、コントローラのフレーム構成を設定します。 SONET フレーム構成 (sonet) がデフォルトです。

コマンドまたはアクション	目的
ステップ 5 <code>sts number</code> 例: <code>RP/0/0/CPU0:router(config-sonet)# sts 1</code>	<code>number</code> により指定された STS ストリームを設定します。有効値の範囲を次に示します。 <ul style="list-style-type: none"> • 1 ポート OC-3/STM-1 SPA では 1 ~ 3 • 1 ポート チャネライズド OC12->DS0 SPA では 1 ~ 12 • 1 ポート チャネライズド OC48->DS3 SPA では 1 ~ 48
ステップ 6 <code>mode mode</code> 例: <code>RP/0/0/CPU0:router(config-stsPath)# mode t3</code>	STS レベルでのインターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • <code>t3</code> : T3 を伝送する SONET パス • <code>vt15-t1</code> : Virtual Tributary 1.5 T1 (VT15 T1; 仮想トリビュタリ 1.5 T1) を伝送する SONET パス • <code>pos</code> : Packet over SONET (OC12 および OC48 のみ)
ステップ 7 <code>width number</code> 例: <code>RP/0/0/CPU0:router(config-stsPath)# width 3</code>	連結される STS ストリーム数を設定します。 <code>number</code> に設定可能な値を次に示します。 <ul style="list-style-type: none"> • 1 : STS ストリーム数 1 を示します。 • 3 : STS ストリーム数 3 を示します (STS-3c)。 • 12 : 12 個の STS ストリームが連結することを示します (STS-12c)。 • 48 : 48 個の STS ストリームが連結することを示します (STS-48c)。 自然境界の STS パスには、幅 3、12、48 が設定されます。これは、次のパス番号と適合します。 <ul style="list-style-type: none"> • STS-3c では 1、4、7、10 など • STS-12c では 1、13、25、37 • STS-48c では 1
ステップ 8 <code>root</code> 例: <code>RP/0/0/CPU0:router(config-stsPath)# root</code>	グローバル コンフィギュレーション モードに戻ります。
ステップ 9 <code>controller controllerName instance</code> 例: <code>RP/0/0/CPU0:router(config)# controller t3 0/1/0/0/0</code>	コントローラ コンフィギュレーション サブモードを開始して、コントローラ名とインスタンス ID を <code>rack/slot/module/port/controllerName</code> 表記で指定します。コントローラ名を次に示します。 <ul style="list-style-type: none"> • <code>t3</code> : T3 を伝送する SONET パス • <code>vt15-t1</code> : Virtual Tributary 1.5 T1 (VT15 T1; 仮想トリビュタリ 1.5 T1) を伝送する SONET パス
ステップ 10 <code>mode mode</code> 例: <code>RP/0/0/CPU0:router(config-t3)# mode t1</code>	このレベルでのインターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • <code>t1</code> : 28 個の T1 にチャネル化 • <code>e1</code> : 21 個の E1 にチャネル化 • <code>serial</code> : HDLC に類似するペイロードを伝送するクリア チャネル

	コマンドまたはアクション	目的
ステップ 11	<code>root</code> 例： RP/0/0/CPU0:router(config-t3)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 12	<code>controller t1 interface-path-id</code> 例： RP/0/0/CPU0:router(config)# controller t1 0/1/0/0/0/0	T1 コントローラ コンフィギュレーション サブモードを開始して、T1 コントローラ名と <i>interface-path-id</i> を <i>rack/slot/module/port/T3Num/T1num</i> 表記で指定します。
ステップ 13	<code>channel-group number</code> 例： RP/0/0/CPU0:router(config-t1)# channel-group 1	タイム スロットの割り当て先となるチャンネル グループ番号を設定します。有効値の範囲は 1 ~ 24 です。
ステップ 14	<code>timeslots num1:num2:num3:num4</code> or <code>timeslots range1-range2</code> 例： RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1:3:7:9 RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-12	インターフェイスのタイム スロットを <i>num1:num2:num3:num4</i> 表記で数字で指定するか、 <i>range1-range2</i> 表記で範囲として指定します。
ステップ 15	<code>show configuration</code> 例： RP/0/0/CPU0:router(config-t1-channel_group)# show configuration	コミットされていない設定の内容を表示します。
ステップ 16	<code>root</code> 例： RP/0/0/CPU0:router(config-t3)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 17	<code>interface serial interface-path-id</code> 例： RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0	完全なインターフェイス番号を <i>rack/slot/module/port/T3Num/T1num:instance</i> 表記で指定します。
ステップ 18	<code>encapsulation frame-relay hdlc ppp</code> 例： Router(config-if)# encapsulation frame-relay hdlc ppp	カプセル化のタイプを、次のいずれかのキーワードを使用して指定します。 <ul style="list-style-type: none">• frame-relay : フレームリレー ネットワーク プロトコル• hdlc : High-level Data Link Control (HDLC; ハイレベル データリンク コントロール) 同期プロトコル• ppp : ポイントツーポイント プロトコル
ステップ 19	<code>ipv4 ip-address mask</code> 例： Router(config-if)# ip address 10.10.10.10 255.255.255.255	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。

コマンドまたはアクション	目的
ステップ 20 <code>no shutdown</code> 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。
ステップ 21 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router (config-sonet)# end または RP/0/0/CPU0:router (config-sonet)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 22 <code>show controllers sonet interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers sonet 0/1/0/0	SONET コントローラの設定を確認します。

Packet over Sonet チャネルの設定

ここでは、Packet over SONET (POS) チャネルを設定する手順について説明します。

前提条件

- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

POS は、次の SPA でのみサポートされます。

- Cisco 1 ポート チャネライズド DS0/OC-12 共有ポート アダプタ

- Cisco 1 ポート チャネライズド DS3/OC-48 共有ポート アダプタ

手順の概要

1. **configure**
2. **controller sonet** *interface-path-id*
3. **clock source** {**internal** | **line**}
4. **framing** {**sdh** | **sonet**}
5. **sts number**
6. **width number**
7. **mode mode**
8. **root**
9. **interface pos** *interface-path-id*
10. **encapsulation** [**hdlc** | **ppp** | **frame-relay** [**IETF**]]
11. **pos crc** {**16** | **32**}
12. **mtu value**
13. **no shutdown**
14. **end**
または
commit
15. **show interfaces pos** *interface-path-id*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller sonet <i>interface-path-id</i> 例： RP/0/0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション サブモードを開始して、SONET コントローラ名と <i>interface-path-id</i> を <i>rack/slot/module/port</i> 表記で指定します。
ステップ 3	clock source { internal line }	SONET ポート転送クロック ソースを設定します。ここで、 internal キーワードは内部クロック、 line キーワードは回線から回収されたクロックを設定します。 <ul style="list-style-type: none"> • ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。2 つのルータがバックツーバックまたは光ファイバで接続されており、クロッキングが得られない場合は、internal キーワードを使用します。 • デフォルト キーワードは line です。 (注) スペース再利用プロトコル (SRP) インターフェイスでは、内部クロッキングが必要です。

■ チャネライズド SONET の設定方法

コマンドまたはアクション	目的
ステップ 4 <code>framing {sdh sonet}</code> 例: RP/0/0/CPU0:router(config-sonet)# framing sonet	(任意) 同期デジタル ハイアラキー (SDH) フレーム構成の場合は sdh キーワード、SONET フレーム構成の場合は sonet キーワードを使用して、コントローラのフレーム構成を設定します。 SONET フレーム構成 (sonet) がデフォルトです。
ステップ 5 <code>sts number</code> 例: RP/0/0/CPU0:router(config-sonet)# sts 1	<code>number</code> により指定された STS ストリームを設定します。有効値の範囲を次に示します。 <ul style="list-style-type: none"> 1 ポート OC-3/STM-1 SPA では 1 ~ 3 1 ポート チャネライズド OC12->DS0 SPA では 1 ~ 12、1 ポート チャネライズド OC48->DS3 SPA では 1 ~ 48
ステップ 6 <code>width number</code> 例: RP/0/0/CPU0:router(config-stsPath)# width 3	連結される STS ストリーム数を設定します。 <code>number</code> に設定可能な値を次に示します。 <ul style="list-style-type: none"> 3 : STS ストリーム数 3 を示します (STS-3c)。 12 : 12 個の STS ストリームが連結することを示します (STS-12c)。 48 : 48 個の STS ストリームが連結することを示します (STS-48c)。 自然境界の STS パスには、幅 3、12、48 が設定されます。これは、次のパス番号と適合します。 <ul style="list-style-type: none"> STS-3c では 1、4、7、10 など STS-12c では 1、13、25、37 STS-48c では 1 (注) 幅が 1 の場合、POS インターフェイスはサポートされません。
ステップ 7 <code>mode mode</code> 例: RP/0/0/CPU0:router(config-stsPath)# mode pos	STS レベルでのインターフェイスのモードを設定します。POS インターフェイスを作成するために、モードを pos に設定します (OC12 および OC48 のみ)。
ステップ 8 <code>root</code> 例: RP/0/0/CPU0:router(config-stsPath)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 9 <code>interface pos interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface POS 0/1/0/0	POS インターフェイス名と <code>rack/slot/module/port</code> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 10 <code>encapsulation [hdlc ppp frame-relay [IETF]]</code> 例: RP/0/0/CPU0:router(config-if)# encapsulation hdlc	(任意) インターフェイス カプセル化パラメータおよび HDLC や Point-to-Point Protocol (PPP; ポイントツーポイントプロトコル) などの詳細を設定します。 (注) デフォルトのカプセル化は hdlc です。 (注) frame-relay オプションは、Cisco XR 12000 シリーズ ルータでのみ使用可能です。

コマンドまたはアクション	目的
ステップ 11 <code>pos crc {16 32}</code> 例: RP/0/0/CPU0:router(config-if)# pos crc 32	(任意) インターフェイスの Cyclic Redundancy Check (CRC; 巡回冗長検査) 値を設定します。16 ビットの CRC モードを指定するには 16 キーワード、32 ビットの CRC モードを指定するには 32 キーワードを入力します。 デフォルト CRC は 32 です。
ステップ 12 <code>mtu value</code> 例: RP/0/0/CPU0:router(config-if)# mtu 4474	(任意) POS Maximum Transmission Unit (MTU; 最大伝送ユニット) 値を設定します。 有効値の範囲は 64 ~ 65,535 です。
ステップ 13 <code>no shutdown</code> 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。
ステップ 14 <code>end</code> または commit 例: RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 15 <code>show interfaces pos interface-path-id</code> 例: RP/0/0/CPU0:router# show interfaces pos 0/1/0/0	(任意) インターフェイスの設定を表示します。

クリア チャネル T3 の設定

ここでは、SONET 回線をクリア チャネルと呼ばれる 1 つの T3 シリアル チャネルに設定する手順について説明します。

クリア チャネルは、T3 コントローラ モードを `serial` に設定することにより確立されます。

前提条件

- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタでは、クリア チャネルは STS 1 および STS 2 でのみサポートされます。

手順の概要

1. **configure**
2. **controller sonet interface-path-id**
3. **clock source {internal | line}**
4. **framing {sdh | sonet}**
5. **sts number**
6. **mode mode**
7. **root**
8. **controller t3 interface-path-id**
9. **mode mode**
10. **root**
11. **interface serial interface-path-id**
12. **encapsulation frame-relay | hdlc | ppp**
13. **ipv4 ip-address mask**
14. **no shutdown**
15. **end**
または
commit
16. **show controllers sonet interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller sonet interface-path-id 例： RP/0/0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション サブモードを開始して、SONET コントローラ名と <i>interface-path-id</i> を <i>rack/slot/module/port</i> 表記で指定します。

コマンドまたはアクション	目的
ステップ 3 <code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-sonet)# clock source internal	SONET ポート転送クロック ソースを設定します。ここで、 internal キーワードは内部クロック、 line キーワードは回線から回収されたクロックを設定します。 <ul style="list-style-type: none"> ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。2つのルータがバックツーバックまたは光ファイバで接続されており、クロッキングが得られない場合は、internal キーワードを使用します。 デフォルト キーワードは line です。 (注) Spatial Reuse Protocol (SRP; スペース再利用プロトコル) インターフェイスでは、内部クロッキングが必要です。
ステップ 4 <code>framing {sdh sonet}</code> 例: RP/0/0/CPU0:router(config-sonet)# framing sonet	(任意) Synchronous Digital Hierarchy (SDH; 同期デジタル ハイアラキ) フレーム構成の場合は sdh キーワード、SONET フレーム構成の場合は sonet キーワードを使用して、コントローラのフレーム構成を設定します。 SONET フレーム構成 (sonet) がデフォルトです。
ステップ 5 <code>sts number</code> 例: RP/0/0/CPU0:router(config-sonet)# sts 1	<code>number</code> により指定された STS ストリームを設定します。有効値の範囲を次に示します。 <ul style="list-style-type: none"> 1 ポート OC-3/STM-1 SPA では 1 ~ 3 1 ポート チャネライズド OC12->DS0 SPA では 1 ~ 12 1 ポート チャネライズド OC48->DS3 SPA では 1 ~ 48
ステップ 6 <code>mode mode</code> 例: RP/0/0/CPU0:router(config-stsPath)# mode t3	STS レベルでのインターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> t3 : T3 を伝送する SONET パス
ステップ 7 <code>root</code> 例: RP/0/0/CPU0:router(config-stsPath)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 8 <code>controller t3 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0/0	T3 コントローラ コンフィギュレーション サブモードを開始して、T3 コントローラ名と <code>interface-path-id</code> の ID を <code>rack/slot/module/port/T3Num</code> 表記で指定します。
ステップ 9 <code>mode mode</code> 例: RP/0/0/CPU0:router(config-t3)# mode serial	インターフェイスのモードを設定します。クリア チャネルを確立するには、モードを serial に設定します。
ステップ 10 <code>root</code> 例: RP/0/0/CPU0:router(config-t3)# root	グローバル コンフィギュレーション モードに戻ります。

コマンドまたはアクション	目的
ステップ 11 <code>interface serial interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0:0	完全なインターフェイス番号を <code>rack/slot/module/port/T3Num/T1num:instance</code> 表記で指定します。
ステップ 12 <code>encapsulation frame-relay hdlc ppp</code> 例: Router(config-if)# encapsulation frame-relay hdlc ppp	カプセル化のタイプを、次のいずれかのキーワードを使用して指定します。 <ul style="list-style-type: none"> • frame-relay : フレームリレー ネットワーク プロトコル • hdlc : High-level Data Link Control (HDLC; ハイレベル データリンク コントロール) 同期プロトコル • ppp : ポイントツーポイント プロトコル
ステップ 13 <code>ipv4 ip-address mask</code> 例: Router(config-if)# ip address 10.10.10.10 255.255.255.255	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 14 <code>no shutdown</code> 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。
ステップ 15 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 16 <code>show controllers sonet interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers sonet 0/1/0/0	SONET コントローラの設定を確認します。

コマンドまたはアクション	目的
<p>ステップ 17</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 18</p> <pre>show controllers sonet interface-path-id</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# show controllers sonet 0/1/0/0</pre>	<p>SONET コントローラの設定を確認します。</p>

チャネライズド SONET 自動保護スイッチング (APS) の設定

ここでは、チャネライズド SONET 回線で Automatic Protection Switching (APS; 自動保護スイッチング) を設定する手順について説明します。ローカル (ルータ 1 つ) とリモート (ルータ 2 つ) の 2 つのオプションがあります。

前提条件

- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。
- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[SONET APS の設定](#)」に示す SONET APS の設定方法を理解している必要があります。

手順の概要

1. `aps group number`
2. `channel {0 | 1} local sonet interface`
または
`channel {0 | 1} remote ip-address`

3. **channel {0 | 1} local sonet interface**
または
channel {0 | 1} remote ip-address
4. **signalling {sonet | sdh}**
5. **end**
または
commit
6. **show aps**
7. **show aps group [number]**

詳細手順

コマンドまたはアクション	目的
<p>ステップ 1 aps group number</p> <p>例: RP/0/0/CPU0:router(config)# aps group 1</p>	<p>指定した番号を持つ APS グループを追加して、APS グループ コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> • aps group コマンドは、グローバル コンフィギュレーション モードで使用します。 • グループを削除するには、no aps group number のように、このコマンドの no 形式を使用します。有効値の範囲は 1 ~ 255 です。 <p>(注) aps group コマンドを使用するには、aps コマンドの適切なタスク ID に関連付けられたユーザ グループのメンバーでなければなりません。</p> <p>(注) aps group コマンドは、設定する保護グループが 1 つだけの場合でも使用します。</p>
<p>ステップ 2 channel {0 1} local sonet interface or channel {0 1} remote ip-address</p> <p>例: RP/0/0/CPU0:router(config-aps)# channel 0 local SONET 0/0/0/1 or RP/0/0/CPU0:router(config-aps)# channel 0 remote 172.18.69.123</p>	<p>APS グループのチャンネルを作成します。</p> <ul style="list-style-type: none"> • 0 : 保護チャンネルを指定します。 • 1 : アクティブ チャンネルを指定します。 <p>(注) アクティブ チャンネルを割り当てる前に、保護チャンネルを割り当てる必要があります。</p> <p>(注) 両方のチャンネルが 1 つのルータにある APS を設定するには、保護チャンネルとアクティブ チャンネルの両方で channel local コマンドを使用します。2 つの個別のルータを使用し、アクティブ チャンネルと保護チャンネルが別々のルータにある APS を設定するには、保護チャンネルかアクティブ チャンネルのいずれかで channel local コマンドを使用したら、もう一方のチャンネルでは channel remote コマンドを使用します。</p>

コマンドまたはアクション	目的
<p>ステップ 3 <code>channel {0 1} local sonet interface</code> or <code>channel {0 1} remote ip-address</code></p> <p>例: RP/0/0/CPU0:router(config-aps)# channel 1 local SONET 0/0/0/2 or RP/0/0/CPU0:router(config-aps)# channel 1 remote 172.18.69.123</p>	<p>APS グループのチャンネルを作成します。</p> <ul style="list-style-type: none"> • 0 : 保護チャンネルを指定します。 • 1 : アクティブチャンネルを指定します。 <p>(注) アクティブチャンネルの割り当ては、保護チャンネルが割り当てられてから行う必要があります。</p> <p>(注) 両方のチャンネルが1つのルータにあるAPSを設定するには、保護チャンネルとアクティブチャンネルの両方で channel local コマンドを使用します。2つの個別のルータを使用し、アクティブチャンネルと保護チャンネルが別々のルータにあるAPSを設定するには、保護チャンネルかアクティブチャンネルのいずれかで channel local コマンドを使用したら、もう一方のチャンネルでは channel remote コマンドを使用します。</p>
<p>ステップ 4 <code>signalling {sonet sdh}</code></p> <p>例: RP/0/0/CPU0:router(config-aps)# signalling sonet</p>	<p>自動保護スイッチング (APS) で使用される K1K2 オーバーヘッドバイトシグナリングプロトコルを設定します。使用可能なキーワードを次に示します。</p> <ul style="list-style-type: none"> • sonet : シグナリングを SONET に設定します。 • sdh : シグナリングを同期デジタルハイアラキー (SDH) に設定します。
<p>ステップ 5 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

コマンドまたはアクション	目的
ステップ 6 <code>show aps</code> 例: RP/0/0/CPU0:router# show aps	(任意) 設定済みのすべての SONET APS グループの動作ステータスを表示します。
ステップ 7 <code>show aps group [number]</code> 例: RP/0/0/CPU0:router# show aps group 3	(任意) 設定済みの SONET APS グループの動作ステータスを表示します。 (注) 複数のグループを定義する場合は、 <code>show aps group</code> コマンドのほうが <code>show aps</code> コマンドよりも有用です。

SDH AU-3 の設定

ここでは、SDH AU-3 を T1 チャネルに設定する手順について説明します。

前提条件

- 「[Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定](#)」モジュールの「[クリア チャネル SONET コントローラの設定方法](#)」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

チャネライズド SDH は、Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ (SPA) でのみサポートされます。

このリリースでは、AU-3 パスは C11-T1 にのみマッピングできます。

手順の概要

- `configure`
- `controller sonet interface-path-id`
- `clock source {internal | line}`
- `framing {sdh | sonet}`
- `au number`
- `mode mode`
- `root`
- `controller t1 interface-path-id`
- `channel-group number`
- `timeslots num1:num2:num3:num4`
または
`timeslots range1-range2`
- `show configuration`
- `root`
- `interface serial interface-path-id`
- `encapsulation frame-relay | hdlc | ppp`

15. `ipv4 ip-address mask`
16. `no shutdown`
17. `end`
または
`commit`
18. `show controllers sonet interface-path-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller sonet interface-path-id</code> 例： RP/0/0/CPU0:router(config)# <code>controller sonet 0/1/0/0</code>	SONET コントローラ コンフィギュレーション サブモードを開始して、SONET コントローラ名と <code>interface-path-id</code> の ID を <code>rack/slot/module/port</code> 表記で指定します。
ステップ 3	<code>clock source {internal line}</code> 例： RP/0/0/CPU0:router(config-sonet)# <code>clock source internal</code>	SONET ポート転送クロック ソースを設定します。ここで、 internal キーワードは内部クロック、 line キーワードは回線から回収されたクロックを設定します。 <ul style="list-style-type: none"> ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。2つのルータがバックツアバックまたは光ファイバで接続されており、クロッキングが得られない場合は、internal キーワードを使用します。 デフォルト キーワードは line です。 (注) スペース再利用プロトコル (SRP) インターフェイスでは、内部クロッキングが必要です。
ステップ 4	<code>framing {sdh sonet}</code> 例： RP/0/0/CPU0:router(config-sonet)# <code>framing sdh</code>	(任意) 同期デジタル ハイアラキー (SDH) フレーム構成の場合は sdh キーワード、SONET フレーム構成の場合は sonet キーワードを使用して、コントローラのフレーム構成を設定します。 SONET フレーム構成 (sonet) がデフォルトです。
ステップ 5	<code>au number</code> 例： RP/0/0/CPU0:router(config-sonet)# <code>au 3</code>	管理ユニット (AU) の <i>number</i> を指定して、 <code>config-auPath</code> モードを開始します。AU-3 では、 <i>number</i> の有効値は 3 です。
ステップ 6	<code>mode mode</code> 例： RP/0/0/CPU0:router(config-auPath)# <code>mode c11-t1</code>	AU レベルでのインターフェイスのモードを設定します。現在サポートされているのは C11-T1 のみです。
ステップ 7	<code>root</code> 例： RP/0/0/CPU0:router(config-auPath)# <code>root</code>	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	<pre>controller t1 interface-path-id</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config)# controller T1 0/1/0/0/0/0/0</pre>	T1 コントローラ コンフィギュレーション サブモードを開始して、T1 コントローラ名と <i>interface-path-id</i> を <i>rack/slot/module/port/auNum/t1Num</i> 表記で指定します。
ステップ 9	<pre>channel-group number</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-t1)# channel-group 0</pre>	タイム スロットの割り当て先となるチャンネル グループ番号を設定します。有効値の範囲は 1 ~ 28 です。
ステップ 10	<pre>timeslots num1:num2:num3:num4</pre> <p>or</p> <pre>timeslots range1-range2</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-t1-channel_gro up)# timeslots 1:3:7:9 RP/0/0/CPU0:router(config-t1-channel_gro up)# timeslots 1-12</pre>	インターフェイスのタイム スロットを <i>num1:num2:num3:num4</i> 表記で数字で指定するか、 <i>range1-range2</i> 表記で範囲として指定します。
ステップ 11	<pre>show configuration</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-t1-channel_gro up)# show configuration</pre>	コミットされていない設定の内容を表示します。
ステップ 12	<pre>root</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-t3)# root</pre>	グローバル コンフィギュレーション モードに戻ります。
ステップ 13	<pre>interface serial interface-path-id</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0</pre>	完全なインターフェイス番号を <i>rack/slot/module/port/T3Num/T1num:instance</i> 表記で指定します。
ステップ 14	<pre>encapsulation frame-relay hdlc ppp</pre> <p>例:</p> <pre>Router(config-if)# encapsulation frame-relay hdlc ppp</pre>	カプセル化のタイプを、次のいずれかのキーワードを使用して指定します。 <ul style="list-style-type: none"> • frame-relay : フレームリレー ネットワーク プロトコル • hdlc : ハイレベル データリンク コントロール (HDLC) 同期 プロトコル • ppp : ポイントツーポイント プロトコル
ステップ 15	<pre>ipv4 ip-address mask</pre> <p>例:</p> <pre>Router(config-if)# ip address 10.10.10.10 255.255.255.255</pre>	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。

コマンドまたはアクション	目的
<p>ステップ 16 <code>no shutdown</code></p> <p>例： RP/0/0/CPU0:router (config-if)# no shutdown</p>	<p>shutdown 設定を削除します。</p> <p>(注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます（親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします）。</p>
<p>ステップ 17 <code>end</code> または <code>commit</code></p> <p>例： RP/0/0/CPU0:router(config-sonet)# end または RP/0/0/CPU0:router(config-sonet)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 18 <code>show controllers sonet interface-path-id</code></p> <p>例： RP/0/0/CPU0:router# show controllers sonet 0/1/0/0</p>	<p>SONET コントローラの設定を確認します。</p>

SDH AU-4 の設定

ここでは、SDH AU-4 ストリームを E3 にマッピングされた TUG3 チャンネルに設定する手順について説明します。

前提条件

- 「Cisco IOS XR ソフトウェアでのクリア チャンネル SONET コントローラの設定」モジュールの「クリア チャンネル SONET コントローラの設定方法」に示す SONET コントローラの設定方法を理解している必要があります。

制約事項

チャネライズド SDH は、Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポートアダプタ (SPA) のみサポートされます。

このリリースでは、AU-4 パスのチャネル化は TUG3 にのみ行えます。

手順の概要

1. **configure**
2. **controller sonet** *interface-path-id*
3. **clock source** {**internal** | **line**}
4. **framing** {**sdh** | **sonet**}
5. **au** *number*
6. **mode** *mode*
7. **width** *number*
8. **tug3** *number*
9. **mode** *mode*
10. **root**
11. **controller name** *interface-path-id*
12. **mode** *mode*
13. **root**
14. **controller name** *instance*
15. **channel-group** *number*
16. **timeslots** *num1:num2:num3:num4*
または
timeslots *range1-range2*
17. **show configuration**
18. **root**
19. **interface serial** *interface-path-id*
20. **encapsulation** **frame-relay** | **hdlc** | **ppp**
21. **ipv4** *ip-address mask*
22. **no shutdown**
23. **end**
または
commit
24. **show controllers sonet** *interface-path-id*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller sonet interface-path-id 例： RP/0/0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション サブモードを開始して、SONET コントローラ名と <i>interface-path-id</i> を <i>rack/slot/module/port</i> 表記で指定します。
ステップ 3	clock source {internal line} 例： RP/0/0/CPU0:router(config-sonet)# clock source internal	SONET ポート転送クロック ソースを設定します。ここで、 internal キーワードは内部クロック、 line キーワードは回線から回収されたクロックを設定します。 <ul style="list-style-type: none"> ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。2つのルータがバックツーバックまたは光ファイバで接続されており、クロッキングが得られない場合は、internal キーワードを使用します。 デフォルトキーワードは line です。 (注) スペース再利用プロトコル (SRP) インターフェイスでは、内部クロッキングが必要です。
ステップ 4	framing {sdh sonet} 例： RP/0/0/CPU0:router(config-sonet)# framing sdh	(任意) 同期デジタル ハイアラキー (SDH) フレーム構成の場合は sdh キーワード、SONET フレーム構成の場合は sonet キーワードを使用して、コントローラのフレーム構成を設定します。 SONET フレーム構成 (sonet) がデフォルトです。
ステップ 5	au number 例： RP/0/0/CPU0:router(config-sonet)# au 4	管理ユニット (AU) の <i>number</i> を指定して、 config-auPath モードを開始します。AU-4 では、 <i>number</i> の有効値は 4 です。
ステップ 6	mode mode 例： RP/0/0/CPU0:router(config-auPath)# mode tug3	AU レベルでのインターフェイスのモードを設定します。現在サポートされているのは TUG3 のみです。
ステップ 7	width number 例： RP/0/0/CPU0:router(config-auPath)# width 3	AU ストリーム数を設定します。有効値の範囲は 1 ~ 3 です。
ステップ 8	tug3 number 例： RP/0/0/CPU0:router(config-auPath)#tug3 1	Tributary Unit Group (TUG; トリビュタリ ユニット グループ) の <i>number</i> を指定して、 config-tug3Path モードを開始します。有効値の範囲は 1 ~ 3 です。

コマンドまたはアクション	目的
ステップ 9 <code>mode mode</code> 例: RP/0/0/CPU0:router(config-tug3Path)# mode e3	tug3 レベルでのインターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • e3 : E3 を伝送する TUG3 パス • t3 : T3 を伝送する TUG3 パス • c1 : TU-12 を伝送する TUG3 パス • c12-e1 : c12 ~ e1 を伝送する TUG3 パス
ステップ 10 <code>root</code> 例: RP/0/0/CPU0:router(config-tug3Path)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 11 <code>controller name instance</code> 例: RP/0/0/CPU0:router(config)# controller e3 0/1/0/0/0/0	コントローラ コンフィギュレーション サブモードを開始して、コントローラ名とインスタンス ID を <i>rack/slot/module/port/name/instance</i> 表記で指定します。コントローラ名を次に示します。 <ul style="list-style-type: none"> • e3 : E3 を伝送する TUG3 パス • t3 : T3 を伝送する TUG3 パス • e1 : チャネライズド E1 ポート (注) このステップでは、E3 または T3 コントローラを作成してその下にステップ 14 に示すように E1 または T1 チャネルを追加するか、またはこの時点でチャネライズド E1 ポートを作成することができます。
ステップ 12 <code>mode mode</code> 例: RP/0/0/CPU0:router(config-e3)#mode e1	インターフェイスのモードを設定します。使用可能なモードを次に示します。 <ul style="list-style-type: none"> • e1 : 21 個の E1 にチャネル化 • serial : hdlc に類似するペイロードを伝送するクリア チャネル • t1 : 28 個の T1 にチャネル化
ステップ 13 <code>root</code> 例: RP/0/0/CPU0:router(config-e3)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 14 <code>controller name instance</code> 例: RP/0/0/CPU0:router(config)# controller E1 0/1/0/0/0/0/0	コントローラ コンフィギュレーション サブモードを開始して、コントローラ名とインスタンス ID を <i>rack/slot/module/port/name/instance1/instance2</i> 表記で指定します。コントローラ名を次に示します。 <ul style="list-style-type: none"> • e1 : 32 個の E1 にチャネル化 • serial : hdlc に類似するペイロードを伝送するクリア チャネル • t1 : 24 個の T1 にチャネル化

コマンドまたはアクション	目的
ステップ 15 <code>channel-group number</code> 例: RP/0/0/CPU0:router(config-e1)# channel-group 0	タイムスロットの割り当て先となるチャンネルグループ番号を設定します。 <ul style="list-style-type: none"> • t1 の場合、有効値の範囲は 1 ~ 24 です。 • e1 の場合、有効値の範囲は 1 ~ 32 です。
ステップ 16 <code>timeslots num1:num2:num3:num4</code> or <code>timeslots range1-range2</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# timeslots 1:3:7:9 RP/0/0/CPU0:router(config-e1-channel_group)# timeslots 1-12	インターフェイスのタイムスロットを <code>num1:num2:num3:num4</code> 表記で数字で指定するか、 <code>range1-range2</code> 表記で範囲として指定します。
ステップ 17 <code>show configuration</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# show configuration	コミットされていない設定の内容を表示します。
ステップ 18 <code>root</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# root	グローバル コンフィギュレーション モードに戻ります。
ステップ 19 <code>interface serial interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0	完全なインターフェイス番号を <code>rack/slot/module/port/T3Num/T1num:instance</code> 表記で指定します。
ステップ 20 <code>encapsulation frame-relay hdlc ppp</code> 例: Router(config-if)# encapsulation frame-relay hdlc ppp	カプセル化のタイプを、次のいずれかのキーワードを使用して指定します。 <ul style="list-style-type: none"> • frame-relay : フレームリレー ネットワーク プロトコル • hdlc : ハイレベル データリンク コントロール (HDLC) 同期 プロトコル • ppp : ポイントツーポイント プロトコル
ステップ 21 <code>ipv4 ip-address mask</code> 例: Router(config-if)# ip address 10.10.10.10 255.255.255.255	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 22 <code>no shutdown</code> 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。

コマンドまたはアクション	目的
<p>ステップ 23 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router (config-sonet) # <code>end</code> または RP/0/0/CPU0:router (config-sonet) # <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 24 <code>show controllers sonet interface-path-id</code></p> <p>例: RP/0/0/CPU0:router# <code>show controllers sonet 0/1/0/0</code></p>	<p>SONET コントローラの設定を確認します。</p>

チャネライズド SONET の設定例

ここでは、次の例について説明します。

- 「[チャネライズド SONET T3 から T1 への設定 : 例](#)」 (P.114)
- 「[チャネライズド Packet over SONET の設定 : 例](#)」 (P.115)
- 「[クリア チャネル T3 の設定 : 例](#)」 (P.115)
- 「[チャネライズド SONET APS の設定 : 例](#)」 (P.116)
- 「[チャネライズド SDH AU-3 の設定 : 例](#)」 (P.116)
- 「[チャネライズド SDH AU-4 の設定 : 例](#)」 (P.116)

チャネライズド SONET T3 から T1 への設定 : 例

次に、SONET T3 から T1 への設定例を示します。

```
configure
controller sonet 0/1/0/0
clock source internal
framing sonet
sts 1
mode t3
```

```
width 3
root
controller t3 0/1/0/0/0
mode t1
framing auto-detect
root
controller t1 0/1/0/0/0/0
framing esf
channel-group 0
timeslots 1:3:7:9
show configuration
root
interface serial 0/1/0/0/0/0:0
encapsulation frame-relay | hdlc | ppp
ip address 10.10.10.10 255.255.255.255
no shutdown
commit
show controllers sonet 0/1/0/0
```

チャネライズド Packet over SONET の設定 : 例

次に、チャネライズド Packet over SONET の設定例を示します。

```
configure
controller sonet 0/1/0/0
clock source internal
framing sonet
sts 1
mode pos
width 3
root
interface POS 0/1/0/0
encapsulation hdlc
pos crc 32
mtu 4474
no shutdown
commit
show interfaces pos 0/1/0/0
```

クリア チャネル T3 の設定 : 例

次に、SONET クリア チャネルの設定例を示します。

```
configure
controller sonet 0/1/0/0
clock source internal
framing sonet
sts 1
mode t3
root
controller t3 0/1/0/0/0
mode serial
root
interface serial 0/1/0/0/0/0:0
encapsulation frame-relay | hdlc | ppp
ip address 10.10.10.10 255.255.255.255
no shutdown
commit
show controllers sonet 0/1/0/0
```

チャネライズド SONET APS の設定 : 例

次に、SONET ローカル（1 台のルータ）APS の設定例を示します。

```
aps group 1
 channel 0 local SONET 0/0/0/1
 channel 1 local SONET 0/0/0/2
 signalling sonet
 commit
show aps
show aps group 3
```

次に、SONET リモート（2 台のルータ）APS の設定例を示します。

```
aps group 1
 channel 0 local SONET 0/0/0/1
 channel 1 remote 172.18.69.123
 signalling sonet
 commit
show aps
show aps group 3
```

チャネライズド SDH AU-3 の設定 : 例

次に、SDH AU-3 の設定例を示します。

```
configure
 controller sonet 0/1/0/0
  clock source internal
  framing sdh
  au 3
  mode c11-t1
  root
 controller T1 0/1/0/0/0/0/0
  channel-group 0
  timeslots 1-12
  show configuration
  root
 interface serial 0/1/0/0/0/0:0
  encapsulation frame-relay | hdlc | ppp
  ip address 10.10.10.10 255.255.255.255
  no shutdown
  commit
show controllers sonet 0/1/0/0
```

チャネライズド SDH AU-4 の設定 : 例

次に、SDH AU-4 の設定例を示します。

```
configure
 controller sonet 0/1/0/0
  clock source internal
  framing sdh
  au 4
  mode tug3
  width 3
  tug3 1
  mode e3
  root
 controller e3 0/1/0/0/0/0
```

```
mode e1
root
controller E1 0/1/0/0/0/0/0/0
channel-group 0
timeslots 1-12
show configuration
root
interface serial 0/1/0/0/0/0:0
encapsulation frame-relay | hdlc | ppp
ip address 10.10.10.10 255.255.255.255
no shutdown
commit
show controllers sonet 0/1/0/0
```

関連情報

SONET チャネルの設定後は、次に示す項で説明するように、クリア チャネル T3/E3 やチャネライズド T3 コントローラおよびインターフェイスなどの他のコントローラやインターフェイスを設定することができます。

- [「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」](#)

その他の参考資料

ここでは、チャネライズド SONET の設定に関する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュール
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、ルータのインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの高密度波長分割多重コントローラの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータでの Dense Wavelength Division Multiplexing (DWDM; 高密度波長分割多重) コントローラの設定について説明します。DWDM は、既存の光ファイバに基づいて、帯域幅を増やすために使用される光のテクノロジーです。DWDM は、サポートされる 10-Gigabit Ethernet (GE; ギガビットイーサネット) または Packet-over-SONET/SDH Physical Layer Interface Module (PLIM; 物理レイヤ インターフェイス モジュール) で設定できます。DWDM コントローラを設定した後は、関連する POS または 10 ギガビットイーサネット インターフェイスを設定できます。

POS インターフェイスを設定する方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。

10 ギガビットイーサネット インターフェイスを設定する方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのイーサネット インターフェイスの設定](#)」モジュールを参照してください。

DWDM コントローラ インターフェイス設定の機能履歴

リリース	変更点
リリース 3.3.0	Cisco CRS-1 ルータにこの機能が追加されました。 OC-768c/STM-256c DWDM PLIM および 10-GE DWDM PLIM のサポートが追加されました。
リリース 3.4.0	レーザー、TTI 文字列、BDI 挿入だけでなく、パフォーマンス モニタリングのサポートも追加されました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「[DWDM コントローラ インターフェイスを設定するための前提条件](#)」 (P.122)
- 「[DWDM コントローラに関する情報](#)」 (P.122)
- 「[DWDM コントローラの設定方法](#)」 (P.123)
- 「[DWDM コントローラでパフォーマンス モニタリングを実行する方法](#)」 (P.128)
- 「[その他の参考資料](#)」 (P.134)

DWDM コントローラ インターフェイスを設定するための前提条件

DWDM コントローラを設定する前に、次のタスクと条件を満たしていることを確認します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 使用している PLIM は DWDM をサポートしています。OC-768c/STM-256c DWDM PLIM および 10-GE DWDM PLIM は DWDM をサポートしています。

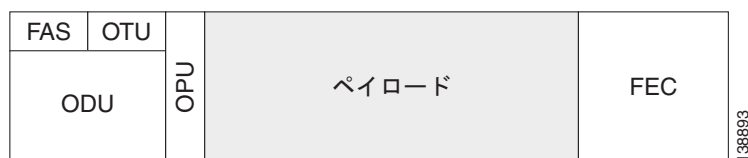
DWDM コントローラに関する情報

Cisco IOS XR ソフトウェアの DWDM のサポートは、ITU-T G.709 に規定されている Optical Transport Network (OTN; 光トランスポート ネットワーク) プロトコルに基づいています。この規格は、SONET/SDH テクノロジーと DWDM の多波長ネットワークの利点を兼ね備えています。また、使用するリジェネレータの数を減らすことで、ネットワーク コストを減らすことができる、Forward Error Correction (FEC; 前方誤り訂正) の機能も備えています。

マルチサービス トランスポートを使用するために、OTN はラップされた OverHead (OH; オーバーヘッド) の概念を使用します。この構造について説明します。

- Optical channel Payload Unit (OPU; 光チャネル ペイロード ユニット) の OH 情報が情報ペイロードに追加され、OPU が形成されます。OPU OH には、クライアント信号のアダプテーションをサポートする情報が含まれます。
- Optical channel Data Unit (ODU; 光チャネル データ ユニット) の OH が OPU に追加され、ODU が形成されます。ODU OH には、光チャネルをサポートするメンテナンス機能と操作機能の情報が含まれます。
- Optical channel Transport Unit (OTU; 光チャネル トランスポート ユニット) の OH と FEC が追加され、OTU が形成されます。OTU OH には、1 つまたは複数の光チャネル接続を経由するトランスポートをサポートする操作機能の情報が含まれます。
- Optical Channel (OCh; 光チャネル) の OH が追加され、OCh が形成されます。OCh には OTN 管理機能があり、OPU、ODU、OTU、および Frame Alignment Signal (FAS; フレーム整列信号) という 4 つのパートが含まれます。図 7 を参照してください。

図 7 OTN 光チャネルの構造



DWDM コントローラの設定方法

DWDM コントローラは、Cisco IOS XR ソフトウェアのコンフィギュレーション スペースの物理層のコントロール要素で設定します。この設定を実行するには、**controller dwdm** コマンドを使用します。設定については、次のタスクで説明します。

- 「光パラメータの設定」(P.123)
- 「G.709 パラメータの設定」(P.125)



(注) POS または GE インターフェイスのすべてのインターフェイス コンフィギュレーション タスクは、インターフェイス コンフィギュレーション モードで実行する必要があります。詳細については、「*Configuring POS Interfaces on Cisco IOS XR Software*」モジュールおよび「*Configuring Ethernet Interfaces on Cisco IOS XR Software*」モジュールを参照してください。

光パラメータの設定

ここでは、DWDM コントローラの受信電力のしきい値および波長パラメータの設定方法について説明します。DWDM インストールの光パラメータが正しく設定されていることを確認し、必要に応じて次のタスクを実行します。

前提条件

rx-los-threshold、**wavelength**、および **transmit-power** の各コマンドを使用できるのは、コントローラがシャットダウン状態の場合のみです。**shutdown** コマンドを使用してください。

制約事項

伝送パワー レベルとレシーブ LOS のしきい値は、OC-768c/STM-256c DWDM PLIM でのみ設定できます。

手順の概要

1. **configure**
2. **controller dwdm interface-path-id**
3. **shutdown**
4. **commit**
5. **rx-los-threshold power-level**
6. **wavelength channel-number**
7. **transmit-power power-level**
8. **no shutdown**
9. **laser {on | off}**
10. **end**
または
commit
11. **show controllers dwdm interface-path-id [optics | wavelength-map]**

DWDM コントローラの設定方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller dwdm interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# controller dwdm 0/1/0/0	<code>rack/slot/module/port</code> 表記で DWDM コントローラ名を指定し、DWDM コンフィギュレーション モードを開始します。
ステップ 3	<code>shutdown</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# shutdown	DWDM コントローラをディセーブルにします。DWDM コンフィギュレーション コマンドを使用する前に、コントローラをディセーブルにする必要があります。
ステップ 4	<code>commit</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# commit	設定変更を保存します。これで、前のステップのシャットダウンが実行されます。コントローラがシャットダウンすると、設定を進めることができます。
ステップ 5	<code>rx-los-threshold power-level</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# rx-los-threshold -10	トランスポンダの受信電力のしきい値を設定します。値の単位は 0.1 dBm で、範囲は -350 ~ 50 です。これは -35 dBm ~ 5 dBm の範囲に対応します。
ステップ 6	<code>wavelength channel-number</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# wavelength 1	最初の波長に対応するチャンネル番号を設定します。値の範囲は 1 ~ 185 ですが、すべてのチャンネルはすべての PLIM でサポートされます。特定のコントローラでサポートするチャンネルおよび波長を決定するには、 show controller dwdm コマンドと wavelength-map キーワードを使用します。 (注) 同じ PLIM 上の別のポート、または同じシステムの別の PLIM で選択した波長が使用されているかどうかを決定するクロスチェック機能はありません。
ステップ 7	<code>transmit-power power-level</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# transmit-power 10	トランスポンダの伝送パワーを設定します。値の単位は 0.1 dBm で、範囲は -190 ~ +10 です。これは -19 dBm ~ +1 dBm の範囲に対応します。
ステップ 8	<code>no shutdown</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# no shutdown	<code>shutdown</code> 設定を削除します。 • <code>shutdown</code> 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 9	<code>laser off</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# laser off	レーザーをオフにします。コントローラが G.709 以外のモードで <code>shutdown</code> 状態の場合、またはコントローラが <code>internal loopback</code> モードの場合、レーザーをオンにすることはできません。

コマンドまたはアクション	目的
<p>ステップ 10</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# end または RP/0/RP0/CPU0:router(config-dwdm)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 11</p> <pre>show controllers dwdm interface-path-id [optics wavelength-map]</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show controller dwdm 0/1/0/0 optics</pre>	<p>出力電力レベル、入力電力レベル、波長、およびレーザーバイアス電流のモニタリング情報を表示します。</p>

トラブルシューティングのヒント

アップ状態で設定変更をコントローラにコミットしようとする、エラーメッセージが表示されます。DWDM コンフィギュレーション コマンドを使用する前に、コントローラをシャットダウンする必要があります。

G.709 パラメータの設定

ここでは、アラートと FEC のアラーム表示およびしきい値をカスタマイズする方法について説明します。デフォルト値が実際のインストールに合っていない場合にのみ、このタスクを使用してください。

前提条件

g709 disable, loopback, コマンドおよび **g709 fec** コマンドは、コントローラがシャットダウン状態の場合にのみ使用できます。**shutdown** コマンドを使用してください。

手順の概要

1. **configure**
2. **controller dwdm interface-path-id**

DWDM コントローラの設定方法

3. `shutdown`
4. `commit`
5. `g709 disable`
6. `loopback internal`
7. `g709 fec {disable | enhanced | standard}`
8. `g709 {odu | otu} alarm disable`
9. `g709 otu overhead tti {expected | sent} {ascii | hex} tti-string`
10. `no shutdown`
11. `end`
または
`commit`
12. `show controllers dwdm interface-path-id g709`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller dwdm interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# <code>controller dwdm 0/1/0/0</code>	<code>rack/slot/module/port</code> 表記で DWDM コントローラ名を指定し、DWDM コンフィギュレーション モードを開始します。
ステップ 3	<code>shutdown</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# <code>shutdown</code>	DWDM コントローラをディセーブルにします。DWDM コンフィギュレーション コマンドを使用する前に、コントローラをディセーブルにする必要があります。
ステップ 4	<code>commit</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# <code>commit</code>	設定変更を保存します。これで、前のステップのシャットダウンが実行されます。コントローラがシャットダウンすると、設定を進めることができます。
ステップ 5	<code>g709 disable</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# <code>g709 disable</code>	(任意) G.709 ラッパーをディセーブルにします。このラッパーはデフォルトでイネーブルです。 (注) <code>g709 disable</code> コマンドは、10-GE DWDM PLIM のみ使用できます。
ステップ 6	<code>loopback {internal line}</code> 例: RP/0/RP0/CPU0:router(config-dwdm)# <code>loopback internal</code>	(任意) ループバック モードの DWDM コントローラを設定します。

	コマンドまたはアクション	目的
ステップ 7	<pre>g709 fec {disable standard}</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# g709 fec disable</pre>	(任意) DWDM コントローラの FEC を設定します。デフォルトでは、拡張 FEC がイネーブルです。
ステップ 8	<pre>g709 {odu otu} alarm disable</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# g709 odu bdi disable</pre>	(任意) DWDM コントローラのコンソールに対する、選択した ODU アラームまたは OTU アラームのロギングをディセーブルにします。デフォルトでは、すべてのアラームがコンソールにロギングされます。
ステップ 9	<pre>g709 otu overhead tti {expected sent} {ascii hex} tti-string</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# g709 otu overhead tti expected ascii test OTU 5678</pre>	show controller dwdm コマンドで表示される伝送または予想の Trail Trace Identifier (TTI) を設定します。
ステップ 10	<pre>no shutdown</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# no shutdown</pre>	shutdown 設定を削除します。 <ul style="list-style-type: none"> shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 11	<pre>end</pre> <p>または</p> <pre>commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# end</pre> <p>または</p> <pre>RP/0/RP0/CPU0:router(config-dwdm)# commit</pre>	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 12	<pre>show controllers dwdm interface-path-id g709</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show controller dwdm 0/1/0/0 optics</pre>	G.709 OTN プロトコルのアラームおよびビット エラーのカウンタと共に、FEC 統計情報としきい値ベースのアラートを表示します。

例

次の例は、コンフィギュレーション コマンドを使用する前に、DWDM コントローラをダウン状態にする方法です。

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# controller dwdm 0/0/0/0
RP/0/RP0/CPU0:Router(config-dwdm)# shutdown
RP/0/RP0/CPU0:Router(config-dwdm)# commit
RP/0/RP0/CPU0:Router(config-dwdm)# rx-los-threshold 0
RP/0/RP0/CPU0:Router(config-dwdm)# wavelength 1
RP/0/RP0/CPU0:Router(config-dwdm)# transmit-power 0
RP/0/RP0/CPU0:Router(config-dwdm)# no shutdown
RP/0/RP0/CPU0:Router(config-dwdm)# end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

RP/0/RP0/CPU0:Oct 15 12:35:54.299 : config[65732]: %MGBL-LIBTARCFG-6-COMMIT :
Configuration committed by user 'lab'. Use 'show configuration commit changes
1000000312' to view the changes.
RP/0/RP0/CPU0:Oct 15 12:35:54.403 : config[65732]: %MGBL-SYS-5-CONFIG_I : Configured from
console by lab
```

次の例は、アラートおよび FEC のアラーム表示としきい値をカスタマイズする方法です。

```
RP/0/RP0/CPU0:routerr# configure
RP/0/RP0/CPU0:routerr(config)# controller dwdm 0/1/0/0
RP/0/RP0/CPU0:routerr(config-dwdm)# shutdown
RP/0/RP0/CPU0:routerr(config-dwdm)# commit
RP/0/RP0/CPU0:routerr(config-dwdm)# g709 disable
RP/0/RP0/CPU0:routerr(config-dwdm)# loopback internal
RP/0/RP0/CPU0:routerr(config-dwdm)# g709 fec standard
RP/0/RP0/CPU0:routerr(config-dwdm)# g709 odu bdi disable
RP/0/RP0/CPU0:routerr(config-dwdm)# no shutdown
RP/0/RP0/CPU0:routerr(config-dwdm)# commit
```

次に行う作業

POS または GE インターフェイスのすべてのインターフェイス コンフィギュレーション タスクは、インターフェイス コンフィギュレーション モードで実行する必要があります。詳細については、「*Configuring POS Interfaces on Cisco IOS XR Software*」モジュールおよび「*Configuring Ethernet Interfaces on Cisco IOS XR Software*」モジュールを参照してください。

DWDM コントローラでパフォーマンス モニタリングを実行する方法

パフォーマンス モニタリング パラメータは、問題を早期に検出するためのパフォーマンス データを収集および格納し、しきい値を設定し、報告するために使用されます。しきい値は、各パフォーマンス モニタリング パラメータのエラー レベルを設定するために使用されます。蓄積サイクルで、パフォー

パフォーマンス監視 モニタリング パラメータの現在の値が、対応するしきい値に達した場合、または超過した場合、Threshold Crossing Alert (TCA; しきい値超過アラート) を生成できます。TCA によって、パフォーマンス低下を早期に検出できます。

パフォーマンス モニタリングの統計情報は 15 分ベースで蓄積され、各 15 分の開始時に同期されます。また、深夜 12 時に始まる日次単位でも統計情報は蓄積されます。履歴カウントは、33 回の 15 分インターバルと 2 回の日次インターバルで維持されます。

パフォーマンス モニタリングについては、次のタスクで説明します。

- 「DWDM コントローラのパフォーマンス モニタリングの設定」(P.129)

DWDM コントローラのパフォーマンス モニタリングの設定

ここでは、DWDM コントローラでパフォーマンス モニタリングを設定する方法とパフォーマンス パラメータを表示する方法について説明します。

手順の概要

1. `configure`
2. `controller dwdm interface-path-id`
3. `pm {15-min | 24-hour} fec threshold {ec-bits | uc-words} threshold`
4. `pm {15-min | 24-hour} optics threshold {lbc | opr | opt} {max | min} threshold`
5. `pm {15-min | 24-hour} otn threshold otn-parameter threshold`
6. `pm {15-min | 24-hour} fec report {ec-bits | uc-words} enable`
7. `pm {15-min | 24-hour} optics report {lbc | opr | opt} {max-tca | min-tca} enable`
8. `pm {15-min | 24-hour} otn report otn-parameter enable`
9. `end`
10. `show controllers dwdm interface-path-id pm history [15-min | 24-hour | fec | optics | otn]`
11. `show controllers dwdm interface-path-id pm interval {15-min | 24-hour} [fec | optics | otn] index`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller dwdm interface-path-id</code> 例： RP/0/RP0/CPU0:router(config)# <code>controller dwdm 0/1/0/0</code>	<code>rack/slot/module/port</code> 表記で DWDM コントローラ名を指定し、DWDM コンフィギュレーション モードを開始します。

DWDM コントローラでパフォーマンス モニタリングを実行する方法

コマンドまたはアクション	目的
<p>ステップ 3 <code>pm {15-min 24-hour} fec threshold {ec-bits uc-words} threshold</code></p> <p>例: RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min fec threshold ec-bits 49000000 RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min fec threshold uc-words xxxxxxx</p>	<p>FEC 層で特定のパラメータのパフォーマンス モニタリングを設定します。</p>
<p>ステップ 4 <code>pm {15-min 24-hour} optics threshold {lbc opr opt} {max min} threshold</code></p> <p>例: RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min optics threshold opt max xxx RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min optics threshold lbc min xxx</p>	<p>光ファイバ層で特定のパラメータのパフォーマンス モニタリングを設定します。</p>

コマンドまたはアクション	目的
<p>ステップ 5 <code>pm {15-min 24-hour} otn threshold</code> <code>otn-parameter threshold</code></p> <p>例 : RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min otn threshold bbe-pm-ne xxx RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min otn threshold es-sm-fe xxx</p>	<p>OTN 層で特定のパラメータのパフォーマンス モニタリングを設定します。次の OTN パラメータを指定できます。</p> <ul style="list-style-type: none"> • bbe-pm-fe - 遠端のパス モニタリングのバックグラウンドブロック エラー (BBE-PM) • bbe-pm-ne - 近端のパス モニタリングのバックグラウンドブロック エラー (BBE-PM) • bbe-sm-fe - 遠端のセクション モニタリングのバックグラウンドブロック エラー (BBE-SM) • bbe-sm-ne - 近端のセクション モニタリングのバックグラウンドブロック エラー (BBE-SM) • bber-pm-fe - 遠端のパス モニタリングのバックグラウンドブロック エラー率 (BBER-PM) • bber-pm-ne - 近端のパス モニタリングのバックグラウンドブロック エラー率 (BBER-PM) • bber-sm-fe - 遠端のセクション モニタリングのバックグラウンドブロック エラー率 (BBER-SM) • bber-sm-ne - 近端のセクション モニタリングのバックグラウンドブロック エラー率 (BBER-SM) • es-pm-fe - 遠端のパス モニタリングのエラー秒数 (ES-PM) • es-pm-ne - 近端のパス モニタリングのエラー秒数 (ES-PM) • es-sm-fe - 遠端のセクション モニタリングのエラー秒数 (ES-SM) • es-sm-ne - 近端のセクション モニタリングのエラー秒数 (ES-SM) • esr-pm-fe - 遠端のパス モニタリングのエラー秒数比 (ESR-PM) • esr-pm-ne - 近端のパス モニタリングのエラー秒数比 (ESR-PM) • esr-sm-fe - 遠端のセクション モニタリングのエラー秒数比 (ESR-SM) • esr-sm-ne - 近端のセクション モニタリングのエラー秒数比 (ESR-SM) • fc-pm-fe - 遠端のパス モニタリングの障害カウント (FC-PM) • fc-pm-ne - 近端のパス モニタリングの障害カウント (FC-PM) • fc-sm-fe - 遠端のセクション モニタリングの障害カウント (FC-SM) • fc-sm-ne - 近端のセクション モニタリングの障害カウント (FC-SM)

DWDM コントローラでパフォーマンス モニタリングを実行する方法

コマンドまたはアクション	目的
	<ul style="list-style-type: none"> • ses-pm-fe - 遠端のパス モニタリングの重大エラー秒数 (SES-PM) • ses-pm-ne - 近端のパス モニタリングの重大エラー秒数 (SES-PM) • ses-sm-fe - 遠端のセクション モニタリングの重大エラー秒数 (SES-SM) • ses-sm-ne - 近端のセクション モニタリングの重大エラー秒数 (SES-SM) • sesr-pm-fe - 遠端のパス モニタリングの重大エラー秒数比 (SESR-PM) • sesr-pm-ne - 近端のパス モニタリングの重大エラー秒数比 (SESR-PM) • sesr-sm-fe - 遠端のセクション モニタリングの重大エラー秒数比 (SESR-SM) • sesr-sm-ne - 近端のセクション モニタリングの重大エラー秒数比 (SESR-SM) • uas-pm-fe - 遠端のパス モニタリングの使用不可秒数 (UAS-PM) • uas-pm-ne - 近端のパス モニタリングの使用不可秒数 (UAS-PM) • uas-sm-fe - 遠端のセクション モニタリングの使用不可秒数 (UAS-SM) • uas-sm-ne - 近端のセクション モニタリングの使用不可秒数 (UAS-SM)
ステップ 6 <code>pm {15-min 24-hour} fec report {ec-bits uc-words} enable</code> 例: <pre>RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min fec report ec-bits enable RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min fec report uc-words enable</pre>	FEC 層で特定のパラメータについて TCA の生成を設定します。
ステップ 7 <code>pm {15-min 24-hour} optics report {lbc opr opt} {max-tca min-tca} enable</code> 例: <pre>RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min optics report opt enable RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min optics report lbc enable</pre>	光ファイバ層で特定のパラメータについて TCA の生成を設定します。
ステップ 8 <code>pm {15-min 24-hour} otn report otn-parameter enable</code> 例: <pre>RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min otn report bbe-pm-ne enable RP/0/RP0/CPU0:router(config-dwdm)# pm 15-min otn report es-sm-fe enable</pre>	OTN 層で特定のパラメータについて TCA の生成を設定します。OTN パラメータについては、 P.131 の ステップ 5 を参照してください。

	コマンドまたはアクション	目的
ステップ 9	<pre>end または commit 例： RP/0/RP0/CPU0:router(config-dwdm)# end または RP/0/RP0/CPU0:router(config-dwdm)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 10	<pre>show controllers dwdm interface-path-id pm history [15-min 24-hour fec optics otn]</pre> <p>例：</p> <pre>RP/0/RP0/CPU0:router# show controllers dwdm 0/2/0/0 pm history 24-hour fec RP/0/RP0/CPU0:router# show controllers dwdm 0/2/0/0 pm history</pre>	DWDM コントローラに関するすべてのパフォーマンス測定情報と TCA 生成情報を表示します。
ステップ 11	<pre>show controllers dwdm interface-path-id pm interval {15-min 24-hour} [fec optics otn] index</pre> <p>例：</p> <pre>RP/0/RP0/CPU0:router# show controllers dwdm 0/2/0/0 pm interval 24-hour 0 RP/0/RP0/CPU0:router# show controllers dwdm 0/2/0/0 pm interval 15-min optics 1</pre>	特定のインターバルに関するパフォーマンス測定情報と TCA 生成情報を表示します。

例

次の例は、**optics** パラメータのパフォーマンス モニタリングを設定し、設定と現在の統計情報を表示する方法です。

```
RP/0/RP1/CPU0:roma# config
RP/0/RP1/CPU0:roma(config)# controller dwdm 0/2/0/0

RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold opt max 2000000
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold opt min 200
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold lbc max 3000000
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold lbc min 300
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold opr max 4000000
```

```

RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics threshold opr min 400
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report opt max-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report opt min-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report opr max-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report opr min-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report lbc max-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# pm 15-min optics report lbc min-tca enable
RP/0/RP1/CPU0:roma(config-dwdm)# exit
RP/0/RP1/CPU0:roma(config)# exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:y

LC/0/2/CPU0:Jul 12 04:10:47.252 : plim_4p_10ge_dwdm[194]: %L1-PMENGINE-4-TCA : Port DWDM
0/2/0/0 reports OPTICS TX-PWR-MIN(NE) PM TCA with current value 0, threshold 200 in
current 15-min interval window
LC/0/2/CPU0:Jul 12 04:10:47.255 : plim_4p_10ge_dwdm[194]: %L1-PMENGINE-4-TCA : Port DWDM
0/2/0/0 reports OPTICS RX-PWR-MIN(NE) PM TCA with current value 68, threshold 400 in
current 15-min interval window
RP/0/RP1/CPU0:Jul 12 04:09:05.443 : config[65678]: %MGBL-CONFIG-6-DB_COMMIT :
Configuration committed by user 'lab'. Use 'show configuration commit changes 1000000001'
to view the changes.
RP/0/RP1/CPU0:Jul 12 04:09:05.604 : config[65678]: %MGBL-SYS-5-CONFIG_I : Configured from
console by lab

RP/0/RP1/CPU0:roma# show controllers dwdm 0/2/0/0 pm interval 15-min optics 0

Optics in the current interval [ 4:15:00 - 04:26:02 Wed Jul 12 2006]
          MIN      AVG      MAX  Threshold  TCA  Threshold  TCA
                   (min)  (enable)  (max)  (enable)
LBC[mA ] :   3605    4948    6453     300      YES    3000000    YES
OPT[uW]  :   2593    2593    2593     200      YES    2000000    YES
OPR[uW]  :    69      69      70      400      YES    4000000    YES

```

その他の参考資料

ここでは、DWDM コントローラ設定に関する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システムブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
ITU-T G.709/Y.1331	Interfaces for the optical transport network (OTN)

MIB

MIB	MIB リンク
この機能によりサポートされた新規 MIB または改訂 MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル セント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR の診断の実行

ここでは、Cisco IOS XR のオンライン診断について説明します。この機能を使用すると、稼働中のネットワークに接続したままでハードウェアの機能をテストおよび検証できます。

Cisco IOS XR 診断の機能履歴

リリース	変更点
リリース 3.3.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.4.0	<p>モニタ syslog、モニタ間隔、テストの障害カウントしきい値のサポートなど、ヘルス モニタリング診断の設定が追加されました。</p> <p>show diagnostic result コマンドおよび show diagnostic content コマンドの出力は、診断のモニタリングのサポートを含むように変更されました。</p> <p>FDIAG RUNNING ステートのノードに関する show diagnostic content コマンドの出力は、オフラインテストスイートの実行をより制御できるように拡張されました。</p> <p>次のオンライン診断テストが追加されました。</p> <ul style="list-style-type: none">• Control Ethernet Inactive Link Test• Self-Ping over Fabric• RommonRevision• Fabric Diagnostic Test
リリース 3.5.0	<p>次のコマンドがサポートされました。</p> <ul style="list-style-type: none">• diagnostic ondemand action-on-failure• diagnostic ondemand iterations• show diagnostics ondemand settings• show diagnostic status <p>物理レイヤ インターフェイス モジュール (PLIM) テストのサポートが追加されました。</p> <p>次のオンライン診断テストがサポートされました。</p> <ul style="list-style-type: none">• File System Functionality Verification Test• Scratch Register Test
リリース 3.6.0	変更ありません。
リリース 3.6.2	Fabric Multicast Diagnostic Test がサポートされ、Fabric Diagnostic Test が強化されました。

リリース 3.7.0	Cisco CRS-1 ルータについて、オンライン診断が Diagnostics パッケージから Base パッケージに移行されました（オフライン診断は Diagnostics パッケージのままです）。
リリース 3.8.0	Cisco CRS-1 ルータの次の機能のサポートが追加されました。 <ul style="list-style-type: none"> ヘルス モニタリング診断でテスト番号ではなくテスト名。 Modular Service Card (MSC; モジュラ サービス カード) の最小限のブートアップ診断。 MSC のメンテナンス モード。

この章の構成

- 「診断を実行するための前提条件」 (P.138)
- 「診断の実行に関する制約事項」 (P.138)
- 「使用できるオンライン診断テスト」 (P.139)

診断を実行するための前提条件

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンドタスク ID は、各コマンドリファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。
タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。
- Cisco CRS-1 ルータでオンライン診断を実行するには、ルータに **Base** パッケージをロードし、アクティブ化する必要があります。
- Cisco CRS-1 ルータでオフライン診断を実行するには、ルータに **Diagnostics** パッケージをロードし、アクティブ化する必要があります。

診断の実行に関する制約事項

- Cisco XR 12000 シリーズ ルータは、**diag** という診断コマンドをサポートします。**diagnostic** コマンドおよび **show diagnostic** コマンドが Cisco XR 12000 シリーズ ルータの **help** ストリングに表示されますが、このルータではサポートしていません。Cisco XR 12000 シリーズ ルータで **diagnostic** コマンドおよび **show diagnostic** コマンドを実行すると、エラー メッセージが表示されるか、要求した出力が表示されません。
- diag** コマンドが Cisco XR 12000 シリーズ ルータの **help** ストリングに表示されるのは、**c12k-diags.pie** をロードし、アクティブ化した場合のみです。**c12k-diags.pie** がロードされていない場合、**help** ストリングに「**diag**」と入力すると **diagnostic** コマンドにマップされますが、**diagnostic** コマンドは Cisco XR 12000 シリーズ ルータではサポートされません。
- Cisco CRS-1 のオンライン診断は、共有ポート アダプタ (SPA) をサポートしません。

- Cisco CRS-1 のオフライン診断の場合、SPA、PLIM、および Service Processor (SP; サービスプロセッサ) のサポートは MSC ノードからのみ使用できます。つまり、SPA、PLIM、または SP でオフライン診断を実行するには、R/S/CPU0 ノードにロードしている必要があります。

使用できるオンライン診断テスト

Cisco CRS-1 ルータでは、次のオンライン診断テストがサポートされます。

- **Control Ethernet Ping Test** - テストを開始するノードからシャーシ内の各 Control Ethernet ノードに「ping」を実行する、中断を伴わないテスト。このテストは同時に 1 つのノードに ping を 1 回送信し、応答が返されるか、2 秒の最大タイムアウトに達するまで待機してから、次のノードへの ping 送信処理に進みます。返された ping 応答は、送信した ping とバイト単位で比較して検証されます。ping の送信対象は、テストを実行するノードと同じシャーシ内にあるアクティブなノードのみです。1 つのノードにつき ping が 1 度だけ送信されます。各 ping のペイロードは 100 バイトです。すべてのノードが ping に応答し、応答が送信した ping と一致する場合、テスト結果は PASS です。
- **Fabric Ping Test** - テストを開始したノードからシャーシ内のファブリック上の各ノードに「ping」を実行する、中断を伴わないテスト。このテストは同時に 1 つのノードに ping を 1 回送信し、応答が返されるか、2 秒の最大タイムアウトに達するまで待機してから、次のノードへの ping 送信処理に進みます。返された ping 応答は、送信した ping とバイト単位で比較して検証されます。ping の送信対象は、テストを実行するノードと同じシャーシ内にあるアクティブなノードのみです。ファブリック内にはトラフィックの未確定のパスがあるため、カバレッジを最大限にするために、1 ノードにつき 72 個の ping が送信されます。各 ping のペイロードは 1 キロバイトです。すべてのノードがすべての ping に応答し、すべての応答が送信した ping と一致する場合、テスト結果は PASS です。応答がない ping には 2 秒間の待機タイムアウトがあるため、到達不能のノードと断続的に機能するノードは、テストの合計実行時間に影響を及ぼします。そのため、ノードがすべてのファブリックの接続を失うという最悪の場合、そのノードのテスト時間は最長 2.5 分に達する可能性があります。合計テスト時間は、テスト対象のラックに含まれるアクティブノードの数と、失敗するファブリック接続があるノードの数に左右されます。
- **Control Ethernet Inactive Link Test** - 同じラック内のスタンバイ Route Processor (RP; ルートプロセッサ) とその他すべてのノード間の非アクティブな Control Ethernet リンクを検証する、中断を伴わないテスト。RP でのみこのテストを利用できます。また、スタンバイ RP でのみ開始できます。このテストは、非アクティブな Control Ethernet (CE) リンクを使用して、「ping」パケットをテスト対象ノードに送信します。各ノードからは、次の 3 つの応答が返されます。
 - 非アクティブな CE の返信リンクに沿って返される応答。テスト対象からスタンバイ RP への非アクティブな CE リンクを検証します。
 - アクティブな RP CPU を経由してスタンバイ RP へのアクティブな CE リンクに沿って返される応答。スタンバイ RP からテスト対象への外部の非アクティブ CE リンクを検証します。
 - スタンバイ RP へのアクティブな CE リンクに沿って返される応答。内部のテスト対象 CE パスを検証します。返された各応答は、送信した ping とバイト単位で比較して検証されます。テスト対象から 3 つの応答がすべて返されるか、2 秒のタイムアウトに達すると、ラック内の次のノードに対して同じ手順を繰り返します。非アクティブな CE リンクに接続するすべてのノードがテストされるまで手順は繰り返されます。すべてのノードがすべての ping に応答し、すべての応答が送信した ping と一致する場合、テスト結果は PASS です。このテストは、切り替え前に、スタンバイ RP の CE の接続を「有効と判断する」ために使用できます。
- **Self-Ping over Fabric Test** - ノードが自身に対してファブリック上で「ping」を実行する、中断を伴わないテスト。このテストは、ファブリック ping をノード自身に送信し、応答が返されるか、2 秒の最大タイムアウトに達するまで待機します。返された ping 応答は、送信した ping とバイト単

位で比較して検証されます。各 ping のペイロードは 100 バイトです。この単一の ping 送信は、300 ms 間隔で 100 回繰り返されます。すべての ping に応答があり、すべての応答が送信した ping と一致する場合、テスト結果は PASS です。このテストの通常の実行時間は 30 秒です。応答がない ping には 2 秒間の待機タイムアウトがあるため、失敗したファブリックの接続があるノードのテスト時間は、最長 3.5 分に達する可能性があります。

- **RommonRevision Test** - ノードがサポートする最低限のバージョンの ROM Monitor (ROMMON) を実行していることを検証する、中断を伴わないテスト。ノードをリブートすると、現在のバージョンの ROMMON が取得され、共有メモリ スペースに保存されます。この共有メモリ スペースには ROMMON の実行バージョンがクエリーされ、そのバージョンは、サポートする最低限のバージョンの ROMMON と比較されます。実行バージョンが最低限のバージョン以下の場合、テストは失敗します。
- **File System Functionality Verification Test** - ハードディスクまたはフラッシュ ディスク上のファイルの作成、書き込み、読み取り、削除を行う基本的な機能を検証する、中断を伴わないテスト。テスト対象のディスクで障害が検出されない場合、テスト結果は PASS になります。障害が検出された場合、単一のディスク デバイスの場所が特定されます。このテストは、RP ノードおよび Distributed Route Processor (DRP; 分散型ルート プロセッサ) ノード上で実行されます。
- **Scratch Register Test** - テスト対象の ASIC が適切に動作しているように見えることをアサートする、中断を伴わないテスト。このテストは、RP、DRP、および Line Card (LC; ラインカード) で実行されます。このテストでは、対象の ASIC が選択した内部の場所で値の書き込みと読み取りを適切に実行しているかどうかをチェックします。選択される場所は、現在適切に動作している ASIC の機能には影響がない値の場所です。テスト対象のレジスタで障害が検出されない場合、テスト結果は PASS になります。障害が検出された場合、単一の ASIC の場所が特定されます。
- **Fabric Diagnostic Test** - ユニキャスト アドレスを使用して、スタンバイ RP ノードから、シャーシ内の各 RP/LC/DRP ノードにユニキャスト ping を実行する、中断を伴わない障害分離テスト。このテストは、異なるファブリック プレーンを経由するユニキャスト ping テスト パケットを操作し、ファブリック プレーン情報を含むユニキャスト ping (PASS または FAIL) の結果を集約し、その結果を分析し、シャーシ内で最も論理的な障害ポイントがあればそれを指摘します。テストを実行できるのは、次のテストタイプのいずれかのスタンバイ RP からのみです。
 - On demand
 - Scheduled
 - Configured to monitor health

Cisco CRS-1 マルチシェルフ システムでこのテストを実行すると、システムのユニキャストの最も論理的な障害ポイントであるファブリック ステージ (S1、S2、または S3) を判断するときに役立ちます。

このテストは、システム内の各 LC ラック スタンバイ RP で実行する必要があります。たとえば、テストが複数の LC ラックでの障害を報告し、障害情報が同じファブリック プレーンを指す場合、最も可能性が高い障害ポイントは S2 ステージです。これはシステムのファブリック シャーシのカードです。

- **Fabric Multicast Diagnosis Test** - マルチキャスト アドレスを使用して、スタンバイ RP ノードから、シャーシ内の各 RP/LC/DRP ノードに「マルチキャスト ping」を実行する、中断を伴わない障害分離テスト。このテストは、異なるファブリック プレーンを経由するマルチキャスト ping テスト パケットを操作し、ファブリック プレーン情報を含むマルチキャスト ping (PASS または FAIL) の結果を集約し、その結果を分析し、シャーシ内で最も論理的な障害ポイントがあればそれを指摘します。テストを実行できるのは、次のテストタイプのいずれかのスタンバイ RP からのみです。
 - On demand
 - Scheduled
 - Configured to monitor health

Cisco CRS-1 マルチシェルフ システムでこのテストを実行すると、システムのマルチキャストの最も論理的な障害ポイントであるファブリック ステージ (S1、S2、または S3) を判断するときに役立ちます。

このテストは、システム内の各 LC ラック スタンバイ RP で実行する必要があります。たとえば、テストが複数の LC ラックでの障害を報告し、障害情報が同じファブリック プレーンを指す場合、最も可能性が高い障害ポイントは S2 ステージです。これはシステムのファブリック シャーシのカードです。

その他の参考資料

Cisco IOS XR の診断に関連する参考資料を示します。

関連資料

内容	参照先
診断コマンド：詳細なコマンド構文、コマンドモード、コマンド履歴、デフォルト、使用方法のガイドライン、および例	『Cisco IOS XR Interface and Hardware Component Command Reference』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
<p>シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。</p>	<p>http://www.cisco.com/techsupport</p>



Cisco IOS XR ソフトウェアでのイーサネット インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータでのイーサネット インターフェイスの設定について説明します。

分散型ギガビット イーサネット、10 ギガビット イーサネット、およびファスト イーサネットのアーキテクチャと機能によって、ネットワークのスケラビリティとパフォーマンスが向上しました。さらに、サービス プロバイダーは、POP でルータと他のシステムを相互接続できるように設計された高密度で広帯域幅のネットワーク ソリューションを提供できるようになりました。たとえば、コア ルータ、エッジ ルータ、レイヤ 2 およびレイヤ 3 スイッチなどです。



(注)

ここでは、[Management Ethernet] インターフェイスの設定情報は説明しません。[Management Ethernet] インターフェイスをセットアップし、telnet サーバをイネーブルにする方法については、『Cisco IOS XR Getting Started Guide』を参照してください。ルーティングの [Management Ethernet] インターフェイスを設定する方法、または [Management Ethernet] インターフェイスの設定を変更する方法については、「[Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイスの高度な設定と変更](#)」モジュールを参照してください。

Cisco IOS XR ソフトウェアのイーサネット インターフェイス設定の機能履歴

リリース	変更点
リリース 3.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.2	Cisco XR 12000 シリーズ ルータのサポートが追加されました。 SIP-800 について、Cisco CRS-1 ルータのサポートが追加されました。 次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。 <ul style="list-style-type: none">1 ポート 10 ギガビット イーサネット SPA5 ポート ギガビット イーサネット SPA10 ポート ギガビット イーサネット SPA 8 ポート ギガビット イーサネット シングル SPA が Cisco CRS-1 ルータに導入されました。

リリース 3.3.0	<p>8 ポート 10 ギガビット イーサネット PLIM での出力 MAC アカウンティングのサポートが追加されました。</p> <p>次の SIP について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p> <ul style="list-style-type: none"> • Cisco XR 12000 SIP-401 • Cisco XR 12000 SIP-501 • Cisco XR 12000 SIP-601 <p>8 ポート ファスト イーサネット SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p>
リリース 3.4.0	<p>レイヤ 2 バーチャル プライベート ネットワーク (L2VPN) 機能が最初にサポートされたのは、Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータのイーサネット インターフェイスでした。</p> <p>8 ポート 1 ギガビット イーサネット SPA について、Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p>
リリース 3.4.1	2 ポート ギガビット イーサネット SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。
リリース 3.5.0	1 ポート 10 ギガビット イーサネット WAN SPA について、Cisco CRS-1 ルータでのサポートが追加されました。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「イーサネット インターフェイスの前提条件」 (P.144)
- 「イーサネット インターフェイスの設定に関する情報」 (P.145)
- 「イーサネット インターフェイスの設定方法」 (P.153)
- 「イーサネット インターフェイスの設定例」 (P.163)
- 「関連情報」 (P.165)
- 「その他の参考資料」 (P.166)

イーサネット インターフェイスの前提条件

イーサネット インターフェイスを設定する前に、次のタスクと条件を満たしていることを確認します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

- Cisco CRS-1 ルータの場合、次のカードの 1 枚以上がルータに装着されていることを確認します。
 - 8 ポート 10 ギガビット イーサネット (8 × 10 GE) 物理レイヤ インターフェイス モジュール (PLIM)
 - 8 ポート 1 ギガビット イーサネット共有ポート アダプタ (SPA)
- Cisco XR 12000 シリーズ ルータの場合、次のカードの 1 枚以上がルータに装着されていることを確認します。
 - 4 ポート 1 ギガビット イーサネット PLIM
 - 2 ポート ギガビット イーサネット SPA
 - 5 ポート 1 ギガビット イーサネット SPA
 - 10 ポート ギガビット イーサネット SPA
 - 1 ポート 10 ギガビット イーサネット SPA
 - 1 ポート 10 ギガビット イーサネット WAN SPA
 - 8 ポート ギガビット イーサネット SPA
 - 8 ポート ファスト イーサネット SPA
- インターフェイスの IP アドレスを知っています。
- 汎用インターフェイス名に汎用表記法の *rack/slot/module/port* を適用する方法を理解しています。
- 10-GE DWDM PLIM で 10 ギガビット イーサネット インターフェイスを設定する場合、このマニュアルの「[Cisco IOS XR ソフトウェアでの高密度波長分割多重コントローラの設定](#)」モジュールの説明に従って、DWDM コントローラの設定が完了している必要があります。

イーサネット インターフェイスの設定に関する情報

イーサネット インターフェイスを設定するには、次の概念について理解する必要があります。

- 「[イーサネットテクノロジーの概要](#)」 (P.146)
- 「[ギガビット イーサネットおよび 10 ギガビット イーサネットのデフォルト設定値](#)」 (P.146)
- 「[ギガビット イーサネット プロトコル規格の概要](#)」 (P.148)
- 「[MAC アドレス](#)」 (P.149)
- 「[MAC アカウンティング](#)」 (P.149)
- 「[イーサネット MTU](#)」 (P.149)
- 「[イーサネット インターフェイスでのフロー制御](#)」 (P.150)
- 「[802.1Q VLAN](#)」 (P.150)
- 「[VRRP](#)」 (P.150)
- 「[HSRP](#)」 (P.151)
- 「[ファスト イーサネット インターフェイスでのデュプレックス モード](#)」 (P.151)
- 「[ファスト イーサネット インターフェイスの速度](#)」 (P.151)
- 「[イーサネット インターフェイスでのリンクのオートネゴシエーション](#)」 (P.152)
- 「[イーサネット インターフェイスでのキャリア遅延](#)」 (P.153)

イーサネット テクノロジーの概要

イーサネットは IEEE 802.3 国際規格によって定義されています。イーサネットによって、同軸ケーブル、ツイストペアケーブル、または光ファイバケーブルで、最大 1024 ノードの接続が可能になります。

Cisco CRS-1 ルータは、ギガビット イーサネット (1000 Mbps) インターフェイスおよび 10 ギガビット イーサネット (10 Gbps) インターフェイスをサポートしています。

Cisco XR 12000 シリーズ ルータは、ファスト イーサネット (100 Mbps) インターフェイス、ギガビット イーサネット (1000 Mbps) インターフェイス、および 10 ギガビット イーサネット (10 Gbps) インターフェイスをサポートしています。

ギガビット イーサネットおよび 10 ギガビット イーサネットのデフォルト設定値

表 4 は、ギガビット イーサネットまたは 10 ギガビット イーサネットのモジュラ サービス カードおよび PC の脅威対策 PLIM でインターフェイスをイネーブルにしたときに表示される、デフォルトのインターフェイス設定パラメータを示します。



(注)

インターフェイスを管理上のダウン状態にするには、**shutdown** コマンドを使用する必要があります。インターフェイスのデフォルトは **no shutdown** です。ルータにモジュラ サービス カードを初めて挿入したときに、プリコンフィギュレーションが行われていない場合、設定マネージャによって **shutdown** 項目が設定に追加されます。この **shutdown** を削除できるのは、**no shutdown** コマンドを入力している場合のみです。

表 4 ギガビット イーサネットおよび 10 ギガビット イーサネット モジュラ サービス カードのデフォルト設定値

パラメータ	コンフィギュレーションファイルのエントリ	デフォルト値
MAC accounting	mac-accounting	off
Flow control	flow-control	egress on ingress off
MTU	MTU	通常フレームの場合は 1514 bytes 802.1Q タグ付きフレーム の場合は 1518 bytes Q-in-Q フレームの場合は 1522 bytes
MAC address	mac address	Hardware burned-in address (BIA ¹)

1. 組み込みのアドレス

ファスト イーサネットのデフォルト設定値

表 5 は、ファスト イーサネット SPA カードおよびその関連 PLIM でインターフェイスをイネーブルにしたときに表示されるデフォルトのインターフェイス設定パラメータを示します。



(注)

インターフェイスを管理上のダウン状態にするには、**shutdown** コマンドを指定する必要があります。インターフェイスのデフォルトは **no shutdown** です。ルータにモジュラ サービス カードを初めて挿入したときに、プリコンフィギュレーションが行われていない場合、設定マネージャによって **shutdown** 項目が設定に追加されます。この **shutdown** を削除できるのは、**no shutdown** コマンドを入力している場合のみです。

表 5 ファストイーサネットのデフォルト設定値

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト値
MAC accounting	mac-accounting	off
Duplex operation	duplex full duplex half	Auto-negotiates duplex operation
MTU	mtu	1500 bytes
Interface speed	speed	100 Mbps
Auto-negotiation	negotiation auto	disable

イーサネット インターフェイスでのレイヤ 2 VPN

L2VPN 接続は、IP または MPLS 対応 IP ネットワーク間の LAN の動作をエミュレートすることで、イーサネット デバイス間が共通の LAN セグメントに接続した場合と同様に通信できるようになります。

L2VPN の機能によって、サービス プロバイダ (SP) は地理的に離れたカスタマー サイトにもレイヤ 2 サービスを提供できるようになります。通常、SP はアクセス ネットワークを使用して、カスタマーをコア ネットワークに接続します。このアクセス ネットワークでは、イーサネット、ATM、フレームリレーなどのレイヤ 2 テクノロジーが併用される場合があります。カスタマー サイトと近接した SP エッジルータ間の接続は、接続回路 (AC) と呼ばれます。カスタマーからのトラフィックは、このリンク上で SP コア ネットワークのエッジへ伝送されます。次に、SP コア ネットワーク上の疑似接続のトンネルを介して、別のエッジルータへ伝送されます。このトラフィックはエッジルータによって別の AC へと伝送され、そこからカスタマーのリモート サイトへ伝送されます。

L2VPN の機能によって、異なる種類のレイヤ 2 接続回路と疑似接続間の接続が可能になります。その結果、ユーザはさまざまなエンドツーエンド サービスを実装できるようになります。

Cisco IOS XR ソフトウェアは、ポイントツーポイント エンドツーエンド サービスをサポートしています。つまり、2 つのイーサネット回路が相互に接続されます。L2VPN イーサネット ポートは、次の 2 モードのいずれかで動作します。

- **Port Mode** - このモードでは、ポートに到達するすべてのパケットは、パケットに指定されている VLAN タグに関係なく、疑似接続上で送信されます。VLAN モードでは、**l2transport** コンフィギュレーション モードで設定が実行されます。
- **VLAN Mode** - PE リンクに対する CE またはアクセス ネットワーク上の各 VLAN は、(VC タイプ 4 または VC タイプ 5 を使用して) 個別の L2VPN 接続として設定できます。VLAN 上で L2VPN を設定する方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定](#)」モジュールを参照してください。VLAN モードでは、個別のサブインターフェイスで設定を実行します。

切り替えは次の 3 つの方法で実行できます。

- AC-to-PW : Provider Edge (PE; プロバイダー エッジ) に到達したトラフィックは 疑似接続 (PW) を介してトンネリングされます。また、それとは反対に疑似接続 (PW) を介して到達したトラフィックは AC を介して送信されます。これが最も一般的なシナリオです。
- ローカルの切り替え - 1 つの AC 上で到達するトラフィックは、疑似接続を介さずに別の AC へ送出されます。
- PW 切り替え - PW に到達するトラフィックは AC へ送信されませんが、別の PW 上でコアに返信されます。

イーサネット インターフェイスで L2VPN を設定する場合、次の点に気を付けてください。

- L2VPN リンクは QoS および MTU の設定をサポートしています。
- ネットワークでパケットを透過的に伝送することを必須にしている場合、必要に応じて、SP ネットワークのエッジでパケットの宛先 MAC アドレスを変更します。こうすることで、SP ネットワークのデバイスによるパケットの消費が回避されます。
- Cisco IOS XR ソフトウェアは、ラインカードごとに最大 4,000 AC をサポートしています。ただし、4,000 AC をサポートしていないラインカードもあります。サポートされる最大 AC 数の詳細については、個々のラインカードの仕様を参照してください。

AC および疑似接続の情報を表示するには、**show interfaces** コマンドを使用します。



(注)

L2VPN ネットワークの設定の詳細については、『*Cisco IOS XR Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs*」モジュールを参照してください。

ギガビット イーサネット プロトコル規格の概要

ギガビット イーサネット インターフェイスは次のプロトコル規格をサポートしています。

- [IEEE 802.3 物理イーサネット インフラストラクチャ](#)
- [IEEE 802.3ab 1000BASE-T ギガビット イーサネット](#)
- [IEEE 802.3z 1000 Mbps ギガビット イーサネット](#)
- [IEEE 802.3ae 10 Gbps イーサネット](#)

各規格の詳細については、このマニュアルで後述します。

IEEE 802.3 物理イーサネット インフラストラクチャ

IEEE 802.3 プロトコル規格では、接続するイーサネットの物理層とデータリンク層の MAC 下位層が定義されています。IEEE 802.3 では、多様な物理メディアで、また多様な速度で Carrier Sense Multiple Access with Collision Detection (CSMA/CD; キャリア検知多重アクセス/衝突検出) アクセスを使用します。IEEE 802.3 規格は 10 Mbps イーサネットに対応します。IEEE 802.3 規格の拡張では、ギガビット イーサネット、10 ギガビット イーサネット、およびファスト イーサネットの実装を規定しています。

IEEE 802.3ab 1000BASE-T ギガビット イーサネット

IEEE 802.3ab プロトコル規格、つまり銅線上のギガビット イーサネット (別名 1000BaseT) は、既存のファスト イーサネット規格の拡張です。この拡張は、すでに設置されているカテゴリ 5e/6 ケーブル配線システム上のギガビット イーサネットの動作を規定しており、費用有効性の高いソリューション

を実現できます。結果として、ファストイーサネットを実行する銅線ベースの環境では既存のインフラストラクチャ上でギガビットイーサネットも実行できるため、要求の厳しいアプリケーションでもネットワークのパフォーマンスが大幅に向上します。

IEEE 802.3z 1000 Mbps ギガビットイーサネット

ギガビットイーサネットはイーサネットプロトコルの上で構築されますが、速度はファストイーサネットの10倍で、1000 Mbps (1 Gbps) に上がります。ギガビットイーサネットを使用すると、デスクトップで10 Mbps または100 Mbps、データセンターで最高1000 Mbps までイーサネットを拡張できます。ギガビットイーサネットはIEEE 802.3z プロトコル規格に準拠します。

ネットワーク管理者は、現在のイーサネット規格と、すでに設置されているイーサネットおよびファストイーサネットのスイッチおよびルータのベースを利用することで、ギガビットイーサネットをサポートするために新しいテクノロジーのトレーニングや学習をし直す必要はなくなります。

IEEE 802.3ae 10 Gbps イーサネット

国際標準化組織の Open Systems Interconnection (OSI; 開放型システム間相互接続) モデルでは、イーサネットは基本的にレイヤ2プロトコルです。10ギガビットイーサネットでは、IEEE 802.3 イーサネット MAC プロトコル、IEEE 802.3 イーサネット フレーム形式、および IEEE 802.3 の最小および最大フレームサイズを使用します。10 Gbps イーサネットは IEEE 802.3ae プロトコル規格に準拠します。

イーサネットモデルに忠実だった 1000BASE-X と 1000BASE-T (ギガビットイーサネット) と同様に、10ギガビットイーサネットも速度と距離の点でイーサネットが自然に発展した結果です。10ギガビットイーサネットは全二重方式でファイバのみのテクノロジーなので、低速で半二重方式のイーサネットテクノロジーを定義する CSMA/CD プロトコルを使用した、通信事業者に影響される多重アクセスは必要ありません。他のどの点でも、10ギガビットイーサネットは元のイーサネットモデルに忠実です。

MAC アドレス

MAC アドレスは、レイヤ2のインターフェイスを識別する固有の6バイトアドレスです。

MAC アカウンティング

MAC アドレス アカウンティング機能を使用すると、LAN インターフェイスの発信元および宛先の MAC アドレスに基づいた IP トラフィックのアカウント情報がわかります。この機能では、LAN インターフェイスが固有の MAC アドレスとの間で送受信する IP パケットの合計パケット数および合計バイト数が計算されます。また、最終受信または最終送信のタイムスタンプも記録されます。

イーサネット MTU

イーサネットの最大伝送ユニット (MTU) は、最大フレームのサイズから4バイトの Frame Check Sequence (FCS; フレームチェックシーケンス) を引いた値です。この MTU がイーサネットネットワークで伝送できるサイズです。パケットの宛先に到達するまでに経由する各物理ネットワークは、MTU が異なる可能性があります。

Cisco IOS XR ソフトウェアは、2種類のフレーム転送プロセスをサポートしています。

- IPv4 パケットのフラグメンテーション。このプロセスでは、次ホップの物理ネットワークの MTU 内に収まるように、必要に応じて IPv4 パケットが分割されます。



(注) IPv6 はフラグメンテーションをサポートしません。

- MTU の検出プロセスによる最大パケット サイズの決定。このプロセスは、すべての IPv6 デバイスと発信側の IPv4 デバイスに使用できます。このプロセスでは、分割せずに送信できる IPv6 または IPv4 パケットの最大サイズを、発信側の IP デバイスが決定します。最大パケットは、IP 発信元デバイスおよび IP 宛先デバイス間にあるすべてのネットワークの中で、最小 MTU と等値です。このパス内にあるすべてのネットワークの最小 MTU よりもパケットが大きい場合、そのパケットは必要に応じて分割されます。このプロセスによって、発信側のデバイスから大きすぎる IP パケットが送信されなくなります。

標準フレーム サイズを超えるフレームの場合、ジャンボ フレームのサポートが自動的にイネーブルになります。デフォルト値は標準フレームの場合は 1514、802.1Q タグ付きフレームの場合は 1518 です。この数値に 4 バイトの FCS は含まれません。

イーサネット インターフェイスでのフロー制御

10 ギガビット イーサネット インターフェイスでのフロー制御は、フロー制御ポーズ フレームを定期的 に送信する処理で構成されます。この処理は、標準の管理インターフェイスで使用される通常の全二重 および半二重のフロー制御とは根本的に異なります。フロー制御は、入トラフィックについてのみアクティブ化または非アクティブ化することができます。出トラフィックについては自動的に実装されま ず。

802.1Q VLAN

VLAN とは、実際は異なる LAN セグメント上のデバイスでも、同じセグメントで接続している場合と 同様に通信できるように設定された、1 つまたは複数の LAN 上にあるデバイスのグループです。

VLAN は、物理接続ではなく論理接続に基づいているため、ユーザ管理、ホスト管理、帯域割り当て、 およびリソースの最適化がとて柔軟です。

IEEE の 802.1Q プロトコル規格では、ブロードキャストおよびマルチキャストのトラフィックが必要以 上の帯域を消費しないように、大規模なネットワークを小規模なパーツに分割することで問題に対処し ています。また、内部ネットワークのセグメント間に、より高レベルのセキュリティを実現できます。

802.1Q 仕様は、イーサネット フレームに VLAN メンバーシップ情報を挿入する標準方式を確立します。

VLAN 設定については、このマニュアルの「[Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定](#)」モジュールを参照してください。

VRRP

Virtual Router Redundancy Protocol (VRRP; 仮想ルータ冗長プロトコル) によって、静的なデフォルト のルーティング環境に固有の単一障害点が除外されます。VRRP は、仮想ルータの役割を LAN 上の VPN コンセントレータの 1 つに動的に割り当てるといふ、選択プロトコルを規定します。仮想ルータ に割り当てる IP アドレスを制御する VRRP VPN コンセントレータはマスターと呼ばれ、送信された パケットをその IP アドレスに転送します。マスターが使用不可になると、バックアップ VPN コンセン トレータがマスターの役割を引き継ぎます。

VRRP の詳細については、『[Cisco IOS XR IP Addresses and Services Configuration Guide](#)』の 「[Implementing VRRP on Cisco IOS XR Software](#)」モジュールを参照してください。

HSRP

Hot Standby Routing Protocol (HSRP) はシスコの独自プロトコルです。HSRP は障害の発生時にルータのバックアップを用意するルーティング プロトコルです。複数のルータが同じセグメントのイーサネット、FDDI、またはトークンリング ネットワークに接続し、LAN 上にある単一の仮想ルータとして連携します。これらのルータは同じ IP アドレスおよび MAC アドレスを共有するため、ルータのいずれかに障害が発生した場合でも、LAN 上のホストはそのまま同じ IP アドレスおよび MAC アドレスにパケットを転送できます。ルーティングの担当デバイスの切り替えは、ユーザには検知されません。

HSRP は、特定の状況で IP トラフィックを中断しない切り替えをサポートし、ホストからは単一のルータを使用しているように見え、使用している第 1 ホップのルータに障害が発生した場合でも接続を維持できるように設計されています。つまり、HSRP は、発信元のホストが第 1 ホップのルータの IP アドレスを動的に取得できない場合でも、第 1 ホップのルータの障害に対処できます。複数のルータが HSRP に参加し、連携して単一の仮想ルータであるように見えます。HSRP によって、確実に単一のルータが仮想ルータの代わりにパケットを転送します。エンドホストがそのパケットを仮想ルータに転送します。

パケットを転送するルータは、アクティブルータと呼ばれます。アクティブルータに障害が発生した場合、代わりになるスタンバイルータが選択されます。HSRP には、参加するルータの IP アドレスを使用して、アクティブルータとスタンバイルータを決定するメカニズムがあります。アクティブルータに障害が発生した場合、スタンバイルータが引き継ぐことができます。ホストの接続が長く切断することはありません。

HSRP は User Datagram Protocol (UDP; ユーザ データグラム プロトコル) 上で実行され、ポート番号 1985 を使用します。ルータは、プロトコルパケットの発信元アドレスとして仮想アドレスではなく実際の IP アドレスを使用するため、HSRP ルータは相互を識別できます。

HSRP の詳細については、『Cisco IOS XR IP Addresses and Services Configuration Guide』の「Implementing HSRP on Cisco IOS XR Software」モジュールを参照してください。

ファスト イーサネット インターフェイスでのデュプレックス モード

ファスト イーサネット ポートはデュプレックス伝送タイプをサポートしています。全二重モードでは、送信側ステーションと受信側ステーション間で同時にデータ伝送が可能ですが、半二重モードでは、同時に 1 方向のみのデータ伝送が可能です。

ファスト イーサネット インターフェイスでデュプレックス モードを設定する場合、次の点に注意してください。

- インターフェイスでオートネゴシエーションがイネーブルの場合、デフォルトではデュプレックスモードがネゴシエーションされます。
- インターフェイスでオートネゴシエーションがディセーブルの場合、デフォルトは全二重モードです。



(注)

デュプレックス モードを設定できるのはファスト イーサネット インターフェイスのみです。ギガビット イーサネットおよび 10 ギガビット イーサネットは常に全二重モードで実行されます。

ファスト イーサネット インターフェイスの速度

ファスト イーサネット インターフェイスではインターフェイス速度を設定できます。ファスト イーサネット インターフェイスの速度を設定する場合、次の点に注意してください。

- オートネゴシエーションがインターフェイスでイネーブルの場合、デフォルトでは速度がネゴシエーションされます。
- オートネゴシエーションがインターフェイスでディセーブルの場合、デフォルトの速度はそのインターフェイスに可能な最高速度です。



(注)

リンクの両エンドは、同じインターフェイス速度にする必要があります。手動で設定したインターフェイス速度は、オートネゴシエーションされた速度よりも優先されます。そのため、リンクの一方のエンドで設定したインターフェイス速度が、もう一方のエンドのインターフェイス速度と異なる場合、リンク速度の上昇を防ぐことができます。

イーサネット インターフェイスでのリンクのオートネゴシエーション

リンクのオートネゴシエーションによって、リンク セグメントを共有するデバイスは、最高のパフォーマンス モードの相互運用で自動的に設定されます。イーサネット インターフェイスでリンクのオートネゴシエーションをイネーブルにするには、インターフェイス コンフィギュレーション モードで **negotiation auto** コマンドを使用します。ラインカードのイーサネット インターフェイスでは、リンクのオートネゴシエーションはデフォルトでディセーブルです。



(注)

negotiation auto コマンドを使用できるのは、ギガビット イーサネットおよびファスト イーサネットのインターフェイスのみです。

表 6 は、デュプレックス モードおよび速度モードのさまざまな組み合わせに関するシステムのパフォーマンスです。インターフェイスでオートネゴシエーションが設定されているという前提で、指定した **duplex** コマンドと指定した **speed** コマンドによって結果のシステム処理が実行されます。

表 6 duplex コマンドと speed コマンドの関係

duplex コマンド	speed コマンド	結果のシステム処理
no duplex	no speed	速度モードとデュプレックス モードの両方がオートネゴシエーションされます。
no duplex	speed 1000	デュプレックス モードがオートネゴシエーションされ、強制的に 1000 Mbps が指定されます。
no duplex	speed 100	デュプレックス モードがオートネゴシエーションされ、強制的に 100 Mbps が指定されます。
no duplex	speed 10	デュプレックス モードがオートネゴシエーションされ、強制的に 10 Mbps が指定されます。
full-duplex	no speed	強制的に全二重モードが指定され、速度はオートネゴシエーションされます。
full-duplex	speed 1000	強制的に全二重モードと 1000 Mbps が指定されます。
full-duplex	speed 100	強制的に全二重モードと 100 Mbps が指定されます。
full-duplex	speed 10	強制的に全二重モードと 10 Mbps が指定されます。
half-duplex	no speed	強制的に半二重モードが指定され、速度はオートネゴシエーションされます。
half-duplex	speed 1000	強制的に半二重モードと 1000 Mbps が指定されます。

表 6 duplex コマンドと speed コマンドの関係 (続き)

duplex コマンド	speed コマンド	結果のシステム処理
half-duplex	speed 100	強制的に半二重モードと 100 Mbps が指定されます。
half-duplex	speed 10	強制的に半二重モードと 10 Mbps が指定されます。

イーサネット インターフェイスでのキャリア遅延

イーサネット インターフェイスでキャリア遅延機能をイネーブルにすると、リンク アップ イベントまたはリンク ダウン イベントに応じてシステムの応答速度が遅くなります。インターフェイスでキャリア遅延のアップとキャリア遅延のダウンを同時に設定できます。

リンクがダウンした後、回復し、再度ダウンする場合、キャリア遅延のアップによって、短期のリンクフラップを抑制できます。過去にダウンしたリンクが **carrier-delay up** コマンドに指定した期間よりも長くアップ状態が継続してから、インターフェイスが回復したことをシステムに通知します。

carrier-delay up コマンドに指定した期間よりも短いフラップは、すべて抑制されます。

キャリア遅延のアップを設定すると、リンクが十分に安定してから、リンクが回復し、トラフィックを転送する準備が整ったことをシステムに通知することができます。

キャリア遅延のダウンによって、リンクがアップした後にダウンし、再度アップした場合、短期のリンクフラップを抑制できます。過去にアップしたリンクが **carrier-delay down** コマンドに指定した期間よりも長くダウン状態が継続してから、インターフェイスがダウンしたことをシステムに通知します。**carrier-delay down** コマンドに指定した値よりも短いフラップは、すべて抑制されます。

キャリア遅延のダウンの設定は、非常に短期のリンクフラップを抑制し、結果的にインターフェイスフラップを防ぐことに役立ちます。また、この機能を設定することで、他のリンク保護機器が介入する時間的な余裕ができるという利点もあります。

イーサネット インターフェイスの設定方法

ここでは、次のタスクについて説明します。

- 「[ギガビット イーサネット インターフェイスまたは 10 ギガビット イーサネット インターフェイスの設定](#)」 (P.153)
- 「[ファストイーサネット インターフェイスの設定](#)」 (P.157)
- 「[イーサネット インターフェイスでの MAC アカウンティングの設定](#)」 (P.159)

ギガビット イーサネット インターフェイスまたは 10 ギガビット イーサネット インターフェイスの設定

基本的なギガビット イーサネット インターフェイスまたは 10 ギガビット イーサネット インターフェイスの設定を作成するには、次の手順で操作します。

手順の概要

1. `show version`
2. `show interfaces [GigabitEthernet | TenGigE] interface-path-id`
3. `configure`

4. `interface [GigabitEthernet | TenGigE] interface-path-id`
5. `ipv4 address ip address mask`
6. `flow-control {bidirectional | egress | ingress}`
7. `mtu bytes`
8. `mac-address value1.value2.value3`
9. `negotiation auto` (ギガビット イーサネット インターフェイスのみ)
10. `no shutdown`
11. `end`
または
`commit`
12. `show interfaces [GigabitEthernet | TenGigE] interface-path-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>show version</code> 例: RP/0/RP0/CPU0:router# show version	(任意) 現在のソフトウェア バージョンを表示します。また、ルータがモジュラ サービス カードを認識していることを確認する場合にも使用できます。
ステップ 2	<code>show interfaces [GigabitEthernet TenGigE] interface-path-id</code> 例: RP/0/RP0/CPU0:router# show interfaces TenGigE 0/1/0/0	(任意) 設定済みのインターフェイスを表示し、各インターフェイス ポートのステータスを確認します。 このステップで使用できるインターフェイスの種類は次のとおりです。 <ul style="list-style-type: none"> • GigabitEthernet • TenGigE
ステップ 3	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 4	<code>interface [GigabitEthernet TenGigE] interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイス名と <code>rack/slot/module/port</code> 表記を指定します。このステップで使用できるインターフェイスの種類は次のとおりです。 <ul style="list-style-type: none"> • GigabitEthernet • TenGigE <p>(注) この例は、モジュラ サービス カード スロット 1 の 8 ポート 10 ギガビット イーサネット インターフェイスです。</p>

コマンドまたはアクション	目的
<p>ステップ 5 <code>ipv4 address ip-address mask</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224</p>	<p>インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p> <ul style="list-style-type: none"> • <i>ip-address</i> をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • <i>mask</i> を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクを指定するには、次の 2 つの方法があります。 <ul style="list-style-type: none"> – ネットワーク マスクには、4 パートのドット付き 10 進アドレスを指定できます。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – ネットワーク マスクはスラッシュ (/) と数字で指定できます。たとえば、/8 は、マスクの先頭 8 ビットが 1 で、アドレスの対応するビットがネットワーク アドレスであることを示します。
<p>ステップ 6 <code>flow-control {bidirectional egress ingress}</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# flow control ingress</p>	<p>(任意) フロー制御のポーズ フレームの送信および処理をイネーブルにします。</p> <ul style="list-style-type: none"> • egress - 送信時にフロー制御のポーズ フレームをイネーブルにします。 • ingress - 受信時に受信したポーズ フレームの処理をイネーブルにします。 • bidirectional - 送信時のフロー制御のポーズ フレームと受信時の受信したポーズ フレームの処理をイネーブルにします。
<p>ステップ 7 <code>mtu bytes</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# mtu 1448</p>	<p>(任意) インターフェイスの MTU 値を設定します。</p> <ul style="list-style-type: none"> • 通常フレームのデフォルトは 1514 バイト、802.1Q タグ付き フレームのデフォルトは 1518 バイトです。 • ギガビット イーサネットおよび 10 ギガビット イーサネットの mtu 値の範囲は 64 ~ 65535 バイトです。
<p>ステップ 8 <code>mac-address value1.value2.value3</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# mac address 0001.2468.ABCD</p>	<p>(任意) [Management Ethernet] インターフェイスの MAC レイヤ アドレスを設定します。</p> <ul style="list-style-type: none"> • 値は、それぞれ MAC アドレスの上位、中間、および下位の 2 バイト (16 進) です。各 2 バイト値の範囲は 0 ~ ffff です。

イーサネット インターフェイスの設定方法

コマンドまたはアクション	目的
<p>ステップ 9 <code>negotiation auto</code></p> <p>例: RP/0/0/CPU0:router(config-if)# negotiation auto</p>	<p>(任意) ギガビット イーサネット インターフェイスでのオートネゴシエーションをイネーブルにします。</p> <ul style="list-style-type: none"> オートネゴシエーションは接続の両エンドで明示的にイネーブルにするか、接続の両エンドで速度とデュプレックス モードを手動設定する必要があります。 オートネゴシエーションがイネーブルの場合、手動で設定した速度またはデュプレックス モードの設定の方が優先されます。 <p>(注) <code>negotiation auto</code> コマンドを使用できるのは、ギガビット イーサネットおよびファスト イーサネットのインターフェイスのみです。</p>
<p>ステップ 10 <code>no shutdown</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# no shutdown</p>	<p><code>shutdown</code> 設定を削除します。こうすることでインターフェイスが強制的に管理上のダウン状態になります。</p>
<p>ステップ 11 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 12 <code>show interfaces [GigabitEthernet TenGigE] interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router# show interfaces TenGigE 0/3/0/0</p>	<p>(任意) ルータ上のインターフェイスに関する統計情報を表示します。</p>

次に行う作業

- イーサネット インターフェイスで MAC アカウンティングを設定する方法については、このモジュールで後述する「イーサネット インターフェイスでの MAC アカウンティングの設定」を参照してください。

- イーサネット インターフェイスで 802.1Q VLAN サブインターフェイスを設定する方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定](#)」モジュールを参照してください。
- レイヤ 2 VPN 実装のイーサネット ポートで AC を設定する方法については、このモジュールで後述する「[イーサネット ポートでの接続回路の設定](#)」を参照してください。
- マルチプロトコル ラベル スイッチング (MPLS) やサービス品質 (QoS) など、レイヤ 3 サービス ポリシーをイーサネット インターフェイスに付加する方法については、該当する Cisco IOS XR ソフトウェアのコンフィギュレーション ガイドを参照してください。

ファスト イーサネット インターフェイスの設定

基本的なファスト イーサネット インターフェイス設定を作成するには、次の手順で操作します。



(注)

ファスト イーサネットは Cisco XR 12000 シリーズ ルータでのみサポートされています。

手順の概要

- `configure`
- `interface fastethernet interface-path-id`
- `ipv4 address ip-address mask`
- `mtu bytes`
- `duplex full`
- `speed speed`
- `negotiation auto`
- `no shutdown`
- `end`
または
`commit`
- `show interfaces fastethernet interface-path-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/0/CPU0:router# <code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface fastethernet interface-path-id</code> 例： RP/0/0/CPU0:router(config)# <code>interface fastethernet 0/0/2/1</code>	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイス名と <code>rack/slot/module/port</code> 表記を指定します。 <ul style="list-style-type: none"> この例は、スロット 0 の 8 ポート ファスト イーサネット SPA のセカンドインターフェイスと、SPA サブスロット 2 です。

コマンドまたはアクション	目的
<p>ステップ 3 <code>ipv4 address ip-address mask</code></p> <p>例: RP/0/0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224</p>	<p>インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p> <ul style="list-style-type: none"> • <i>ip-address</i> をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • <i>mask</i> を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクを指定するには、次の 2 つの方法があります。 <ul style="list-style-type: none"> – ネットワーク マスクには、4 パートのドット付き 10 進アドレスを指定できます。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – ネットワーク マスクはスラッシュ (/) と数字で指定できます。たとえば、/8 は、マスクの先頭 8 ビットが 1 で、アドレスの対応するビットがネットワーク アドレスであることを示します。
<p>ステップ 4 <code>mtu bytes</code></p> <p>例: RP/0/0/CPU0:router(config-if)# mtu 1448</p>	<p>(任意) インターフェイスの MTU 値を設定します。</p> <ul style="list-style-type: none"> • デフォルトは 1500 バイトです。 • ファスト イーサネットの mtu 値の範囲は 0 ~ 10240 バイトです。
<p>ステップ 5 <code>duplex full</code></p> <p>例: RP/0/0/CPU0:router(config-if)# duplex full</p>	<p>(任意) 全二重モードをイネーブルにします。半二重モードをイネーブルにするには、duplex half コマンドを使用します。</p> <ul style="list-style-type: none"> • デュプレックス操作はデフォルトでオートネゴシエーションされます。 • オートネゴシエーションがイネーブルの場合、ユーザ定義の duplex 値はすべて無視されます。
<p>ステップ 6 <code>speed speed</code></p> <p>例: RP/0/0/CPU0:router(config-if)# speed 100</p>	<p>(任意) インターフェイスの速度を設定します。有効なオプションは 10 Mbps または 100 Mbps です。</p> <ul style="list-style-type: none"> • デフォルトは 100 Mbps です。 • オートネゴシエーションがイネーブルの場合、ユーザ定義の speed 値はすべて無視されます。
<p>ステップ 7 <code>negotiation auto</code></p> <p>例: RP/0/0/CPU0:router(config-if)# negotiation auto</p>	<p>(任意) インターフェイスでオートネゴシエーションをイネーブルにします。</p> <ul style="list-style-type: none"> • オートネゴシエーションは接続の両エンドで明示的にイネーブルにするか、接続の両エンドで速度とデュプレックス モードを手動設定する必要があります。 • オートネゴシエーションがイネーブルの場合、手動で設定した速度またはデュプレックス モードの設定の方が優先されます。
<p>ステップ 8 <code>no shutdown</code></p> <p>例: RP/0/0/CPU0:router(config-if)# no shutdown</p>	<p><code>shutdown</code> 設定を削除します。その結果、インターフェイスに強制されていた管理上のダウン状態が解除され、アップ状態またはダウン状態に移行できるようになります。</p>

コマンドまたはアクション	目的
<p>ステップ 9</p> <pre>end または commit</pre> <p>例: RP/0/0/CPU0:router(config-if)# end または RP/0/0/CPU0:router(config-if)# commit </p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 10</p> <pre>show interfaces fastethernet interface-path-id</pre> <p>例: RP/0/0/CPU0:router# show interfaces fastethernet 0/0/1/1 </p>	<p>(任意) ルータ上のインターフェイスに関する統計情報を表示します。</p>

次に行う作業

- ファスト イーサネット インターフェイスで MAC アカウンティングを設定する方法については、このモジュールで後述する「[イーサネット インターフェイスでの MAC アカウンティングの設定](#)」を参照してください。
- ファスト イーサネット インターフェイスで 802.1Q VLAN サブインターフェイスを設定する方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定](#)」モジュールを参照してください。
- レイヤ 2 VPN 実装のファスト イーサネット ポートで AC を設定する方法については、このモジュールで後述する「[イーサネット ポートでの接続回路の設定](#)」を参照してください。
- MPLS や QoS など、レイヤ 3 サービス ポリシーをファスト イーサネット インターフェイスに附加する方法については、該当する Cisco IOS XR ソフトウェアのコンフィギュレーション ガイドを参照してください。

イーサネット インターフェイスでの MAC アカウンティングの設定

このタスクでは、イーサネット インターフェイスでの MAC アカウンティングの設定方法について説明します。MAC アカウントには、この手順で説明する特殊な **show** コマンドがあります。**show** コマンド以外は、基本的なイーサネット インターフェイスの設定と同じなので、手順を 1 回のコンフィギュ

■ イーサネット インターフェイスの設定方法

レーション セッションにまとめることができます。イーサネット インターフェイスの他の一般的なパラメータの設定方法については、このモジュールの「[ギガビット イーサネット インターフェイスまたは 10 ギガビット イーサネット インターフェイスの設定](#)」を参照してください。

手順の概要

1. **configure**
2. **interface [GigabitEthernet | TenGigE | fastethernet] interface-path-id**
3. **ipv4 address ip-address mask**
4. **mac-accounting {egress | ingress}**
5. **end**
または
commit
6. **show mac-accounting type location instance**

詳細手順

コマンドまたはアクション	目的
ステップ 1 configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 interface [GigabitEthernet TenGigE fastethernet] interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイス名と <i>rack/slot/module/port</i> 表記の <i>interface-path-id</i> を指定します。 <ul style="list-style-type: none"> • この例は、モジュラ サービス カード スロット 1 の 8 ポート 10 ギガビット イーサネット インターフェイスです。
ステップ 3 ipv4 address ip-address mask 例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 <ul style="list-style-type: none"> • <i>ip-address</i> をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • <i>mask</i> を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクを指定するには、次の 2 つの方法があります。 <ul style="list-style-type: none"> – ネットワーク マスクには、4 パートのドット付き 10 進アドレスを指定できます。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – ネットワーク マスクはスラッシュ (/) と数字で指定できます。たとえば、/8 は、マスクの先頭 8 ビットが 1 で、アドレスの対応するビットがネットワーク アドレスであることを示します。

コマンドまたはアクション	目的
ステップ 4 <code>mac-accounting {egress ingress}</code> 例: RP/0/RP0/CPU0:router(config-if)# mac-accounting egress	LAN インターフェイス上の発信元 MAC アドレスと宛先 MAC アドレスに基づいて、IP トラフィックのアカウントリング情報を生成します。 <ul style="list-style-type: none"> MAC アカウンティングをディセーブルにするには、このコマンドの no フォームを使用します。
ステップ 5 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 6 <code>show mac-accounting type location instance</code> 例: RP/0/RP0/CPU0:router# show mac-accounting TenGigE location 0/2/0/4	インターフェイスの MAC アカウンティングの統計情報を表示します。

イーサネット ポートでの接続回路の設定

ギガビットイーサネット、10ギガビットイーサネット、またはファストイーサネットポートで接続回路を設定するには、次の手順で操作します。



(注) この手順の各操作では、ポートモードで操作する L2VPN イーサネットポートを設定します。

手順の概要

1. `configure`
2. `interface [GigabitEthernet | TenGigE] interface-path-id`
3. `l2transport`
4. `l2protocol {cdp | pvst | stp | vtp} {[tunnel] experimental bits | drop}`

5. end

または
commit

6. show interfaces [GigabitEthernet | TenGigE] interface-path-id

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface [GigabitEthernet TenGigE] interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイス名と <i>rack/slot/module/port</i> 表記を指定します。このステップで使用できるインターフェイスの種類は次のとおりです。 <ul style="list-style-type: none"> • GigabitEthernet • TenGigE
ステップ 3	l2transport 例: RP/0/RP0/CPU0:router(config-if)# l2transport	ポートでレイヤ 2 トランスポート モードをイネーブルにし、レイヤ 2 トランスポート コンフィギュレーション モードを開始します。
ステップ 4	l2protocol {cdp pvst stp vtp} {[tunnel] experimental bits drop} 例: RP/0/RP0/CPU0:router(config-if-12)#	インターフェイスでレイヤ 2 プロトコルのトンネリングおよびデータ ユニットのパラメータを設定します。 使用できるプロトコルは次のとおりです。 <ul style="list-style-type: none"> • cdp : Cisco Discovery Protocol (CDP; シスコ検出プロトコル) のトンネリングおよびデータ ユニットのパラメータ。 • pvst : VLAN スパニング ツリー プロトコルのトンネリングおよびデータ ユニットのパラメータを設定します。 • stp : スパニング ツリー プロトコルのトンネリングおよびデータ ユニットのパラメータ。 • vtp : VLAN Trunk Protocol のトンネリングおよびデータ ユニットのパラメータ。 指定したプロトコルに関連するパケットをトンネル処理するには、 tunnel オプションを含めます。 指定したプロトコルの MPLS EXP ビットを変更するには、 experimental bits キーワードの引数を含めます。 指定したプロトコルに関連するパケットをドロップするには、 drop キーワードを含めます。

コマンドまたはアクション	目的
<p>ステップ 5 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if-12)# <code>end</code> または RP/0/RP0/CPU0:router(config-if-12)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 6 <code>show interfaces [GigabitEthernet TenGigE]</code> <code>interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show interfaces TenGigE</code> 0/3/0/0</p>	<p>(任意) ルータ上のインターフェイスに関する統計情報を表示します。</p>

次に行う作業

- AC でポイントツーポイントの疑似接続 XConnect を設定する方法については、『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』の「Implementing MPLS Layer 2 VPNs」モジュールを参照してください。
- QoS など、レイヤ 2 サービス ポリシーをイーサネット インターフェイスに付加する方法については、該当する Cisco IOS XR ソフトウェアのコンフィギュレーション ガイドを参照してください。

イーサネット インターフェイスの設定例

ここでは、次の設定例について説明します。

- [イーサネット インターフェイスの設定：例](#)
- [ファスト イーサネット インターフェイスの設定：例](#)
- [MAC アカウンティングの設定：例](#)
- [レイヤ 2 VPN AC：例](#)

イーサネット インターフェイスの設定 : 例

次の例は、10 ギガビット イーサネットのモジュラ サービス カードのインターフェイスを設定する方法です。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# flow-control ingress
RP/0/RP0/CPU0:router(config-if)# mtu 1448
RP/0/RP0/CPU0:router(config-if)# mac-address 0001.2468.ABCD
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/RP0/CPU0:router# show interfaces TenGigE 0/0/0/1
```

```
TenGigE0/0/0/1 is down, line protocol is down
Hardware is TenGigE, address is 0001.2468.abcd (bia 0001.81a1.6b23)
Internet address is 172.18.189.38/27
MTU 1448 bytes, BW 10000000 Kbit
    reliability 0/255, txload Unknown, rxload Unknown
Encapsulation ARPA,
Full-duplex, 10000Mb/s, LR
output flow control is on, input flow control is on
loopback not set
ARP type ARPA, ARP timeout 01:00:00
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

ファスト イーサネット インターフェイスの設定 : 例

次の例は、ファスト イーサネット SPA のインターフェイスを設定する方法です。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface fastethernet 0/0/2/0
RP/0/0/CPU0:router(config-if)# ipv4 address 172.30.1.2 255.255.255.224
RP/0/0/CPU0:router(config-if)# duplex full
RP/0/0/CPU0:router(config-if)# mtu 1514
RP/0/0/CPU0:router(config-if)# speed 100
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/0/CPU0:router# show interfaces fastethernet 0/0/2/0

FastEthernet0/0/2/0 is up, line protocol is up
Hardware is FastEthernet, address is 000f.f83b.30c8 (bia 000f.f83b.30c8)
Internet address is 172.30.1.2/24
MTU 1514 bytes, BW 1000000 Kbit
```

```
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA,
Duplex unknown, 100Mb/s, TX, link type is force-up
output flow control is off, input flow control is off
loopback not set
ARP type ARPA, ARP timeout 04:00:00
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
  0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
  0 runts, 0 giants, 0 throttles, 0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

MAC アカウンティングの設定 : 例

次の例は、イーサネット インターフェイスで MAC アカウンティングを設定する方法です。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/2
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# mac-accounting egress
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# exit
```

レイヤ 2 VPN AC : 例

次の例は、イーサネット インターフェイスでレイヤ 2 VPN AC を設定する方法です。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/2
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# l2transport
RP/0/RP0/CPU0:router(config-if-l2)# l2protocol cdp drop
RP/0/RP0/CPU0:router(config-if-l2)# commit
```

関連情報

イーサネット インターフェイスの設定が完了したら、イーサネット インターフェイスで各 VLAN サブ インターフェイスを設定できます。VLAN サブインターフェイスの設定方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定](#)」モジュールを参照してください。

Shelf Controller (SC; シェルフ コントローラ)、ルートプロセッサ (RP)、および分散型 RP のイーサネット管理インターフェイスの変更方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイスの高度な設定と変更](#)」モジュールを参照してください。

IPv6 については、『Cisco IOS XR IP Addresses and Services Configuration Guide』の「Implementing Access Lists and Prefix Lists on Cisco IOS XR Software」モジュールを参照してください。

その他の参考資料

ギガビット イーサネット、10 ギガビット イーサネット、およびファスト イーサネットの各インターフェイスに関連する参考資料です。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport/



Cisco IOS XR ソフトウェアでのフレームリレーの設定

ここでは、フレームリレーのカプセル化で設定した Packet-over-SONET/SDH (POS)、マルチリンク、およびシリアル インターフェイスで使用できる、任意設定のフレームリレー パラメータについて説明します。

Cisco IOS XR ソフトウェアのフレームリレー インターフェイス設定の機能履歴

リリース	変更点
リリース 3.4.0	Cisco XR 12000 シリーズ ルータにこの機能が追加されました。
リリース 3.5.0	この機能は、IPv6 をサポートするために更新されました。 Layer 2 Tunnel Protocol Version 3 (L2TPv3) は、フレームリレーのカプセル化を設定したシリアル インターフェイスおよび POS インターフェイスでサポートされていました。
リリース 3.6.0	マルチリンク フレームリレー (FRF.16) およびエンドツーエンド フラグメンテーション (FRF.12) は、Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ (SPA) で導入され、2 ポートおよび 4 ポート チャネライズド T3 SPA は Cisco XR 12000 シリーズ ルータで導入されました。
リリース 3.7.0	変更ありません。
リリース 3.8.0	FRF.16 および FRF.12 は、Cisco XR 12000 シリーズ ルータ の Cisco 1 ポート チャネライズド OC-12 SPA で導入されました。 フレームリレーのサポートは Cisco XR 12000 シリーズ ルータの次のラインカードで導入されました。 <ul style="list-style-type: none">• 1 ポート チャネライズド OC-12/STM-1 ラインカード• 4 ポート チャネライズド OC-12/STM-4 ラインカード

この章の構成

- 「フレームリレー設定の前提条件」 (P.170)
- 「フレームリレー インターフェイスに関する情報」 (P.170)
- 「フレームリレーの設定」 (P.177)
- 「フレームリレーの設定例」 (P.192)
- 「その他の参考資料」 (P.196)

フレームリレー設定の前提条件

フレームリレーを設定する前に、次の条件を満たしていることを確認します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 使用しているハードウェアが POS インターフェイスまたはシリアル インターフェイスをサポートしている必要があります。
- 対応するモジュールの説明に従って、**encapsulation frame relay** コマンドを使用し、インターフェイスでフレームリレーのカプセル化をイネーブルにしました。
 - マルチリンク バンドル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このモジュールの「[マルチリンク フレームリレー バンドル インターフェイスの設定 \(P.182\)](#)」を参照してください。
 - POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
 - シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

フレームリレー インターフェイスに関する情報

ここでは、フレームリレー インターフェイスを設定する際のさまざまな側面について説明します。

- 「[フレームリレーのカプセル化 \(P.170\)](#)」
- 「[マルチリンク フレームリレー \(P.173\)](#)」
- 「[エンドツーエンド フラグメンテーション \(FRF.12\) \(P.177\)](#)」

フレームリレーのカプセル化

Cisco XR 12000 シリーズ ルータでは、フレームリレーは、POS インターフェイス、シリアル メイン インターフェイス、およびそれらのインターフェイスで設定された POV でサポートされます。フレームリレーのカプセル化をインターフェイスでイネーブルにするには、インターフェイス コンフィギュレーション モードで **encapsulation frame-relay** コマンドを使用します。

フレームリレー インターフェイスは、次の 2 種類のカプセル化されたフレームをサポートします。

- Cisco (これがデフォルト値です)
- IETF

PVC で Cisco または IETF のカプセル化を設定するには、フレームリレー PVC コンフィギュレーション モードで **encap** コマンドを使用します。



(注) **encap** コマンドで PVC のカプセル化タイプを設定しない場合、その PVC はメイン インターフェイスからカプセル化タイプを継承します。

encapsulation frame relay コマンドおよび **encap** コマンドについては、次のモジュールを参照してください。

- POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

フレームリレーのカプセル化でインターフェイスを設定し、その他の追加コンフィギュレーション コマンドを適用しない場合、表 7 のデフォルト インターフェイス設定が示されます。これらのデフォルト設定は、このモジュールの説明に従って設定で変更できます。

表 7 フレームリレーのカプセル化のデフォルト設定

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定	コマンド モード
PVC Encapsulation	encap [cisco ietf]	cisco (注) encap コマンドを設定しない場合、PVC のカプセル化タイプはフレームリレーのメイン インターフェイスから継承されます。	PVC の設定
Type of support provided by the interface	frame-relay intf-type [dce dte]	dte .	インターフェイス コンフィギュレーション
LMI type supported on the interface	frame-relay lmi-type {ansi cisco q933a}	DCE の場合、デフォルト設定は cisco です。 DTE の場合、デフォルト設定は DCE でサポートされる LMI タイプに合わせて同期されます。 (注) インターフェイスをデフォルトの LMI タイプに戻すには、 no frame-relay lmi-type {ansi cisco q933a} コマンドを使用します。	インターフェイス コンフィギュレーション
Disable or enable LMI	frame-relay lmi disable	デフォルトでフレームリレー インターフェイスの LMI はイネーブルです。 LMI をディセーブルにした後で改めてイネーブルにするには、 no frame-relay lmi disable コマンドを使用します。	インターフェイス コンフィギュレーション



(注) LMI ポーリング関連のコマンドのデフォルト設定については、表 8 (P.173) と表 9 (P.173) を参照してください。

LMI

Local Management Interface (LMI; ローカル管理インターフェイス) プロトコルは、PVC の追加、削除、およびステータスをモニタリングします。また、フレームリレーの User-Network Interface (UNI; ユーザネットワーク インターフェイス) を構成するリンクの完全性も検証します。

フレームリレー インターフェイスは、UNI で次のタイプの LMI をサポートします。

- ANSI-ANSI T1.617 Annex D
- Q.933-ITU-T Q.933 Annex A
- Cisco

インターフェイスで使用する LMI タイプを設定するには、**frame-relay lmi-type** コマンドを使用します。



(注) 使用する LMI タイプは、メイン インターフェイスに設定されている PVC と対応している必要があります。フレームリレー接続の両エンドの LMI タイプは一致する必要があります。

使用しているルータが別の非フレームリレー ルータに接続するスイッチとして機能する場合、**frame-relay intf-type dce** コマンドを使用して、Data Communication Equipment (DCE; データ通信機器) をサポートする LMI タイプを設定します。

ルータがフレームリレー ネットワークに接続している場合、**frame-relay intf-type dte** コマンドを使用して、Data Terminal Equipment (DTE; データ端末装置) をサポートする LMI タイプを設定します。



(注) DTE インターフェイスでは、デフォルトで LMI タイプの自動検知がサポートされています。

システムのフレームリレー インターフェイスに関する統計情報を表示するには、EXEC モードで **show frame-relay lmi** コマンドを使用します (このコマンドの実際の構文は、**show frame-relay lmi interface type interface-path-id** です。type 引数および *interface-path-id* 引数を指定する場合、メイン インターフェイスの情報を指定する必要があります)。エラーのしきい値、イベント カウント、およびポーリングの検証タイマーを変更し、**show frame-relay lmi** コマンドを使用して、フレームリレー インターフェイスをモニタリングし、発生した問題を解決するときに役立つ情報を収集できます。

LMI タイプが **cisco** (デフォルトの LMI タイプ) の場合、単一のインターフェイスでサポートできる PVC の最大数は、メイン インターフェイスの MTU サイズに関連します。次の式を使用して、カードまたは SPA でサポートされる PVC の最大数を計算します。

$$(MTU - 13) / 8 = \text{PVC の最大数}$$

cisco LMI で設定した POS PVC でサポートされる PVC のデフォルトの数は 557 です。また、**cisco** LMI で設定したシリアル PVC でサポートされる PVC のデフォルトの数は 186 です。

シスコ製ではない LMI タイプの場合、単一のメイン インターフェイスで最大 992 PVC がサポートされます。



(注) 特定の LMI タイプをインターフェイスに設定する場合、**no frame-relay lmi-type {ansi | cisco | q933a}** コマンドを使用して、インターフェイスをデフォルトの LMI タイプに戻します。

表 8 は、DCE 用に設定した PVC で LMI ポーリング オプションを変更するときに使用できるコマンドです。

表 8 DCE の LMI ポーリング コンフィギュレーション コマンド

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定
Sets the error threshold on a DCE interface.	lmi-n392dce threshold	3
Sets the monitored event count.	lmi-n393dce events	4
Sets the polling verification timer on the DCE end.	lmi-t392dce seconds	15

表 9 は、DTE 用に設定した PVC で LMI ポーリング オプションを変更するときに使用できるコマンドです。

表 9 DTE の LMI ポーリング コンフィギュレーション コマンド

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定
Set the number of Line Integrity Verification (LIV) exchanges performed before requesting a full status message.	lmi-n391dte polling-cycles	6
Sets the error threshold.	lmi-n392dte threshold	3
Sets the monitored event count.	lmi-n393dte events	4
Sets the polling interval (in seconds) between each status inquiry from the DTE end.	frame-relay lmi-t391dte seconds	10

マルチリンク フレームリレー

マルチリンク フレームリレーは、次の SPA でのみサポートされます。

- Cisco 1 ポート チャネライズド STM-1/OC-3 SPA
- 2 ポートおよび 4 ポート チャネライズド T3 SPA
- Cisco 1 ポート チャネライズド OC-12 SPA

マルチリンク フレームリレー インターフェイスは、インターフェイスでフレームリレーのカプセル化を可能にするマルチリンク バンドルの一部です。マルチリンク フレームリレー インターフェイスを作成するには、次のコンポーネントを設定します。

- MgmtMultilink コントローラ
- フレームリレーのカプセル化を可能にするマルチリンク バンドル インターフェイス
- バンドル ID 名
- マルチリンク フレームリレー サブインターフェイス
- バンドル インターフェイスの帯域幅クラス
- シリアル インターフェイス

MgmtMultilink コントローラ

次のコマンドを使用して、コントローラのマルチリンク バンドルを設定します。

```
controller MgmtMultilink rack/slot/bay/controller-id
bundle bundleId
```

この設定で、汎用マルチリンク バンドルのコントローラが作成されます。コントローラ ID 番号はコントローラ チップのゼロベース インデックスです。現在、マルチリンク フレームリレーをサポートする SPA には、1 ペイごとに 1 コントローラしかないため、コントローラの ID 番号は常にゼロ (0) です。

マルチリンク バンドル インターフェイス

マルチリンク バンドルを作成した後は、次のコマンドを使用して、フレームリレーのカプセル化を可能にするマルチリンク バンドル インターフェイスを作成します。

```
interface Multilink rack/slot/bay/controller-id/bundleId
encapsulation frame-relay
```

この設定で、マルチリンク バンドル インターフェイスにマルチリンク フレームリレー サブインターフェイスを作成できます。



(注)

マルチリンク バンドル インターフェイス上のカプセル化をフレームリレーに設定した後は、マルチリンク バンドルに関連付けられたメンバー リンクがインターフェイスにある場合、カプセル化は変更できません。

バンドル ID 名



(注)

バンドル ID 名は、Frame Relay Forum 16.1 (FRF 16.1; フレームリレー フォーラム 16.1) でのみ設定できます。

バンドル ID (**bid**) 名は、インターフェイスの両エンドポイントのバンドル インターフェイスを識別します。バンドル ID 名は、一貫したリンクの割り当てを確保するために情報要素で交換されます。

デフォルトで、インターフェイス名 (たとえば **Multilink 0/4/1/0/1**) がバンドル ID 名として使用されます。ただし、オプションで **frame-relay multilink bid** コマンドを使用して名前を作成することもできます。



(注)

デフォルトの名前を使用するか、**frame-relay multilink bid** コマンドを使用して名前を作成するかにかかわらず、各バンドルには固有の名前を指定する必要があります。異なるバンドルで同じ名前が使用されると、バンドルは永続的にフラップします。

バンドル ID 名の長さは、ヌルの終端文字を含めて 50 文字までです。バンドル ID 名はバンドル インターフェイス レベルで設定され、各メンバー リンクに適用されます。

バンドル ID 名を設定するには、次のコマンドを使用します。

```
interface Multilink rack/slot/bay/controller-id/bundleId
frame-relay multilink bid bundle-id-name
```

マルチリンク フレームリレー サブインターフェイス

マルチリンク フレームリレー サブインターフェイスを設定するには、次のコマンドを使用します。


```
interface Multilink rack/slot/bay/controller-id/bundleId.subinterface
[point-to-point | l2transport ]
```

1 つのマルチリンク バンドル インターフェイスには最大 992 サブインターフェイスを設定できます。



(注) サブインターフェイス レベルで特定のフレームリレー インターフェイス機能を設定します。

マルチリンク フレームリレー サブインターフェイス機能

次のコマンドは、マルチリンク フレームリレー バンドル サブインターフェイスで特定の機能を設定するために使用できます。

- `mtu MTU size`
- `description`
- `shutdown`
- `bandwidth bandwidth`
- `service-policy {input | output} policymap-name`



(注) `service-policy` コマンドを入力すると、ポリシー マップをマルチリンク フレームリレー バンドル サブインターフェイスに付加できるようになりますが、フレームリレー PVC コンフィギュレーション モードで次の操作を実行する必要があります。詳細については、「[マルチリンク フレームリレー バンドル インターフェイスの設定](#)」(P.182) を参照してください。

バンドル インターフェイスの帯域幅クラス



(注) 帯域幅クラスは、マルチリンク バンドル インターフェイスでのみ設定できます。

マルチリンク フレームリレー インターフェイスでは、3 タイプの帯域幅クラスのいずれかを設定できます。

- a - 帯域幅クラス A
- b - 帯域幅クラス B
- c - 帯域幅クラス C

帯域幅クラス A を設定し、1 つまたは複数のメンバー リンクがアップ (PH_ACTIVE) の場合、バンドル インターフェイスもアップで、BL_ACTIVATE がフレームリレー接続にシグナリングされます。すべてのメンバー リンクがダウンの場合、バンドル インターフェイスはダウンで、BL_DEACTIVATE がフレームリレー接続にシグナリングされます。

帯域幅クラス B を設定し、すべてのメンバー リンクがアップ (PH_ACTIVE) の場合、バンドル インターフェイスはアップで、BL_ACTIVATE がフレームリレー接続にシグナリングされます。いずれかのメンバー リンクがダウンの場合、バンドル インターフェイスはダウンで、BL_ACTIVATE がフレームリレー接続にシグナリングされません。

帯域幅クラス C を設定する場合、バンドルのリンクのしきい値を 1 ~ 255 に設定する必要があります。このしきい値は、バンドル インターフェイスをアップにするため、およびフレームリレー接続に BL_ACTIVATE をシグナリングするために必要な、アップ (PH_ACTIVE) にするリンクの最小数です。アップ状態のリンク数がこのしきい値未満の場合、バンドル インターフェイスがダウンになり、BL_DEACTIVATE がフレームリレー接続にシグナリングされます。しきい値に 1 を入力した場合の動作は、帯域幅クラス A と同じです。アップ状態のメンバー リンク数よりも多いしきい値を入力すると、バンドルはダウンのままです。

フレームリレー マルチリンク バンドル インターフェイスの帯域幅クラスを設定するには、次のコマンドを使用します。

```
interface Multilink rack/slot/bay/controller-id/bundleId
frame-relay multilink bandwidth-class a | b | c threshold
```

デフォルトは a (帯域幅クラス A) です。

シリアル インターフェイス

T3 コントローラと T1 コントローラを設定した後は、マルチリンク フレームリレー バンドル サブインターフェイスにシリアル インターフェイスを追加できます。この場合、シリアル インターフェイスを設定し、マルチリンク フレームリレー (mfr) としてカプセル化し、それをバンドル インターフェイス (マルチリンク グループ番号によって指定されます) に割り当て、リンクの名前を設定します。

マルチリンク フレームリレー シリアル インターフェイスを設定するには、次のコマンドを使用します。

```
interface serial rack/slot/module/port/t1-num:channel-group-number
encapsulation mfr
multilink group group number
frame-relay multilink lid link-id name
```



(注)

MFR バンドルのすべてのシリアル リンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MFR バンドルのメンバーとして設定する前に、シリアル インターフェイスで **mtu** コマンドを設定しないでください。デフォルト以外の MTU 値を設定している場合、MFR バンドルのメンバーとしてシリアル インターフェイスを設定しようとすると、Cisco IOS XR ソフトウェアによってブロックされます。また、MFR バンドルのメンバーとして設定されているシリアル インターフェイスの **mtu** コマンド値を変更しようとした場合もブロックされます。

表示コマンド

マルチリンク フレームリレー シリアル インターフェイスの設定を検証するには、次の **show** コマンドを使用します。

```
show frame-relay multilink location
show frame-relay multilink interface serial
```

次の例は、**show frame-relay multilink location** コマンドの表示出力です。

```
RP/0/0/CPU0:router# show frame-relay multilink location 0/4/cpu0
Member interface: Serial0/4/2/0/9:0, ifhandle 0x05007b00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

Bundle interface: Multilink0/4/2/0/2, ifhandle 0x05007800
  Member Links: 4 active, 0 inactive
  State = Up, BW Class = C (threshold 3)
  Member Links:
    Serial0/4/2/0/12:0, HW state = Up, link state = Up
    Serial0/4/2/0/11:0, HW state = Up, link state = Up
    Serial0/4/2/0/10:0, HW state = Up, link state = Up
    Serial0/4/2/0/9:0, HW state = Up, link state = Up

Member interface: Serial0/4/2/0/10:0, ifhandle 0x05007c00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800
```

```
Member interface: Serial0/4/2/0/11:0, ifhandle 0x05007d00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800
```

```
Member interface: Serial0/4/2/0/12:0, ifhandle 0x05007e00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800
```

次の例は、コマンドの表示出力です。

```
RP/0/0/CPU0:router# show frame-relay multilink interface serial 0/4/2/0/10:0

Member interface: Serial0/4/2/0/10:0, ifhandle 0x05007c00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800
```

エンドツーエンド フラグメンテーション (FRF.12)

Data-Link Connection Identifier (DLCI; データリンク接続識別子) を使用して、FRF.12 エンドツーエンドフラグメンテーション接続を設定することができます。ただし、チャネライズドフレームリレーシリアルインターフェイスで設定する必要があります。



(注) **fragment end-to-end** コマンドは、POS インターフェイス、またはマルチリンク フレームリレー バンドルインターフェイスの DLCI では使用できません。

FRF.12 エンドツーエンドフラグメンテーションを DLCI 接続で設定するには、次のコマンドを使用します。

```
fragment end-to-end fragment-size
```

fragment-size 引数には、シリアルインターフェイスのフラグメント サイズをバイト単位で定義します。



(注) DLCI 接続では、優先順位の高いフラグメントと低いフラグメントのインターリーピングが発生するように、パケットを優先順位の高低で分類する出力サービス ポリシーを設定することを強くお勧めします。

フレームリレーの設定

次の項では、フレームリレー インターフェイスの設定方法について説明します。

- 「インターフェイスでのデフォルト フレームリレー設定の変更」 (P.178)
- 「フレームリレーのカプセル化を設定したインターフェイスでの LMI のディセーブル」 (P.180)
- 「マルチリンク フレームリレー バンドル インターフェイスの設定」 (P.182)
- 「チャネライズドフレームリレー シリアルインターフェイスでの FRF.12 エンドツーエンドフラグメンテーションの設定」 (P.188)

インターフェイスでのデフォルト フレームリレー設定の変更

フレームリレーのカプセル化を設定した POS インターフェイス、マルチリンク インターフェイス、またはシリアル インターフェイスでデフォルトのフレームリレー パラメータを変更するには、次のタスクを実行します。

前提条件

デフォルトのフレームリレー設定を変更する前に、次のモジュールの説明に従ってインターフェイスでフレームリレーをイネーブルにする必要があります。

- POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。



(注) POS インターフェイスまたはシリアル インターフェイスでフレームリレーのカプセル化をイネーブルにする前に、そのインターフェイスに IP アドレスを以前に割り当てたことがないことを確認します。IP アドレスが割り当て済みの場合、フレームリレーのカプセル化をイネーブルにすることはできません。フレームリレーの場合、IP アドレスとサブネット マスクはサブインターフェイスで設定します。

制約事項

- LMI タイプは、アクティブにする接続の両エンドで一致する必要があります。
- インターフェイスでフレームリレーのカプセル化を削除し、そのインターフェイスを PPP または HDLC のカプセル化で設定し直す前に、すべてのインターフェイス、サブインターフェイス、LMI、およびそのインターフェイスのフレームリレー設定を削除する必要があります。

手順の概要

1. `configure`
2. `interface type interface-path-id`
3. `frame-relay intf-type [dce | dte]`
4. `frame-relay lmi-type {ansi | cisco | q933a}`
5. `encap [cisco | ietf]`
6. `end`
または
`commit`
7. `show interfaces type [interface-path-id]`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface type interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface pos 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>frame-relay intf-type [dce dte]</code> 例: RP/0/0/CPU0:router(config-if)# frame-relay intf-type dce	インターフェイスで提供するサポートのタイプを設定します。 <ul style="list-style-type: none"> • 使用しているルータが別のルータに接続するスイッチとして機能する場合、frame-relay intf-type dce コマンドを使用して、DCE をサポートする LMI タイプを設定します。 • ルータがフレームリレー ネットワークに接続している場合、frame-relay intf-type dte コマンドを使用して、DTE をサポートする LMI タイプを設定します。 (注) デフォルトのインターフェイス タイプは DTE です。
ステップ 4	<code>frame-relay lmi-type {ansi q933a cisco}</code> 例: RP/0/0/CPU0:router(config-if)# frame-relay lmi-type ansi	インターフェイスでサポートする LMI タイプを選択します。 <ul style="list-style-type: none"> • ANSI T1.617a-1994 Annex D の定義に従って LMI を使用するには、frame-relay lmi-type ansi コマンドを入力します。 • Cisco の定義 (標準ではありません) に従って LMI を使用するには、frame-relay lmi-type cisco コマンドを使用します。 • ITU-T Q.933 (02/2003) Annex A の定義に従って LMI を使用するには、frame-relay lmi-type q933a コマンドを使用します。 (注) デフォルトの LMI タイプは Cisco です。
ステップ 5	<code>encap [cisco ietf]</code> 例: RP/0/0/CPU0:router (config-subif)# encap ietf	フレームリレー PVC のカプセル化を設定します。 (注) PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン インターフェイスからカプセル化タイプを継承します。

■ フレームリレーの設定

コマンドまたはアクション	目的
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-if)# <code>end</code> または RP/0/0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7 <code>show interfaces type [interface-path-id]</code></p> <p>例: RP/0/0/CPU0:router# <code>show interface pos 0/4/0/1</code></p>	<p>(任意) 指定したインターフェイスの設定を検証します。</p>

フレームリレーのカプセル化を設定したインターフェイスでの LMI のディセーブル

フレームリレーのカプセル化が設定されたインターフェイスで LMI をディセーブルにするには、次のタスクを実行します。



(注)

フレームリレーのカプセル化がイネーブルなインターフェイスでは、デフォルトで LMI がイネーブルです。インターフェイスの LMI をディセーブルにした後で改めてイネーブルにするには、インターフェイス コンフィギュレーション モードで **no frame-relay lmi disable** コマンドを使用します。

手順の概要

1. **configure**
2. **interface type interface-path-id**
3. **frame-relay lmi disable**
4. **end**
または
commit
5. **show interfaces type [interface-path-id]**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface type interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface POS 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>frame-relay lmi disable</code> 例: RP/0/0/CPU0:router(config-if)# frame-relay lmi disable	指定したインターフェイスで LMI をディセーブルにします。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 5	<code>show interfaces type interface-path-id</code> 例: RP/0/0/CPU0:router# show interfaces POS 0/1/0/0	(任意) 指定したインターフェイスで LMI がディセーブルになっていることを確認します。

マルチリンク フレームリレー バンドル インターフェイスの設定

マルチリンク フレームリレー バンドル インターフェイスとそのサブインターフェイスを設定するには、次の手順で操作します。

制約事項

- マルチリンク フレームリレー バンドル インターフェイスのすべてのメンバー リンクは、同じタイプにする必要があります。メンバー リンクは、ポイントツーポイントなど、同じフレーム構成タイプにし、同じ帯域幅クラスにする必要があります。
- すべてのリンクは、遠端でも同じラインカードまたは SPA に接続している必要があります。
- MFR バンドルのすべてのシリアル リンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MFR バンドルのメンバーとして設定する前に、シリアル インターフェイスで **mtu** コマンドを設定しないでください。Cisco IOS XR ソフトウェアは次の処理をブロックします。
 - インターフェイスがデフォルト以外の MTU 値で設定されている場合、MFR バンドルのメンバーとしてシリアル インターフェイスを設定しようとする処理。
 - MFR バンドルのメンバーとして設定されているシリアル インターフェイスの **mtu** コマンド値を変更しようとする処理。

手順の概要

1. **config**
2. **controller MgmtMultilink rack/slot/bay/controller-id**
3. **exit**
4. **controller t3 interface-path-id**
5. **mode type**
6. **clock source {internal | line}**
7. **exit**
8. **controller t1 interface-path-id**
9. **channel-group channel-group-number**
10. **timeslots range**
11. **exit**
12. **exit**
13. **interface Multilink rack/slot/bay/controller-id/bundleId**
14. **encapsulation frame-relay**
15. **frame-relay multilink bid bundle-id-name**
16. **frame-relay multilink bandwidth-class a | b | c threshold**
17. **multilink fragment-size size**
18. **exit**
19. **interface Multilink rack/slot/bay/controller-id/bundleId.**
subinterface [point-to-point | l2transport]

20. `ipv4 address ip-address`
21. `pvc dlc`
22. `service-policy {input | output} policy-map`
23. `exit`
24. `exit`
25. `interface serial interface-path-id`
26. `encapsulation mfr`
27. `multilink group group-id`
28. `frame-relay multilink lid link-id name`
29. `exit`
30. `end`
または
`commit`
31. `exit`
32. `show frame-relay interface multilink rack/slot/bay/controller-id/bundleId`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>config</code> 例： RP/0/0/CPU0:router# config	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller MgmtMultilink rack/slot/bay/controller-id</code> 例： RP/0/0/CPU0:router(config)# controller MgmtMultilink 0/1/0/0	<code>rack/slot/bay/controller-id</code> 表記で汎用マルチリンクバンドルのコントローラを作成し、マルチリンク管理コンフィギュレーションモードを開始します。コントローラ ID 番号はコントローラチップのゼロベース インデックスです。現在、マルチリンクフレームリレーをサポートする SPA には、1 ベイごとに 1 コントローラしかないため、コントローラの ID 番号は常にゼロ (0) です。
ステップ 3	<code>exit</code> 例： RP/0/0/CPU0:router(config-mgmtmultilink)# exit	マルチリンク管理コンフィギュレーションモードを終了します。
ステップ 4	<code>controller t3 interface-path-id</code> 例： RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<code>rack/slot/module/port</code> 表記で T3 コントローラ名を指定し、T3 コンフィギュレーションモードを開始します。
ステップ 5	<code>mode type</code> 例： RP/0/0/CPU0:router(config-t3)# mode t1	チャネライズするマルチリンクのタイプを設定します (たとえば、28 T1)。

■ フレームリレーの設定

コマンドまたはアクション	目的
ステップ 6 <code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-t3)# clock source internal	(任意) 個々の E3 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアルリンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 7 <code>exit</code> 例: RP/0/0/CPU0:router(config-t3)# exit	T3/E3 コントローラ コンフィギュレーション モードを終了します。
ステップ 8 <code>controller t1 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t1 0/1/0/0/0	T1 コンフィギュレーション モードを開始します。
ステップ 9 <code>channel-group channel-group-number</code> 例: RP/0/0/CPU0:router(config-t1)# channel-group 0	T1 チャンネル グループを作成し、そのチャンネル グループのチャンネル グループ コンフィギュレーション モードを開始します。
ステップ 10 <code>timeslots range</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-24	1 つまたは複数の DS0 タイムスロットをチャンネル グループに関連付け、関連付けたシリアル サブインターフェイスをそのチャンネル グループに作成します。 <ul style="list-style-type: none"> • 範囲は 1 ~ 24 タイムスロットです。 • 24 タイムスロットすべてを単一のチャンネル グループに割り当てることも、タイムスロットを複数のチャンネル グループに分割することもできます。 (注) 個々の T1 コントローラは、24 DS0 タイムスロットの合計をサポートします。
ステップ 11 <code>exit</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# exit	チャンネル グループ コンフィギュレーション モードを終了します。
ステップ 12 <code>exit</code> 例: RP/0/0/CPU0:router(config-t1)# exit	T1 コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 13	<pre>interface Multilink rack/slot/bay/controller-id/bundleId</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config)# interface Multilink 0/1/0/0/100</pre>	バンドルのフレームリレーのカプセル化を指定するマルチリンク バンドル インターフェイスを作成します。マルチリンク バンドル インターフェイスにマルチリンク フレームリレー サブインターフェイスを作成します。
ステップ 14	<pre>encapsulation frame-relay</pre> <p>例:</p> <pre>Router(config-if)# encapsulation frame-relay</pre>	フレームリレーのカプセル化タイプを指定します。
ステップ 15	<pre>frame-relay multilink bid bundle-id-name</pre> <p>例:</p> <pre>Router(config-if)# frame-relay multilink bid MFRBundle</pre>	<p>(任意) デフォルトで、インターフェイス名 (たとえば Multilink 0/4/1/0/1) がバンドル ID 名として使用されます。ただし、オプションで frame-relay multilink bid コマンドを使用して名前を作成することもできます。</p> <p>(注) デフォルトの名前を使用するか、frame-relay multilink bid コマンドを使用して名前を作成するかにかかわらず、各バンドルには固有の名前を指定する必要があります。異なるバンドルで同じ名前が使用されると、バンドルは永続的にフラップします。</p>
ステップ 16	<pre>frame-relay multilink bandwidth-class a b c threshold</pre> <p>例:</p> <pre>Router(config-if)# frame-relay multilink bandwidth-class a</pre>	<p>マルチリンク フレームリレー インターフェイスでは、3 タイプの帯域幅クラスのいずれかを設定します。</p> <ul style="list-style-type: none"> • a - 帯域幅クラス A • b - 帯域幅クラス B • c — 帯域幅クラス C <p>デフォルトは a (帯域幅クラス A) です。</p>
ステップ 17	<pre>multilink fragment-size size</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-if)# multilink fragment-size 256</pre>	<p>(任意) マルチリンク フラグメントのサイズを指定します。デフォルトは no fragments です。</p>
ステップ 18	<pre>exit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router(config-if)# exit</pre>	インターフェイス コンフィギュレーション モードを終了します。

■ フレームリレーの設定

コマンドまたはアクション	目的
ステップ 19 <code>interface Multilink</code> <code>rack/slot/bay/controller-id/bundleId.</code> <code>subinterface [point-to-point l2transport]</code> 例: RP/0/0/CPU0:router(config)# interface Multilink 0/1/0/0/100.16 point-to-point	<code>rack/slot/bay/controller-id bundleId.subinterace</code> [point-to-point l2transport] 表記でマルチリンクサブインターフェイスを作成し、サブインターフェイス コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • l2transport - 接続回路として扱います。 • point-to-point - ポイントツーポイント リンクとして扱います。 1 つのマルチリンク バンドル インターフェイスには最大 992 サブインターフェイスを設定できます。DLCI は 16 ~ 1007 です。
ステップ 20 <code>ipv4 address ip-address</code> 例: RP/0/0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0	次の形式でインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 A.B.C.D/prefix or A.B.C.D/mask
ステップ 21 <code>pvc dlci</code> 例: RP/0/0/CPU0:router (config-subif)# pvc 16	POS 相手先固定接続 (PVC) を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。 <code>dlci</code> を 16 から 1007 の範囲の PVC ID に置き換えます。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。
ステップ 22 <code>service-policy {input output} policy-map</code> 例: RP/0/0/CPU0:router(config-fr-vc) # service-policy output policy-mapA	ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。 (注) ポリシー マップの作成と設定については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。
ステップ 23 <code>exit</code> 例: RP/0/0/CPU0:router(config-fr-vc) # exit	フレームリレー仮想回線モードを終了します。
ステップ 24 <code>exit</code> 例: RP/0/0/CPU0:router(config-subif) # exit	サブインターフェイス コンフィギュレーション モードを終了します。
ステップ 25 <code>interface serial interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0	完全なインターフェイス番号を <code>rack/slot/module/port/T3Num/T1num:instance</code> 表記で指定します。
ステップ 26 <code>encapsulation mfr</code> 例: RP/0/0/CPU0:router(config) # encapsulation mfr	シリアル インターフェイスでマルチリンク フレームリレーをイネーブルにします。

コマンドまたはアクション	目的
ステップ 27 <code>multilink group group-id</code> 例: RP/0/0/CPU0:router(config-if)# multilink group 100	このインターフェイスのマルチリンク グループ ID を指定します。
ステップ 28 <code>frame-relay multilink lid link-id name</code> 例: RP/0/0/CPU0:router(config-if)# frame-relay multilink lid sj1	フレームリレー マルチリンク バンドル リンクの名前を設定します。 (注) バンドル内の各リンクには固有の名前を指定する必要があります。同じバンドル内の異なるリンクに同じ名前が使用されている場合、バンドルは永続的にフラップします。
ステップ 29 <code>exit</code> 例: RP/0/0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了します。
ステップ 30 <code>end</code> または commit 例: RP/0/0/CPU0:router(config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 31 <code>exit</code> 例: RP/0/0/CPU0:router(config)# exit	グローバル コンフィギュレーション モードを終了します。
ステップ 32 <code>show frame-relay interface multilink rack/slot/bay/controller-id/bundleId</code> 例: RP/0/0/CPU0:router# show frame-relay interface Multilink 0/5/1/0/1	バンドル固有の情報やフレームリレー情報など、Interface Description Block (IDB; インターフェイス デスクリプション ブロック) から取得した情報を表示します。

チャネライズド フレームリレー シリアル インターフェイスでの FRF.12 エンドツーエンド フラグメンテーションの設定

チャネライズド フレームリレー シリアル インターフェイスで FRF.12 エンドツーエンド フラグメンテーションを設定するには、次の手順で操作します。

手順の概要

1. **config**
2. **controller t3** *interface-path-id*
3. **mode** *type*
4. **clock source** {**internal** | **line**}
5. **exit**
6. **controller t1** *interface-path-id*
7. **channel-group** *channel-group-number*
8. **timeslots** *range*
9. **exit**
10. **exit**
11. **interface serial** *interface-path-id*
12. **encapsulation** **frame-relay**
13. **exit**
14. **interface serial** *interface-path-id*
15. **ipv4 address** *ip-address*
16. **pvc** *dci*
17. **service-policy** {**input** | **output**} *policy-map*
18. **fragment end-to-end** *fragment-size*
19. **exit**
20. **exit**
21. **exit**
22. **end**
または
commit
23. **exit**
24. **show frame-relay pvc** [**dci** | **interface** | **location**]

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>config</code> 例: RP/0/0/CPU0:router# config	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller t3 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<code>rack/slot/module/port</code> 表記で T3 コントローラ名を指定し、T3 コンフィギュレーション モードを開始します。
ステップ 3	<code>mode type</code> 例: RP/0/0/CPU0:router(config-t3)# mode t1	チャネライズするマルチリンクのタイプを設定します (たとえば、28 T1)。
ステップ 4	<code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-t3)# clock source internal	(任意) 個々の E3 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアルリンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 5	<code>exit</code> 例: RP/0/0/CPU0:router(config-t3)# exit	T3/E3 または T1/E1 コントローラ コンフィギュレーション モードを終了します。
ステップ 6	<code>controller t1 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t1 0/1/0/0/0	T1 コンフィギュレーション モードを開始します。
ステップ 7	<code>channel-group channel-group-number</code> 例: RP/0/0/CPU0:router(config-t1)# channel-group 0	T1 チャネル グループを作成し、そのチャネルグループのチャネルグループ コンフィギュレーション モードを開始します。

■ フレームリレーの設定

コマンドまたはアクション	目的
ステップ 8 <code>timeslots range</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-24	1 つまたは複数の DS0 タイムスロットをチャンネルグループに関連付け、関連付けたシリアル サブインターフェイスをそのチャンネルグループに作成します。 <ul style="list-style-type: none"> • 範囲は 1 ~ 24 タイムスロットです。 • 24 タイムスロットすべてを単一のチャンネルグループに割り当てることも、タイムスロットを複数のチャンネルグループに分割することもできます。 (注) 個々の T1 コントローラは、24 DS0 タイムスロットの合計をサポートします。
ステップ 9 <code>exit</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# exit	チャンネルグループ コンフィギュレーション モードを終了します。
ステップ 10 <code>exit</code> 例: RP/0/0/CPU0:router(config-t1)# exit	T1 コンフィギュレーション モードを終了します。
ステップ 11 <code>interface serial interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0	完全なインターフェイス番号を <code>rack/slot/module/port/T3Num/T1num:instance</code> 表記で指定します。
ステップ 12 <code>encapsulation frame-relay</code> 例: RP/0/0/CPU0:Router(config-if)# encapsulation frame-relay	フレームリレーのカプセル化タイプを指定します。
ステップ 13 <code>exit</code> 例: RP/0/0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了します。
ステップ 14 <code>interface serial interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 1/0/0/0/0:0.1	<code>rack/slot/module/port[/channel-num:channel-group-number].subinterface</code> 表記で、完全なサブインターフェイス番号を指定します。
ステップ 15 <code>ipv4 address ip-address</code> 例: RP/0/0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0	次の形式でインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 A.B.C.D/prefix or A.B.C.D/mask

コマンドまたはアクション	目的
ステップ 16 <code>pvc dldci</code> 例: RP/0/0/CPU0:router (config-subif)# pvc 100	POS PVC を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。 <i>dldci</i> を 16 から 1007 の範囲の PVC ID に置き換えます。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。
ステップ 17 <code>service-policy {input output} policy-map</code> 例: RP/0/0/CPU0:router (config-fr-vc)# service-policy output policy-mapA	ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。 (注) 効率的な FRF.12 機能（具体的にはインターリーブ）のためには、出力サービス ポリシーに優先順位を設定します。 (注) ポリシー マップの作成と設定については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。
ステップ 18 <code>fragment end-to-end fragment-size</code> 例: RP/0/0/CPU0:router (config-fr-vc)# fragment end-to-end 100	インターフェイスでフレームリレー フレームのフラグメンテーションをイネーブルにします。 <i>fragment-size</i> を、発信元フレームリレー フレームのペイロード バイト数に置き換えます。これが各フラグメントのバイト数になります。この数値には、元のフレームのフレームリレー ヘッダーは含まれません。 有効な値は 16 ~ 1600 です。デフォルト値は 53 です。
ステップ 19 <code>exit</code> 例: RP/0/0/CPU0:router (config-fr-vc)# exit	フレームリレー仮想回線モードを終了します。
ステップ 20 <code>exit</code> 例: RP/0/0/CPU0:router (config-subif)# exit	サブインターフェイス コンフィギュレーション モードを終了します。
ステップ 21 <code>exit</code> 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了します。

■ フレームリレーの設定例

コマンドまたはアクション	目的
<p>ステップ 22 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-if)# <code>end</code> または RP/0/0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 23 <code>exit</code></p> <p>例: RP/0/0/CPU0:router(config)# <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了します。</p>
<p>ステップ 24 <code>show frame-relay pvc [dlci interface location]</code></p> <p>例: RP/0/0/CPU0:router# <code>show frame-relay pvc 100</code></p>	<p>指定した PVC DLCI、インターフェイス、または場所の情報を表示します。</p>

フレームリレーの設定例

ここでは、次の設定例について説明します。

- 「オプションのフレームリレー パラメータ : 例」 (P.193)
- 「マルチリンク フレームリレー : 例」 (P.195)
- 「エンドツーエンド フラグメンテーション : 例」 (P.196)

オプションのフレームリレー パラメータ : 例

次の例は、フレームリレーのカプセル化を設定した POS インターフェイスを始動および設定する方法です。この例では、インターフェイスが DCE で ANSI T1.617a-1994 Annex D LMI をサポートするように、デフォルトのフレームリレー設定を変更します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface POS 0/3/0/0
RP/0/0/CPU0:router (config-if)# encapsulation frame-relay IETF
RP/0/0/CPU0:router (config-if)# frame-relay intf-type dce
RP/0/0/CPU0:router (config-if)# frame-relay lmi-type ansi
RP/0/0/CPU0:router (config-if)# no shutdown
RP/0/0/CPU0:router (config-if)# end
```

```
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface pos 0/3/0/0.10 point-to-point
RP/0/0/CPU0:router (config-subif)# ipv4 address 10.46.8.6/24
RP/0/0/CPU0:router (config-subif)# pvc 20
RP/0/0/CPU0:router (config-fr-vc)# encap ietf
RP/0/0/CPU0:router (config-subif)# commit
```

次の例は、フレームリレーのカプセル化を設定したシリアル インターフェイスで LMI をディセーブルにする方法です。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface
RP/0/0/CPU0:router (config)# interface serial 0/3/0/0/0:0
RP/0/0/CPU0:router (config-if)# frame-relay lmi disable
RP/0/0/CPU0:router (config-if)# end
```

```
Uncommitted changes found, commit them? [yes]: yes
```

次の例は、シリアル インターフェイスで LMI を再度イネーブルにする方法です。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface
RP/0/0/CPU0:router (config)# interface serial 0/3/0/0/0:0
RP/0/0/CPU0:router (config-if)# no frame-relay lmi disable
RP/0/0/CPU0:router (config-if)# end
```

```
Uncommitted changes found, commit them? [yes]: yes
```

次の例は、POS インターフェイス上の LMI に関するフレームリレーの統計情報を表示する方法です。

```
RP/0/0/CPU0:router# show frame-relay lmi

LMI Statistics for interface POS0/1/0/0/ (Frame Relay DCE) LMI TYPE = ANSI
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 9
Invalid Information ID 0           Invalid Report IE Len 0
Invalid Report Request 0           Invalid Keep IE Len 0
Num Status Enq. Rcvd 9444          Num Status Msgs Sent 9444
Num Full Status Sent 1578          Num St Enq. Timeouts 41
Num Link Timeouts 7

LMI Statistics for interface POS0/1/0/1/ (Frame Relay DCE) LMI TYPE = CISCO
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 0
Invalid Information ID 0           Invalid Report IE Len 0
```

■ フレームリレーの設定例

```

Invalid Report Request 0          Invalid Keep IE Len 0
Num Status Enq. Rcvd 9481        Num Status Msgs Sent 9481
Num Full Status Sent 1588        Num St Enq. Timeouts 16
Num Link Timeouts 4

```

次の例は、メイン シリアル インターフェイス上の PVC でシリアル サブインターフェイスを作成する方法です。

```

RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0:0.10 point-to-point
RP/0/0/CPU0:router (config-subif)# ipv4 address 10.46.8.6/24
RP/0/0/CPU0:router (config-subif)# pvc 20
RP/0/0/CPU0:router (config-fr-vc)# encapsulation ietf
RP/0/0/CPU0:router(config-subif)# commit

```

次の例は、システムに設定されているすべての PVC に関する情報を表示する方法です。

```
RP/0/0/CPU0router# show frame-relay pvc
```

```
PVC Statistics for interface Serial0/3/2/0 (Frame Relay DCE)
```

	Active	Inactive	Deleted	Static
Local	4	0	0	0
Switched	0	0	0	0
Dynamic	0	0	0	0

```

DLCI = 612, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTIVE, INTERFACE = Serial0/3/2/0.1

```

```

input pkts 0          output pkts 0          in bytes 0
out bytes 0          dropped pkts 0          in FECN packets 0
in BECN pkts 0        out FECN pkts 0          out BECN pkts 0
in DE pkts 0          out DE pkts 0
out bcast pkts 0      out bcast bytes 0
pvc create time 00:00:00      last time pvc status changed 00:00:00

```

```

DLCI = 613, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTIVE, INTERFACE = Serial0/3/2/0.2

```

```

input pkts 0          output pkts 0          in bytes 0
out bytes 0          dropped pkts 0          in FECN packets 0
in BECN pkts 0        out FECN pkts 0          out BECN pkts 0
in DE pkts 0          out DE pkts 0
out bcast pkts 0      out bcast bytes 0
pvc create time 00:00:00      last time pvc status changed 00:00:00

```

```

DLCI = 614, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTIVE, INTERFACE = Serial0/3/2/0.3

```

```

input pkts 0          output pkts 0          in bytes 0
out bytes 0          dropped pkts 0          in FECN packets 0
in BECN pkts 0        out FECN pkts 0          out BECN pkts 0
in DE pkts 0          out DE pkts 0
out bcast pkts 0      out bcast bytes 0
pvc create time 00:00:00      last time pvc status changed 00:00:00

```

```

DLCI = 615, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTIVE, INTERFACE = Serial0/3/2/0.4

```

```

input pkts 0          output pkts 0          in bytes 0
out bytes 0          dropped pkts 0          in FECN packets 0
in BECN pkts 0        out FECN pkts 0          out BECN pkts 0
in DE pkts 0          out DE pkts 0
out bcast pkts 0      out bcast bytes 0
pvc create time 00:00:00      last time pvc status changed 00:00:00

```

次の例は、DTE 用に設定した PVC で LMI ポーリング オプションを変更し、**show frame-relay lmi** コマンドを使用して、モニタリングしてインターフェイスの問題を解決するための情報を表示する方法です。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0/0
RP/0/0/CPU0:router(config-if)# frame-relay lmi-n391dte 10
RP/0/0/CPU0:router(config-if)# frame-relay lmi-n391dte 5
RP/0/0/CPU0:router(config-if)# frame-relay lmi-t391dte 15
RP/0/0/CPU0:router(config-subif)# commit

RP/0/0/CPU0:router# show frame-relay lmi

LMI Statistics for interface serial 0/3/0/0/0 (Frame Relay DTE) LMI TYPE = ANSI
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 9
Invalid Information ID 0          Invalid Report IE Len 0
Invalid Report Request 0          Invalid Keep IE Len 0
Num Status Enq. Rcvd 9444         Num Status Msgs Sent 9444
Num Full Status Sent 1578         Num St Enq. Timeouts 41
Num Link Timeouts 7
```

マルチリンク フレームリレー : 例

次の例は、シリアル インターフェイスでマルチリンク フレームリレーを設定する方法です。

```
RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# controller MgmtMultilink 0/3/1/0
RP/0/0/CPU0:router(config-mgmtmultilink)# bundle 100
RP/0/0/CPU0:router(config-mgmtmultilink)# exit

RP/0/0/CPU0:router(config)# controller T3 0/3/1/0
RP/0/0/CPU0:router(config-t3)# mode tl
RP/0/0/CPU0:router(config-t3)# clock source internal
RP/0/0/CPU0:router(config-t3)# exit

RP/0/0/CPU0:router(config)# controller T1 0/3/1/0/0
RP/0/0/CPU0:router(config-t1)# channel-group 0
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-24
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1)# exit

RP/0/0/CPU0:router(config)# interface Multilink 0/3/1/0/100
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# exit

RP/0/0/CPU0:router(config)# interface Multilink 0/3/1/0/100.16 point-to-point
RP/0/0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0
RP/0/0/CPU0:router(config-subif)# pvc 16
RP/0/0/CPU0:router(config-fr-vc)# service-policy output policy-mapA
RP/0/0/CPU0:router(config-fr-vc)# exit
RP/0/0/CPU0:router(config-subif)# exit

RP/0/0/CPU0:router(config)# interface Serial 0/3/1/0/0:0
RP/0/0/CPU0:router(config-if)# encapsulation mfr
RP/0/0/CPU0:router(config-if)# multilink group 100
RP/0/0/CPU0:router(config-if)# frame-relay multilink lid sj1
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config)#
```

エンドツーエンド フラグメンテーション：例

次の例は、チャネライズド フレームリレー シリアル インターフェイスで FRF.12 エンドツーエンド フラグメンテーションを設定する方法です。

```
RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# controller T30/3/1/0
RP/0/0/CPU0:router(config-t3)# mode t1
RP/0/0/CPU0:router(config-t3)# clock source internal
RP/0/0/CPU0:router(config-t3)# exit
RP/0/0/CPU0:router(config-t3)# controller T10/3/1/0/0
RP/0/0/CPU0:router(config-t1)# channel-group 0
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-24
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1-channel_group)# interface Serial 0/3/1/0/0:0
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config-if)# interface Serial 0/3/1/0/0:0.100 point-to-point
RP/0/0/CPU0:router(config-subif)# ipv4 address 3.1.1.1 255.255.255.0
RP/0/0/CPU0:router(config-subif)# pvc 100
RP/0/0/CPU0:router(config-fr-vc)# service-policy output LFI
RP/0/0/CPU0:router(config-fr-vc)# fragment end-to-end 256
```

その他の参考資料

フレームリレーに関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システムブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
FRF.12	フレームリレー フォーラム .12
FRF.16	フレームリレー フォーラム .16

規格	タイトル
ANSI T1.617 Annex D	American National Standards Institute T1.617 Annex D
ITU Q.933 Annex A	International Telecommunication Union Q.933 Annex A

MIB

MIB	MIB リンク
FRF.16 MIB	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC 1294	Multiprotocol Interconnect Over Frame Relay
RFC 1315	Management Information Base for Frame Relay DTEs
RFC 1490	Multiprotocol Interconnect Over Frame Relay
RFC 1586	Guidelines for Running OSPF Over Frame Relay Networks
RFC 1604	Definitions of Managed Objects for Frame Relay Service
RFC 2115	Management Information Base for Frame Relay DTEs Using SMIv2
RFC 2390	Inverse Address Resolution Protocol
RFC 2427	Multiprotocol Interconnect Over Frame Relay
RFC 2954	Definitions of Managed Objects for Frame Relay Service
RFC 3020	RFC for FRF.16 MIB

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのリンクバンドルの設定

バンドルは、1つ以上のポートグループを集約し、1つのリンクとして扱うようにしたものです。1つのバンドル内の各リンクの速度は異なってもよく、最も高速なリンクの速度は、最も低速なリンクの最大4倍とすることができます。各バンドルには、1つのMAC、1つのIPアドレス、1つの設定セット（サービス品質（QoS）など）があります。

CRS-1 シリーズ ルータおよび Cisco XR 12000 シリーズ ルータでは、イーサネット インターフェイスでリンクバンドルがサポートされています。CRS-1 シリーズ ルータでは、Packet-over-SONET/SDH (POS) インターフェイスでもバンドルがサポートされています。



(注) バンドルには、モジュラ サービス カードとの1対1の関連付けはありません。

Cisco IOS XR ソフトウェアでのリンクバンドル設定機能の履歴

リリース	変更点
リリース 3.2	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.3.0	この機能は次のように更新されました。 <ul style="list-style-type: none">1:N 冗長性機能をサポートするため、bundle minimum-active links コマンドを使用して最低アクティブリンク数を設定できるようになりました。1:N 冗長性機能をサポートするため、bundle minimum-active links コマンドを使用して最低帯域幅 (kbps 単位) を設定できるようになりました。イーサネットリンクバンドルでVLANサブインターフェイスの追加がサポートされました。show bundle bundle-Ether コマンドと show bundle bundle-POS コマンドの出力が変更されました。reasons キーワードが show bundle bundle-Ether コマンドと show bundle bundle-POS コマンドに追加されました。bundle id コマンドが bundle-id から変更されました。スタティックルートをを使用した、バンドルされたVLAN上のBFD。
リリース 3.4.0	このモジュールの設定手順が拡張されました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	この機能が Cisco XR 12000 シリーズ ルータで初めてサポートされました。

リリース 3.7.0	リンク バンドルがマルチシェルフ Cisco CRS-1 ルータでサポートされることを示す注記が追加されました。
リリース 3.8.0	この機能は次のように更新されました。 <ul style="list-style-type: none"> • reasons キーワードが show bundle bundle-Ether コマンドおよび show bundle bundle-POS コマンドから削除されました。ポートが分散状態以外の状態になっている場合、どちらのコマンドでも理由が表示されるようになりました。 • hot-standby キーワードが bundle maximum-active links コマンドに追加されました。 • lACP fast-switchover コマンドが追加されました。

この章の構成

この章で説明する内容は、次のとおりです。

- 「リンク バンドルを設定するための前提条件」 (P.200)
- 「リンク バンドルを設定する際の制約事項」 (P.202)
- 「リンク バンドルの設定に関する情報」 (P.202)
- 「リンク バンドルの設定方法」 (P.208)
- 「リンク バンドルの設定例」 (P.225)
- 「その他の参考資料」 (P.226)

リンク バンドルを設定するための前提条件

リンク バンドルの前提条件は、この機能を設定しようとしているプラットフォームに依存します。ここで説明する内容は、次のとおりです。

- 「Cisco CRS-1 ルータでリンク バンドルを設定するための前提条件」 (P.200)
- 「Cisco XR 12000 シリーズ ルータでリンク バンドルを設定するための前提条件」 (P.201)

Cisco CRS-1 ルータでリンク バンドルを設定するための前提条件

Cisco IOS XR ルータでリンク バンドルを設定する前に、次の作業が終了し条件が満たされていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。インターフェイスの IP アドレスを把握しておいてください。

- 設定するバンドルに含めるリンクがわかっていること。

- イーサネットリンクバンドルを設定する場合、ルータに少なくとも次のイーサネットカードのいずれかが搭載されていること。
 - 1ポート10ギガビットイーサネットSPA (LANおよびWANPHY)
 - 4ポート10ギガビットイーサネットチューニング可能WDMPHY物理レイヤインターフェイスモジュール (PLIM)
 - 5ポートギガビットイーサネットSPA
 - 8ポートギガビットイーサネットSPA (バージョン1および2)
 - 8ポート10ギガビットイーサネット物理レイヤインターフェイスモジュール (PLIM)
 - 10ポートギガビットイーサネットSPA
- POSリンクバンドルを設定する場合、POSラインカードまたはSPAが、Cisco IOS XRソフトウェアが動作するルータに搭載されていること。



(注)

物理インターフェイス、PLIM、モジュラサービスカードの詳細については、『Cisco CRS-1 Carrier Routing System 8-Slot Line Card Chassis System Description』を参照してください。

Cisco XR 12000 シリーズ ルータでリンクバンドルを設定するための前提条件

Cisco XR 12000 シリーズ ルータでリンクバンドルを設定する前に、次の作業が終了し条件が満たされていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスクIDを含むタスクグループに関連付けられたユーザグループにユーザを割り当てる必要があります。すべてのコマンドタスクIDは、各コマンドリファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスクグループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザグループおよびタスクIDの詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。
- インターフェイスのIPアドレスがわかっていること。
- 設定するバンドルに含めるリンクがわかっていること。
- 少なくとも次のいずれかのSIPがルータに搭載されていること。
 - Cisco XR 12000 SIP-401
 - Cisco XR 12000 SIP-501
 - Cisco XR 12000 SIP-601
- イーサネットリンクバンドルを設定する場合、ルータに少なくとも次のイーサネットカードのいずれかが搭載されていること。
 - 1ポート10ギガビットイーサネットSPA
 - 10ポートギガビットイーサネットSPA
 - 8ポートギガビットイーサネットSPA
 - 5ポートギガビットイーサネットSPA

リンクバンドルを設定する際の制約事項



(注)

システムのアップグレードまたはリロード後、現在のリンクバンドル設計により、QoS など、その下で設定されている機能にエラーがある場合、バンドルされたインターフェイスがアップ状態でなくなります。この動作は、バンドルされたインターフェイスだけに適用されます。物理インターフェイスには影響ありません。

Cisco XR 12000 シリーズ ルータでは、リンクバンドルで次の機能がサポートされません。

- 双方向フォワーディング検出 (BFD)
- QoS
- Access Control List (ACL; アクセス制御リスト)
- QinQ カプセル化
- MPLS Traffic Engineering (TE; トラフィック エンジニアリング)
- マルチキャスト VPN
- POS インターフェイス
- E3 インターフェイス
- 1:1 保護回線
- IPv6



(注)

Cisco XR 12000 シリーズ ルータは E3 インターフェイスをリンクバンドルのメンバーとしてサポートしていませんが、E3 インターフェイスを通じて受信したトラフィックは、リンクバンドルインターフェイスから出力できます。

リンクバンドルの設定に関する情報

リンクバンドルを実装するには、次の概念について理解する必要があります。

- 「リンクバンドルの概要」 (P.202)
- 「リンクバンドルの特性」 (P.203)
- 「LACP を通じたリンク集約」 (P.205)
- 「Cisco CRS-1 ルータでの QoS とリンクバンドル」 (P.206)
- 「リンクバンドルの設定の概要」 (P.207)
- 「RP スイッチオーバー時のノンストップフォワーディング」 (P.207)
- 「リンクスイッチオーバー」 (P.207)

リンクバンドルの概要

リンクバンドルは、1 つに束ねられたポートのグループであり、1 つのリンクとして振る舞います。リンクバンドルには次の利点があります。

- 複数のリンクが複数のラインカードと SPA にまたがり、1つのインターフェイスを形成します。そのため、1つのリンクが障害になっても接続が失われません。
- バンドル インターフェイスでは、トラフィックがバンドルのすべてのメンバーを通じて転送されるため、使用可能な帯域幅が増えます。そのため、バンドル内のいずれかのリンクが障害になっても、別のリンク上でトラフィックが転送されます。パケットフローを中断することなく帯域幅を追加または削除できます。たとえば、トラフィックを中断せずに、OC-48c PLIM モジュラ サービスカードを OC-192 PLIM モジュラ サービスカードにアップグレードできます。

1つのバンドル内のすべてのリンクの種類は同じである必要があります。たとえば、1つのバンドルに含まれるインターフェイスは、すべてイーサネット インターフェイスであるか、すべて POS インターフェイスになります。イーサネット インターフェイスと POS インターフェイスを同時に含めることはできません。

Cisco IOS XR ソフトウェアでは、次の方法でイーサネット インターフェイスおよび POS インターフェイスのバンドルを形成できます。

- IEEE 802.3ad : バンドル内のすべてのメンバー リンクの互換性を確保するため、Link Aggregation Control Protocol (LACP) を採用した標準テクノロジー。互換性がないリンクや障害になったリンクは、バンドルから自動的に削除されます。
- EtherChannel または POS チャネル : ユーザがリンクを設定してバンドルに参加させることができるシスコの専用テクノロジー。バンドル内のリンクに互換性があるかどうかを確認するための仕組みはありません。EtherChannel はイーサネット インターフェイスに適用され、POS チャネルは POS インターフェイスに適用されます。



(注)

リンクバンドルはすべてのマルチシェルフ Cisco CRS-1 ルータでサポートされています。

リンクバンドルの特性

ここでは、Cisco CRS-1 ルータと Cisco XR 12000 シリーズ ルータに共通するリンクバンドルの特性について説明します。各プラットフォーム固有の特性については、「[CRS-1 シリーズルータのリンクバンドルの特性](#)」(P.204) および「[Cisco XR 12000 シリーズルータのリンクバンドルの特性](#)」(P.205) を参照してください。

リンクバンドルの特性と制限を次に示します。

- 1つのバンドルには、LACP がイネーブルまたはディセーブルにされた、複数のリンクが含まれます。1つのバンドルに、LACP がイネーブルになっているリンクと LACP がディセーブルになっているリンクが含まれる場合、LACP がディセーブルになっているリンクはバンドルに集約されません。
- バンドルの各メンバーは、1台のルータに搭載されている複数のモジュラ サービスカードや、同じサービスカードの複数の SPA にまたがることができます。
- イーサネット リンクバンドルでは、バンドルに追加されるすべてのポートとインターフェイスの速度と帯域幅が同じである必要があります。
- 物理層とリンク層の設定は、バンドルの個々のメンバーリンクに対して実行します。
- ネットワーク層プロトコルおよび上位レイヤのアプリケーションの設定は、バンドル自体に対して実行します。
- バンドルは、管理上イネーブルまたはディセーブルにできます。
- バンドル内のそれぞれのリンクは、管理上イネーブルまたはディセーブルにできます。
- バンドルに対して設定された MAC アドレスは、そのバンドル内の各リンクの MAC アドレスになります。

- バンドルに MAC アドレスが設定されていない場合、バンドルおよびバンドルメンバーは、最初のメンバーのアドレスを継承します。
- バンドル内の各リンクでは、異なるメンバーに対して異なるキープアライブ周期を設定できます。
- ロードバランシング（メンバーリンク間のデータの分散）は、パケットではなくフロー単位で実行されます。
- ルーティングアップデートや hello などの上位レイヤプロトコルは、インターフェイスバンドルのどのメンバーリンク上でも送信されます。
- 1つのバンドル内のすべてのリンクは、同じ2台のシステム上で終端する必要があります。どちらのシステムも直接接続されている必要があります。
- バンドルされたインターフェイスはポイントツーポイントです。
- バンドルには物理リンクだけを含むことができます。トンネルおよび VLAN サブインターフェイスはバンドルのメンバーにできません。ただし、VLAN をバンドルのサブインターフェイスとして作成することはできます。
- リンクバンドル上の IPv4 アドレスの設定は、通常のインターフェイス上の IPv4 アドレスの設定と同じです。
- マルチキャストトラフィックは、バンドルのメンバー上でロードバランスされます。特定のフローに対し、内部処理によってメンバーリンクが選択され、そのフローのすべてのトラフィックがそのメンバー上で送信されます。

CRS-1 シリーズルータのリンクバンドルの特性

CRS-1 シリーズに固有のリンクバンドルの特性と制限事項を次に示します。

- リンクバンドルはすべてのマルチシェルフ Cisco CRS-1 ルータでサポートされています。
- 1つのバンドルに含まれるインターフェイスは、すべてイーサネットインターフェイスであるか、すべて POS インターフェイスです。イーサネットインターフェイスと POS インターフェイスは混在できません。
- 1つのバンドルは最大 32 個の物理リンクをサポートします。32 個を超えるリンクをバンドルに追加すると、32 個のリンクだけが機能し、残りのリンクは自動的にディセーブルになります。
- CRS-1 シリーズルータは、最大 32 個のバンドルをサポートします。
- イーサネットおよび POS リンクバンドルは、イーサネットチャネルおよび POS チャネルと同様の方法で作成され、両方のエンドシステムで同じコンフィギュレーションを入力します。
- POS リンクバンドルの場合、1つのバンドル内でリンク速度が異なっていてもよく、バンドルのメンバー間で許容される速度の差は、最大 4 倍です。
- HDLC は、Cisco IOS XR ソフトウェアで POS リンクバンドルに対してサポートされている唯一のカプセル化タイプです。他のカプセル化タイプが設定された POS リンクはバンドルに参加できません。バンドルで HDLC を動作させるためには、すべての POS リンクバンドルメンバーで HDLC が動作している必要があることに注意してください。
- QoS がサポートされており、各バンドルメンバーに均等に適用されます。
- CDP キープアライブや HDLC キープアライブなどのリンク層プロトコルは、バンドル内の各リンク上で独立して動作します。
- 1つのバンドル内のすべてのリンクは、POS チャネルまたは 802.3ad のいずれかを実行するように設定されている必要があります。これらが混在するバンドルはサポートされていません。

Cisco XR 12000 シリーズ ルータのリンクバンドルの特性

Cisco XR 12000 シリーズ ルータに固有のリンクバンドルの特性と制限事項を次に示します。

- 1つのバンドルは最大 8 個の物理リンクをサポートします。8 個を超えるリンクをバンドルに追加すると、8 個のリンクだけが機能し、残りのリンクは自動的にディセーブルになります。
- Cisco XR 12000 シリーズ ルータ は、最大 16 個のバンドルをサポートします。
- イーサネット リンクバンドルは、イーサネット チャネルと同様の方法で作成され、両方のエンドシステムで同じコンフィギュレーションを入力します。

LACP を通じたリンク集約

異なるモジュラ サービス カードおよび同じサービス カード内の SPA 上のインターフェイスを集約することで、冗長性が提供され、インターフェイスまたはモジュラ サービス カードで障害が発生したときに、トラフィックをすばやく他のメンバー リンクにリダイレクトできます。

オプションの Link Aggregation Control Protocol (LACP) は IEEE 802 規格で定義されています。LACP では、2 台の直接接続されたシステム (ピア) 間で通信し、バンドル メンバーの互換性が確認されます。ピアは、別のルータまたはスイッチのいずれかです。LACP は、リンクバンドルの動作状態を監視し、次のことを確認します。

- すべてのリンクが同じ 2 台のシステム上で終端していること。
- 両方のシステムがリンクを同じバンドルの一部と見なしていること。
- すべてのリンクがピア上で適切に設定されていること。

LACP は、ローカル ポート状態と、パートナー システムの状態のローカルなビューが格納されたフレームを送信します。これらのフレームが解析され、両方のシステムが同調していることが確認されます。

IEEE 802.3ad 規格

IEEE 802.3ad 規格では、一般にイーサネット リンクバンドルを構成する方法が定義されています。Cisco IOS XR ソフトウェアでは、イーサネット リンクバンドルと POS リンクバンドルの両方で IEEE 802.3ad 規格が使用されています。

バンドル メンバーとして設定された各リンクに対し、リンクバンドルの各エンドをホストするシステム間で、次の情報が交換されます。

- グローバルに一意のローカル システム ID
- リンクがメンバーになっているバンドルの ID (動作キー)
- リンクの ID (ポート ID)
- リンクの現在の集約ステータス

この情報は、Link Aggregation Group Identifier (LAG ID; リンク集約グループ ID) を構成するために使用されます。共通の LAG ID を共有するリンクは集約できます。個々のリンクには固有の LAG ID があります。

システム ID はルータを区別し、その一意性はシステムの MAC アドレスを使用することで保証されません。バンドル ID とリンク ID は、それを割り当てるルータでだけ意味を持ち、2 つのリンクが同じ ID を持たないことと、2 つのバンドルが同じ ID を持たないことが保証される必要があります。

ピアシステムからの情報はローカル システムの情報と組み合わせられ、バンドルのメンバーとして設定されたリンクの互換性が判断されます。

バンドルに追加されている最初のリンクの MAC アドレスがバンドル自体の MAC アドレスになります。そのリンク（バンドルに追加されている最初のリンク）がバンドルから削除されるか、ユーザが別の MAC アドレスを設定するまで、この MAC アドレスが使用されます。バンドルの MAC アドレスは、バンドルトラフィックを通過させる際にすべてのメンバーリンクによって使用されます。バンドルに対して設定されたすべてのユニキャストアドレスまたはマルチキャストアドレスも、すべてのメンバーリンクで設定されます。



(注)

MAC アドレスを変更するとパケットのフォワーディングに影響を与えるおそれがあるため、MAC アドレスは変更しないことをお勧めします。

Cisco CRS-1 ルータでの QoS とリンクバンドル

入力方向では、バンドルのローカルインスタンスに QoS が適用されます。各バンドルはキューのセットに関連付けられます。QoS は、バンドル上で設定されているさまざまなネットワーク層プロトコルに適用されます。

出方向では、メンバーリンクへの参照を持つバンドルに QoS が適用されます。QoS は、メンバーの帯域幅の合計に基づいて適用されます。

リンクバンドルは次の QoS 機能をサポートしています。

- 高優先順位/低優先順位：最大帯域幅は、バンドルインターフェイスの帯域幅のパーセンテージとして計算されます。このパーセンテージは出力上のすべてのメンバーリンクに適用されるか、入力上のローカルバンドルインスタンスに適用されます。
- 保証される帯域幅：パーセンテージで提供され、すべてのメンバーリンクに適用されます。
- トラフィックシェーピング：パーセンテージで提供され、すべてのメンバーリンクに適用されます。
- WRED：最小および最大パラメータは、メンバーリンクまたはバンドルインスタンスごとの正しい比率に変換され、バンドルに適用されます。
- マーキング：ポリシーに従ったパケットの QoS レベルの変更プロセス。
- テールドロップ：キューが一杯のときにパケットはドロップされます。

イーサネットリンクバンドル上の VLAN

802.1Q VLAN サブインターフェイスを 802.3ad イーサネットリンクバンドル上で設定できます。イーサネットリンクバンドル上に VLAN を追加するときには、次の点に注意してください。

- バンドルあたりの VLAN の最大数は、CRS-1 シリーズルータでは 128、Cisco XR 12000 シリーズルータでは 100 です。
- ルータあたりにバンドルできる VLAN の最大数は、CRS-1 シリーズルータでは 4000、Cisco XR 12000 シリーズルータでは 1600 です。



(注)

バンドル VLAN のメモリ要件は、標準の物理インターフェイスよりも若干多くなります。

バンドル上で VLAN サブインターフェイスを作成するには、次のように、**interface Bundle-Ether** コマンドを使用して VLAN サブインターフェイスインスタンスを追加します。

```
interface Bundle-Ether interface-bundle-id.subinterface
```


イーサネットリンクバンドル上で VLAN を作成した後、すべての物理 VLAN サブインターフェイスコンフィギュレーションがそのリンクバンドル上でサポートされます。

リンクバンドルの設定の概要

リンクバンドルの設定の一般的な概要を次の手順に示します。リンクをバンドルに追加する前に、リンクから以前のネットワーク層コンフィギュレーションをすべてクリアする必要があることに注意してください。

1. グローバルコンフィギュレーションモードで、リンクバンドルを作成します。イーサネットリンクバンドルを作成するには、**interface Bundle-Ether** コマンドを入力します。POS リンクバンドルを作成するには、**interface Bundle-POS** コマンドを入力します。
2. **ipv4 address** コマンドを使用して、IP アドレスとサブネットマスクを仮想インターフェイスに割り当てます。
3. インターフェイスコンフィギュレーションサブモードで **bundle id** コマンドを使用し、ステップ 1 で作成したバンドルにインターフェイスを追加します。1 つのバンドルに追加できるリンクの最大数は、CRS-1 シリーズルータでは 32 個、Cisco XR 12000 シリーズルータでは 8 個です。
4. CRS-1 シリーズルータでは、バンドルに対してオプションで 1:1 保護回線を実装できます。そのためには、**bundle maximum-active links** コマンドに 1 を設定します。この設定を行うと、バンドルで優先順位が最も高いリンクがアクティブになり、優先順位が 2 番目に高いリンクがスタンバイになります（リンクの優先順位は、**bundle port-priority** コマンドの値で決まります）。アクティブなリンクが障害になると、スタンバイリンクがすぐにアクティブリンクになります。



(注) リンクは、そのリンクのインターフェイスコンフィギュレーションサブモードからバンドルのメンバーに設定できます。

RP スイッチオーバー時のノンストップフォワーディング

Cisco IOS XR ソフトウェアは、アクティブおよびスタンバイ RP カード間でのスイッチオーバー時のノンストップフォワーディングをサポートしています。ノンストップフォワーディングを使用すると、スイッチオーバーが発生したときにリンクバンドルの状態が変化しません。

たとえば、アクティブな RP が障害になった場合、スタンバイ RP が動作可能になります。障害になった RP のコンフィギュレーション、ノードの状態、チェックポイントデータは、スタンバイ RP に複製されます。スタンバイ RP がアクティブ RP になったとき、バンドルされたインターフェイスはすべて存在します。



(注) スタンバイインターフェイスコンフィギュレーションが維持されることを保証するために何かを設定する必要はありません。

リンクスイッチオーバー

デフォルトでは、1 つのバンドル内でアクティブにトラフィックを伝送できる最大リンク数は、Cisco CRS-1 ルータでは 32 個、Cisco XR 12000 シリーズルータでは 8 個です。バンドル内の 1 つのメンバーリンクが障害になると、トラフィックは動作可能な残りのメンバーリンクにリダイレクトされます。

Cisco CRS-1 ルータでは、バンドルに対してオプションで 1:1 保護回線を実装できます。そのためには、**bundle maximum-active links** コマンドに 1 を設定します。そうすることで、1 つのアクティブリンクと 1 つ以上の専用のスタンバイリンクが指定されます。アクティブリンクが障害になるとスイッチオーバーが発生し、スタンバイリンクがすぐにアクティブになり、中断のないトラフィックが保証されます。

アクティブリンクとスタンバイリンクで LACP が動作している場合、IEEE 規格に基づくスイッチオーバー（デフォルト）か、専用の高速な最適化されたスイッチオーバーを選択できます。アクティブリンクとスタンバイリンクで LACP が動作していない場合、専用の最適化されたスイッチオーバーオプションが使用されます。

使用するスイッチオーバーの種類にかかわらず、**wait-while** タイマーをディセーブルにできます。これにより、スタンバイリンクの状態ネゴシエーションが高速になり、障害になったアクティブリンクからスタンバイリンクへのスイッチオーバーが高速になります。そのためには、**lACP fast-switchover** コマンドを使用します。

リンクバンドルの設定方法

ここでは、次の手順について説明します。

- 「イーサネットリンクバンドルの設定」(P.208)
- 「VLANバンドルの設定」(P.213)
- 「POSリンクバンドルの設定」(P.220)

イーサネットリンクバンドルの設定

ここでは、イーサネットリンクバンドルの設定方法について説明します。



(注) イーサネットリンクバンドルでは MAC アカウンティングはサポートされていません。



(注) イーサネットバンドルをアクティブにするためには、バンドルの両方の接続ポイントで同じ設定を行う必要があります。

制約事項

Cisco XR 12000 シリーズルータでは、現在 1:1 保護回線がサポートされていません。そのため、**bundle maximum-active links** コマンドと **lACP fast-switchover** コマンドは、Cisco XR 12000 シリーズルータではサポートされていません。

手順の概要

イーサネットリンクバンドルを作成するには、次の手順のように、バンドルを作成し、そのバンドルにメンバーインターフェイスを追加します。

1. **configure**
2. **interface Bundle-Ether bundle-id**
3. **ipv4 address ipv4-address mask**

4. **bundle minimum-active bandwidth** *kbps*
5. **bundle minimum-active links** *links*
6. **bundle maximum-active links** *links* [**hot-standby**]
7. **lacp fast-switchover**
8. **exit**
9. **interface** {**GigabitEthernet** | **TenGigE**} *interface-path-id*
10. **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}
11. **bundle port-priority** *priority*
12. **no shutdown**
13. **exit**
14. ステップ 2 で作成したバンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。
15. **end**
または
commit
16. **exit**
17. **exit**
18. 接続のリモート エンドでステップ 1 から 15 を実行します。
19. **show bundle Bundle-Ether** *bundle-id*
20. **show lacp bundle Bundle-Ether** *bundle-id*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface Bundle-Ether <i>bundle-id</i> 例： RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3	新しいイーサネット リンク バンドルを作成し名前を付与します。 この interface Bundle-Ether コマンドを実行すると、インターフェイス コンフィギュレーション サブモードが開始されます。このモードでは、インターフェイス固有のコンフィギュレーション コマンドを入力できます。インターフェイス コンフィギュレーション サブモードを終了して通常のグローバル コンフィギュレーション モードに戻るには、 exit コマンドを使用します。
ステップ 3	ipv4 address <i>ipv4-address mask</i> 例： RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0	ipv4 address コンフィギュレーション コマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。

■ リンクバンドルの設定方法

コマンドまたはアクション	目的
ステップ 4 <code>bundle minimum-active bandwidth kbps</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000	(任意) ユーザがバンドルをアップ状態にする前に必要な最小帯域幅を設定します。
ステップ 5 <code>bundle minimum-active links links</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2	(任意) 特定のバンドルをアップ状態にする前に必要なアクティブリンク数を設定します。
ステップ 6 <code>bundle maximum-active links links [hot-standby]</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	(任意) バンドルで 1:1 保護回線を実装します。これにより、バンドル内で最も優先順位が高いリンクがアクティブになり、2 番目に優先順位が高いリンクがスタンバイになります。また、アクティブおよびスタンバイの LACP 対応のリンクの間でのスイッチオーバーが、専用の最適化に従って実装されることを指定します。 (注) Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線はサポートされていません。そのため、 bundle maximum-active links コマンドは、Cisco XR 12000 シリーズ ルータではサポートされていません。 (注) アクティブおよびスタンバイリンクの優先順位は、 bundle port-priority コマンドの値で決まります。
ステップ 7 <code>lACP fast-switchover</code> 例: RP/0/RP0/CPU0:router(config-if)# lACP fast-switchover	(任意) LACP が動作するメンバーリンクを持つバンドル上で 1:1 保護回線をイネーブルにすると (bundle maximum-active links コマンドの値に 1 を設定します)、LACP 状態マシンの wait-while タイマーをディセーブルにできます。このタイマーをディセーブルにすると、スタンバイモードのバンドルメンバーリンクで、正常状態のネゴシエーションが高速になるため、障害になったアクティブリンクからスタンバイリンクへのスイッチオーバーが高速になります。 (注) Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線はサポートされていません。そのため、 lACP fast-switchover コマンドは Cisco XR 12000 シリーズ ルータでサポートされていません。
ステップ 8 <code>exit</code> 例: RP/0/RP0/CPU0:router(config-if)# exit	イーサネットリンクバンドルのインターフェイスコンフィギュレーションサブモードを終了します。

コマンドまたはアクション	目的
<p>ステップ 9 <code>interface {GigabitEthernet TenGigE}</code> <code>interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0</p>	<p>特定のインターフェイスのインターフェイス コンフィギュレーション モードを開始します。</p> <p>GigabitEthernet キーワードまたは TenGigE キーワードを入力して、インターフェイスの種類を指定します。<code>interface-path-id</code> 引数には、<code>rack/slot/module</code> 形式のノード ID を指定します。</p>
<p>ステップ 10 <code>bundle id bundle-id [mode {active on passive}]</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# bundle-id 3</p>	<p>指定したバンドルにリンクを追加します。</p> <p>バンドル上でアクティブ LACP またはパッシブ LACP をイネーブルにするには、オプションの mode active キーワードまたは mode passive キーワードをコマンド文字列に追加します。</p> <p>LACP をサポートせずにバンドルにリンクを追加するには、オプションの mode on キーワードをコマンド文字列に追加します。</p> <p>(注) mode キーワードを指定しない場合、デフォルトのモードは on になります (LACP はポート上で動作しません)。</p>
<p>ステップ 11 <code>bundle port-priority priority</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# bundle port-priority 1</p>	<p>(任意) bundle maximum-active links コマンドに 1 を設定する場合、アクティブリンクの優先順位を最も高くし (最も小さい値)、スタンバイリンクの優先順位を 2 番目に高く (次に小さい値) する必要があります。たとえば、アクティブリンクの優先順位を 1 に設定し、スタンバイリンクの優先順位を 2 に設定します。</p>
<p>ステップ 12 <code>no shutdown</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# no shutdown</p>	<p>(任意) リンクがダウン状態の場合はアップ状態にします。no shutdown コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。</p>
<p>ステップ 13 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# exit</p>	<p>イーサネット インターフェイスのインターフェイス コンフィギュレーション サブモードを終了します。</p>

■ リンクバンドルの設定方法

コマンドまたはアクション	目的
<p>ステップ 14</p> <pre>interface {GigabitEthernet TenGigE} number bundle id bundle-id [mode {active passive on}] no shutdown exit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 1/0/2/1</pre> <pre>RP/0/RP0/CPU0:router(config-if)# bundle id 3</pre> <pre>RP/0/RP0/CPU0:router(config-if)# bundle port-priority 2</pre> <pre>RP/0/RP0/CPU0:router(config-if)# no shutdown</pre> <pre>RP/0/RP0/CPU0:router(config-if)# exit</pre> <pre>RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 1/0/2/3</pre> <pre>RP/0/RP0/CPU0:router(config-if)# bundle id 3</pre> <pre>RP/0/RP0/CPU0:router(config-if)# no shutdown</pre> <pre>RP/0/RP0/CPU0:router(config-if)# exit</pre>	<p>(任意) バンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。</p>
<p>ステップ 15</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 16</p> <pre>exit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# exit</pre>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 17	<code>exit</code> 例： RP/0/RP0/CPU0:router(config)# exit	グローバル コンフィギュレーション モードを終了します。
ステップ 18	接続のリモート エンドでステップ 1 から 15 を実行します。	リンク バンドルの他端をアップ状態にします。
ステップ 19	<code>show bundle Bundle-Ether bundle-id</code> 例： RP/0/RP0/CPU0:router# show bundle Bundle-Ether 3	(任意) 指定したイーサネット リンク バンドルに関する情報を表示します。
ステップ 20	<code>show lacp bundle Bundle-Ether bundle-id</code> 例： RP/0/RP0/CPU0:router# show lacp bundle Bundle-Ether 3	(任意) LACP ポートとそのピアに関する詳細情報を表示します。

VLAN バンドルの設定

ここでは、VLAN バンドルの設定方法について説明します。VLAN バンドルの作成では、主に次の 3 つの作業を行います。

1. イーサネット バンドルを作成します。
2. VLAN サブインターフェイスを作成し、イーサネット バンドルに割り当てます。
3. イーサネット リンクをイーサネット バンドルに割り当てます。

これらの作業について、以降の手順で詳しく説明します。



(注) VLAN バンドルをアクティブにするには、バンドル接続の両端で同じ設定を行う必要があります。

制約事項

Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線がサポートされていません。そのため、**bundle maximum-active links** コマンドと **lacp fast-switchover** コマンドは、Cisco XR 12000 シリーズ ルータではサポートされていません。

手順の概要

VLAN リンク バンドルの作成について、次の手順で説明します。

1. `configure`
2. `interface Bundle-Ether bundle-id`
3. `ipv4 address ipv4-address mask`
4. `bundle minimum-active bandwidth kbps`
5. `bundle minimum-active links links`
6. `bundle maximum-active links links [hot-standby]`
7. `lacp fast-switchover`
8. `exit`

9. **interface Bundle-Ether** *bundle-id.vlan-id*
10. **dot1q vlan** *vlan-id*
11. **ipv4 address** *ipv4-address mask*
12. **no shutdown**
13. **exit**
14. ステップ 2 で作成したバンドルにさらに VLAN を追加するには、ステップ 9 から 12 を繰り返します
15. **end**
または
commit
16. **exit**
17. **exit**
18. **configure**
19. **interface** {GigabitEthernet | TenGigE} *interface-path-id*
20. **bundle id** *bundle-id* [mode {active | on | passive}]
21. **bundle port-priority** *priority*
22. **no shutdown**
23. ステップ 2 で作成したバンドルにさらにイーサネット インターフェイスを追加するには、ステップ 19 から 21 を繰り返します
24. **end**
または
commit
25. 接続のリモート エンドでステップ 1 から 23 を実行します。
26. **show bundle Bundle-Ether** *bundle-id*
27. **show vlan interface**
28. **show vlan trunks** [{GigabitEthernet | TenGigE | Bundle-Ether} *interface-path-id*] [/brief | summary] [location *node-id*]

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface Bundle-Ether bundle-id 例： RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3	新しいイーサネット リンクバンドルを作成し名前を付与します。 この interface Bundle-Ether コマンドを実行すると、インターフェイス コンフィギュレーション サブモードが開始されます。このモードでは、インターフェイス固有のコンフィギュレーション コマンドを入力できます。インターフェイス コンフィギュレーション サブモードを終了して通常のグローバル コンフィギュレーション モードに戻るには、 exit コマンドを使用します。
ステップ 3	ipv4 address ipv4-address mask 例： RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0	ipv4 address コンフィギュレーション コマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。
ステップ 4	bundle minimum-active bandwidth kbps 例： RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000	(任意) ユーザがバンドルをアップ状態にする前に必要な最小帯域幅を設定します。
ステップ 5	bundle minimum-active links links 例： RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2	(任意) 特定のバンドルをアップ状態にする前に必要なアクティブ リンク数を設定します。
ステップ 6	bundle maximum-active links links [hot-standby] 例： RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	(任意) バンドルで 1:1 保護回線を実装します。これにより、バンドル内で最も優先順位が高いリンクがアクティブになり、2 番目に優先順位が高いリンクがスタンバイになります。また、アクティブおよびスタンバイの LACP 対応のリンクの間でのスイッチオーバーが、専用の最適化に従って実装されることを指定します。 (注) Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線はサポートされていません。そのため、 bundle maximum-active links コマンドは、Cisco XR 12000 シリーズ ルータではサポートされていません。 (注) アクティブおよびスタンバイ リンクの優先順位は、 bundle port-priority コマンドの値で決まります。

■ リンクバンドルの設定方法

	コマンドまたはアクション	目的
ステップ 7	<pre>lacp fast-switchover</pre> <p>例: RP/0/RP0/CPU0:router(config-if)# lacp fast-switchover</p>	<p>(任意) LACP が動作するメンバー リンクを持つバンドル上で 1:1 保護回線をイネーブルにすると (bundle maximum-active links コマンドの値に 1 を設定します)、LACP 状態マシンの wait-while タイマーをディセーブルにできます。このタイマーをディセーブルにすると、スタンバイモードのバンドルメンバーリンクで、正常状態のネゴシエーションが高速になるため、障害になったアクティブリンクからスタンバイリンクへのスイッチオーバーが高速になります。</p> <p>(注) Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線はサポートされていません。そのため、lacp fast-switchover コマンドは Cisco XR 12000 シリーズ ルータでサポートされていません。</p>
ステップ 8	<pre>exit</pre> <p>例: RP/0/RP0/CPU0:router(config-if)# exit</p>	<p>インターフェイス コンフィギュレーション サブモードを終了します。</p>
ステップ 9	<pre>interface Bundle-Ether bundle-id.vlan-id</pre> <p>例: RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3.1</p>	<p>新しい VLAN を作成し、その VLAN をステップ 2 で作成したイーサネットバンドルに割り当てます。</p> <p>bundle-id 引数には、ステップ 2 で作成した bundle-id を指定します。</p> <p>vlan-id にはサブインターフェイス ID を指定します。範囲は 1 ~ 4094 です (0 と 4095 は予約されています)。</p> <p>(注) .vlan-id 引数を interface Bundle-Ether bundle-id コマンドに追加すると、サブインターフェイス コンフィギュレーションモードが開始されます。</p>
ステップ 10	<pre>dot1q vlan vlan-id</pre> <p>例: RP/0/RP0/CPU0:router#(config-subif)# dot1q vlan 10</p>	<p>VLAN をサブインターフェイスに割り当てます。</p> <p>vlan-id 引数にはサブインターフェイス ID を指定します。範囲は 1 ~ 4094 です (0 と 4095 は予約されています)。</p>
ステップ 11	<pre>ipv4 address ipv4-address mask</pre> <p>例: RP/0/RP0/CPU0:router#(config-subif)# ipv4 address 10.1.2.3/24</p>	<p>サブインターフェイスに IP アドレスおよびサブネットマスクを割り当てます。</p>
ステップ 12	<pre>no shutdown</pre> <p>例: RP/0/RP0/CPU0:router#(config-subif)# no shutdown</p>	<p>(任意) リンクがダウン状態の場合はアップ状態にします。no shutdown コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。</p>
ステップ 13	<pre>exit</pre> <p>例: RP/0/RP0/CPU0:router(config-subif)# exit</p>	<p>VLAN サブインターフェイスのサブインターフェイス コンフィギュレーションモードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 14	<p>ステップ 2 で作成したバンドルにさらに VLAN を追加するには、ステップ 9 から 12 を繰り返します。</p> <pre>interface Bundle-Ether bundle-id.vlan-id dot1q vlan vlan-id ipv4 address ipv4-address mask no shutdown exit</pre> <p>例 :</p> <pre>RP/0/RP0/CPU0:router(config-subif)# interface Bundle-Ether 3.1 RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 20 RP/0/RP0/CPU0:router(config-subif)# ipv4 address 20.2.3.4/24 RP/0/RP0/CPU0:router(config-subif)# no shutdown exit</pre>	<p>(任意) バンドルにさらにサブインターフェイスを追加します。</p>
ステップ 15	<pre>end または commit</pre> <p>例 :</p> <pre>RP/0/RP0/CPU0:router(config-subif)# end または RP/0/RP0/CPU0:router(config-subif)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 16	<pre>exit</pre> <p>例 :</p> <pre>RP/0/RP0/CPU0:router(config-subif)# end</pre>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>
ステップ 17	<pre>exit</pre> <p>例 :</p> <pre>RP/0/RP0/CPU0:router(config)# exit</pre>	<p>グローバル コンフィギュレーション モードを終了します。</p>

■ リンクバンドルの設定方法

	コマンドまたはアクション	目的
ステップ 18	<p><code>configure</code></p> <p>例: RP/0/RP0/CPU0:router # configure</p>	グローバル コンフィギュレーション モードを開始します。
ステップ 19	<p><code>interface {GigabitEthernet TenGigE} interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0</p>	<p>バンドルに追加するイーサネット インターフェイスのインターフェイス コンフィギュレーション モードを開始します。</p> <p>GigabitEthernet キーワードまたは TenGigE キーワードを入力して、インターフェイスの種類を指定します。<i>interface-path-id</i> 引数には、<i>rack/slot/module</i> 形式のノード ID を指定します。</p> <p>(注) リンクバンドルの両端にイーサネット インターフェイスを追加するまでは、VLAN バンドルはアクティブになりません。</p>
ステップ 20	<p><code>bundle id bundle-id [mode {active on passive}]</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# bundle-id 3</p>	<p>ステップ 2 から 13 で設定したバンドルにイーサネット インターフェイスを追加します。</p> <p>バンドル上でアクティブ LACP またはパッシブ LACP をイネーブルにするには、オプションの mode active キーワードまたは mode passive キーワードをコマンド文字列に追加します。</p> <p>LACP をサポートせずにバンドルにインターフェイスを追加するには、オプションの mode on キーワードをコマンド文字列に追加します。</p> <p>(注) mode キーワードを指定しない場合、デフォルトのモードは on になります (LACP はポート上で動作しません)。</p>
ステップ 21	<p><code>bundle port-priority priority</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# bundle port-priority 1</p>	(任意) bundle maximum-active links コマンドに 1 を設定する場合、アクティブリンクの優先順位を最も高くし (最も小さい値)、スタンバイリンクの優先順位を 2 番目に高く (次に小さい値) する必要があります。たとえば、アクティブリンクの優先順位を 1 に設定し、スタンバイリンクの優先順位を 2 に設定します。
ステップ 22	<p><code>no shutdown</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# no shutdown</p>	(任意) リンクがダウン状態の場合はアップ状態にします。 no shutdown コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。
ステップ 23	-	VLAN バンドルにさらにイーサネット インターフェイスを追加するには、ステップ 19 から 21 を繰り返します。

	コマンドまたはアクション	目的
ステップ 24	<pre>end または commit 例： RP/0/RP0/CPU0:router(config-subif)# end または RP/0/RP0/CPU0:router(config-subif)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 25	<p>VLAN バンドル接続のリモート エンドでステップ 1 から 23 を実行します。</p>	<p>リンク バンドルの他端をアップ状態にします。</p>
ステップ 26	<pre>show bundle Bundle-Ether bundle-id 例： RP/0/RP0/CPU0:router# show bundle Bundle-Ether 3</pre>	<p>(任意) 指定したイーサネットリンクバンドルに関する情報を表示します。</p> <p>show bundle Bundle-Ether コマンドを実行すると、指定したバンドルに関する情報が表示されます。バンドルが正しく設定されており、トラフィックを伝送している場合は、show bundle Bundle-Ether コマンドの出力の State フィールドに数値 4 が表示されます。これは、指定された VLAN バンドルポートが「分散している」ことを意味します。</p>
ステップ 27	<pre>show vlan interface 例： RP/0/RP0/CPU0:router # show vlan interface</pre>	<p>現在の VLAN インターフェイスとステータス コンフィギュレーションを表示します。</p>

	コマンドまたはアクション	目的
ステップ 28	<pre>show vlan trunks [{GigabitEthernet TenGigE Bundle-Ether} interface-path-id] [brief summary] [location node-id]</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show vlan trunk summary</pre>	<p>(任意) 各 VLAN トランク インターフェイスに関するサマリー情報を表示します。</p> <ul style="list-style-type: none"> • キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> – brief : 簡潔なサマリーを表示します。 – summary : 完全なサマリーを表示します。 – location : 指定したスロット上の VLAN トランク インターフェイスに関する情報を表示します。 – interface : 指定したインターフェイスまたはサブインターフェイスに関する情報を表示します。 <p>show vlan trunks コマンドを使用し、イーサネットバンドル上で設定されているすべての VLAN サブインターフェイスが「up」になっていることを確認します。</p>
ステップ 29	<pre>lACP fast-switchover</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# lACP fast-switchover</pre>	<p>(任意) LACP が動作するメンバー リンクを持つバンドル上で 1:1 保護回線をイネーブルにすると (bundle maximum-active links コマンドの値に 1 を設定します)、LACP 状態マシンの wait-while タイマーをディセーブルにできます。このタイマーをディセーブルにすると、スタンバイモードのバンドルメンバーリンクで、正常状態のネゴシエーションが高速になるため、障害になったアクティブリンクからスタンバイリンクへのスイッチオーバーが高速になります。</p>

POS リンクバンドルの設定

ここでは、POS リンクバンドルの設定方法について説明します。



(注) POS バンドルをアクティブにするためには、POS バンドルの両方の接続ポイントで同じ設定を行う必要があります。

制約事項

Cisco XR 12000 シリーズ ルータでは、現在 POS インターフェイスと POS トランクバンドルがサポートされていません。

手順の概要

バンドルされた POS インターフェイスの作成では、次のステップに示すように、バンドルとメンバーインターフェイスの両方を設定します。

1. **configure**
2. **interface Bundle-POS bundle-id**

3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*
5. **bundle minimum-active links** *links*
6. **bundle maximum-active links** *links [hot-standby]*
7. **lACP fast-switchover**
8. **exit**
9. **interface POS** *number*
10. **bundle id** *bundle-id [mode {active | on | passive}]*
11. **bundle port-priority** *priority*
12. **no shutdown**
13. **exit**
14. ステップ 2 で作成したバンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。
15. **end**
または
commit
16. **exit**
17. **exit**
18. 接続のリモートエンドでステップ 1 から 15 を実行します。
19. **show bundle Bundle-POS** *bundle-id*
20. **show lACP bundle Bundle-POS** *bundle-id*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface Bundle-POS bundle-id</code> 例: RP/0/RP0/CPU0:router#(config)#interface Bundle-POS 2	名前と新たにバンドルされた POS インターフェイスを設定します。 このインターフェイス コマンドを実行すると、インターフェイス コンフィギュレーション サブモードが開始されます。このモードでは、インターフェイス固有のコンフィギュレーション コマンドを入力します。インターフェイス コンフィギュレーション サブモードを終了して通常のグローバル コンフィギュレーション モードに戻るには、 exit コマンドを使用します。
ステップ 3	<code>ipv4 address ipv4-address mask</code> 例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0	<code>ip address</code> コンフィギュレーション サブコマンドを使用して、IP アドレスとサブネット マスクを仮想インターフェイスに割り当てます。
ステップ 4	<code>bundle minimum-active bandwidth kbps</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 620000	(任意) ユーザがバンドルをアップ状態にする前に必要な最小帯域幅を設定します。
ステップ 5	<code>bundle minimum-active links links</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2	(任意) 特定のバンドルをアップ状態にする前に必要なアクティブ リンク数を設定します。
ステップ 6	<code>bundle maximum-active links links [hot-standby]</code> 例: RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	(任意) バンドルで 1:1 保護回線を実装します。これにより、バンドル内で最も優先順位が高いリンクがアクティブになり、2 番目に優先順位が高いリンクがスタンバイになります。また、アクティブおよびスタンバイの LACP 対応のリンクの間でのスイッチオーバーが、専用の最適化に従って実装されることを指定します。 (注) アクティブおよびスタンバイ リンクの優先順位は、 bundle port-priority コマンドの値で決まります。
ステップ 7	<code>lACP fast-switchover</code> 例: RP/0/RP0/CPU0:router(config-if)# lACP fast-switchover	(任意) LACP が動作するメンバー リンクを持つバンドル上で 1:1 保護回線をイネーブルにすると (bundle maximum-active links コマンドの値に 1 を設定します)、LACP 状態マシンの wait-while タイマーをディセーブルにできます。このタイマーをディセーブルにすると、スタンバイ モードのバンドル メンバー リンクで、正常状態のネゴシエーションが高速になるため、障害になったアクティブ リンクからスタンバイ リンクへのスイッチオーバーが高速になります。

	コマンドまたはアクション	目的
ステップ 8	<code>exit</code>	インターフェイス コンフィギュレーション サブモードを終了します。
ステップ 9	<code>interface POS interface-path-id</code> 例： RP/0/RP0/CPU0:router(config)# interface POS 0/1/0/0	POS インターフェイス コンフィギュレーション モードを開始し、POS インターフェイス名を指定します。 <code>interface-path-id</code> は、 <code>rack/slot/module/port</code> の形式で指定します。
ステップ 10	<code>bundle id bundle-id [mode {active passive on}]</code> 例： RP/0/RP0/CPU0:router(config-if)# bundle-id 3	指定したバンドルにリンクを追加します。 バンドル上でアクティブ LACP またはパッシブ LACP をイネーブルにするには、オプションの mode active キーワードまたは mode passive キーワードをコマンド文字列に追加します。 LACP をサポートせずにバンドルにリンクを追加するには、オプションの mode on キーワードをコマンド文字列に追加します。 (注) mode キーワードを指定しない場合、デフォルトのモードは on になります (LACP はポート上で動作しません)。
ステップ 11	<code>bundle port-priority priority</code> 例： RP/0/RP0/CPU0:router(config-if)# bundle port-priority 1	(任意) bundle maximum-active links コマンドに 1 を設定する場合、アクティブリンクの優先順位を最も高くし (最も小さい値)、スタンバイリンクの優先順位を 2 番目に高く (次に小さい値) する必要があります。たとえば、アクティブリンクの優先順位を 1 に設定し、スタンバイリンクの優先順位を 2 に設定します。
ステップ 12	<code>no shutdown</code> 例： RP/0/RP0/CPU0:router(config-if)# no shutdown	シャットダウン コンフィギュレーションを削除します。これにより、インターフェイスが管理上ダウンになります。 no shutdown コマンドは、コンフィギュレーションとリンクの状態に応じて、リンクをアップ状態またはダウン状態に戻します。
ステップ 13	<code>exit</code> 例： RP/0/RP0/CPU0:router# exit	POS インターフェイスのインターフェイス コンフィギュレーション サブモードを終了します。
ステップ 14	バンドルにさらにリンクを追加するには、ステップ 8 から 11 を繰り返します。	(任意) ステップ 2 で作成したバンドルにさらにリンクを追加します。

■ リンクバンドルの設定方法

コマンドまたはアクション	目的
<p>ステップ 15 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 16 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>exit</code></p>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>
<p>ステップ 17 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config)# <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了します。</p>
<p>ステップ 18 接続のリモートエンドでステップ 1 から 15 を実行します。</p>	<p>リンクバンドルの他端をアップ状態にします。</p>
<p>ステップ 19 <code>show bundle Bundle-POS number</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show bundle Bundle-POS 1</code></p>	<p>(任意) 指定した POS リンクバンドルに関する情報を表示します。</p>
<p>ステップ 20 <code>show lacp bundle Bundle-POS bundle-id</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show lacp bundle Bundle-POS 3</code></p>	<p>(任意) LACP ポートとそのピアに関する詳細情報を表示します。</p>

リンクバンドルの設定例

次に、2つのポートを結合して、LACPが動作するEtherChannelバンドルを構成する例を示します。



(注) Cisco XR 12000 シリーズ ルータでは、現在 1:1 保護回線はサポートされていません。そのため、**bundle maximum-active links** コマンドと **lACP fast-switchover** コマンドは、Cisco XR 12000 シリーズ ルータではサポートされていません。

```
RP/0/RP0/CPU0:Router# config
RP/0/RP0/CPU0:Router(config)# interface Bundle-Ether 3
RP/0/RP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RP0/CPU0:Router(config-if)# bundle maximum-active links 1 hot-standby
RP/0/RP0/CPU0:Router(config-if)# lACP fast-switchover
RP/0/RP0/CPU0:Router(config-if)# exit
RP/0/RP0/CPU0:Router(config)# interface TenGigE 0/3/0/0
RP/0/RP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RP0/CPU0:Router(config-if)# bundle port-priority 1
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config)# exit
RP/0/RP0/CPU0:Router(config)# interface TenGigE 0/3/0/1
RP/0/RP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RP0/CPU0:Router(config-if)# bundle port-priority 2
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config-if)# exit
```

次に、イーサネットバンドル上で2つのVLANを作成しアップ状態にする例を示します。

```
RP/0/RP0/CPU0:Router# config
RP/0/RP0/CPU0:Router(config)# interface Bundle-Ether 1
RP/0/RP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RP0/CPU0:Router(config-if)# exit
RP/0/RP0/CPU0:Router(config)# interface Bundle-Ether 1.1
RP/0/RP0/CPU0:Router(config-subif)# dot1q vlan 10
RP/0/RP0/CPU0:Router(config-subif)# ip addr 10.2.3.4/24
RP/0/RP0/CPU0:Router(config-subif)# no shutdown
RP/0/RP0/CPU0:Router(config-subif)# exit
RP/0/RP0/CPU0:Router(config)# interface Bundle-Ether 1.2
RP/0/RP0/CPU0:Router(config-subif)# dot1q vlan 20
RP/0/RP0/CPU0:Router(config-subif)# ip addr 20.2.3.4/24
RP/0/RP0/CPU0:Router(config-subif)# no shutdown
RP/0/RP0/CPU0:Router(config-subif)# exit
RP/0/RP0/CPU0:Router(config)# interface gig 0/1/5/7
RP/0/RP0/CPU0:Router(config-if)# bundle-id 1 mode act
RP/0/RP0/CPU0:Router(config-if)# commit
RP/0/RP0/CPU0:Router(config-if)# exit
RP/0/RP0/CPU0:Router(config)# exit
RP/0/RP0/CPU0:Router # show vlan trunks
```

次に、2つのポートを結合し、POSリンクバンドルを構成する例を示します。

```
Router# config
Router(config)# interface Bundle-POS 5
Router(config-if)# ipv4 address 1.2.3.4/24
Router(config-if)# bundle minimum-active bandwidth 620000
Router(config-if)# bundle minimum-active bandwidth 620000
Router(config-if)# exit
Router(config)# interface POS 0/0/1/0
```

```

Router(config-if)# bundle id 5
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface POS 0/0/1/1
Router(config-if)# bundle id 5
Router(config-if)# no shutdown
Router(config-if)# exit

```

その他の参考資料

ここでは、リンクバンドルの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』

規格

規格	タイトル
IEEE 802.3ad (802.3-2002 に Annex 43 として併合)	-

MIB

MIB	MIB リンク
リンク集約のための IEEE で定義された MIB (802.3 Annex 30C で定義)	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの仮想のループバック インターフェイスおよびヌル インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータでのループバック インターフェイスおよびヌル インターフェイスの設定について説明します。ループバック インターフェイスとヌル インターフェイスは、仮想インターフェイスと見なされます。

仮想インターフェイスは、ルータ内部の論理パケット スイッチング エンティティです。仮想インターフェイスは、グローバル スコープを持ちますが、関連付けられた位置を持ちません。代替として、仮想インターフェイスは名前のあとにグローバルに一意な数字による ID を持ちます。たとえば、Loopback 0、Loopback1、Loopback 99999 のようになります。この ID は仮想インターフェイスのタイプごとに固有であるため、Loopback 0 と Null 0 の両方を持つことができ、全体として固有な文字列の名前を形成します。

ループバック インターフェイスとヌル インターフェイスのコントロールプレーンは、アクティブ ルート プロセッサ (RP) 上に存在します。設定とコントロールプレーンは、スタンバイ RP 上にミラーリングされ、スイッチオーバーが発生した場合には、仮想インターフェイスがそれまでのスタンバイ RP に移り、このスタンバイ RP が新たにアクティブ RP となります。

Cisco IOS XR ソフトウェアでのループバック インターフェイスおよびヌル インターフェイス設定機能の履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.0	変更ありません。
リリース 3.2	Cisco XR 12000 シリーズ ルータのサポートが追加されました。
リリース 3.3.0	変更ありません。
リリース 3.4.0	このモジュールが更新され、仮想 IPv4 管理インターフェイスの設定に関する情報が含まれました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「仮想インターフェイス設定の前提条件」 (P.230)
- 「仮想インターフェイスの設定に関する情報」 (P.230)
- 「仮想インターフェイスの設定方法」 (P.232)
- 「仮想インターフェイスの設定例」 (P.236)
- 「その他の参考資料」 (P.237)

仮想インターフェイス設定の前提条件

この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

仮想インターフェイスの設定に関する情報

仮想インターフェイスを設定するには、次の概念を理解している必要があります。

- 「仮想ループバック インターフェイスの概要」 (P.230)
- 「ヌル インターフェイスの概要」 (P.231)
- 「仮想管理インターフェイスの概要」 (P.231)
- 「アクティブ/スタンバイ RP および仮想インターフェイスの設定」 (P.231)

仮想ループバック インターフェイスの概要

仮想ループバック インターフェイスは、単一のエンドポイントを持つ常時アップ状態の仮想インターフェイスです。仮想ループバック インターフェイスで転送されるパケットは、ただちに同一インターフェイスによって受信されます。ループバック インターフェイスは物理インターフェイスをエミュレートします。

Cisco IOS XR ソフトウェアでは、仮想ループバック インターフェイスは次の機能を実行します。

- ループバック インターフェイスはルーティング プロトコル セッションの終端アドレスとして機能できます。これにより、アウトバウンド インターフェイスがダウンしても、ルーティング プロトコル セッションをアップ状態に維持することができます。
- ルータ IP スタックが適切に動作していることを確認するには、ループバック インターフェイスに対して ping を実行します。

他のルータまたはアクセス サーバが仮想ループバック インターフェイスにアクセスを試みるようなアプリケーションでは、ルーティング プロトコルを設定して、ループバック アドレスに割り当てられるサブネットを分散させる必要があります。

ループバック インターフェイスにルーティングされたパケットは、ルータまたはアクセス サーバに再ルーティングされ、ローカルで処理されます。ループバック インターフェイスからルーティングされ、ループバック インターフェイスを宛先としない IP パケットは廃棄されます。この 2 つの条件下では、ループバック インターフェイスはヌル インターフェイスと同じように動作できます。

ヌル インターフェイスの概要

ヌル インターフェイスは、ほとんどのオペレーティング システムで使用可能なヌル装置と同様に機能します。このインターフェイスは常時アップ状態であり、トラフィックの転送や受信は行えず、カプセル化は常に失敗します。ヌル インターフェイスは、トラフィックをフィルタリングするための代替的な方法として使用できます。不要なネットワーク トラフィックをヌル インターフェイスに送ることによって、アクセス リストを使用する場合に伴うオーバーヘッドを回避できます。

ヌル インターフェイスに指定できるインターフェイス コンフィギュレーション コマンドは **ipv4 unreachable** コマンドのみです。**ipv4 unreachable** コマンドを使用した場合、ソフトウェアは、認識できないプロトコルが使用されている自分宛の非ブロードキャスト パケットを受信すると、**Internet Control Message Protocol (ICMP; インターネット制御メッセージプロトコル)** プロトコル到達不能メッセージを送信元に送ります。宛先アドレスまでのルートが不明なため最終的な宛先に配信できないデータグラムを受信した場合、ソフトウェアはそのデータグラムの発信者に **ICMP ホスト到達不能** メッセージで応答します。

Null0 インターフェイスは、起動時にデフォルトで RP 上に作成され、削除はできません。このインターフェイスに **ipv4 unreachable** コマンドを設定することは可能ですが、このインターフェイスは送られてきたすべてのパケットを廃棄するだけなので、ほとんどの設定は不要です。

Null0 インターフェイスを表示するには、**show interfaces null0** コマンドを使用します。

仮想管理インターフェイスの概要

IPv4 仮想アドレスを設定することにより、どの RP がアクティブであるかを事前に把握していなくても、管理ネットワークでの単一の仮想アドレスからルータにアクセスすることができます。IPv4 仮想アドレスは、ルート プロセッサ (RP) のスイッチオーバーが発生した状況でも存続します。そのためには、IPv4 仮想アドレスは両方の RP の管理イーサネット インターフェイスと共通の IPv4 サブネットを共有する必要があります。

各 RP が複数の管理イーサネット インターフェイスを持つ Cisco XR 12000 シリーズ ルータでは、IPv4 仮想アドレスは同じ IP サブネットを共有するアクティブ RP の管理イーサネット インターフェイスにマッピングされます。

アクティブ/スタンバイ RP および仮想インターフェイスの設定

スタンバイ RP は、必要時に使用可能になり、アクティブ RP から作業を引き継げる状態になります。スタンバイ RP がアクティブ RP となり、アクティブ RP の役割を引き継ぐ必要がある状況を次に示します。

- ウォッチドッグによる障害検出
- 管理コマンドの引き継ぎ
- シャーシからのアクティブ RP の取り外し

セカンダリ RP がシャーシに搭載されていなかった場合、プライマリ RP の稼動中にセカンダリ RP を搭載すると、自動的にスタンバイ RP になります。シャーシからスタンバイ RP を取り外しても、RP の冗長性が失われるだけで、システムに影響はありません。

スイッチオーバー後、すべての仮想インターフェイスはスタンバイ（新たにアクティブになった）RP に存在します。仮想インターフェイスのステートと設定は変更されず、スイッチオーバー時にインターフェイス経由の転送（トンネルの場合）が失われることはありません。ルータは、ホスト RP のスイッチオーバーを通じて、バンドルおよびトンネルで上で Nonstop Forwarding（NSF; 無停止転送）を使用します。

**(注)**

スタンバイ インターフェイスの設定維持を保証するために、ユーザ側で何かを設定する必要はありません。

仮想インターフェイスの設定方法

ここでは、次の手順について説明します。

- 「[仮想ループバック インターフェイスの設定](#)」(P.232) (必須)
- 「[ヌル インターフェイスの設定](#)」(P.233) (必須)
- 「[IPv4 仮想インターフェイスの設定](#)」(P.235) (必須)

仮想ループバック インターフェイスの設定

ここでは、基本的なループバック インターフェイスの設定手順について説明します。

制約事項

ループバック インターフェイスの IP アドレスは、ネットワーク上のすべてのルータ間で固有である必要があります。この IP アドレスは、ルータ上の他のインターフェイスでは使用できません。また、ネットワーク上のいかなるルータのインターフェイスでも使用できません。

手順の概要

1. **configure**
2. **interface loopback *interface-path-id***
3. **ipv4 address *ip-address***
4. **end**
または
commit
5. **show interfaces *type interface-path-id***

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface loopback interface-path-id</code> 例: RP/0/RP0/CPU0:router#(config)# interface Loopback 3	インターフェイス コンフィギュレーション モードを開始して、新しいループバック インターフェイスの名前を指定します。
ステップ 3	<code>ipv4 address ip-address</code> 例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32	<code>ipv4 address</code> コンフィギュレーション コマンドを使用して、仮想ループバック インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – <code>yes</code> と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – <code>no</code> と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – <code>cancel</code> と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、<code>commit</code> コマンドを使用します。
ステップ 5	<code>show interfaces type interface-path-id</code> 例: RP/0/RP0/CPU0:router# show interfaces Loopback 3	(任意) ループバック インターフェイスの設定を表示します。

ヌル インターフェイスの設定

ここでは、基本的なヌル インターフェイスの設定手順について説明します。

手順の概要

1. **configure**
2. **interface null 0**
3. **ipv4 unreachable**
4. **end**
または
commit
5. **show interfaces type interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface null 0 例： RP/0/RP0/CPU0:router#(config)# interface null 0	null0 インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ipv4 unreachable 例： RP/0/RP0/CPU0:router#(config-null0)# ipv4 unreachable	IPv4 ICMP 到達不能メッセージを生成します。 <ul style="list-style-type: none">• このコマンドには、引数やキーワードはありません。
ステップ 4	end または commit 例： RP/0/RP0/CPU0:router(config-null0)# end または RP/0/RP0/CPU0:router(config-null0)# commit	設定変更を保存します。 <ul style="list-style-type: none">• end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:<ul style="list-style-type: none">– yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。– no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。– cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。• 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

コマンドまたはアクション	目的
ステップ 5 <code>show interfaces null 0</code> 例: RP/0/RP0/CPU0:router# show interfaces null0	ヌル インターフェイスの設定を確認します。

IPv4 仮想インターフェイスの設定

ここでは、IPv4 仮想インターフェイスの設定手順について説明します。

手順の概要

1. `configure`
2. `ipv4 address virtual address ip-address subnet mask`
3. `end`
または
`commit`

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 2 <code>ipv4 address virtual address ipv4-address/mask</code> 例: RP/0/RP0/CPU0:router(config)# <code>ipv4 virtual address 10.3.32.154/8</code>	管理イーサネット インターフェイスの IPv4 仮想アドレスを定義します。
ステップ 3 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-null0)# <code>end</code> または RP/0/RP0/CPU0:router(config-null0)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

仮想インターフェイスの設定例

ここでは、次の設定例について説明します。

- 「ループバック インターフェイスの設定：例」(P.236)
- 「ヌル インターフェイスの設定：例」(P.237)

ループバック インターフェイスの設定：例

次に、ループバック インターフェイスの設定例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Loopback 3
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RP0/CPU0:router# show interfaces Loopback 3
```

```
Loopback3 is up, line protocol is up
Hardware is Loopback interface(s)
Internet address is 172.18.189.38/32
MTU 1514 bytes, BW Unknown
reliability 0/255, txload Unknown, rxload Unknown
```

```
Encapsulation Loopback, loopback not set
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
  0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
```

ヌル インターフェイスの設定 : 例

次に、ヌル インターフェイスの設定例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Null 0
RP/0/RP0/CPU0:router(config-null0)# ipv4 unreachable
RP/0/RP0/CPU0:router(config-null0)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RP0/CPU0:router# show interfaces Null 0
```

```
Null0 is up, line protocol is up
Hardware is Null interface
Internet address is Unknown
MTU 1500 bytes, BW Unknown
  reliability 0/255, txload Unknown, rxload Unknown
Encapsulation Null, loopback not set
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
  0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
```

仮想 IPv4 インターフェイスの設定 : 例

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
RP/0/RP0/CPU0:router(config-null0)# commit
```

その他の参考資料

ここでは、ループバック インターフェイスおよびヌル インターフェイスの設定に関連する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイスの高度な設定と変更

LAN IP アドレスを介してスイッチに Telnet で接続するには、『Cisco IOS XR Getting Started Guide』の「*Configuring General Router Features*」の説明に従って管理イーサネット インターフェイスをセットアップし、Telnet サーバーをイネーブルにする必要があります。ここでは、『Cisco IOS XR Getting Started Guide』の説明に従って管理イーサネット インターフェイスを設定した後に、そのデフォルト設定を変更する手順について説明します。



(注) 物理レイヤ インターフェイス モジュール (PLIM) と管理イーサネット インターフェイス ポート間のフォワーディングは、デフォルトではディセーブルに設定されています。PLIM ポートと管理イーサネット インターフェイス ポート間のフォワーディングをイネーブルにするには、**rp mgmtethernet forwarding** コマンドを使用します。



(注) システムの管理イーサネット インターフェイスはデフォルトで表示されますが、これらのインターフェイスを使用してルータにアクセスしたり、Simple Network Management Protocol (SNMP; 簡易ネットワーク管理プロトコル)、Common Object Request Broker Architecture (CORBA)、HTTP、Extensible Markup Language (XML)、TFTP、Telnet、コマンドライン インターフェイス (CLI) などのプロトコルやアプリケーションを使用したりするには、インターフェイスを設定する必要があります。

Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイス設定の機能履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.0	変更ありません。
リリース 3.2	この機能が Cisco XR 12000 シリーズ ルータで初めてサポートされました。
リリース 3.3.0	管理イーサネット インターフェイスの手動設定という選択肢のみありました。当初は、ソフトウェアのインストール時にユーザが管理イーサネット インターフェイスの設定を行うように初期プロンプトが表示されていましたが、このプロンプトが削除されました。
リリース 3.4.0	変更ありません。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「管理イーサネット インターフェイス設定の前提条件」 (P.242)
- 「管理イーサネット インターフェイスの設定に関する情報」 (P.243)
- 「高度な管理イーサネット インターフェイス設定の実行方法」 (P.243)
- 「管理イーサネット インターフェイスの設定例」 (P.250)
- 「その他の参考資料」 (P.251)

管理イーサネット インターフェイス設定の前提条件

ここで説明する管理イーサネット インターフェイスの設定手順を実行する前に、次の作業を実施し、条件を満たしていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 『*Cisco IOS XR Getting Started Guide*』の「*Configuring General Router Features*」モジュールの説明に従って、管理イーサネット インターフェイスの初期設定を実行します。
- **show running-config** コマンドを使用するには、設定管理コマンドの適切なタスク ID を含むタスク グループに関連付けられたユーザ グループに属している必要があります。**show running-config** コマンドに使用できるタスク ID は、『*Cisco IOS XR System Management Command Reference*』に一覧が記載されています。
- 汎用インターフェイス名の仕様である *rack/slot/module/port* の適用方法を理解しています。
インターフェイスの命名規則の詳細については、『*Cisco IOS XR Getting Started Guide*』を参照してください。



(注)

トランスペアレント スイッチオーバーの場合、アクティブおよびスタンバイの管理イーサネット インターフェイスが両方とも、物理的に同じ LAN または スイッチに接続されている必要があることに注意してください。

管理イーサネット インターフェイスの設定に関する情報

管理イーサネット インターフェイスを設定するには、次の概念について理解している必要があります。

- 「[デフォルト インターフェイス設定](#)」 (P.243)

デフォルト インターフェイス設定

表 10 に、デフォルトの管理イーサネット インターフェイス設定を示します。これらの設定は、手動設定により変更できます。デフォルト設定は、`show running-config` コマンド出力には表示されません。

表 10 管理イーサネット インターフェイスのデフォルト設定

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
速度 (Mbps 単位)	速度はオートネゴシエーションされます。	<code>speed [10 100 1000]</code> システムをオートネゴシエーションされた速度に戻すには、 <code>no speed [10 100 1000]</code> コマンドを使用します。
デュプレックス モード	デュプレックス モードはオートネゴシエーションされます。	<code>duplex {full half}</code> システムをオートネゴシエーションされたデュプレックス操作に戻すには、必要に応じて <code>no duplex {full half}</code> コマンドを使用します。
MAC アドレス	MAC アドレスは、ハードウェアに組み込みのアドレス (BIA) から読み取られます。	<code>mac-address address</code> デバイスをデフォルトの MAC アドレスに戻すには、 <code>no mac-address address</code> コマンドを使用します。

高度な管理イーサネット インターフェイス設定の実行方法

ここでは、次の手順について説明します。

- 「[管理イーサネット インターフェイスの設定](#)」 (P.243) (必須)
- 「[管理イーサネット インターフェイスのデュプレックス モードの設定](#)」 (P.245) (任意)
- 「[管理イーサネット インターフェイスの速度の設定](#)」 (P.247) (任意)
- 「[管理イーサネット インターフェイスの MAC アドレスの変更](#)」 (P.248) (任意)
- 「[管理イーサネット インターフェイス設定の確認](#)」 (P.249) (任意)

管理イーサネット インターフェイスでのキャリア遅延を設定することもできます。キャリア遅延の詳細については、「[イーサネット インターフェイスでのキャリア遅延](#)」 (P.153) を参照してください。

管理イーサネット インターフェイスの設定

管理イーサネット インターフェイスを設定するには、次の作業を行います。この手順では、管理イーサネット インターフェイスに必要な最小限の設定について説明します。



(注) 『Cisco IOS XR Getting Started Guide』の「Configuring General Router Features」モジュールの説明に従って、すでに管理イーサネット インターフェイスをセットアップし、Telnet サーバをイネーブルにしている場合は、この作業を行う必要はありません。

手順の概要

1. **configure**
2. **interface MgmtEth interface-path-id**
3. **ipv4 address ip-address mask**
4. **mtu bytes**
5. **no shutdown**
6. **end**
または
commit
7. **show interfaces MgmtEth interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface MgmtEth interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイス名と <i>rack/slot/module/port</i> 表記を指定します。 この例では、スロット 0 にインストールされた RP カードのポート 0 を示しています。
ステップ 3	ipv4 address ip-address mask 例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 <ul style="list-style-type: none"> • <i>ip-address</i> をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • <i>mask</i> を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクを指定するには、次の 2 つの方法があります。 <ul style="list-style-type: none"> – ネットワーク マスクには、4 パートのドット付き 10 進アドレスを指定できます。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – ネットワーク マスクはスラッシュ (/) と数字で指定できます。たとえば、/8 は、マスクの先頭 8 ビットが 1 で、アドレスの対応するビットがネットワーク アドレスであることを示します。

コマンドまたはアクション	目的
ステップ 4 <code>mtu bytes</code> 例: RP/0/RP0/CPU0:router(config-if# mtu 1448	(任意) インターフェイスの最大伝送ユニット (MTU) 値を設定します。 <ul style="list-style-type: none"> デフォルトは 1514 バイトです。 管理イーサネット インターフェイスの mtu 値は 64 ~ 1514 バイトの範囲で指定します。
ステップ 5 <code>no shutdown</code> 例: RP/0/RP0/CPU0:router(config-if)# no shutdown	<code>shutdown</code> 設定を削除します。その結果、インターフェイスに強制されていた管理上のダウン状態が解除され、アップ状態またはダウン状態に移行できるようになります。
ステップ 6 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>show interfaces MgmtEth interface-path-id</code> 例: RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0	(任意) ルータ上のインターフェイスに関する統計情報を表示します。

管理イーサネット インターフェイスのデュプレックス モードの設定

RP に対応した管理イーサネット インターフェイスのデュプレックス モードを設定するには、次の作業を行います。

手順の概要

1. `configure`
2. `interface MgmtEth interface-path-id`

3. duplex [full | half]

4. end

または

commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface MgmtEth interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0	インターフェース コンフィギュレーション モードを開始し、管理イーサネット インターフェ이스の名前とインスタンスを指定します。
ステップ 3	<code>duplex [full half]</code> 例: RP/0/RP0/CPU0:router(config-if)# duplex full	インターフェースのデュプレックス モードを設定します。有効なオプションは full または half です。 (注) システムをオートネゴシエーションされたデュプレックス操作に戻すには、 no duplex コマンドを使用します。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーションセッションを継続するには、 commit コマンドを使用します。

管理イーサネット インターフェイスの速度の設定

RP に対応した管理イーサネット インターフェイスの速度を設定するには、次の作業を行います。

手順の概要

1. **configure**
2. **interface MgmtEth interface-path-id**
3. **speed {10 | 100 | 1000}**
4. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface MgmtEth interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0	インターフェイス コンフィギュレーション モードを開始し、管理イーサネット インターフェイスの名前とインスタンスを指定します。

コマンドまたはアクション	目的
<p>ステップ 3 <code>speed {10 100 1000}</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# speed 100</p>	<p>インターフェイス速度 <code>speed</code> パラメータを設定します。</p> <p>有効な <code>speed</code> オプションは、10、100、または 1000 Mbps です。</p> <p>(注) デフォルトの管理イーサネット インターフェイス速度はオートネゴシエーションされます。</p> <p>(注) システムをオートネゴシエーションされたデフォルトの速度に戻すには、no speed コマンドを使用します。</p>
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

管理イーサネット インターフェイスの MAC アドレスの変更

RP に対応した管理イーサネット インターフェイスの MAC レイヤ アドレスを設定するには、次の作業を行います。

手順の概要

1. `configure`
2. `interface MgmtEth interface-path-id`
3. `mac-address address`
4. `end`
または
`commit`

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface MgmtEth interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# <code>interface MgmtEth 0/RP0/CPU0/0</code>	インターフェイス コンフィギュレーション モードを開始し、管理イーサネット インターフェイスの名前とインスタンスを指定します。
ステップ 3 <code>mac-address address</code> 例: RP/0/RP0/CPU0:router(config-if)# <code>mac-address 0001.2468.ABCD</code>	管理イーサネット インターフェイスの MAC レイヤ アドレスを設定します。 (注) デバイスをデフォルトの MAC アドレスに戻すには、 no mac-address address コマンドを使用します。
ステップ 4 <code>end</code> または commit 例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

管理イーサネット インターフェイス設定の確認

RP に対応した管理イーサネット インターフェイスの設定変更を確認するには、次の作業を行います。

手順の概要

1. `show interfaces MgmtEth interface-path-id`
2. `show running-config`

ステップ 1 <code>show interfaces MgmtEth interface-path-id</code> 例: RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0	管理イーサネット インターフェイス設定を表示します。
ステップ 2 <code>show running-config interface MgmtEth interface-path-id</code> 例: RP/0/RP0/CPU0:router# show running-config interface MgmtEth 0/RP0/CPU0/0	実行コンフィギュレーションを表示します。

管理イーサネット インターフェイスの設定例

ここでは、次の設定例について説明します。

- 「管理イーサネット インターフェイスの設定：例」(P.250)

管理イーサネット インターフェイスの設定：例

次に、RP での管理イーサネット インターフェイスの高度な設定とその確認を行う例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:router(config)# ipv4 address 172.29.52.70 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# speed 100
RP/0/RP0/CPU0:router(config-if)# duplex full
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:Mar 26 01:09:28.685 :ifmgr[190]:%LINK-3-UPDOWN :Interface
MgmtEth0/RP0/CPU0/0, changed state to Up
RP/0/RP0/CPU0:router(config-if)# end

RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0

MMgmtEth0/RP0/CPU0/0 is up, line protocol is up
  Hardware is Management Ethernet, address is 0011.93ef.e8ea (bia 0011.93ef.e8ea
)
  Description: Connected to Lab LAN
  Internet address is 172.29.52.70/24
  MTU 1514 bytes, BW 100000 Kbit
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set,
  ARP type ARPA, ARP timeout 04:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 3000 bits/sec, 7 packets/sec
  5 minute output rate 0 bits/sec, 1 packets/sec
    30445 packets input, 1839328 bytes, 64 total input drops
    0 drops for unrecognized upper-level protocol
    Received 23564 broadcast packets, 0 multicast packets
      0 runts, 0 giants, 0 throttles, 0 parity
    57 input errors, 40 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    171672 packets output, 8029024 bytes, 0 total output drops
    Output 16 broadcast packets, 0 multicast packets
    0 output errors, 0 underruns, 0 applique, 0 resets
    0 output buffer failures, 0 output buffers swapped out
```

```

1 carrier transitions

RP/0/RP0/CPU0:router# show running-config interface MgmtEth 0/RP0/CPU0/0

interface MgmtEth0/RP0/CPU0/0
 description Connected to Lab LAN
 ipv4 address 172.29.52.70 255.255.255.0
!
```

その他の参考資料

管理イーサネット インターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの NetFlow の設定

NetFlow フローは、単一のインターフェイス（またはサブインターフェイス）に到着し、同じキーフィールドの値を持つ、単方向のパケット シーケンスです。

NetFlow は、次の目的で使用されます。

- アカウンティング/課金：NetFlow データを使用すると、非常に柔軟で詳細なリソース使用率アカウンティングのための、きめ細かな計測が可能です。
- ネットワーク プランニングと分析：NetFlow データは、戦略的ネットワーク プランニングのための重要な情報を提供します。
- ネットワーク モニタリング：NetFlow データを使用すると、ほぼリアルタイムなネットワーク モニタリングが可能です。

Cisco IOS XR ソフトウェアでの NetFlow の設定機能の履歴

リリース	変更点
リリース 3.2	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.3.0	<ul style="list-style-type: none">• この機能が Cisco XR 12000 シリーズ ルータで初めてサポートされました。• NetFlow BGP データ エクスポート機能を有効にするための bgp attribute-download コマンドの使用に関する情報が追加されました。• サブインターフェイスに対する NetFlow のサポートが、Cisco CRS-1 ルータに追加されました。
リリース 3.3.1	<ul style="list-style-type: none">• マルチプロトコル ラベル スイッチング (MPLS) 対応の NetFlow のサポートが、Cisco CRS-1 ルータに追加されました。• MPLS 対応の NetFlow をサポートするために、mpls flow コマンドに追加されました。• MPLS 対応の NetFlow をサポートするために、mpls record コマンドに追加されました。

リリース 3.4.0	<ul style="list-style-type: none"> 次のコマンドが、フロー エクスポート マップ コンフィギュレーション モードに移動されました。 <ul style="list-style-type: none"> destination dscp source transport udp 1 つのフロー モニタ マップで最大 8 個のエクスポートをサポートするように、NetFlow が更新されました。 キーとして使用する MPLS ラベルの個数を指定できるように NetFlow が更新されました。
リリース 3.4.1	<ul style="list-style-type: none"> Cisco CRS-1 ルータからの show コマンドの出力に、record mpls フィールドが追加されました。
リリース 3.5.0	<ul style="list-style-type: none"> record ipv6 コマンドが Cisco CRS-1 ルータに追加されました。 record mpls コマンドが Cisco XR 12000 シリーズ ルータに追加されました。 MPLS 対応の NetFlow が Cisco XR 12000 シリーズ ルータでサポートされるようになりました。 IPv6 対応の NetFlow のサポートが Cisco CRS-1 ルータに追加されました。
リリース 3.6.0	変更ありません。
リリース 3.7.0	宛先ベースの NetFlow アカウンティングが Cisco XR 12000 シリーズ ルータでサポートされるようになりました。
リリース 3.8.0	変更ありません。

この章の構成

この章で説明する内容は、次のとおりです。

- 「NetFlow を設定するための前提条件」(P.254)
- 「NetFlow を設定するための制約事項」(P.255)
- 「NetFlow の設定に関する情報」(P.255)
- 「Cisco IOS XR ソフトウェアでの NetFlow の設定方法」(P.262)
- 「NetFlow の設定例」(P.279)
- 「その他の参考資料」(P.282)
- 「RFC」(P.283)

NetFlow を設定するための前提条件

この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

NetFlow を設定するための制約事項

NetFlow を Cisco IOS XR ソフトウェアで設定するには、次の制約事項に留意してください。

- 送信元インターフェイスを設定する必要があります。送信元インターフェイスを設定しないと、エクスポートがディセーブル状態のままになります。
- Cisco IOS XR ソフトウェアは、エクスポート フォーマット バージョン 9 だけをサポートします。
- すべてのフロー モニタ マップに対して有効なレコード マップを設定する必要があります。



ヒント

管理インターフェイスを使用して NetFlow パケットをエクスポートしないことをお勧めします。管理インターフェイスのエクスポートは、Cisco XR 12000 シリーズ ルータでは動作せず、Cisco CRS-1 ルータでは非効率的です。

NetFlow の設定に関する情報

NetFlow を実装するには、次の概念について理解する必要があります。

- [「NetFlow の概要」 \(P.255\)](#)
- [「モニタ マップの概要」 \(P.256\)](#)
- [「サンプリング マップの概要」 \(P.256\)](#)
- [「エクスポート マップの概要」 \(P.257\)](#)
- [「NetFlow コンフィギュレーション サブモード」 \(P.257\)](#)
- [「IPv4 と IPv6 をサポートする MPLS フロー モニタ」 \(P.260\)](#)
- [「宛先ベースの NetFlow アカウンティング」 \(P.261\)](#)

NetFlow の概要

フローは、次の状況で、NetFlow エクスポート ユーザ データグラム プロトコル (UDP) データグラムの一部としてエクスポートされます。

- フローが長時間非アクティブまたはアクティブになっている場合
- フロー キャッシュがフルになりつつある場合
- いずれかのカウンタ (パケットまたはバイト) が一巡した場合
- ユーザがフローのエクスポートを実行した場合

NetFlow エクスポート UDP データグラムは、外部フロー コレクタ装置に送信され、そこで NetFlow エクスポート データのフィルタリングと集計が行われます。データのエクスポートは、期限満了フローと制御情報からなります。

NetFlow インフラストラクチャは、次のマップのコンフィギュレーションと使用に基づいています。

- モニタ マップ
- サンプラ マップ
- エクスポート マップ

これらのマップについて、以降の項で説明します。

モニタ マップの概要

モニタ マップには、フロー レコード マップとフロー エクスポート マップへの名前参照が含まれています。モニタ マップは、インターフェイスに適用されます。次のモニタ マップ属性を設定できます。

- フロー キャッシュ内のエントリ数。
- キャッシュの種類（パーマネントまたは通常）。パーマネント キャッシュのエントリは、ユーザが明示的にクリアするまでキャッシュから削除されません。
- アクティブ フロー タイムアウト。
- 非アクティブ フロー タイムアウト。
- アップデート タイムアウト。
- デフォルト タイムアウト。
- サンプリングおよび収集されるパケットのレコード タイプ。



(注) レコード名は、パケットがルータを通過するときに NetFlow がサンプリングするパケットの種類を示します。現在、MPLS、IPv4、および IPv6 パケットのサンプリングがサポートされています。



(注) アクティブ フロー タイムアウトおよび非アクティブ フロー タイムアウトは、通常のキャッシュ タイプに関連付けられます。アップデート タイムアウトは、パーマネント キャッシュ タイプに関連付けられます。

サンプラ マップの概要

サンプラ マップは、パケットのサンプリング率（ n 個のパケットのうち 1 個）を指定します。帯域幅が大きいインターフェイスでは、NetFlow 処理をすべてのパケットに適用すると、CPU 使用率が大幅に高くなります。サンプラ マップ コンフィギュレーションは、一般にそのような高速インターフェイス向けに作られています。NetFlow を双方向に適用する場合、各方向でフロー レコード パケットがポリシングされる速度は、Cisco CRS-1 ルータで 35,000 パケット/秒、Cisco XR 12000 シリーズ ルータで 25,000 パケット/秒です。NetFlow を単方向だけに適用する場合、各方向でフロー レコード パケットがポリシングされる速度は、Cisco CRS-1 ルータで 70,000 パケット/秒、Cisco XR 12000 シリーズ ルータで 50,000 パケット/秒です。



(注) これらの値は MSC-Rev.A ラインカード用です。Cisco CRS-1 ルータ用の MSC-Rev.B ラインカードでは、ポリシング速度が 1 つの方向で 62,500 パケット/秒と高速になっており、単方向で 125,000 パケット/秒が適用されます。

エクスポート マップの概要

エクスポート マップには、NetFlow エクスポート パケットに対する、ユーザ ネットワーク 指定とトランスポート レイヤの詳細が含まれています。 **flow exporter-map** コマンドを使用すると、コレクタ属性とバージョン属性を設定できます。次のコレクタ情報を設定できます。

- エクスポートの宛先 IP アドレス
- エクスポート パケットの DSCP 値
- 送信元インターフェイス
- UDP ポート番号 (コレクタが NetFlow パケットを受信するポート)
- エクスポート パケットの転送プロトコル



(注) Cisco IOS XR ソフトウェアでは、エクスポート パケットの転送プロトコルとして UDP だけがサポートされています。



(注) NetFlow エクスポート パケットは、送信元インターフェイスに割り当てられている IP アドレスを使用します。送信元インターフェイスに IP アドレスが割り当てられていない場合、エクスポートは非アクティブになります。

次のエクスポート バージョン属性も設定できます。

- テンプレート タイムアウト
- テンプレート データ タイムアウト
- テンプレート オプション タイムアウト
- インターフェイス テーブル タイムアウト
- サンプラ テーブル タイムアウト



(注) 1 つのフロー モニタ マップは、最大 8 つのエクスポートをサポートできます。

NetFlow コンフィギュレーション サブモード

Cisco IOS XR では、NetFlow マップの設定は、マップ固有のサブモードで行います。Cisco IOS XR では、次の NetFlow マップ コンフィギュレーション サブモードがサポートされています。

- 「フロー エクスポート マップ コンフィギュレーション サブモード」 (P.258)
- 「フロー エクスポート マップ バージョン コンフィギュレーション サブモード」 (P.258)
- 「フロー モニタ マップ コンフィギュレーション サブモード」 (P.259)
- 「サンプラ マップ コンフィギュレーション サブモード」 (P.259)



ヒント

Cisco IOS XR ソフトウェアでは、サブモードで使用できるコマンドのほとんどを、グローバル コンフィギュレーション モードで 1 つのコマンド文字列として実行できます。たとえば、**record ipv4** コマンドをフロー モニタ マップ コンフィギュレーション サブモードから、次のように実行できます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm
RP/0/RP0/CPU0:router(config-fmm)# record ipv4
```

また、同じコマンドを、グローバル コンフィギュレーション モードで、次のように実行できます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm record ipv4
```

フロー エクスポート マップ コンフィギュレーション サブモード

flow exporter-map fem-name コマンドをグローバル コンフィギュレーション モードで実行すると、コマンドライン インターフェイス (CLI) プロンプトが「**config-fem**」に変化し、フロー エクスポート マップ コンフィギュレーション サブモードが開始されたことが示されます。

次の出力例で、疑問符 (?) のオンライン ヘルプ機能により、フロー エクスポート マップ コンフィギュレーション サブモードで使用できるすべてのコマンドが表示されます。

```
RP/0/RP0/CPU0:router(config)# flow exporter-map fem

RP/0/RP0/CPU0:router(config-fem)# ?

  commit          Commit the configuration changes to running
  describe        Describe a command without taking real actions
  destination      Export destination configuration
  do              Run an exec command
  dscp            Specify DSCP value for export packets
  exit            Exit from this submode
  no              Negate a command or set its defaults
  root            Exit to the global configuration mode
  show            Show contents of configuration
  source          Source interface
  transport        Specify the transport protocol for export packets
  version         Specify export version parameters
```

```
RP/0/RP0/CP0:router(config-fem)#
```



(注) **version** コマンドを入力すると、フロー エクスポート マップ バージョン コンフィギュレーション サブモードが開始されます。



(注) 1 つのフロー モニタ マップは、最大 8 つのエクスポートをサポートできます。

フロー エクスポート マップ バージョン コンフィギュレーション サブモード

フロー エクスポート マップ コンフィギュレーション サブモードで **version v9** コマンドを実行すると、CLI プロンプトが「**config-fem-ver**」に変化し、フロー エクスポート マップ バージョン コンフィギュレーション サブモードが開始されたことが示されます。

次の出力例で、疑問符 (?) のオンライン ヘルプ機能により、フロー エクスポート マップ バージョン コンフィギュレーション サブモードで使用できるすべてのコマンドが表示されます。

```
RP/0/RP0/CPU0:router(config-fem)# version v9

RP/0/RP0/CPU0:router(config-fem-ver)# ?

  commit          Commit the configuration changes to running
  describe        Describe a command without taking real actions
```

```
do          Run an exec command
exit        Exit from this submode
no          Negate a command or set its defaults
options    Specify export of options template
show       Show contents of configuration
template   Specify template export parameters
```

```
RP/0/RP0/CPU0:router(config-fem-ver)#
```

フロー モニタ マップ コンフィギュレーション サブモード

flow monitor-map *map_name* コマンドをグローバル コンフィギュレーション モードで実行すると、CLI プロンプトが「**config-fmm**」に変化し、フロー モニタ マップ コンフィギュレーション サブモードが開始されたことが示されます。

次の出力例で、疑問符 (?) のオンライン ヘルプ機能により、フロー モニタ マップ コンフィギュレーション サブモードで使用できるすべてのコマンドが表示されます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm
```

```
RP/0/RP0/CPU0:router(config-fmm)# ?
```

```
cache      Specify flow cache attributes
commit     Commit the configuration changes to running
describe   Describe a command without taking real actions
do         Run an exec command
exit       Exit from this submode
exporter   Specify flow exporter map name
no         Negate a command or set its defaults
record     Specify a flow record map name
show       Show contents of configuration
```

```
RP/0/RP0/CP0:router(config-fmm)#
```

サンブラ マップ コンフィギュレーション サブモード

sampler-map *map_name* コマンドをグローバル コンフィギュレーション モードで実行すると、CLI プロンプトが「**config-sm**」に変化し、サンブラ マップ コンフィギュレーション サブモードが開始されたことが示されます。

次の出力例で、疑問符 (?) のオンライン ヘルプ機能により、サンブラ マップ コンフィギュレーション サブモードで使用できるすべてのコマンドが表示されます。

```
RP/0/RP0/CPU0:router(config)# sampler-map fmm
```

```
RP/0/RP0/CPU0:router(config-sm)# ?
```

```
commit     Commit the configuration changes to running
describe   Describe a command without taking real actions
do         Run an exec command
exit       Exit from this submode
no         Negate a command or set its defaults
random     Use random mode for sampling packets
show       Show contents of configuration
```

```
RP/0/RP0/CP0:router(config-sm)#
```

NetFlow BGP データ エクスポート機能のイネーブル化

NetFlow BGP ルーティング属性の収集をイネーブルにするには、**bgp attribute-download** コマンドを使用します。これによりルーティング属性がエクスポートされます。ルーティング属性が収集されない場合、ゼロ (0) がエクスポートされます。

BGP 属性のダウンロードがイネーブルになっている場合、BGP はプレフィックス (コミュニティ、拡張コミュニティ、AS パス) の属性情報を Routing Information Base (RIB; ルーティング情報ベース) および Forwarding Information Base (FIB; 転送情報ベース) にダウンロードします。これにより FIB は、プレフィックスを属性に関連付け、NetFlow 統計情報と関連する属性を送信できます。

IPv4 と IPv6 をサポートする MPLS フロー モニタ

Cisco IOS XR ソフトウェア リリース 3.8.0 は、MPLS パケットの NetFlow 収集をサポートしています。また、IPv4、IPv6、または IPv4 と IPv6 の両方のペイロードを伝送する MPLS パケットの NetFlow 収集もサポートしています。



(注) MPLS IPv6 は、Cisco XR 12000 シリーズ ルータではサポートされていません。

IPv4 と IPv6 をサポートするための MPLS キャッシュの再構成

Cisco IOS XR ソフトウェア リリース 3.8.0 では、同時に 1 つのインターフェイスで実行できる MPLS フロー モニタは 1 つだけです。追加の MPLS フロー モニタをインターフェイスに適用する場合、NetFlow モニタは既存のモニタを上書きします。

IPv4 フィールド、IPv6 フィールド、IPv4-IPv6 フィールドを収集するように MPLS フロー モニタを設定できます。IPv4-IPv6 コンフィギュレーションは、1 つの MPLS フロー モニタを使用して、IPv4 アドレスと IPv6 アドレスの両方を収集します。IPv4 コンフィギュレーションは、IPv4 アドレスだけを収集します。IPv6 コンフィギュレーションは、IPv6 アドレスだけを収集します。

MPLS フロー モニタは、最大 1,000,000 個のキャッシュ エントリをサポートします。NetFlow エントリには、次の種類のフィールドが含まれます。

- IPv4 フィールド
- IPv6 フィールド
- IPv4 フィールドがある MPLS
- IPv6 フィールドがある MPLS

NetFlow キャッシュ エントリあたりの最大バイト数は次のとおりです。

- IPv4 : 88 バイト/エントリ
- MPLS : 88 バイト/エントリ
- IPv6 : 108 バイト/エントリ
- IPv4 フィールドがある MPLS : 108 バイト/エントリ
- IPv6 フィールドがある MPLS : 128 バイト/エントリ



(注) 種類の異なる NetFlow エントリは、個別のキャッシュに格納されます。その結果、ラインカードの NetFlow エントリの数は、ラインカードの使用可能メモリ量に大きな影響を与えます。また、IPv6 のサンプリング レートが IPv4 のサンプリング レートと同じ場合でも、IPv6 フィールドでは長いキーが使用されるため、IPv6 の CPU 使用率は高くなります。

IPv6 フローがある MPLS パケット

MPLS パケット中の IPv6 フローの収集は任意です。CPU は、各 IPv6 フィールドに対し 128 バイトを使用します。IPv6 フローには、次の種類の情報が含まれます。

- 送信元 IP アドレス
- 宛先 IP アドレス
- トラフィック クラス値
- レイヤ 4 プロトコル番号
- レイヤ 4 送信元ポート番号
- レイヤ 4 宛先ポート番号
- フロー ID
- ヘッダー オプション マスク

MPLS パケット中の IPv6 フィールドを収集するには、**record mpls ipv6-fields** コマンドを実行して、MPLS レコードタイプ、**ipv6** フィールドをアクティブ化する必要があります。また、このコマンドで、集約で使用するラベルの数も指定できます。

宛先ベースの NetFlow アカウンティング

宛先ベースの NetFlow アカウンティングは、使用量ベースの課金アプリケーションです。宛先に従ってトラフィックを記録し、サービス プロバイダーが宛先固有のアカウンティングと課金を行えるようにします。宛先ベースの NetFlow アカウンティング レコードには、宛先ピアの Autonomous System (AS; 自律システム番号) と BGP ネクストホップ IP アドレスが含まれます。

宛先ベースの NetFlow アカウンティングでは、次のフィールドが収集およびエクスポートされます。

- 宛先ピアの AS 番号
- BGP ネクストホップ IP アドレス
- 入力インターフェイス
- 出力インターフェイス
- フォワーディング ステータス
- プッシュされるトップ ラベルの EXP ビット
- フローの方向 (ここでは常に入力)

宛先ベースの NetFlow アカウンティングは、Cisco IOS XR ソフトウェア リリース 3.8.0 上で次の機能をサポートしています。

- Cisco XR 12000 シリーズ ルータだけをサポート
- Cisco XR 12000 シリーズ ルータ上ですべてのシャーシ、ルート プロセッサ、ラインカード、共有ポート アダプタ (SPA) をサポート

- IPv4 アドレスだけをサポート
- 物理インターフェイス、バンドル インターフェイス、論理サブインターフェイス上のコンフィギュレーションをサポート
- IPv4 ユニキャスト トラフィックとマルチキャスト トラフィックをサポート
- 入トラフィックだけをサポート
- フル モード NetFlow だけをサポート
- ユーザ データグラム プロトコル (UDP) 上の NetFlow エクスポート フォーマット バージョン 9 をサポート

宛先ベースの NetFlow アカウンティングでは、Cisco IOS XR ソフトウェア リリース 3.8.0 上で次の機能をサポートしていません。

- IPv6 アドレスは未サポート
- MPLS IPv6 は未サポート
- Cisco CRS-1 ルータは未サポート
- 個別の Modular QoS Command-Line Interface (MQC) クラスは未サポート
- 同じインターフェイス上の同じ方向に対する、宛先ベースの NetFlow アカウンティングと、IPv4 でサンプリングされた NetFlow の同時コンフィギュレーションは未サポート
- レイヤ 2 スイッチド MPLS トラフィックは未サポート
- 出トラフィックは未サポート
- サンプリング モードの NetFlow は未サポート
- NetFlow エクスポート フォーマット バージョン 5、バージョン 8、IP Flow Information Export (IPFIX)、Stream Control Transmission Protocol (SCTP) は未サポート

Cisco IOS XR ソフトウェアでの NetFlow の設定方法

NetFlow の設定手順の概要は次のとおりです。

-
- ステップ 1** エクスポータ マップを作成および設定します。
- ステップ 2** モニタ マップとサンブラ マップを作成および設定します。



(注) モニタ マップは、ステップ 1 で作成するエクスポータ マップを参照する必要があります。エクスポータ マップをモニタ マップに適用しない場合、フロー レコードはエクスポートされず、エージングはモニタ マップで指定されたキャッシュ パラメータに従って実行されます。

- ステップ 3** モニタ マップとサンブラ マップをインターフェイスに適用します。
-

これらの手順について、次の項で詳しく説明します。

- 「エクスポータ マップの設定」(P.263)
- 「サンブラ マップの設定」(P.265)
- 「モニタ マップの設定」(P.267)
- 「モニタ マップとサンブラ マップのインターフェイスへの適用」(P.270)

- 「NetFlow データのクリア」 (P.271)
- 「IPv6 フィールドがある MPLS パケットの NetFlow 収集の設定」 (P.272)
- 「宛先ベースの NetFlow アカウンティングの設定」 (P.277)

エクスポート マップの設定

flow monitor-map map_name exporter map_name コマンドを使用して、エクスポート マップを設定し、モニタ マップに適用します。エクスポート マップは、モニタ マップを設定する前に設定できます。また、最初にモニタ マップを設定し、後でエクスポート マップを設定および適用することもできます。



(注) Cisco IOS XR ソフトウェアは、エクスポート マップ内でだけ単一のコレクタの設定をサポートします。

エクスポート マップを作成および設定する方法を次の手順に示します。

手順の概要

1. **configure**
2. **flow exporter-map map_name**
3. **destination hostname_or_IP_address**
4. **dscp dscp_value**
5. **source type interface-path-id**
6. **transport udp port**
7. **version v9**
8. **options {interface-table | sampler-table} [timeout seconds]**
9. **template [data | options] timeout seconds**
10. **end**
または
commit
11. **show flow exporter-map map_name**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow exporter-map map_name 例： RP/0/RP0/CPU0:router(config)# flow exporter-map fem	エクスポート マップを作成し、エクスポート マップ名を設定し、フロー エクスポート マップ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	destination <i>hostname_or_IP_address</i> 例: RP/0/RP0/CPU0:router(config-fem) # destination nnn.nnn.nnn.nnn	フロー エクスポート マップに対しエクスポートの宛先を設定します。宛先はホスト名でも IP アドレスでも構いません。
ステップ 4	dscp <i>dscp_value</i> 例: RP/0/RP0/CPU0:router(config-fem) # dscp 55	(任意) エクスポート パケットの Differentiated Services Codepoint (DSCP) 値を指定します。 <i>dscp_value</i> 引数には、0 ~ 63 の範囲の値を指定します。
ステップ 5	source <i>type interface-path-id</i> 例: RP/0/RP0/CPU0:router(config-fem) # source pos 0/1/0/1	送信元インターフェイスを、 <i>type interface-path-id</i> の形式で指定します。
ステップ 6	transport udp <i>port</i> 例: RP/0/RP0/CPU0:router(config-fem) # transport udp 9991	(任意) UDP パケットの宛先ポートを指定します。 <i>port</i> には、1024 ~ 65535 の範囲の宛先 UDP ポート値を指定します。
ステップ 7	version v9 例: RP/0/RP0/CPU0:router(config-fem-ver) # version v9	(任意) フロー エクスポート マップ バージョン コンフィギュレーション サブモードを開始します。
ステップ 8	options { interface-table sampler-table } [timeout <i>seconds</i>] 例: RP/0/RP0/CPU0:router(config-fem-ver) # options sampler-table timeout 2000	(任意) サンプラ テーブルのエクスポート タイムアウト値を設定します。 <i>seconds</i> には、1 ~ 604800 秒の範囲のエクスポート タイムアウト値を指定します。 デフォルト値は 1800 秒です。
ステップ 9	template [data options] timeout <i>seconds</i> 例: RP/0/RP0/CPU0:router(config-fem-ver) # template data timeout 10000	(任意) データ パケットのエクスポート期間を設定します。 <i>seconds</i> には、1 ~ 604800 秒の範囲のエクスポート タイムアウト値を指定します。

コマンドまたはアクション	目的
<p>ステップ 10 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router (config-fem-ver)# <code>end</code> または RP/0/RP0/CPU0:router(config-fem-ver)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <code>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</code> <code>[cancel]:</code> <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 11 <code>show flow exporter-map map_name</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show flow exporter-map fem</code></p>	<p>エクスポータ マップ データを表示します。</p>

サンプラ マップの設定

サンプラ マップを作成および設定する方法を次の手順に示します。

手順の概要

1. `configure`
2. `sampler-map map_name`
3. `random 1 out-of sampling_interval`
4. `end`
または
`commit`
5. `exit`
6. `exit`
7. `show sampler-map map_name`

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>sampler-map map_name</code> 例: RP/0/RP0/CPU0:router(config)# <code>sampler-map fsm</code> RP/0/RP0/CPU0:router(config-sm)#	<p>サンプラ マップを作成し、サンプラ マップ コンフィギュレーション モードを開始します。</p> <p>サンプラ マップを設定するときには、次の点に注意してください。</p> <ul style="list-style-type: none"> • Cisco CRS-1 ルータでは、NetFlow は個々のラインカードに対し、1 つの方向で 35,000 パケット/秒の速度でポリシングを実行できます。 • Cisco XR 12000 シリーズ ルータでは、SNF が 1 つの方向（入力または出力）でイネーブルになっている場合、NetFlow は個々のラインカードに対し、1 つの方向で 50,000 パケット/秒の速度でポリシングを実行できます。なお、この制限は、SNF が双方向でイネーブルになっている場合には適用されません。SNF が双方向でイネーブルになっている場合、NetFlow は個々のラインカードに対し、1 つの方向で 25,000 パケット/秒をサポートします。
ステップ 3 <code>random 1 out-of sampling_interval</code> 例: RP/0/RP0/CPU0:router(config-sm)# <code>random 1 out-of 10</code>	パケットのサンプリングで、サンプリング間隔にランダムモードを使用することを設定します。 <code>sampling_interval</code> 引数には、1 ~ 65535 の範囲の数値を指定します。

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router (config-sm) # <code>end</code> または RP/0/RP0/CPU0:router (config-sm) # <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config-sm) # <code>exit</code></p>	<p>サンプラ マップ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。</p>
<p>ステップ 6 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config) # <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。</p>
<p>ステップ 7 <code>show sampler-map map_name</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show sampler-map fsm</code></p>	<p>サンプラ マップ データを表示します。</p>

モニタ マップの設定

モニタ マップを作成および設定する方法を次の手順に示します。

手順の概要

1. **configure**
2. **flow monitor-map map_name**
3. **record ipv4**
または
record ipv6
または
record mpls
または

```
record mpls [ipv4-fields] [labels number]
または
record mpls [ipv6-fields] [labels number]
または
record mpls [ipv4-ipv6-fields] [labels number]
```

4. `cache entries number`
5. `cache permanent`
6. `cache timeout {active | inactive | upate} timeout_value`
7. `exporter map_name`
8. `end`
または
`commit`
9. `exit`
10. `exit`
11. `show flow monitor-map map_name`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>flow monitor-map map_name</code> 例: RP/0/RP0/CPU0:router(config)# flow monitor-map fmm RP/0/RP0/CPU0:router(config-fmm) #	モニタ マップを作成し、モニタ マップ名を設定して、フロー モニタ マップ コンフィギュレーション サブモードを開始します。

コマンドまたはアクション	目的
<p>ステップ 3</p> <pre>record ipv4 or record ipv6 or record mpls or record mpls [ipv4-fields] [labels number] or record mpls [ipv6-fields] [labels number] or record mpls [ipv4-ipv6-fields] [labels number]</pre> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# record ipv4</p>	<p>IPv4、IPv6、または MPLS のフロー レコード マップ名を設定します。</p> <ul style="list-style-type: none"> • record ipv4 コマンドを使用して、IPv4 のフロー レコード マップ名を設定します。 • record ipv6 コマンドを使用して、IPv6 のフロー レコード マップ名を設定します。 • record mpls labels コマンドを <i>number</i> 引数とともに使用して、集約するラベルの数を指定します。デフォルトでは、MPLS 対応の NetFlow は、MPLS ラベルスタックの上位 6 個のラベルを集約します。最大値は 6 です。 • record mpls ipv4-fields コマンドを使用し、MPLS 対応 NetFlow 中の IPv4 フィールドを収集します。 • record mpls ipv6-fields コマンドを使用し、MPLS 対応 NetFlow 中の IPv6 フィールドを収集します。 • record mpls ipv4-ipv6-fields コマンドを使用し、MPLS 対応 NetFlow 中の IPv4 および IPv6 フィールドを収集します。
<p>ステップ 4</p> <pre>cache entries number</pre> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# cache entries 10000</p>	<p>(任意) フロー キャッシュ中のエントリ数を設定します。 <i>number</i> 引数には、フロー キャッシュへの格納を許可するフロー エントリ数を、4096 ~ 1000000 の範囲で指定します。</p> <p>キャッシュ エントリのデフォルトの数は 65535 です。</p>
<p>ステップ 5</p> <pre>cache permanent</pre> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# flow monitor-map fmm cache permanent</p>	<p>(任意) フロー キャッシュからのエントリの削除をディセーブルにします。</p>
<p>ステップ 6</p> <pre>cache timeout {active timeout_value inactive timeout_value upate timeout_value}</pre> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 1000</p>	<p>(任意) アクティブ、非アクティブ、アップデート フロー キャッシュ タイムアウト値を設定します。</p> <ul style="list-style-type: none"> • 非アクティブ フロー キャッシュのデフォルトのタイムアウト値は 15 秒です。 • アクティブ フロー キャッシュのデフォルトのタイムアウト値は 1800 秒です。 • アップデート フロー キャッシュのデフォルトのタイムアウト値は 1800 秒です。 <p>(注) update timeout_value キーワード引数は、パーマネント キャッシュでだけ使用されます。このキーワードは、パーマネント キャッシュからエントリをエクスポートするために使用するタイムアウト値を指定します。この例では、エントリはエクスポートされますが、キャッシュに残ります。</p>
<p>ステップ 7</p> <pre>exporter map_name</pre> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# exporter fem</p>	<p>エクスポート マップをモニタ マップに関連付けます。</p> <p>(注) 1 つのフロー モニタ マップは、最大 8 つのエクスポートをサポートできます。</p>

コマンドまたはアクション	目的
<p>ステップ 8 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-fmm) # <code>end</code> または RP/0/RP0/CPU0:router(config-fmm) # <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 9 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config-fmm) # <code>exit</code></p>	<p>フロー モニタ マップ コンフィギュレーション サブモードを終了します。</p>
<p>ステップ 10 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config) # <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了します。</p>
<p>ステップ 11 <code>show flow monitor-map map_name</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show flow monitor-map fmm</code></p>	<p>フロー モニタ マップ データを表示します。</p>

モニタ マップとサンプラ マップのインターフェイスへの適用

手順の概要

モニタ マップとサンプラ マップをインターフェイスに適用するための手順を次に示します。

1. `configure`
2. `interface type number`
3. `flow [ipv4 | ipv6 | mpls] monitor monitor_map sampler sampler_map {egress | ingress}`
4. `end`
または
`commit`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface type number</code> 例: RP/0/RP0/CPU0:router(config)# interface POS 0/0/0/0 RP/0/RP0/CPU0:router(config-if)#	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>flow [ipv4 ipv6 mpls] monitor monitor_map sampler sampler_map {egress ingress}</code> 例: RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor fmm sampler fsm egress	モニタ マップとサンプラ マップをインターフェイスに関連付けます。 IPV4 NetFlow を指定したインターフェイス上でイネーブルにするには、 ipv4 と入力します。IPV6 NetFlow を指定したインターフェイス上でイネーブルにするには、 ipv6 と入力します。MPLS 対応の NetFlow を指定したインターフェイス上でイネーブルにするには、 mpls と入力します。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

NetFlow データのクリア

フロー エクスポート マップとフロー モニタ マップのデータをクリアする方法を次の手順で説明します。

手順の概要

1. `clear flow exporter [exporter_name] {restart | statistics} location node-id`

2. clear flow monitor [monitor_name] cache [force-export | statistics] location node-id}

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<pre>clear flow exporter [exporter_name] {restart statistics} location node-id</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# clear flow exporter statistics location 0/0/CPU0</pre>	<p>フロー エクスポート データをクリアします。</p> <p>エクスポート統計情報をクリアするには、statistics オプションを指定します。指定したノードで現在設定されているすべてのテンプレートをエクスポートするには、restart オプションを指定します。</p>
ステップ 2	<pre>clear flow monitor [monitor_name] cache [force-export statistics] location node-id</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# clear flow monitor cache force-export location 0/0/CPU0</pre>	<p>フロー モニタ データをクリアします。</p> <p>キャッシュ統計情報をクリアするには、statistics オプションを指定します。データをまずキャッシュからサーバにエクスポートし、次にキャッシュからエントリをクリアするには、force-export オプションを指定します。</p>

IPv6 フィールドがある MPLS パケットの NetFlow 収集の設定

IPv6 フィールドがある MPLS パケットの NetFlow 収集を設定する方法を、次の手順に示します。

手順の概要

1. **configure**
2. **flow exporter-map map_name**
3. **version v9**
4. **options {interface-table | sampler-table} [timeout seconds]**
5. **template [data | options] timeout seconds**
6. **exit**
7. **transport udp port**
8. **source type interface-path-id**
9. **destination hostname_or_IP_address**
10. **exit**
11. **flow monitor-map map_name**
12. **record mpls [ipv4-ipv6-fields] [labels number]**
13. **exporter map_name**
14. **cache entries number**
15. **cache timeout {active timeout_value | inactive timeout_value | upate timeout_value}**
16. **cache permanent**
17. **sampler-map map_name**
18. **random 1 out-of sampling_interval**
19. **interface type number**

20. `flow [ipv4 | ipv6 | mpls] monitor monitor_map sampler sampler_map {egress | ingress}`
21. `end`
または
`commit`
22. `exit`
23. `exit`
24. `show flow monitor-map map_name`
25. `show flow exporter-map map_name`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/RP0/CPU0:router# config	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>flow exporter-map map_name</code> 例： RP/0/RP0/CPU0:router(config)# flow exporter-map expl	エクスポート マップを作成し、エクスポート マップ名を設定し、フロー エクスポート マップ コンフィギュレーション モードを開始します。
ステップ 3	<code>version v9</code> 例： RP/0/RP0/CPU0:router(config-fem)# version v9	(任意) フロー エクスポート マップ バージョン コンフィギュレーション サブモードを開始します。
ステップ 4	<code>options {interface-table sampler-table} [timeout seconds]</code> 例： RP/0/RP0/CPU0:router(config-fem-ver)# options interface-table timeout 300	(任意) インターフェイス テーブルまたはサンプル テーブルのエクスポート タイムアウト値を設定します。 <code>seconds</code> には、1 ~ 604800 秒の範囲のエクスポート タイムアウト値を指定します。インターフェイス テーブルとサンプル テーブルの両方のデフォルトは 1800 秒です。 インターフェイス テーブルとサンプル テーブルの両方のエクスポート タイムアウト値を設定するには、このステップを 2 回実行する必要があります。
ステップ 5	<code>template [data options] timeout seconds</code> 例： RP/0/RP0/CPU0:router(config-fem-ver)# template data timeout 300	(任意) データ パケットまたはオプション パケットのエクスポート周期を設定します。 <code>seconds</code> には、1 ~ 604800 秒の範囲のエクスポート タイムアウト値を指定します。 データ パケットとオプション パケットの両方のエクスポート周期を設定するには、このステップを 2 回実行する必要があります。
ステップ 6	<code>exit</code> 例： RP/0/RP0/CPU0:router(config-fem-ver)# exit	フロー エクスポート マップ バージョン コンフィギュレーション モードを終了し、フロー エクスポート マップ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 7	<code>transport udp port</code> 例: RP/0/RP0/CPU0:router(config-fem)# transport udp 12515	(任意) UDP パケットの宛先ポートを指定します。 <i>port</i> には、1024 ~ 65535 の範囲の宛先 UDP ポート値を指定します。
ステップ 8	<code>source type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config-fem)# source Loopback0	送信元インターフェイスを、 <i>type interface-path-id</i> の形式で指定します。例: POS 0/1/0/1 または Loopback0
ステップ 9	<code>destination hostname_or_IP_address</code> 例: RP/0/RP0/CPU0:router(config-fem)# destination 170.1.1.11	フロー エクスポート マップに対しエクスポートの宛先を設定します。宛先はホスト名でも IP アドレスでも構いません。
ステップ 10	<code>exit</code> 例: RP/0/RP0/CPU0:router(config-fem)# exit	フロー エクスポート マップ コンフィギュレーション モードを終了します。
ステップ 11	<code>flow monitor-map map_name</code> 例: RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv6-fmm	モニタ マップを作成し、モニタ マップ名を設定して、フロー モニタ マップ コンフィギュレーション サブモードを開始します。
ステップ 12	<code>record mpls [ipv4-ipv6-fields] [labels number]</code> 例: RP/0/RP0/CPU0:router(config-fmm)# record mpls ipv6-fields labels 3	IPv4、IPv6、または MPLS のフロー レコード マップ名を設定します。 ipv4-ipv6-fields キーワードを使用し、MPLS 対応 NetFlow で IPv4 フィールドと IPv6 フィールドを収集します。
ステップ 13	<code>exporter map_name</code> 例: RP/0/RP0/CPU0:router(config-fmm)# exporter expl	エクスポート マップをモニタ マップに関連付けます。 (注) 1 つのフロー モニタ マップは、最大 8 つのエクスポートをサポートできます。
ステップ 14	<code>cache entries number</code> 例: RP/0/RP0/CPU0:router(config-fmm)# cache entries 10000	(任意) フロー キャッシュ中のエントリ数を設定します。 <i>number</i> 引数には、フロー キャッシュへの格納を許可するフロー エントリ数を、4096 ~ 1000000 の範囲で指定します。 キャッシュ エントリのデフォルトの数は 65535 です。

コマンドまたはアクション	目的
<p>ステップ 15 <code>cache timeout {active timeout_value inactive timeout_value upate timeout_value}</code></p> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 1800</p>	<p>(任意) アクティブ、非アクティブ、アップデートフロー キャッシュ タイムアウト値を設定します。</p> <ul style="list-style-type: none"> 非アクティブ フロー キャッシュのデフォルトのタイムアウト値は 15 秒です。 アクティブ フロー キャッシュのデフォルトのタイムアウト値は 1800 秒です。 アップデート フロー キャッシュのデフォルトのタイムアウト値は 1800 秒です。 <p>(注) <code>inactive</code> キーワードと <code>active</code> キーワードはパーマネント キャッシュには適用されません。</p> <p>(注) <code>update</code> キーワードは、パーマネント キャッシュでだけ使用されます。このキーワードは、パーマネント キャッシュからエントリをエクスポートするために使用するタイムアウト値を指定します。この例では、エントリはエクスポートされますが、キャッシュに残ります。</p>
<p>ステップ 16 <code>cache permanent</code></p> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# flow monitor-map fmm cache permanent</p>	<p>(任意) フロー キャッシュからのエントリの削除をディセーブルにします</p>
<p>ステップ 17 <code>exit</code></p> <p>例 : RP/0/RP0/CPU0:router(config-fmm)# exit</p>	<p>フロー モニタ マップ コンフィギュレーション サブモードを終了します。</p>
<p>ステップ 18 <code>sampler-map map_name</code></p> <p>例 : RP/0/RP0/CPU0:router(config)# sampler-map fsm RP/0/RP0/CPU0:router(config-sm)#</p>	<p>サンブラ マップを作成し、サンブラ マップ コンフィギュレーション モードを開始します。</p> <p>サンブラ マップを設定するときには、次の点に注意してください。</p> <ul style="list-style-type: none"> Cisco CRS-1 ルータでは、NetFlow は個々のラインカードに対し、1 つの方向で 35,000 パケット/秒の速度でポリシングを実行できます。 Cisco XR 12000 シリーズ ルータでは、SNF が 1 つの方向（入力または出力）でイネーブルになっている場合、NetFlow は個々のラインカードに対し、1 つの方向で 50,000 パケット/秒の速度でポリシングを実行できます。なお、この制限は、SNF が双方向でイネーブルになっている場合には適用されません。SNF が双方向でイネーブルになっている場合、NetFlow は個々のラインカードに対し、1 つの方向で 25,000 パケット/秒をサポートします。

コマンドまたはアクション	目的
ステップ 19 <code>random 1 out-of sampling_interval</code> 例: RP/0/RP0/CPU0:router(config-sm)# random 1 out-of 1000	パケットのサンプリングで、サンプリング間隔にランダムモードを使用することを設定します。 <i>sampling_interval</i> 引数には、1 ~ 65535 の範囲の数値を指定します。
ステップ 20 <code>exit</code> 例: RP/0/RP0/CPU0:router(config-sm)# exit	サンプリング マップ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 21 <code>interface type number</code> 例: RP/0/RP0/CPU0:router(config)# interface POS 0/0/0/0 RP/0/RP0/CPU0:router(config-if)#	インターフェイス コンフィギュレーション モードを開始します。
ステップ 22 <code>flow [ipv4 ipv6 mpls] monitor monitor_map sampler sampler_map {egress ingress}</code> 例: RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor MPLS-IPv6-fmm sampler fsm egress	モニタ マップとサンプリング マップをインターフェイスに関連付けます。 IPV4 NetFlow を指定したインターフェイス上でイネーブルにするには、 ipv4 と入力します。IPV6 NetFlow を指定したインターフェイス上でイネーブルにするには、 ipv6 と入力します。MPLS 対応の NetFlow を指定したインターフェイス上でイネーブルにするには、 mpls と入力します。
ステップ 23 <code>end</code> または commit 例: RP/0/RP0/CPU0:router (config-fem-ver)# end または RP/0/RP0/CPU0:router(config-fem-ver)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 24	exit 例： RP/0/RP0/CPU0:router(config-if)# exit	イーサネット インターフェイスのインターフェイス コンフィギュレーション サブモードを終了します。
ステップ 25	exit 例： RP/0/RP0/CPU0:router(config)# exit	グローバル コンフィギュレーション モードを終了します。
ステップ 26	show flow monitor-map <i>map_name</i> 例： RP/0/RP0/CPU0:router# show flow monitor-map fmm	フロー モニタ マップ データを表示します。
ステップ 27	show flow exporter-map <i>map_name</i> 例： RP/0/RP0/CPU0:router# show flow exporter-map fem	エクスポート マップ データを表示します。

宛先ベースの NetFlow アカウンティングの設定

宛先ベースの NetFlow アカウンティングを設定するには、フロー モニタ マップ、フロー レコード、フロー モニタを、次の手順で設定します。

手順の概要

1. **configure**
2. **flow monitor-map** *map_name*
3. **record ipv4 destination**
4. **exit**
5. **interface type interface-path-id**
6. **flow ipv4 monitor name ingress**
7. **end**
または
commit
8. **show flow exporter-map** *map_name*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	flow monitor-map map_name 例: RP/0/RP0/CPU0:router(config)# flow monitor-map map1 RP/0/RP0/CPU0:router(config-fmm)#	モニタ マップを作成し、モニタ マップ名を設定して、フロー モニタ マップ コンフィギュレーション サブモードを開始します。
ステップ 3	record ipv4 destination 例: RP/0/RP0/CPU0:router(config-fmm)# record ipv4 destination.	IPv4 宛先ベース NetFlow アカウンティング レコードのためのフロー レコードを設定します。 destination キーワードは、レコードが IPv4 宛先ベース NetFlow アカウンティング用であることを指定します。
ステップ 4	exit 例: RP/0/RP0/CPU0:router(config-fmm)# exit	フロー モニタ マップ モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 5	interface type interface-path-id 例: RP/0/RP0/CPU0:router# interface POS 0/1/0/0	インターフェイスの <i>type</i> と物理 <i>interface-path-id</i> は、タイプラック/スロット/モジュール/ポートの形式で指定します。 タイプ: POS、Ethernet、ATM など。 ラック: ラックのシャーシ番号。 スロット: ラインカードまたはモジュラ サービス カードの物理スロット番号。 モジュール: モジュール番号。物理レイヤ インターフェイス モジュール (PLIM) は常に 0 です。 ポート: インターフェイスの物理ポート番号。
ステップ 6	flow ipv4 monitor name ingress 例: RP/0/RP0/CPU0:router# flow ipv4 monitor monitor1 ingress	入力方向の IPv4 フロー モニタを設定し、モニタの名前を割り当てます。

	コマンドまたはアクション	目的
ステップ 7	<pre>end または commit 例: RP/0/RP0/CPU0:router(config-fmm)# end または RP/0/RP0/CPU0:router(config-fmm)# commit</pre>	現在のコンフィギュレーションの変更を保存します。
ステップ 8	<pre>show flow exporter-map map_name 例: RP/0/RP0/CPU0:router# show flow exporter-map fem</pre>	エクスポート マップ データを確認します。

NetFlow の設定例

次に NetFlow の設定例を示します。

- 「サンプラ マップ : 例」 (P.279)
- 「エクスポート マップ : 例」 (P.279)
- 「フロー モニタ マップ : 例」 (P.280)
- 「IPv4 と IPv6 をサポートする MPLS フロー モニタ : 例」 (P.280)
- 「宛先ベースの NetFlow アカウンティング : 例」 (P.281)

サンプラ マップ : 例

次に、1000 個のパケットから 1 個をサンプリングする新しいサンプラ マップ「fsm1」を作成する例を示します。

```
RP/0/RP0/CPU0:router(config)# sampler-map fsm1
RP/0/RP0/CPU0:router(config-sm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config)# exit
```

エクスポート マップ : 例

次に、NetFlow エクスポート パケットに対してバージョン 9 (V9) エクスポート フォーマットを使用する、新しいフロー エクスポート マップ「fem1」を作成する例を示します。データ テンプレート フローセットが V9 エクスポート パケットに 10 分ごとに挿入され、オプション インターフェイス テーブル フローセットが V9 エクスポート パケットに挿入されます。エクスポート パケットは、フロー コレクタの宛先 10.1.1.1 に送信されます。送信元アドレスは、ループバック 0 のインターフェイス IP アドレスと同じです。UDP 宛先ポートは 1024 であり、DSCP 値は 10 です。

```
RP/0/RP0/CPU0:router(config)# flow exporter-map fem1
RP/0/RP0/CPU0:router(config-fem)# destination 10.1.1.1
RP/0/RP0/CPU0:router(config-fem)# source Loopback 0
RP/0/RP0/CPU0:router(config-fem)# transport udp 1024
RP/0/RP0/CPU0:router(config-fem)# dscp 10
RP/0/RP0/CPU0:router(config-fem)# exit
```

```
RP/0/RP0/CPU0:router(config-fem)# version v9
RP/0/RP0/CPU0:router(config-fem-ver)# template data timeout 600
RP/0/RP0/CPU0:router(config-fem-ver)# options interface-table
RP/0/RP0/CPU0:router(config-fem-ver)# exit
```

フロー モニタ マップ : 例

次に、新しいフロー モニタマップ「fmm1」を作成する例を示します。このフロー モニタ マップは、フロー エクスポート マップ「fem1」を参照し、フロー キャッシュ属性を 10000 キャッシュ エントリに設定します。キャッシュのアクティブなエントリは、30 秒ごとにエージングされ、キャッシュの非アクティブなエントリは 15 秒ごとにエージングされます。このモニタ マップのレコード マップは IPv4 です。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map fmm1
RP/0/RP0/CPU0:router(config-fmm)# record ipv4
RP/0/RP0/CPU0:router(config-fmm)# exporter fem1
RP/0/RP0/CPU0:router(config-fmm)# cache entries 10000
RP/0/RP0/CPU0:router(config-fmm)# cache timeout active 30
RP/0/RP0/CPU0:router(config-fmm)# cache timeout inactive 15
RP/0/RP0/CPU0:router(config-fmm)# exit
```

次に、フロー モニタ「fmm1」およびサンプラ「fsm1」を TenGigE 0/0/0/0 インターフェイスの入力方向に適用する例を示します。

```
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/0
RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor fmm1 sampler fsm1 ingress
RP/0/RP0/CPU0:router(config-if)# exit
```

次に、IPv6 フィールドのある MPLS パケットを収集するように NetFlow モニタを設定する例を示します。

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# flow exporter-map expl
RP/0/RP0/CPU0:router(config-fem)# version v9
RP/0/RP0/CPU0:router(config-fem-ver)# options interface-table timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)# options sampler-table timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)# template data timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)# template options timeout 300
RP/0/RP0/CPU0:router(config-fem-ver)# exit
RP/0/RP0/CPU0:router(config-fem)# transport udp 12515
RP/0/RP0/CPU0:router(config-fem)# source Loopback0
RP/0/RP0/CPU0:router(config-fem)# destination 170.1.1.11
RP/0/RP0/CPU0:router(config-fem)# exit
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls ipv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)# exporter expl
RP/0/RP0/CPU0:router(config-fmm)# cache entries 10000
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config-fmm)# exit
RP/0/RP0/CPU0:router(config)# sampler-map FSM
RP/0/RP0/CPU0:router(config-sm)# random 1 out-of 1000
RP/0/RP0/CPU0:router(config-sm)# exit
RP/0/RP0/CPU0:router(config)# interface pos0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv6-fmm sampler FSM ingress
```

IPv4 と IPv6 をサポートする MPLS フロー モニタ : 例

次の設定では、MPLS トラフィックが収集されますが、ペイロード情報は収集されません。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls labels 3
```

```
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config)# exit
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-fmm sampler fsm ingress
```

次の設定では、IPv4 ペイロードがある MPLS トラフィックが収集されます。また、IPv4 ペイロードのない MPLS トラフィックも収集されますが、IPv4 フィールドにはゼロ (0) が設定されます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv4-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls IPv4-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config-fmm)# exit
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv4-fmm sampler fsm ingress
```

次の設定では、IPv6 ペイロードがある MPLS トラフィックが収集されます。また、IPv6 ペイロードのない MPLS トラフィックも収集されますが、IPv6 フィールドにはゼロ (0) が設定されます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls IPv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config-fmm)# exit
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv6-fmm sampler fsm ingress
```

次の設定は、IPv6 フィールドと IPv4 フィールドの両方がある MPLS トラフィックが収集されます。また、IPv4 ペイロードも IPv6 ペイロードもない MPLS トラフィックも収集されますが、IPv4 フィールドと IPv6 フィールドにはゼロ (0) が設定されます。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv4-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls IPv4-IPv6-fields labels 3
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config-fmm)# exit
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow mpls monitor MPLS-IPv4-IPv6-fmm sampler fsm ingress
```

次に、Packet-over-SONET (POS) インターフェイス上での IPv6 フィールドの収集を設定する例を示します。

```
RP/0/RP0/CPU0:router(config)# flow monitor-map MPLS-IPv6-fmm
RP/0/RP0/CPU0:router(config-fmm)# record mpls ipv6-fields labels <n>
RP/0/RP0/CPU0:router(config-fmm)# exporter <exporter-name>
RP/0/RP0/CPU0:router(config-fmm)# cache entries <number>
RP/0/RP0/CPU0:router(config-fmm)# cache timeout {active | inactive | update} <timeout>
RP/0/RP0/CPU0:router(config-fmm)# cache permanent
RP/0/RP0/CPU0:router(config-fmm)# exit
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow MPLS monitor <MPLS-IPv6-fields> sampler <name>
<direction>
```



(注)

フロー レコードは、バージョン 9 のフォーマットでエクスポートされます。

宛先ベースの NetFlow アカウンティング : 例

次に、宛先ベースの NetFlow アカウンティング用に IPv4 フロー レコードを設定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# flow monitor-map map1
RP/0/RP0/CPU0:router(config-fmm)# record ipv4 destination
RP/0/RP0/CPU0:router(config-fmm)# exporter fem
RP/0/RP0/CPU0:router(config-fmm)# exit
```

```
RP/0/RP0/CPU0:router(config)# interface pos 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# flow ipv4 monitor map1 ingress
RP/0/RP0/CPU0:router(config-if)# end
RP/0/RP0/CPU0:router# show flow exporter-map fem
RP/0/RP0/CPU0:router# show flow monitor-map map1
```

その他の参考資料

インターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface User Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用している MIB を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの POS インターフェイスの設定

ここでは、Packet-over-SONET/SDH (POS) インターフェイスの設定について説明します。POS インターフェイスは、Cisco ハイレベル データリンク コントロール (HDLC) プロトコルまたは ポイント ツーポイント プロトコル (PPP) カプセル化を使用して、SONET フレームおよび同期デジタル ハイアラキ (SDH) フレームを介した安全で信頼性の高いデータ伝送を実現します。Cisco XR 12000 シリーズ ルータは、Cisco HDLC カプセル化と PPP カプセル化に加え、フレームリレー カプセル化もサポートします。

レイヤ 1 の POS インターフェイスを設定するコマンドについては、『*Cisco IOS XR Interface and Hardware Component Command Reference*』を参照してください。

Cisco IOS XR ソフトウェアの POS インターフェイス設定機能の履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.0	変更ありません。
リリース 3.2	Cisco XR 12000 シリーズ ルータでこの機能がサポートされました。 次の SPA について、Cisco CRS-1 ルータ でのサポートが追加されました。 <ul style="list-style-type: none">• 1 ポート OC-192c/STM-64 POS/RPR XFP SPA• 4 ポート OC-3c/STM-1 POS SPA• Cisco CRS-1 ルータ用 SIP-800
リリース 3.3.0	Cisco CRS-1 ルータに 8 ポート OC-12c/STM-4 POS SPA のサポートが追加されました。 Cisco XR 12000 シリーズ ルータ上で、2 ポート OC-48 POS/RPR SPA のサポートが追加されました。

リリース 3.4.0	<p>次のハードウェアについて、Cisco CRS-1 ルータでのサポートが追加されました。</p> <ul style="list-style-type: none"> • 2 ポート OC-48c/STM16c POS SPA • 4 ポート OC-48c/STM16c POS SPA <p>Cisco XR 12000 シリーズ ルータ上で、次の機能のサポートが追加されました。</p> <ul style="list-style-type: none"> • 相手先固定接続 (PVC) のサブインターフェイス • 次のハードウェア上の POS メインインターフェイスおよび PVC でのフレームリレー カプセル化 <ul style="list-style-type: none"> – 4 ポート OC-3 POS/SDH SPA – 8 ポート OC-3 POS/SDH SPA – 2 ポート OC-12 POS/SDH SPA – 4 ポート OC-12 POS/SDH SPA – 8 ポート OC-12 POS/SDH SPA – 1 ポート OC-48/STM-16 POS/SDH SPA – 2 ポート OC-48/STM-16 POS/SDH SPA – 1 ポート OC-192c/STM-64c POS/SDH SPA – 4 ポート OC-3c/STM-1 POS/SDH ラインカード – 8 ポート OC-3c/STM-1c POS/SDH ラインカード – 16 ポート OC-3c/STM-1c POS/SDH ラインカード – 4 ポート チャネライズド OC-12/STM-4 POS ISE ラインカード – 4 ポート OC-12c/STM-4 POS/SDH ISE ラインカード – 1 ポート チャネライズド OC-48/STM-16 POS ISE ラインカード – 1 ポート OC-48c/STM-16c POS/SDH ISE ラインカード
リリース 3.4.1	<p>1 ポート OC-192c/STM-64 POS/RPR VSR 光ファイバ SPA について、Cisco CRS-1 ルータでのサポートが追加されました。</p>

リリース 3.5.0	<p>次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p> <ul style="list-style-type: none">• 1 ポート チャネライズド OC3 SPA• 1 ポート チャネライズド OC48 SPA• 1 ポート チャネライズド OC12 SPA• 2 ポート OC12 POS• 4 ポート OC12 POS• 8 ポート OC12 POS• 4 ポート OC3 POS• 8 ポート OC3 POS <p>Cisco XR 12000 シリーズ ルータでは、フレームリレー カプセル化を使用する POS インターフェイスに L2TPv3 ベースの L2VPN のサポートが追加されました。</p> <p>Cisco XR 12000 シリーズ ルータでは、interface コマンドに l2transport キーワードが追加されました。</p>
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	Cisco XR 12000 シリーズ ルータ上で、レイヤ 2 サブインターフェイスに対するサービス品質 (QoS) のサポートが追加されました。

この章の構成

- 「POS インターフェイスを設定するための前提事項」 (P.287)
- 「POS インターフェイスの設定に関する情報」 (P.288)
- 「POS インターフェイスの設定方法」 (P.293)
- 「POS インターフェイスの設定例」 (P.312)
- 「その他の参考資料」 (P.315)

POS インターフェイスを設定するための前提事項

POS インターフェイスを設定する前に、次の条件を満たしていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。
- 新しい POS インターフェイス設定に割り当てるインターフェイスの IP アドレスを調べておく必要があります。

- 次のコントローラ タイプのいずれか 1 つが設定済みであることが必要です。
 - SONET コントローラ。詳細については、このマニュアルの前のモジュールを参照してください。
 - DWDM コントローラ。詳細については、このマニュアルの「[Cisco IOS XR ソフトウェアでの高密度波長分割多重コントローラの設定](#)」モジュールを参照してください。



(注) POS DWDM コントローラの設定は、OC-768c/STM-256c DWDM PLIM だけでサポートされています。

POS インターフェイスの設定に関する情報

POS コントローラ インターフェイスを設定するには、次の概念を理解しておく必要があります。

- 「[Cisco HDLC カプセル化](#)」 (P.289)
- 「[PPP カプセル化](#)」 (P.289)
- 「[キープアライブ タイマー](#)」 (P.290)
- 「[フレームリレーのカプセル化](#)」 (P.291)
- 「[POS インターフェイスのデフォルト設定](#)」 (P.288)

Cisco XR 12000 シリーズ ルータでは、1 つの POS インターフェイスにおいて、PPP カプセル化、Cisco HDLC カプセル化、またはフレームリレー カプセル化を使用するデータが伝送されます。

Cisco CRS-1 ルータでは、1 つの POS インターフェイスにおいて、PPP カプセル化または Cisco HDLC カプセル化を使用するデータが伝送されます。フレームリレーは、Cisco CRS-1 ルータではサポートされていません。

ルータは、POS インターフェイス アドレスを識別するために、そのインターフェイスに関連付けられた物理レイヤ インターフェイス モジュール (PLIM) カードのラック番号、スロット番号、ベイ番号、およびポート番号を使用します。POS インターフェイス下にサブインターフェイスおよび相手先固定接続 (PVC) 設定されている場合、ルータは POS インターフェイス パス ID にサブインターフェイス番号を含めます。

POS インターフェイスのデフォルト設定

POS インターフェイスが始動され、追加のコンフィギュレーション コマンドが適用されない場合は、[表 11](#) に示すデフォルト インターフェイス設定が適用されます。これらのデフォルト設定はコンフィギュレーションで変更できます。

表 11 POS モジュラ サービス カードおよび PLIM のデフォルト インターフェイス設定

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定
キープアライブ	<code>keepalive [disable]</code>	10 秒のキープアライブ
カプセル化	Cisco XR 12000 シリーズ ルータの場合： <code>encapsulation [hdlc ppp frame-relay [IETF]]</code> Cisco CRS-1 ルータの場合： <code>encapsulation [hdlc ppp]</code>	<code>hdlc</code>

表 11 POS モジュラ サービス カードおよび PLIM のデフォルト インターフェイス設定 (続き)

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定
最大伝送ユニット (MTU)	<code>mtu bytes</code>	4474 バイト
巡回冗長検査 (CRC)	<code>crc [16 32]</code>	32



(注) デフォルト設定は、`show running-config` コマンドの出力には含まれません。

Cisco HDLC カプセル化

ハイレベル データリンク コントロール (HDLC) は、HDLC を使用して同期シリアル リンク上でデータを送信するシスコ独自のプロトコルです。また、Cisco HDLC は、シリアル リンク キープアライブを維持するために、Serial Line Address Resolution Protocol (SLARP) と呼ばれる単純な制御プロトコルも提供します。HDLC は、Cisco IOS XR ソフトウェアにおける POS インターフェイスのデフォルト カプセル化タイプです。Cisco HDLC は、開放型システム間相互接続 (OSI) スタックのレイヤ 2 (データ リンク) におけるデータ カプセル化のデフォルトであり、効率的なパケット記述およびエラー制御を実現します。



(注) Cisco HDLC は、POS インターフェイスにおいてデフォルトでイネーブルになります。

Cisco HDLC では、「[キープアライブ タイマー](#)」(P.290) で説明するように、キープアライブを使用してリンク ステートをモニタします。



(注) キープアライブ タイマーを設定した後で、ピアに送信される Serial Line Address Resolution Protocol (SLARP) パケットの情報を表示するには、`debug chdlc slarp packet` コマンドを使用します。

PPP カプセル化

PPP は、同期シリアル リンクでのデータ送信に使用される標準プロトコルです。PPP は、リンク プロパティのネゴシエーションを行う Link Control Protocol (LCP; リンク制御プロトコル) も提供します。LCP は、エコー要求および応答を使用して、リンクを継続的に使用できるかどうかをモニタします。



(注) インターフェイスに PPP カプセル化が設定されている場合、ECHOREQ パケットを送信し、ECHOREP 応答を受信しなかった回数が 3 回に達すると、リンク ダウンが宣言され、完全な LCP ネゴシエーションが再度開始されます。

PPP は、リンク上で動作するデータ プロトコルのプロパティをネゴシエーションするプロトコルとして、以下の Network Control Protocol (NCP; ネットワーク制御プロトコル) を提供します。

- IP Control Protocol (IPCP; IP コントロール プロトコル) : IP プロパティのネゴシエーションを行います。
- Multiprotocol Label Switching control processor (MPLSCP; マルチプロトコル ラベル スイッチング コントロール プロセッサ) : MPLS プロパティのネゴシエーションを行います。

- Cisco Discovery Protocol control processor (CDPCP; Cisco Discovery Protocol コントロール プロセッサ) : CDP プロパティのネゴシエーションを行います。
- IPv6CP : IP Version 6 (IPv6) プロパティのネゴシエーションを行います。
- Open Systems Interconnection control processor (OSICP; 開放型システム間相互接続コントロール プロセッサ) : OSI プロパティのネゴシエーションを行います。

PPP では、「[キープアライブ タイマー](#)」(P.290) で説明するように、キープアライブを使用してリンクステートをモニタします。

PPP は、データ トラフィックの伝送を許可する前にリモート装置にアイデンティティの証明を要求する、以下の認証プロトコルをサポートします。

- Challenge Handshake Authentication Protocol (CHAP; チャレンジ ハンドシェイク認証プロトコル) : CHAP 認証では、リモート装置にチャレンジメッセージが送信されます。リモート装置は共有秘密鍵でチャレンジ値を暗号化し、暗号化された値と名前を応答メッセージでローカル ルータに返します。ローカル ルータは、リモート装置の名前をローカル ユーザ名データベースまたはリモートセキュリティ サーバデータベースに保存されている対応する秘密鍵と照合し、保存されている秘密鍵を使用することで元のチャレンジメッセージを暗号化して、暗号化された値と一致することを確認します。
- Microsoft Challenge Handshake Authentication Protocol (MS-CHAP; マイクロソフト チャレンジ ハンドシェイク認証プロトコル) : MS-CHAP は Microsoft バージョンの CHAP です。標準バージョンの CHAP と同様、MS-CHAP も PPP 認証に使用されます。この場合、Microsoft Windows NT または Microsoft Windows 95 を使用しているパーソナル コンピュータとネットワーク アクセス サーバとして動作するシスコのルータまたはアクセス サーバ間で認証が行われます。
- Password Authentication Protocol (PAP; パスワード認証プロトコル) : PAP 認証では、リモート装置が名前とパスワードを送信する必要があり、それらがローカル ユーザ名データベースまたはリモートセキュリティ サーバデータベース内の対応するエン트리と照合されます。



(注) PPP 認証プロトコルのイネーブル化および設定の詳細については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの PPP の設定](#)」モジュールを参照してください。

POS インターフェイスで CHAP、MS-CHAP、および PAP をイネーブルにするには、インターフェイス コンフィギュレーション モードで **ppp authentication** コマンドを使用します。



(注) PPP 認証をイネーブル化またはディセーブル化しても、リモート装置に対して自身の認証を行うローカル ルータの動作には影響しません。

キープアライブ タイマー

シスコのキープアライブはリンク ステートのモニタリングに役立ちます。キープアライブ タイマーの値によって決定される間隔で定期的にキープアライブがピアとの間で送受信されます。ピアから適切なキープアライブ応答を受信しなかったリンクは、ダウン状態に移行します。ピアから適切なキープアライブ応答があった場合、またはキープアライブがディセーブルの場合、リンクはアップ状態に移行します。

ピアにキープアライブを送信し、応答が得られなかった回数が 3 回に達すると、リンクはダウン状態に移行します。ECHOREQ パケットは、LCP ネゴシエーションが完了したとき (LCP のオープン時など) にだけ送信されます。

リンク制御プロトコル (LCP) がピアに ECHOREQ パケットを送信する間隔 (秒数) を設定するには、インターフェイス コンフィギュレーション モードで **keepalive** コマンドを使用します。

デフォルトのキープアライブ インターバルは 10 秒です。

システムをデフォルトのキープアライブ インターバルに戻すには、**no keepalive** コマンドを使用します。

キープアライブをディセーブルにするには、**keepalive disable** コマンドを使用します。



(注) **keepalive** コマンドは、HDLC カプセル化または PPP カプセル化を使用する POS インターフェイスだけに適用されます。

このコマンドはフレームリレー カプセル化を使用する POS インターフェイスには適用されません。



(注) MDR 中は、キープアライブ インターバルが 10 秒以上であることが必要です。

ピア上の LCP は、ECHOREQ パケットを受信すると、ピアでキープアライブがイネーブルであるかどうかにかかわらず、エコー応答 (ECHOREP) パケットで応答します。

キープアライブは 2 つのピア間で独立しています。一方のピアでキープアライブをイネーブルに設定し、もう一方でディセーブルに設定することもできます。キープアライブがローカルでディセーブルに設定されていても、LCP は受信した ECHOREQ パケットに対して ECHOREP パケットで応答します。同様に、キープアライブ インターバルがそれぞれのピアで異なっても LCP には影響しません。



(注) キープアライブ タイマーを設定した後で、ピアに送信される SLARP パケットの情報を表示するには、**debug chdlc slarp packet** コマンドと他の Cisco HDLC **debug** コマンドを使用します。

フレームリレーのカプセル化

Cisco XR 12000 シリーズ ルータでは、フレームリレー カプセル化を使用する POS インターフェイスの設定は階層形式となり、次の要素で構成されます。

1. POS メイン インターフェイスは物理インターフェイスとポートで構成されます。POS インターフェイスが Cisco HDLC カプセル化および PPP カプセル化を使用する接続をサポートしていない場合は、POS メイン インターフェイス下に PVC を持つサブインターフェイスを設定する必要があります。フレームリレー接続は PVC だけでサポートされます。
2. POS サブインターフェイスは POS メイン インターフェイス下に設定されます。POS サブインターフェイスは、その下に PVC を設定しなければトラフィックをアクティブに伝送しません。レイヤ 3 設定は、通常はサブインターフェイス上で行われます。
3. ポイントツーポイント PVC は POS サブインターフェイス下に設定します。PVC は、メイン インターフェイスの直下には設定できません。ポイントツーポイント PVC は、各サブインターフェイスに 1 つだけ設定できます。PVC は定義済みの回線パスを使用し、そのパスが中断されるとエラーになります。PVC は、回線が削除されるまでアクティブのままです。POS PVC 上の接続はフレームリレー カプセル化だけをサポートします。
4. レイヤ 2 PVC AC は、POS サブインターフェイス下に設定します。PVC は、メイン インターフェイスの直下には設定できません。レイヤ 2 PVC AC は、各サブインターフェイスに 1 つだけ設定できます。ポイントツーポイント PVC と同様、レイヤ 2 PVC AC も定義済みの回線パスを使用し、そのパスが中断されるとエラーになります。PVC は、回線が削除されるまでアクティブのままです。POS PVC 上の接続はフレームリレー カプセル化だけをサポートします。



(注) 親インターフェイスの管理ステートによって、サブインターフェイスとその PVC のステートが決まります。親インターフェイスまたはサブインターフェイスの管理ステートが変わると、その親インターフェイスまたはサブインターフェイス下に設定された子 PVC の管理ステートも変更されます。

Cisco XR 12000 シリーズ ルータでは、以下のハードウェアがフレームリレー カプセル化をサポートしています。

- 1 ポート 192c/STM-64c POS/SDH SPA
- 2 ポート OC48/STM16 POS/SDH SPA
- 4 ポート OC-3c/STM-1 POS/SDH ラインカード
- 8 ポート OC-3c/STM-1c POS/SDH ラインカード
- 16 ポート OC-3c/STM-1c POS/SDH ラインカード
- 4 ポート OC-12c/STM4 POS/SDH ISE ラインカード
- 1 ポート OC-48c/STM16c POS/SDH ISE ラインカード



(注) フレームリレー カプセル化は Cisco XR 12000 シリーズ ルータだけでサポートされています。

POS インターフェイスでフレームリレー カプセル化を設定するには、**encapsulation frame-relay** コマンドを使用します。

フレームリレー インターフェイスは、次の 2 種類のカプセル化されたフレームをサポートします。

- Cisco (これがデフォルト値です)
- IETF

PVC に Cisco カプセル化または IETF カプセル化を設定するには、PVC コンフィギュレーション モードで **encap** コマンドを使用します。PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン POS インターフェイスのカプセル化タイプを継承します。



(注) MPLS に設定された POS メイン インターフェイスには、Cisco カプセル化を設定する必要があります。IETF カプセル化は、MPLS ではサポートされません。

インターフェイスにフレームリレー カプセル化を設定する前に、そのインターフェイスから以前のレイヤ 3 設定がすべて削除されていることを確認する必要があります。たとえば、メイン インターフェイスに IP アドレスが設定されていないことが必要です。IP アドレスが設定されている場合、メイン インターフェイス上のフレームリレー設定は無効になります。

フレームリレー インターフェイス上の LMI

ローカル管理インターフェイス (LMI) プロトコルは、PVC の追加、削除、およびステータスをモニタリングします。また、フレームリレー UNI インターフェイスを構成するリンクの完全性も検証します。デフォルトでは、すべての PVC で **cisco LMI** がイネーブルになります。ただし、このマニュアルで後述する「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」モジュールで説明するように、LMI タイプを ANSI または Q.933 に変更できます。

LMI タイプが **cisco** (デフォルトの LMI タイプ) の場合、単一のインターフェイスでサポートできる PVC の最大数は、メイン インターフェイスの MTU サイズに関連します。次の式を使用して、カードまたは SPA でサポートされる PVC の最大数を計算します。

$$(MTU - 13) / 8 = PVC \text{ の最大数}$$



(注) POS インターフェイスの場合、**mtu** コマンドのデフォルト設定は 4474 バイトです。したがって、**cisco LMI** で設定された 1 つの POS インターフェイスでサポートされる PVC のデフォルトの最大数は 557 です。



(注) フレームリレー インターフェイスには LMI インターフェイス タイプを設定する必要があります。そうしなければ、POS インターフェイスはアップ状態になりません。プロバイダー エッジ (PE) ルータと Customer Edge (CE; カスタマー エッジ) ルータとの接続では、LMI がアップ状態になるためには、PE 側が DCE であり、CE 側が DTE であることが必要です。フレームリレー インターフェイスに対する LMI インターフェイス タイプの設定の詳細については、このマニュアルで後述する「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」モジュールを参照してください。

フレームリレー用の Layer 2 Tunnel Protocol Version 3 ベースのレイヤ 2 VPN

Layer 2 Tunnel Protocol Version 3 (L2TPv3) は、IP コア ネットワーク上でレイヤ 2 ペイロードをトンネリングするために使用されるプロトコルです。L2TPv3 は、IP ネットワーク上の L2VPN に対してパケットのシグナリングおよびフォーマットを定義します。

Cisco IOS XR ソフトウェアは、ポイントツーポイント エンドツーエンド サービスをサポートしており、2 つの接続回路 (AC) が相互に接続されます。

L2TPv3 接続の設定では、次のタスクを実行する必要があります。

- 各プロバイダー エッジ (PE) ルータでの AC の設定
- 2 つの PE ルータ間における L2TPv3 でカプセル化された疑似接続の設定

ここでは、フレームリレー カプセル化を使用する POS インターフェイスにレイヤ 2 AC を設定する方法について説明します。ネットワーク内での L2TPv3 疑似接続の設定に関する詳細については、『*Cisco IOS XR MPLS Configuration Guide*』の「*Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software*」モジュールを参照してください。L2VPN の設定の詳細については、『*Cisco IOS XR MPLS Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software*」モジュールを参照してください。

POS インターフェイスの設定方法

ここでは、次の手順について説明します。

- 「[POS インターフェイスの始動](#)」(P.294)
- 「[オプションの POS インターフェイス パラメータの設定](#)」(P.296)
- 「[PVC を持つポイントツーポイント POS サブインターフェイスの作成](#)」(P.299)
- 「[オプションの PVC パラメータの設定](#)」(P.301)
- 「[POS インターフェイスでのキープアライブ インターバルの変更](#)」(P.304)
- 「[PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成](#)」(P.306)

POS インターフェイスの始動

ここでは、POS インターフェイスの始動に使用するコマンドについて説明します。

前提条件

Cisco IOS XR ソフトウェアを実行するルータに POS ラインカードまたは SPA が取り付けられている必要があります。

制約事項

POS インターフェイスがアクティブになるためには、POS 接続の両端の設定が一致している必要があります。

手順の概要

1. **show interfaces**
2. **configure**
3. **interface pos interface-path-id**
4. **ipv4 address ipv4_address/prefix**
5. **no shutdown**
6. **end**
または
commit
7. **exit**
8. **exit**
9. 接続の他端でインターフェイスを始動するために、ステップ 1 ~ 8 を繰り返します。
10. **show ipv4 interface brief**
11. **show interfaces pos interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	show interfaces 例： RP/0/0/CPU0:router# show interfaces	(任意) 設定されているインターフェイスを表示します。 • このコマンドを使用して、ルータが PLIM カードを認識しているのかも確認します。
ステップ 2	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 3 <code>interface pos interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface POS 0/3/0/0	POS インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4 <code>ipv4 address ipv4_address/prefix</code> 例: RP/0/0/CPU0:router (config)#ipv4 address 10.46.8.6/24	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 (注) このインターフェイスにフレームリレー カプセル化を設定する場合は、このステップを省略してください。フレームリレーの場合、IP アドレスとサブネット マスクはサブインターフェイスに設定します。
ステップ 5 <code>no shutdown</code> 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。
ステップ 6 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>exit</code> 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 8 <code>exit</code> 例: RP/0/0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。

■ POS インターフェイスの設定方法

	コマンドまたはアクション	目的
ステップ 9	<pre>show interfaces configure interface pos interface-path-id no shut exit exit commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# show interfaces RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface pos 0/3/0/0 RP/0/0/CPU0:router (config-if)# no shutdown RP/0/0/CPU0:router (config-if)# commit RP/0/0/CPU0:router (config-if)# exit RP/0/0/CPU0:router (config)# exit</pre>	<p>接続の他端でインターフェイスを始動するためにステップ 1～8 を繰り返します。</p> <p>(注) POS 接続の両端で設定が一致している必要があります。</p>
ステップ 10	<pre>show ipv4 interface brief</pre> <p>例:</p> <pre>RP/0/0/CPU0:router # show ipv4 interface brief</pre>	<p>インターフェイスがアクティブであり、適切に設定されていることを確認します。</p> <p>POS インターフェイスが適切に始動されていると、show ipv4 interface brief コマンドの出力結果で、そのインターフェイスの [Status] フィールドに [Up] と表示されます。</p>
ステップ 11	<pre>show interfaces pos interface-path-id</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# show interfaces pos 0/3/0/0</pre>	<p>(任意) インターフェイスの設定を表示します。</p>

次に行う作業

始動した POS インターフェイスのデフォルト設定を変更するには、「[オプションの POS インターフェイス パラメータの設定](#)」(P.296) を参照してください。

オプションの POS インターフェイス パラメータの設定

ここでは、POS インターフェイスのデフォルト設定の変更には使用できるコマンドについて説明します。

前提条件

POS インターフェイスのデフォルト設定を変更する前に、POS インターフェイスを始動して、「[POS インターフェイスの始動](#)」(P.294) で説明するように shutdown 設定を削除することをお勧めします。

制約事項

POS インターフェイスでオプションのパラメータを設定するときには、次の制約事項が適用されます。

- POS インターフェイスがアクティブになるためには、POS 接続の両端の設定が一致している必要があります。
- PPP カプセル化を設定したインターフェイスで MTU 値を変更すると、回線プロトコルがフラップします。

手順の概要

1. `configure`
2. `interface pos interface-path-id`
3. `encapsulation [hdlc | ppp | frame-relay [IETF]]`
4. `pos crc {16 | 32}`
5. `mtu value`
6. `end`
または
`commit`
7. `exit`
8. `exit`
9. `show interfaces pos [interface-path-id]`

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface pos interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# <code>interface POS 0/3/0/0</code>	POS インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>encapsulation [hdlc ppp frame-relay [IETF]]</code> 例: RP/0/RP0/CPU0:router(config-if)# <code>encapsulation hdlc</code>	(任意) インターフェイス カプセル化パラメータおよび HDLC やポイントツーポイント プロトコル (PPP) などの詳細を設定します。 (注) デフォルトのカプセル化は hdlc です。 (注) frame-relay オプションは、Cisco XR 12000 シリーズ ルータでのみ使用可能です。
ステップ 4	<code>pos crc {16 32}</code> 例: RP/0/RP0/CPU0:router(config-if)# <code>pos crc 32</code>	(任意) インターフェイスの巡回冗長検査 (CRC) 値を設定します。16 ビットの CRC モードを指定するには 16 キーワード、32 ビットの CRC モードを指定するには 32 キーワードを入力します。 (注) デフォルト CRC は 32 です。
ステップ 5	<code>mtu value</code> 例: RP/0/RP0/CPU0:router(config-if)# <code>mtu 4474</code>	(任意) MTU 値を設定します。 • デフォルト値は 4474 です。 • POS MTU の範囲は 64 ~ 9216 です。

コマンドまたはアクション	目的
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router (config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router (config-if)# <code>exit</code></p>	<p>インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。</p>
<p>ステップ 8 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router (config)# <code>exit</code></p>	<p>グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。</p>
<p>ステップ 9 <code>show interfaces pos [interface-path-id]</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show interface pos 0/3/0/0</code></p>	<p>(任意) 指定した POS インターフェイスの一般情報を表示します。</p>

次に行う作業

- 始動した POS インターフェイス上に PVC を持つポイントツーポイント フレームリレー サブインターフェイスを作成するには、「[PVC を持つポイントツーポイント POS サブインターフェイスの作成](#)」(P.299) を参照してください。
- PPP カプセル化がイネーブルである POS インターフェイスに PPP 認証を設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの PPP の設定](#)」モジュールを参照してください。
- Cisco HDLC カプセル化または PPP カプセル化がイネーブルである POS インターフェイスのキープアライブ インターバルを変更するには、「[POS インターフェイスでのキープアライブ インターバルの変更](#)」(P.304) を参照してください。

- フレームリレー カプセル化がイネーブルである POS インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルの「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」の「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」モジュールを参照してください。

PVC を持つポイントツーポイント POS サブインターフェイスの作成

ここに記載する手順では、ポイントツーポイント POS サブインターフェイスを作成し、その POS サブインターフェイスに相手先固定接続 (PVC) を設定します。



(注) PVC を持つサブインターフェイスは Cisco XR 12000 シリーズ ルータだけでサポートされます。



(注) サブインターフェイスおよび PVC の作成は、フレームリレー カプセル化だけが設定されたインターフェイスでサポートされます。

前提条件

POS インターフェイスでサブインターフェイスを作成する前に、「[POS インターフェイスの始動 \(P.294\)](#)」で説明するように、フレームリレー カプセル化が設定されたメイン POS インターフェイスを始動する必要があります。

制約事項

PVC は、各ポイントツーポイント POS サブインターフェイスに 1 つだけ設定できます。

手順の概要

1. **configure**
2. **interface pos *interface-path-id.subinterface point-to-point***
3. **ipv4 address *ipv4_address/prefix***
4. **pvc *dldi***
5. **end**
または
commit
6. 接続の他端で POS サブインターフェイスおよび関連付けられている PVC を始動するためにステップ 1 ~ 5 を繰り返します。

■ POS インターフェイスの設定方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface pos interface-path-id.subinterface point-to-point 例: RP/0/0/CPU0:router (config)# interface pos 0/3/0/0.1 point-to-point	POS サブインターフェイス コンフィギュレーション モードを開始します。 <i>subinterface</i> は、1 から 4294967295 の範囲のサブインターフェイス ID に置き換えてください。
ステップ 3	ipv4 address ipv4_address/prefix 例: RP/0/0/CPU0:router (config-subif)#ipv4 address 10.46.8.6/24	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 4	pvc dci 例: RP/0/0/CPU0:router (config-subif)# pvc 20	POS 相手先固定接続 (PVC) を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。 <i>dci</i> を 16 から 1007 の範囲の PVC ID に置き換えます。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

コマンドまたはアクション	目的
<p>ステップ 5</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router (config-fr-vc)# end または RP/0/0/CPU0:router (config-fr-vc)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6</p> <pre>configure interface pos interface-path-id.subinterface pvc dlcid commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface pos 0/3/0/1.1 RP/0/0/CPU0:router (config-subif)# ipv4 address 10.46.8.5/24 RP/0/0/CPU0:router (config-subif)# pvc 20 RP/0/0/CPU0:router (config-fr-vc)# commit</pre>	<p>接続の他端で POS サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 5 を繰り返します。</p> <p>(注) DLCI (PVC ID) は、サブインターフェイス接続の両端で一致している必要があります。</p> <p>(注) 接続の他端のサブインターフェイスに IP アドレスおよびサブネットマスクを割り当てるときには、接続の両端のアドレスが同じサブネットに属している必要があることに注意してください。</p>

次に行う作業

- オプションの PVC パラメータを設定するには、「[オプションの PVC パラメータの設定](#)」(P.301)を参照してください。
- フレームリレー カプセル化がイネーブルである POS インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルの「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」の「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」モジュールを参照してください。
- レイヤ 3 QOS サービス ポリシーを PVC サブモードの PVC に付加するには、該当する Cisco IOS XR ソフトウェアのコンフィギュレーションガイドを参照してください。

オプションの PVC パラメータの設定

ここでは、POS PVC でのデフォルト設定の変更に使用できるコマンドについて説明します。

前提条件

PVC のデフォルト設定を変更する前に、「[PVC を持つポイントツーポイント POS サブインターフェイスの作成](#)」(P.299) で説明するように POS サブインターフェイスで PVC を作成する必要があります。

制約事項

- 接続がアクティブになるためには、DLCI (PVC ID) が PVC の両端で一致している必要があります。
- PVC DLCI を変更するには、PVC を削除し、新しい DLCI を設定して PVC を追加し直す必要があります。

手順の概要

1. **configure**
2. **interface pos** *interface-path-id.subinterface*
3. **pvc** *dlci*
4. **encap** [**cisco** | **ietf**]
5. **service-policy** {**input** | **output**} *policy-map*
6. **end**
または
commit
7. 接続の他端で PVC を設定するために、ステップ 1 ~ 6 を繰り返します。
8. **show frame-relay pvc** *dlci-number*
9. **show policy-map interface pos** *interface-path-id.subinterface* {**input** | **output**}
or
show policy-map type qos interface pos *interface-path-id.subinterface* {**input** | **output**}

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface pos <i>interface-path-id.subinterface</i> 例： RP/0/0/CPU0:router (config)# interface pos 0/3/0/0.1	POS サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc <i>dlci</i> 例： RP/0/0/CPU0:router (config-subif)# pvc 20	PVC に対するサブインターフェイス コンフィギュレーション モードを開始します。 <i>dlci</i> は、PVC の識別に使用される DLCI 番号に置き換えてください。有効値の範囲は 16 ~ 1007 です。

コマンドまたはアクション	目的
ステップ 4 <code>encap [cisco ietf]</code> 例: RP/0/0/CPU0:router (config-fr-vc)# encap ietf	(任意) フレームリレー PVC のカプセル化を設定します。 (注) PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン POS インターフェイスのカプセル化タイプを継承します。
ステップ 5 <code>service-policy {input output} policy-map</code> 例: RP/0/0/CPU0:router (config-fr-vc)# service-policy output policy1	ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。 (注) ポリシー マップの作成と設定については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。
ステップ 6 <code>end</code> または commit 例: RP/0/0/CPU0:router (config-fr-vc)# end または RP/0/0/CPU0:router (config-fr-vc)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>configure</code> interface pos interface-path-id.subinterface pvc dlci encap [cisco ietf] commit 例: RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface pos 0/3/0/1.1 RP/0/0/CPU0:router (config-subif)# pvc 20 RP/0/0/CPU0:router (config-fr-vc)# encap cisco RP/0/0/CPU0:router (config-fr-vc)# commit	接続の他端で POS サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1～6 を繰り返します。 (注) サブインターフェイス接続の両端で設定が一致している必要があります。

■ POS インターフェイスの設定方法

	コマンドまたはアクション	目的
ステップ 8	<pre>show frame-relay pvc dlci-number</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show frame-relay pvc 20</pre>	(任意) 指定した POS インターフェイスの設定を検証します。
ステップ 9	<pre>show policy-map interface pos interface-path-id.subinterface {input output} または show policy-map type qos interface pos interface-path-id.subinterface {input output}</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show policy-map interface pos 0/3/0/0.1 output または RP/0/RP0/CPU0:router# show policy-map type qos interface pos 0/3/0/0.1 output</pre>	(任意) サブインターフェイスに付加された入力ポリシーおよび出力ポリシーの統計情報と設定を表示します。

次に行う作業

フレームリレー カプセル化がイネーブルである POS インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルの「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」の「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」モジュールを参照してください。

POS インターフェイスでのキープアライブ インターバルの変更

Cisco HDLC カプセル化または PPP カプセル化がイネーブルである POS インターフェイスのキープアライブ インターバルを変更するには、次の作業を行います。



(注) POS インターフェイスで Cisco HDLC カプセル化または PPP カプセル化をイネーブルした場合、キープアライブ インターバルはデフォルトで 10 秒に設定されます。デフォルトのキープアライブ インターバルを変更する手順は、次のとおりです。



(注) Cisco HDLC は、POS インターフェイスにおいてデフォルトでイネーブルになります。

前提条件

キープアライブ タイマーの設定を変更する前に、インターフェイスで Cisco HDLC カプセル化または PPP カプセル化がイネーブルになっていることを確認する必要があります。インターフェイスで Cisco HDLC カプセル化または PPP カプセル化をイネーブルにするには、「[オプションの POS インターフェイス パラメータの設定](#)」(P.296) で説明するように **encapsulation** コマンドを使用します。

制約事項

MDR 中は、キープアライブ インターバルが 10 秒以上である必要があります。

手順の概要

1. **configure**
2. **interface pos interface-path-id**
3. **keepalive {seconds | disable}**
4. **end**
または
commit
5. **show interfaces type interface-path-id**

詳細手順

コマンドまたはアクション	目的
ステップ 1 configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 interface pos interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/0	POS インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3 keepalive {seconds disable} 例: RP/0/RP0/CPU0:router(config-if)# keepalive 3 または RP/0/RP0/CPU0:router(config-if)# keepalive disable	リンク制御プロトコル (LCP) がピアに ECHOREQ を送信する頻度 (秒) を指定します。デフォルトのキープアライブ インターバルは 10 秒です。 システムをデフォルトのキープアライブ インターバルに戻すには、 no keepalive コマンドを使用します。 キープアライブ タイマーをディセーブルにするには、 keepalive disable コマンドを使用します。

■ レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
<p>ステップ 4</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5</p> <pre>show interfaces pos interface-path-id</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show interfaces POS 0/3/0/0</pre>	<p>(任意) インターフェイスの設定を確認します。</p>

レイヤ 2 接続回路 (AC) の設定方法

レイヤ 2 接続回路 (AC) の設定作業について、次の手順で説明します。

- [PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成](#)
- [オプションのレイヤ 2 PVC パラメータの設定](#)



(注)

レイヤ 2 スイッチングのためのインターフェイスの設定後は、**ipv4 address** などのルーティング コマンドは使用できません。



(注)

現在、レイヤ 2 AC は、HDLC カプセル化または PPP カプセル化が設定されたインターフェイスではサポートされません。

PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成

ここに記載する手順では、PVC を持つレイヤ 2 フレームリレー サブインターフェイスを作成します。

前提条件

POS インターフェイスでサブインターフェイスを作成する前に、「[POS インターフェイスの始動](#)」(P.294) で説明するように POS インターフェイスを始動する必要があります。



(注)

インターフェイスをレイヤ 2 スイッチング用に設定する場合は、「[POS インターフェイスの始動](#)」設定手順のステップ 4 を省略してください。 **ipv4 address** コマンドは、フレームリレー カプセル化が設定されたインターフェイスでは使用できません。

制約事項

- 各サブインターフェイスで設定できる PVC は 1 つだけです。
- 接続が正しく動作するためには、PVC の両端で設定が一致している必要があります。
- ipv4 address** コマンドは、フレームリレー カプセル化が設定されたインターフェイスでは使用できません。インターフェイスをレイヤ 2 トランスポート モード用に設定する前に、IP アドレスの以前の設定を削除する必要があります。
- レイヤ 2 設定は、フレームリレー PVC だけでサポートされます。レイヤ 2 設定が直接メイン POS インターフェイスに適用されるレイヤ 2 ポート モードはサポートされていません。
- レイヤ 2 設定は Cisco XR 12000 シリーズ ルータだけで使用でき、CRS-1 シリーズでは使用できません。

手順の概要

- configure**
- interface pos interface-path-id.subinterface l2transport**
- pvc dcli**
- end**
または
commit
- AC の他端でサブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface pos interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config)# interface pos 0/3/0/0.1 l2transport	サブインターフェイスを作成して、そのサブインターフェイスに対する POS サブインターフェイス コンフィギュレーション モードを開始します。 (注) <i>subinterface</i> は、1 つのメイン インターフェイスに設定された他のサブインターフェイスに対して一意である必要があります。

■ レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
ステップ 3 <code>pvc dlci</code> 例: RP/0/0/CPU0:router(config-if)# pvc 100	フレームリレー相手先固定接続 (PVC) を作成して、レイヤ 2 転送 PVC コンフィギュレーション モードを開始します。 <i>dlci</i> は、PVC の識別に使用される DLCI 番号に置き換えてください。有効値の範囲は 16 ~ 1007 です。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。
ステップ 4 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-fr-vc)# end または RP/0/0/CPU0:router(config-fr-vc)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 5 AC の他端でサブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 4 を繰り返します。	AC を始動します。 (注) AC の両端で設定が一致している必要があります。

次に行う作業

- オプションのサブインターフェイス パラメータを設定するには、「[オプションのレイヤ 2 サブインターフェイス パラメータの設定](#)」(P.311) を参照してください。
- オプションの PVC パラメータを設定するには、「[オプションのレイヤ 2 PVC パラメータの設定](#)」(P.309) を参照してください。
- 作成した AC にポイントツーポイント疑似接続 XConnect を設定するには、『Cisco IOS XR MPLS Configuration Guide』の「Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software」モジュールを参照してください。
- L2VPN を設定するには、『Cisco IOS XR MPLS Configuration Guide』の「Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software」モジュールを参照してください。

オプションのレイヤ 2 PVC パラメータの設定

ここでは、フレームリレー レイヤ 2 PVC でのデフォルト設定の変更には使用できるコマンドについて説明します。

前提条件

「PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成」(P.306) で説明するように、レイヤ 2 サブインターフェイスで PVC を作成する必要があります。

手順の概要

1. **configure**
2. **interface pos interface-path-id.subinterface l2transport**
3. **pvc dlc**
4. **encap [cisco | ietf]**
5. **service-policy {input | output} policy-map**
6. **end**
または
commit
7. AC の他端で PVC を設定するために、ステップ 1 ~ 5 を繰り返します。
8. **show policy-map interface pos interface-path-id.subinterface {input | output}**
or
show policy-map type qos interface pos interface-path-id.subinterface {input | output}

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface pos interface-path-id.subinterface l2transport 例: RP/0/0/CPU0:router(config)# interface pos 0/6/0/1.10 l2transport	レイヤ 2 フレームリレー サブインターフェイスに対する POS サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc dlc 例: RP/0/0/CPU0:router(config-if)# pvc 100	指定した PVC に対するフレームリレー PVC コンフィギュレーション モードを開始します。 <i>dlci</i> は、PVC の識別に使用される DLCI 番号に置き換えてください。有効値の範囲は 16 ~ 1007 です。
ステップ 4	encap {cisco ietf} 例: RP/0/0/CPU0:router(config-fr-vc)# encap ietf	フレームリレー PVC のカプセル化を設定します。 PVC の両端でカプセル化タイプが一致している必要があります。

レイヤ 2 接続回路 (AC) の設定方法

コマンドまたはアクション	目的
<p>ステップ 5 <code>service-policy {input output} policy-map</code></p> <p>例： RP/0/0/CPU0:router (config-fr-vc)# service-policy output policy1</p>	<p>ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。</p> <p>(注) ポリシー マップの作成と設定については、『<i>Cisco IOS XR Modular Quality of Service Configuration Guide</i>』を参照してください。</p>
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例： RP/0/0/CPU0:router (config-pos-l2transport-pvc)# end または RP/0/0/CPU0:router (config-pos-l2transport-pvc)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 <p>設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。</p>
<p>ステップ 7 AC の他端で PVC を設定するために、ステップ 1 ~ 5 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) 接続の両端で設定が一致している必要があります。</p>
<p>ステップ 8 <code>show policy-map interface pos interface-path-id.subinterface {input output}</code> または <code>show policy-map type qos interface pos interface-path-id.subinterface {input output}</code></p> <p>例： RP/0/0/CPU0:router# show policy-map interface pos 0/6/0/1.10 output または RP/0/0/CPU0:router# show policy-map type qos interface pos 0/6/0/1.10 output</p>	<p>(任意) サブインターフェイスに付加された入力ポリシー および出力ポリシーの統計情報と設定を表示します。</p>

次に行う作業

- 作成した AC にポイントツーポイント疑似接続 XConnect を設定するには、『*Cisco IOS XR MPLS Configuration Guide*』の「*Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software*」モジュールを参照してください。

- L2VPN を設定するには、『Cisco IOS XR MPLS Configuration Guide』の「Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software」モジュールを参照してください。

オプションのレイヤ 2 サブインターフェイス パラメータの設定

ここでは、フレームリレー レイヤ 2 サブインターフェイスでのデフォルト設定の変更に使用できるコマンドについて説明します。

前提条件

PVC のデフォルト設定を変更する前に、「[PVC を持つレイヤ 2 フレームリレー サブインターフェイスの作成](#)」(P.306) で説明するようにレイヤ 2 サブインターフェイスで PVC を作成する必要があります。

制約事項

ほとんどの場合、サブインターフェイスに設定された MTU がメイン インターフェイスに設定された MTU より優先されます。このルールの例外は、サブインターフェイスの MTU がメイン インターフェイスの MTU より大きい場合です。その場合、CLI 出力にはサブインターフェイスの MTU の設定値が表示されますが、実際に有効となる MTU はメイン インターフェイスに設定された値です。レイヤ 2 接続のトラブルシューティングや最適化において混乱を避けるために、メイン インターフェイスに設定する MTU の方を大きくすることをお勧めします。

手順の概要

1. `configure`
2. `interface pos interface-path-id.subinterface`
3. `mtu value`
4. `end`
または
`commit`
5. AC の他端でサブインターフェイスを設定するために、ステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface pos interface-path-id.subinterface</code> 例: RP/0/0/CPU0:router(config)# <code>interface pos 0/3/0/1.1</code>	レイヤ 2 フレームリレー サブインターフェイスに対する POS サブインターフェイス コンフィギュレーション モードを開始します。

■ POS インターフェイスの設定例

	コマンドまたはアクション	目的
ステップ 3	<pre>mtu value</pre> <p>例： RP/0/RP0/CPU0:router(config-if)# mtu 5000 </p>	(任意) MTU 値を設定します。有効値の範囲は 64 ~ 65535 です。
ステップ 4	<pre>end</pre> <p>または</p> <pre>commit</pre> <p>例： RP/0/0/CPU0:router(config-pos-l2transport-pvc)# end</p> <p>または</p> <pre>RP/0/0/CPU0:router(config-pos-l2transport-pvc)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <p>設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。</p>
ステップ 5	AC の他端で PVC を設定するために、ステップ 1 ~ 4 を繰り返します。	<p>AC を始動します。</p> <p>(注) 接続の両端で設定が一致している必要があります。</p>

POS インターフェイスの設定例

ここでは、次の設定例について説明します。

- 「[POS インターフェイスの始動と Cisco HDLC カプセル化の設定：例](#)」(P.312)
- 「[POS インターフェイスでのフレームリレー カプセル化の設定：例](#)」(P.313)
- 「[POS インターフェイスでの PPP カプセル化の設定：例](#)」(P.314)

POS インターフェイスの始動と Cisco HDLC カプセル化の設定：例

次に、Cisco HDLC カプセル化を設定した基本的な POS インターフェイスの始動例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

次に、キープアライブ メッセージの間隔を 10 秒に設定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# keepalive 10
RP/0/RP0/CPU0:router(config-if)# commit
```

POS インターフェイスでのフレームリレー カプセル化の設定 : 例

次に、ルータ 1 でフレームリレー カプセル化が設定された POS インターフェイスと PVC を持つポイントツーポイント POS サブインターフェイスを作成する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface pos 0/3/0/0.1 point-to-point
RP/0/0/CPU0:router (config-subif)# ipv4 address 10.20.3.1/24
RP/0/0/CPU0:router (config-subif)# pvc 100
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/0/CPU0:router# show interface POS 0/3/0/0

Wed Oct  8 04:20:30.248 PST DST
POS0/3/0/0 is up, line protocol is up
  Interface state transitions: 1
  Hardware is Packet over SONET/SDH
  Internet address is 10.20.3.1/24
  MTU 4474 bytes, BW 155520 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation FRAME-RELAY, crc 32, controller loopback not set,
  LMI enq sent 116, LMI stat recvd 76, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
  LMI DLCI 1023 LMI type is CISCO frame relay DTE
  Last clearing of "show interface" counters 00:00:06
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    1 packets input, 13 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
  Received 0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  1 packets output, 13 bytes, 0 total output drops
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
```

次に、ルータ 1 に接続されたルータ 2 でフレームリレー カプセル化が設定された POS インターフェイスと PVC を持つポイントツーポイント POS サブインターフェイスを作成する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface POS 0/3/0/1
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# frame-relay intf-type dce
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

■ POS インターフェイスの設定例

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface pos 0/3/0/1.1 point-to-point
RP/0/0/CPU0:router (config-subif)# ipv4 address 10.20.3.2/24
RP/0/0/CPU0:router (config-subif)# pvc 100
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0/0/CPU0:router# show interface POS 0/3/0/1
```

```
Wed Oct 8 04:20:38.037 PST DST
POS0/3/0/1 is up, line protocol is up
  Interface state transitions: 1
  Hardware is Packet over SONET/SDH
  Internet address is 10.20.3.2/24
  MTU 4474 bytes, BW 155520 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation FRAME-RELAY, crc 32, controller loopback not set,
  LMI enq sent 0, LMI stat recvd 0, LMI upd recvd 0
  LMI enq recvd 77, LMI stat sent 77, LMI upd sent 0, DCE LMI up
  LMI DLCI 1023 LMI type is CISCO frame relay DCE
  Last clearing of "show interface" counters 00:00:14
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    2 packets input, 26 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
    Received 0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    2 packets output, 26 bytes, 0 total output drops
    0 output errors, 0 underruns, 0 applique, 0 resets
    0 output buffer failures, 0 output buffers swapped out
```

次に、メイン POS インターフェイスで PVC を持つレイヤ 2 POS サブインターフェイスを作成する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# interface pos 0/3/0/0.1 l2transport
RP/0/0/CPU0:router (config-subif)# pvc 100
RP/0/0/CPU0:router(config-subif)# commit
```

POS インターフェイスでの PPP カプセル化の設定：例

次に、POS インターフェイスを作成し、PPP カプセル化を設定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# encapsulation ppp
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0/RP0/CPU0:router# show interfaces POS 0/3/0/0
```

```
POS0/3/0/0 is down, line protocol is down
  Hardware is Packet over SONET
  Internet address is 172.18.189.38/27
  MTU 4474 bytes, BW 2488320 Kbit
    reliability 0/255, txload Unknown, rxload Unknown
  Encapsulation PPP, crc 32, controller loopback not set, keepalive set (
  10 sec)
  LCP Closed
  Closed: IPCP
```

```

Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
  0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
  0 runts, 0 giants, 0 throttles, 0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions

```

その他の参考資料

ここでは、POS インターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
FRF.1.2	PVC User-to-Network Interface (UNI) Implementation Agreement - July 2000
ANSI T1.617 Annex D	-
ITU Q.933 Annex A	-

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC 1294	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 1315	<i>Management Information Base for Frame Relay DTEs</i>
RFC 1490	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 1586	<i>Guidelines for Running OSPF Over Frame Relay Networks</i>
RFC 1604	<i>Definitions of Managed Objects for Frame Relay Service</i>
RFC 2115	<i>Management Information Base for Frame Relay DTEs Using SMIPv2</i>
RFC 2390	<i>Inverse Address Resolution Protocol</i>
RFC 2427	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 2954	<i>Definitions of Managed Objects for Frame Relay Service</i>

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの PPP の設定

ここでは、Cisco IOS XR ソフトウェアの POS インターフェイスとシリアル インターフェイスでのポイントツーポイント プロトコル (PPP) に関連した作業について説明します。

- PPP 認証プロトコルのイネーブル化と設定
- PPP 認証のディセーブル化
- オプションの PPP timeout パラメータと retry パラメータの変更
- マルチリンク PPP の設定

PPP インターフェイス設定の機能履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータに PPP 認証が追加されました。
リリース 3.0	変更ありません。
リリース 3.3.0	Cisco XR 12000 シリーズ ルータでは、PPP カプセル化で設定したシリアル インターフェイスがサポートされるようになりました。
リリース 3.4.0	変更ありません。
リリース 3.4.1	Cisco XR 12000 シリーズ ルータでは、マルチリンク PPP がサポートされるようになりました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「[PPP 認証設定の前提条件](#)」 (P.318)
- 「[PPP 認証について](#)」 (P.318)
- 「[PPP 認証の設定方法](#)」 (P.320)
- 「[デフォルトの PPP 設定の変更方法](#)」 (P.329)
- 「[認証プロトコルをディセーブルにする方法](#)」 (P.332)
- 「[マルチリンク PPP について](#)」 (P.337)
- 「[マルチリンク PPP の設定方法](#)」 (P.339)
- 「[PPP の設定例](#)」 (P.346)

- 「その他の参考資料」(P.350)

PPP 認証設定の前提条件

POS インターフェイスまたはシリアル インターフェイスで PPP 認証を設定する前に、次のタスクと条件を満たしていることを確認します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンドタスク ID は、各コマンドリファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 使用しているハードウェアが POS インターフェイスまたはシリアル インターフェイスをサポートしている必要があります。
- 対応するモジュールの説明に従って、**encap ppp** コマンドを使用し、インターフェイスで PPP のカプセル化をイネーブルにしました。
 - POS インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「*Cisco IOS XR ソフトウェアでの POS インターフェイスの設定*」モジュールを参照してください。
 - シリアル インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「*Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定*」モジュールを参照してください。

PPP 認証について

インターフェイスに PPP 認証が設定されている場合、ホストは、PPP 接続を確立する前に他のホストがセキュア パスワードを使用して自身を一意に識別することを求めます。このパスワードは一意で、両方のホストで認識されています。

PPP は、次の認証プロトコルをサポートします。

- チャレンジ ハンドシェイク 認証プロトコル (CHAP)
- Microsoft による CHAP プロトコルの拡張版 (MS-CHAP)
- パスワード認証プロトコル (PAP)

POS インターフェイスまたはシリアル インターフェイス上で初めて PPP をイネーブルにしたときは、対象のインターフェイスで CHAP、MS-CHAP、PAP のいずれかのシークレット パスワードを設定するまで、そのインターフェイスでの認証はイネーブルになりません。インターフェイスで PPP を設定する場合、次の点に気を付けてください。

- CHAP、MS-CHAP、PAP は単一のインターフェイスに設定できますが、一度に使用される認証方式は 1 つだけです。使用される認証プロトコルの順序は、LCP ネゴシエーション中のピアによって決定されます。使用される最初の認証方式は、ピアによってもサポートされます。

- PAP は、POS インターフェイスおよびシリアル インターフェイスで使用可能な最小のセキュア認証プロトコルです。POS インターフェイスおよびシリアル インターフェイス経由で送信される情報について、より高レベルのセキュリティを確保するため、PAP 認証に加えて CHAP または MS-CHAP 認証を設定することをお勧めします。
- PPP 認証をイネーブル化またはディセーブル化しても、リモート装置に対して自身の認証を行うローカル ルータの動作には影響しません。
- **ppp authentication** コマンドは、インターフェイス上で CHAP、MS-CHAP、PAP 認証が選択される順序を指定するときにも使用されます。CHAP、MS-CHAP、PAP は、任意の順序でイネーブル化できます。3 つのすべての方式をイネーブル化すると、リンク ネゴシエーションでは、最初に指定された方式が要求されます。ピアが 2 番目の方式の使用を提案した場合、または最初の方式を拒否した場合は、2 番目の方式が試行されます。リモート装置の中には、1 つの方式しかサポートしないものがあります。方式の順序は、適切な方式で正しくネゴシエーションするためにリモート装置の機能で指定された方式と、求められるデータ ラインセキュリティのレベルに基づいて決定されます。PAP ユーザ名とパスワードはクリア テキスト文字列として送信されます。この文字列は、代行受信や再利用が可能です。

**注意**

aaa authentication ppp コマンドを使わずに設定した *list-name* 値を使用すると、インターフェイスはピアを認証できません。**ppp** キーワードを指定した **aaa authentication** コマンドの実装についての詳細は、『Cisco IOS XR System Security Command Reference』の「Authentication, Authorization, and Accounting Commands on Cisco IOS XR Software」モジュールおよび『Cisco IOS XR System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

PAP 認証

PAP は、リモート ノードに対し、2 ウェイ ハンドシェイクを使用してそのアイデンティティを確立するためのシンプルな方式を提供します。2 台のホスト間で PPP リンクが確立した後、ユーザ名とパスワードのペアは認証が確認されるまで、または接続が終了するまで、リモート ノードによってリンクを経由して（クリア テキストで）繰り返し送信されます。

PAP はセキュアな認証プロトコルではありません。パスワードはリンクを経由してクリア テキストで送信され、プレイバック攻撃やトライアルアンドエラー攻撃からの保護機能はありません。リモート ノードは、ログイン試行の頻度とタイミングを管理しています。

CHAP 認証

CHAP は RFC 1994 で定義され、3 ウェイ ハンドシェイクを使用してピアのアイデンティティを確認します。次の手順に、CHAP プロセスの概要を示します。

- ステップ 1** CHAP オーセンティケータがピアにチャレンジ メッセージを送信します。
- ステップ 2** ピアは 1 ウェイ ハッシュ関数で算出された値で応答します。
- ステップ 3** オーセンティケータは、応答を、独自の計算で予測したハッシュ値と照合します。値が一致すると、認証は成功します。値が一致しないと、接続は終了します。

この認証方式は、オーセンティケータとピアでのみ認識されている CHAP パスワードによって決まります。CHAP パスワードは、リンク経由では送信されません。認証は 1 ウェイですが、相互認証に同じ CHAP パスワードセットを使用することで、CHAP のネゴシエーションを双方向に行うことができます。



(注) 有効な CHAP 認証には、両方のホストの CHAP パスワードが同一である必要があります。

MS-CHAP 認証

Microsoft チャレンジ ハンドシェイク 認証プロトコル (MS-CHAP) は、Microsoft バージョンの CHAP で、RFC 1994 の拡張です。MS-CHAP では、CHAP と同じ認証プロセスが使用されます。ただし、認証は、Microsoft Windows NT または Microsoft Windows 95 を実行する PC と、ネットワーク アクセス サーバ (NAS) として動作する Cisco ルータまたはアクセス サーバの間で行われます。



(注) 有効な MS-CHAP 認証には、両方のホストの MS-CHAP パスワードが同一である必要があります。

PPP 認証の設定方法

ここでは、次の手順について説明します。

- 「PAP、CHAP、MS-CHAP 認証のイネーブル化」(P.320)
- 「PAP 認証パスワードの設定」(P.323)
- 「CHAP 認証パスワードの設定」(P.325)
- 「MS-CHAP 認証パスワードの設定」(P.327)

PAP、CHAP、MS-CHAP 認証のイネーブル化

ここでは、シリアル インターフェイスまたは POS インターフェイスで PAP、CHAP、MS-CHAP 認証をイネーブルにする手順について説明します。

前提条件

次のモジュールの説明に従って、`encapsulation ppp` コマンドを使用し、インターフェイスで PPP のカプセル化をイネーブルにする必要があります。

- POS インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

手順の概要

1. `configure`
2. `interface type interface-path-id`

3. `ppp authentication protocol [protocol [protocol]] [list-name | default]`
4. `end`
または
`commit`
5. `show ppp interfaces {type interface-path-id | all | brief {type interface-path-id | all | location node-id} | detail {type interface-path-id | all | location node-id} | location node-id}`

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3 <code>ppp authentication protocol [protocol [protocol]] [list-name default]</code> 例: RP/0/RP0/CPU0:router(config-if)# ppp authentication chap pap MIS-access	インターフェイスで CHAP、MS-CHAP、または PAP をイネーブルにし、インターフェイスで CHAP、MS-CHAP、PAP 認証が選択される順序を指定します。 <ul style="list-style-type: none"> • <code>protocol</code> 引数を、pap、chap、または ms-chap に置き換えます。 • <code>list name</code> 引数を、使用する認証方式のリストの名前に置き換えます。リストを作成するには、『Cisco IOS XR System Security Command Reference』の「Authentication, Authorization, and Accounting Commands on Cisco IOS XR Software」モジュールに記載されている説明に従って aaa authentication ppp コマンドを使用します。 • リストの名前を指定しないと、デフォルトが使用されます。デフォルトのリストは、『Cisco IOS XR System Security Command Reference』の「Authentication, Authorization, and Accounting Commands on Cisco IOS XR Software」モジュールに記載されている説明に従って aaa authentication ppp コマンドで指定します。

コマンドまたはアクション	目的
<p>ステップ 4</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5</p> <pre>show ppp interfaces {type interface-path-id all brief {type interface-path-id all location node-id} detail {type interface-path-id all location node-id} location node-id}</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show ppp interfaces serial 0/2/0/0</pre>	<p>インターフェイスの PPP ステート情報を表示します。</p> <ul style="list-style-type: none"> type interface-path-id 引数を入力すると、特定のインターフェイスの PPP 情報が表示されます。 brief キーワードを入力すると、ルータのすべてのインターフェイス、特定のインターフェイス インスタンス、または特定のノードのすべてのインターフェイスの簡易出力が表示されます。 all キーワードを入力すると、ルータにインストールされているすべてのノードの詳細な PPP 情報が表示されます。 location node-id キーワード引数を入力すると、指定したノードの詳細な PPP 情報が表示されます。 <p>リンク制御プロトコル (LCP) またはネットワーク制御プロトコル (NCP) に適用される PPP ステートには、7つのステートがあります。</p>

関連情報

対応する項の説明に従って、PAP、CHAP、または MS-CHAP 認証のパスワードを設定します。

- インターフェイスで PAP をイネーブルにする場合は、「[PAP 認証パスワードの設定](#)」(P.323) の説明に従って PAP 認証のユーザ名とパスワードを設定します。
- インターフェイスで CHAP をイネーブルにする場合は、「[CHAP 認証パスワードの設定](#)」(P.325) の説明に従って CHAP 認証のユーザ名とパスワードを設定します。
- インターフェイスで MS-CHAP をイネーブルにする場合は、「[MS-CHAP 認証パスワードの設定](#)」(P.327) の説明に従って MS-CHAP 認証のユーザ名とパスワードを設定します。

PAP 認証パスワードの設定

ここでは、シリアル インターフェイスまたは POS インターフェイスで PAP 認証をイネーブルにして設定する手順について説明します。



(注)

PAP は、POS およびインターフェイスで使用可能な最小のセキュア認証プロトコルです。POS およびインターフェイス経由で送信される情報について、より高レベルのセキュリティを確保するため、PAP 認証に加えて CHAP または MS-CHAP 認証を設定することをお勧めします。

前提条件

「PAP、CHAP、MS-CHAP 認証のイネーブル化」(P.320) の説明に従って、`ppp authentication` コマンドを使用し、インターフェイスで PAP 認証をイネーブルにする必要があります。

手順の概要

1. `configure`
2. `interface type interface-path-id`
3. `ppp pap sent-username username password [clear | encrypted] password`
4. `end`
または
`commit`
5. `show running-config`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： <code>RP/0/RP0/CPU0:router# configure</code>	グローバル コンフィギュレーション モードを開始します。

PPP 認証の設定方法

コマンドまたはアクション	目的
ステップ 2 <code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3 <code>ppp pap sent-username username password [clear encrypted] password</code> 例: RP/0/RP0/CPU0:router(config-if)# ppp pap sent-username xxxx password notified	インターフェイスでリモートのパスワード 認証プロトコル (PAP) サポートをイネーブルにし、ピアに対する PAP 認証要求に sent-username コマンドと password コマンドを含めます。 <ul style="list-style-type: none"> username 引数を、PAP 認証要求で送信するユーザ名に置き換えます。 password clear を入力してパスワードのクリア テキスト暗号化を選択するか、パスワードがすでに暗号化されている場合は password encrypted を入力します。 ppp pap sent--username コマンドを使用すると、複数の username および password コンフィギュレーション コマンドを、インターフェイス上にあるこのコマンドの単一コピーに置き換えることができます。 ppp pap sent-username コマンドは、インターフェイスごとに設定する必要があります。 リモートの PAP サポートでは、デフォルトでディセーブルになっています。
ステップ 4 <code>end</code> または commit 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 5 <code>show running-config</code> 例: RP/0/RP0/CPU0:router# show running-config	PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。

CHAP 認証パスワードの設定

ここでは、CHAP 認証をイネーブルにし、シリアル インターフェイスまたは POS インターフェイスで CHAP パスワードを設定する手順について説明します。

前提条件

「PAP、CHAP、MS-CHAP 認証のイネーブル化」(P.320) の説明に従って、**ppp authentication** コマンドを使用し、インターフェイスで CHAP 認証をイネーブルにする必要があります。

制約事項

両ホストのエンドポイントに同じ CHAP パスワードを設定する必要があります。

手順の概要

1. **configure**
2. **interface type interface-path-id**
3. **ppp chap password [clear | encrypted] password**
4. **end**
または
commit
5. **show running-config**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 2 <code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始 します。
ステップ 3 <code>ppp chap password [clear encrypted] password</code> 例: RP/0/RP0/CPU0:router(config-if)# ppp chap password clear xxxx	指定したインターフェイスで CHAP 認証をイネーブルにし、 インターフェイス固有の CHAP パスワードを定義します。 <ul style="list-style-type: none"> • clear を入力してクリア テキスト暗号化を選択するか、 パスワードがすでに暗号化されている場合は encrypted を入力します。 • password 引数を、クリア テキストまたはすでに暗号 化されているパスワードに置き換えます。このパス ワードは、ルータのコレクション間のセキュアな通信 の認証に使用されます。 • ppp chap password コマンドはリモート CHAP 認証 のみに使用され（ピアに対するルータ認証の場合）、 ローカルの CHAP 認証では有効になりません。このコ マンドは、このコマンドをサポートしないピアを認証 しようとする場合に使用すると便利です（古い Cisco IOS XR ソフトウェア イメージを実行している ルータなど）。 • CHAP シークレット パスワードは、不明なピアからの チャレンジに応答するためにルータによって使用され ます。

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>show running-config</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show running-config</code></p>	<p>PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。</p>

MS-CHAP 認証パスワードの設定

ここでは、MS-CHAP 認証をイネーブルにし、シリアル インターフェイスまたは POS インターフェイスで MS-CHAP パスワードを設定する手順について説明します。

前提条件

「[PAP、CHAP、MS-CHAP 認証のイネーブル化](#)」(P.320) の説明に従って、**ppp authentication** コマンドを使用し、インターフェイスで MS-CHAP 認証をイネーブルにする必要があります。

制約事項

両ホストのエンドポイントに同じ MS-CHAP パスワードを設定する必要があります。

手順の概要

1. `configure`
2. `interface type interface-path-id`
3. `ppp ms-chap password [clear | encrypted] password`

4. `end`
または
`commit`
5. `show running-config`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/RP0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface type interface-path-id</code> 例： RP/0/RP0/CPU0:router(config)# <code>interface serial 0/4/0/1</code>	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>ppp ms-chap password [clear encrypted] password</code> 例： RP/0/RP0/CPU0:router(config-if)# <code>ppp ms-chap password clear xxxx</code>	ルータのコレクションを呼び出すルータをイネーブルにし、共通の Microsoft チャレンジ ハンドシェイク 認証 (MS-CHAP) シークレット パスワードを設定します。 MS-CHAP シークレット パスワードは、不明なピアからのチャレンジに応答するためにルータによって使用されます。
ステップ 4	<code>end</code> または <code>commit</code> 例： RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – <code>yes</code> と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – <code>no</code> と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – <code>cancel</code> と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、<code>commit</code> コマンドを使用します。
ステップ 5	<code>show running-config</code> 例： RP/0/RP0/CPU0:router# <code>show running-config</code>	PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。

デフォルトの PPP 設定の変更方法

インターフェイスで初めて PPP をイネーブルにすると、次のデフォルト設定が適用されます。

- 認証が失敗すると、ただちに、インターフェイスは自身をリセットします。
- 応答がなくても許可される設定要求の最大数は 10 で、この数を超えるとすべての要求が停止されます。
- Negative Acknowledgment (CONFNAK; 否定応答) が連続して返される場合、それが許可される最大数は 5 で、この数を超えるとネゴシエーションが終了されます。
- 応答がなくても許可される Terminate Request (TermReq; 終了要求) の最大数は 2 で、この数を超えるとリンク制御プロトコル (LCP) またはネットワーク制御プロトコル (NCP) は終了されます。
- 認証パケットに対する応答の最大待機時間は 10 秒です。
- PPP ネゴシエーション中の応答の最大待機時間は 3 秒です。

ここでは、PPP カプセル化がイネーブルになっているシリアル インターフェイスまたは POS インターフェイスで基本的な PPP 設定を変更する手順について説明します。ここで使用するコマンドは、PPP (CHAP、MS-CHAP、PAP) によってサポートされるすべての種類の認証に適用されます。

前提条件

encapsulation ppp コマンドを使用し、インターフェイスで PPP カプセル化をイネーブルにする必要があります。

- POS インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- インターフェイスで POS のカプセル化をイネーブルにするには、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

手順の概要

1. **configure**
2. **interface type interface-path-id**
3. **ppp max-bad-auth retries**
4. **ppp max-configure retries**
5. **ppp max-failure retries**
6. **ppp max-terminate number**
7. **ppp timeout authentication seconds**
8. **ppp timeout retry seconds**
9. **end**
または
commit
10. **show ppp interfaces {type interface-path-id | all | brief {type interface-path-id | all | location node-id} | detail {type interface-path-id | all | location node-id} | location node-id}**

■ デフォルトの PPP 設定の変更方法

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface type interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ppp max-bad-auth retries 例: RP/0/RP0/CPU0:router(config-if)# ppp max-bad-auth 3	(任意) PPP 認証が失敗した後、インターフェイスで許可する認証のリトライ回数を設定します。 <ul style="list-style-type: none"> 許可する認証のリトライ回数を指定しない場合、認証が失敗すると、ただちに、ルータは自身をリセットします。 retries 引数を、0 ~ 10 の範囲でリトライ回数に置き換えます。この回数を超えると、インターフェイスは自身をリセットします。 デフォルトのリトライ回数は 0 回です。 ppp max-bad-auth コマンドは、PPP カプセル化がイネーブルになっている任意のインターフェイスに適用できます。
ステップ 4	ppp max-configure retries 例: RP/0/RP0/CPU0:router(config-if)# ppp max-configure 4	(任意) (応答なしで) 試行される設定要求の最大数を指定します。この数を超えると、要求は停止されます。 <ul style="list-style-type: none"> retries 引数を、4 ~ 20 の範囲で設定要求がリトライする最大回数に置き換えます。 デフォルトの設定要求の最大数は 10 です。 設定要求の最大回数分だけ送信されないうちに設定要求メッセージが応答を受け取った場合、以降の設定要求は放棄されます。
ステップ 5	ppp max-failure retries 例: RP/0/RP0/CPU0:router(config-if)# ppp max-failure 3	(任意) 否定応答 (CONFNAK) が連続して返される場合に、それが許可される最大数を設定します。この数を超えるとネゴシエーションは終了されます。 <ul style="list-style-type: none"> retries 引数を、2 ~ 10 の範囲で CONFNAK の最大数に置き換えます。この数を超えるとネゴシエーションは終了されます。 デフォルトの CONFNAK の最大数は 5 です。

コマンドまたはアクション	目的
<p>ステップ 6 <code>ppp max-terminate number</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# ppp max-terminate 5</p>	<p>(任意) 応答がなくても送信される終了要求 (TermReq) の最大数を設定します。この数を超えるとリンク制御プロトコル (LCP) またはネットワーク制御プロトコル (NCP) は終了されます。</p> <ul style="list-style-type: none"> • <i>number</i> 引数を、応答がなくても送信される TermReq の最大数に置き換えます。この数を超えると LCP または NCP は終了されます。範囲は 2 ~ 10 です。 • デフォルトの TermReq の最大数は 2 です。
<p>ステップ 7 <code>ppp timeout authentication seconds</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# ppp timeout authentication 20</p>	<p>(任意) PPP 認証タイムアウト パラメータを設定します。</p> <ul style="list-style-type: none"> • <i>seconds</i> 引数 を、認証パケットに対する応答を待機する最大時間 (秒) に置き換えます。範囲は 3 ~ 30 秒です。 • デフォルトの認証タイムアウトは 10 秒です。この時間には、リモート ルータが接続を認証して許可し、応答するまでの時間を組み込む必要があります。ただし、この処理に 10 秒かからないこともあります。そのような場合は ppp timeout authentication コマンドを使用してタイムアウト時間を短くし、認証応答が失われる場合の接続時間を改善します。
<p>ステップ 8 <code>ppp timeout retry seconds</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# ppp timeout retry 8</p>	<p>(任意) PPP 認証タイムアウト リトライ パラメータを設定します。</p> <ul style="list-style-type: none"> • <i>seconds</i> 引数 を、PPP ネゴシエーション時に応答を待機する最大時間 (秒) に置き換えます。範囲は 1 ~ 10 秒です。 • デフォルトは 3 秒です。

■ 認証プロトコルをディセーブルにする方法

コマンドまたはアクション	目的
<p>ステップ 9</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 10</p> <pre>show ppp interfaces {type interface-path-id all brief {type interface-path-id all location node-id} detail {type interface-path-id all location node-id} location node-id}</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show ppp interfaces serial 0/2/0/0</pre>	<p>インターフェイスまたは PPP カプセル化がイネーブルになっているすべてのインターフェイスの PPP 設定を確認します。</p>

認証プロトコルをディセーブルにする方法

ここでは、次の手順について説明します。

- 「インターフェイスでの PAP 認証のディセーブル化」(P.332)
- 「インターフェイスでの CHAP 認証のディセーブル化」(P.334)
- 「インターフェイスでの MS-CHAP 認証のディセーブル化」(P.336)

インターフェイスでの PAP 認証のディセーブル化

ここでは、シリアル インターフェイスまたは POS インターフェイスで PAP 認証をディセーブルにする手順について説明します。

手順の概要

1. configure

2. `interface type interface-path-id`
3. `ppp pap refuse`
4. `end`
または
`commit`
5. `show running-config`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface type interface-path-id 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ppp pap refuse 例: RP/0/RP0/CPU0:router(config-if)# ppp pap refuse	認証を要求するピアからのパスワード 認証プロトコル (PAP) 認証を拒否します。 <ul style="list-style-type: none"> • 発信チャレンジ ハンドシェイク 認証プロトコル (CHAP) が (ppp authentication コマンドを使用して) 設定されている場合、拒否パケットでの認証方式として CHAP が提案されます。 • PAP 認証は、デフォルトではディセーブルに設定されています。

■ 認証プロトコルをディセーブルにする方法

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>show running-config</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show running-config</code></p>	<p>PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。</p>

インターフェイスでの CHAP 認証のディセーブル化

ここでは、シリアル インターフェイスまたは POS インターフェイスで CHAP 認証をディセーブルにする手順について説明します。

手順の概要

1. `configure`
2. `interface type interface-path-id`
3. `ppp chap refuse`
4. `end`
または
`commit`
5. `show running-config`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>interface type interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<code>ppp chap refuse</code> 例: RP/0/RP0/CPU0:router(config-if)# ppp chap refuse	認証を要求するピアからの CHAP 認証を拒否します。指定したインターフェイスで <code>ppp chap refuse</code> コマンドを入力すると、CHAP を使用してユーザー認証を強制しようとしたピアの試行はすべて拒否されます。 <ul style="list-style-type: none">• CHAP 認証は、デフォルトではディセーブルに設定されています。• 発信パスワード認証プロトコル (PAP) が (<code>ppp authentication</code> コマンドを使用して) 設定されている場合、拒否パケットでの認証方式として PAP が提案されます。
ステップ 4	<code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none">• <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:<ul style="list-style-type: none">– yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。– no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。– cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。• 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、<code>commit</code> コマンドを使用します。
ステップ 5	<code>show running-config</code> 例: RP/0/RP0/CPU0:router# show running-config	PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。

インターフェイスでの MS-CHAP 認証のディセーブル化

ここでは、シリアル インターフェイスまたは POS インターフェイスで MS-CHAP 認証をディセーブルにする手順について説明します。

手順の概要

1. **configure**
2. **interface type interface-path-id**
3. **ppp ms-chap refuse**
4. **end**
または
commit
5. **show running-config**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface type interface-path-id 例： RP/0/RP0/CPU0:router(config)# interface serial 0/4/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ppp ms-chap refuse 例： RP/0/RP0/CPU0:router(config-if)# ppp ms-chap refuse	認証を要求するピアからの MS-CHAP 認証を拒否します。指定したインターフェイスで ppp chap refuse コマンドを入力すると、MS-CHAP を使用してユーザー認証を強制しようとしたピアの試行はすべて拒否されます。 <ul style="list-style-type: none"> • MS-CHAP 認証は、デフォルトではディセーブルに設定されています。 • 発信パスワード認証プロトコル (PAP) が (ppp authentication コマンドを使用して) 設定されている場合、拒否パケットでの認証方式として PAP が提案されます。

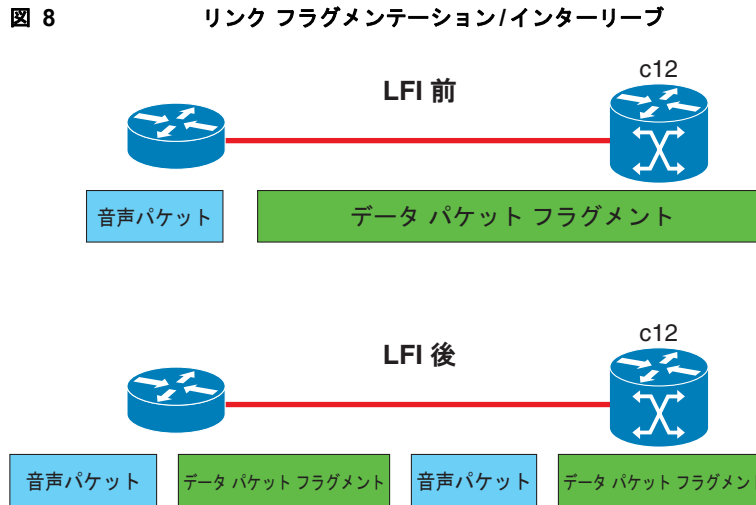
コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>show running-config</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show running-config</code></p>	<p>PPP カプセル化がイネーブルになっているインターフェイスの PPP 認証情報を確認します。</p>

マルチリンク PPP について

Multilink Point-to-Point Protocol (MLPPP; マルチリンク ポイントツーポイント プロトコル) は、複数の物理リンクを組み合わせることで 1 つの論理リンクを構成する機能を持ちます。Cisco XR 12000 シリーズ ルータ上に Cisco IOS XR を実装すると、複数の PPP インターフェイスが 1 つのマルチリンク インターフェイスに結合されます。MLPPP は、複数の PPP リンク間におけるデータグラムの分割、再構成、順序付けを行います。

Link Fragment Interleave (LFI; リンク フラグメンテーション/インターリーブ) は MLPPP インターフェイス用に設計されており、768 Kbps 未満の低速のインターフェイス上で音声とデータを統合する場合に必要になります。

LFI は、データと同じ回線上を移動する音声やビデオなど、遅延の影響を受けやすいトラフィックを安定させます。ネットワークが 768 Kbps 未満の低速のインターフェイス上で大きなパケットを処理しているときに遅延やジッタが生じると、音声は脆弱になります。LFI は、大きなデータグラムを分割（フラグメント）し、これらを低遅延のトラフィック パケットにインターリーブすることで、遅延やジッタを軽減します。



サポートされるカード

MLPPP は、次のラインカードおよび SPA でサポートされています。

- Cisco XR 12000 マルチサービス ラインカード
- 2 ポートおよび 4 ポート チャネライズド T3 SPA (SPA-2XCT3/DS0、SPA-4XCT3/DS0)

LFI は、以下でサポートされています。

- Cisco 1 ポート チャネライズド STM-1/OC-3 共有ポート アダプタ

機能の概要

Cisco IOS XR での MLPPP は、PPP シリアルインターフェイスでサポートされているのと同じ機能（ただし、QoS を除きます）を提供します。加えて、次の機能も提供します。

- フラグメント サイズ (128、256、512 バイト)
- 長いシーケンス番号 (24 ビット)
- 失われたフラグメントの検出タイムアウト時間 (80 ミリ秒)
- 最小アクティブ リンク設定オプション
- マルチリンク インターフェイスでの LCP エコー要求/応答サポート
- Full T1 および E1 のフレーム化されたリンクとフレーム化されていないリンク

制限事項

Cisco IOS XR ソフトウェア対応の MLPPP には、以下の制限事項があります。

- サポートされるのはフルレート T1 のみです。
- バンドルのすべてのリンクは、同じ SPA に属します。
- バンドルのすべてのリンクは、同じ速度で動作する必要があります。
- バンドルごとの最大リンク数は 12 です。

- 2ポート チャネライズド T3 SPA 上の最大バンドル数は 28 です。
- 4ポート チャネライズド T3 SPA 上の最大バンドル数は 56 です。
- ラインカードごとの最大バンドル数は 224 です。
- MLPPP バンドルのすべてのシリアル リンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MLPPP バンドルのメンバーとして設定する前に、シリアル インターフェイスで **mtu** コマンドを設定しないでください。Cisco IOS XR ソフトウェアは、以下をブロックします。
 - インターフェイスがデフォルト以外の MTU 値で設定されている場合、MLPPP バンドルのメンバーとしてシリアル インターフェイスを設定しようとする処理。
 - MLPPP バンドルのメンバーとして設定されているシリアル インターフェイスの **mtu** コマンド値を変更しようとする処理。



(注) PPP カプセル化を設定したインターフェイスで MTU 値を変更すると、回線プロトコルがフラップします。

Cisco IOS XR ソフトウェアでのマルチリンク処理は、マルチリンク コントローラと呼ばれるハードウェア モジュールによって制御されます。このコントローラは、ASIC、ネットワーク プロセッサ、CPU の連携動作で成り立ちます。MgmtMultilink コントローラにより、マルチリンク インターフェイスはチャネライズド SPA のシリアル インターフェイスのように動作します。

マルチリンク PPP の設定方法

ここでは、次の手順について説明します。

- [「コントローラの設定」\(P.339\)](#)
- [「インターフェイスの設定」\(P.342\)](#)
- [「MLPPP オプション機能の設定」\(P.344\)](#)

コントローラの設定

コントローラを設定するには、次の作業を行います。

手順の概要

1. **configure**
2. **controller type interface-path-id**
3. **mode type**
4. **clock source {internal | line}**
5. **exit**
6. **controller t1 interface-path-id**
7. **channel-group channel-group-number**
8. **timeslots range**
9. **exit**

■ マルチリンク PPP の設定方法

10. `exit`
11. `controller mgmtmultilink interface-path-id`
12. `bundle bundle-id`
13. `end`
または
`commit`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例： RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller type interface-path-id</code> 例： RP/0/0/CPU0:router (config)# <code>controller t3 0/1/0/0</code>	コントローラ コンフィギュレーション サブモードを開始して、コントローラ名とインスタンス ID を <code>rack/slot/module/port</code> 表記で指定します。
ステップ 3	<code>mode type</code> 例： RP/0/0/CPU0:router# <code>mode t1</code>	チャネライズするマルチリンクのタイプを設定します (たとえば、28 T1)。
ステップ 4	<code>clock source {internal line}</code> 例： RP/0/0/CPU0:router (config-t3)# <code>clock source internal</code>	(任意) ポートのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。
ステップ 5	<code>exit</code> 例： RP/0/0/CPU0:router (config-t3)# <code>exit</code>	コントローラのコンフィギュレーション モードを終了します。
ステップ 6	<code>controller t1 interface-path-id</code> 例： RP/0/0/CPU0:router (config)# <code>controller t1 0/1/0/0/0</code>	T1 コンフィギュレーション モードを開始します。
ステップ 7	<code>channel-group channel-group-number</code> 例： RP/0/0/CPU0:router (config-t1)# <code>channel-group 0</code>	T1 チャネル グループを作成し、そのチャネル グループのチャネル グループ コンフィギュレーション モードを開始します。チャネル グループ番号は、0 ~ 23 の範囲で設定できます。

コマンドまたはアクション	目的
<p>ステップ 8 <code>timeslots range</code></p> <p>例: RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 7-12</p>	<p>1 つまたは複数の DS0 タイムスロットをチャンネルグループに関連付け、関連付けたシリアル サブインターフェイスをそのチャンネルグループに作成します。</p> <ul style="list-style-type: none"> • 範囲は 1 ~ 24 タイムスロットです。 • 24 タイムスロットすべてを単一のチャンネルグループに割り当てることも、タイムスロットを複数のチャンネルグループに分割することもできます。 <p>(注) タイムスロットの範囲は、1 ~ 24 にする必要があります。これは、結果として構築されるシリアルインターフェイスが MLPPP バンドルに受け入れられるようにするためです。</p>
<p>ステップ 9 <code>exit</code></p> <p>例: RP/0/0/CPU0:router(config-t1-channel_group)# exit</p>	<p>チャンネルグループ コンフィギュレーション モードを終了します。</p>
<p>ステップ 10 <code>exit</code></p> <p>例: RP/0/0/CPU0:router(config-t1)# exit</p>	<p>T1 コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。</p>
<p>ステップ 11 <code>controller mgmtmultilink interface-path-id</code></p> <p>例: RP/0/0/CPU0:router(config)# controller mgmtmultilink 0/1/0/0</p>	<p>マルチリンク インターフェイスの管理用にコントローラ コンフィギュレーション サブモードを開始します。コントローラ名とインスタンス ID を <i>rack/slot/module/port</i> 表記で指定します。</p>

コマンドまたはアクション	目的
ステップ 12 <code>bundle bundle-id</code> 例 : RP/0/0/CPU0:router (config-mgmtmultilink) # <code>bundle 20</code>	指定したバンドル ID でマルチリンク インターフェイスを作成します。
ステップ 13 <code>end</code> または <code>commit</code> 例 : RP/0/0/CPU0:router (config-t3) # <code>end</code> または RP/0/0/CPU0:router (config-t3) # <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

インターフェイスの設定

インターフェイスを設定するには、次の作業を行います。

制約事項

- MLPPP バンドルのすべてのシリアルリンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MLPPP バンドルのメンバーとして設定する前に、シリアルインターフェイスで **mtu** コマンドを設定しないでください。Cisco IOS XR ソフトウェアは、以下をブロックします。
 - インターフェイスがデフォルト以外の MTU 値で設定されている場合、MLPPP バンドルのメンバーとしてシリアルインターフェイスを設定しようとする処理。
 - MLPPP バンドルのメンバーとして設定されているシリアルインターフェイスの **mtu** コマンド値を変更しようとする処理。



(注) PPP カプセル化を設定したインターフェイスで MTU 値を変更すると、回線プロトコルがフラップします。

手順の概要

1. **configure**
2. **interface multilink** *interface-path-id*
3. **ipv4 address** *address/mask*
4. **multilink fragment-size** *size*
5. **keepalive** {*interval* | **disable**}
6. **exit**
7. **interface type** *interface-path-id*
8. **encapsulation type**
9. **multilink group** *group-id*
10. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface multilink <i>interface-path-id</i> 例： RP/0/0/CPU0:router(config)# interface multilink 0/1/0/0/1	マルチリンク インターフェイス名とインスタンス ID を <i>rack/slot/module/port/bundle-id</i> 表記で指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ipv4 address <i>ip-address</i> 例： RP/0/0/CPU0:router(config-if)# ipv4 address 80.170.0.1/24	次の形式でインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 <i>A.B.C.D/prefix</i> or <i>A.B.C.D/mask</i>
ステップ 4	multilink fragment-size <i>size</i> 例： RP/0/0/CPU0:router(config-if)# multilink fragment-size 128	(任意) マルチリンク フラグメントのサイズを指定します (128 バイトなど)。フラグメント サイズによっては、サポートされない場合があります。 デフォルトは no fragments です。
ステップ 5	keepalive { <i>seconds</i> disable }	リンク制御プロトコル (LCP) がピアに ECHOREQ を送信する頻度 (秒) を指定します。デフォルトのキープアライブ インターバルは 10 秒です。 システムをデフォルトのキープアライブ インターバルに戻すには、 no keepalive コマンドを使用します。 キープアライブ タイマーをディセーブルにするには、 keepalive disable コマンドを使用します。

■ マルチリンク PPP の設定方法

コマンドまたはアクション	目的
ステップ 6 <code>exit</code> 例: RP/0/0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 7 <code>interface type interface-path-id</code> 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/1:0	インターフェイス名とインスタンス ID を <i>rack/slot/module/port/t1-number:channel-group</i> 表記で指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8 <code>encapsulation type</code> 例: RP/0/0/CPU0:router(config-if)# encapsulation ppp	カプセル化のタイプを指定します。ここでは、PPP を指定します。 (注) PPP は、Cisco IOS XR リリース 3.4.1 以降でのみサポートされています。
ステップ 9 <code>multilink group group-id</code> 例: RP/0/0/CPU0:router(config-if)# multilink group 1	このインターフェイスのマルチリンク グループ ID を指定します。
ステップ 10 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

MLPPP オプション機能の設定

次のいずれかのオプション機能を設定するには、次のタスクを実行します。

- アクティブ リンクの最大数
- マルチリンク インターリーブ



(注) アクティブ リンクの最大数は、両方のエンドポイントで設定する必要があります。

手順の概要

1. **configure**
2. **interface multilink *interface-path-id***
3. **multilink**
4. **ppp multilink minimum-active links *value***
5. **multilink interleave**
6. **no shutdown**
7. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface multilink <i>interface-path-id</i> 例: RP/0/0/CPU0:router(config)# interface multilink 0/1/0/0/1	マルチリンク インターフェイス名とインスタンス ID を <i>rack/slot/module/port/bundle-id</i> 表記で指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	multilink 例: RP/0/0/CPU0:router(config-if)# multilink	インターフェイス マルチリンク コンフィギュレーション モードを開始します。
ステップ 4	ppp multilink minimum-active links <i>value</i> 例: RP/0/0/CPU0:router(config-if-multilink)# ppp multilink minimum-active links 12	(任意) マルチリンク インターフェイスのアクティブ リンクの最小数を指定します。
ステップ 5	multilink interleave 例: RP/0/0/CPU0:router(config-if-multilink)# multilink interleave	(任意) マルチリンク インターフェイスでインターリーブをイネーブルにします。

コマンドまたはアクション	目的
ステップ 6 <code>no shutdown</code> 例: RP/0/0/CPU0:router(config-if-mutlilink)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> • <code>shutdown</code> 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 7 <code>end</code> または commit 例: RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

PPP の設定例

ここでは、次の設定例について説明します。

- 「[POS インターフェイスでの PPP カプセル化の設定：例](#)」 (P.346)
- 「[シリアルインターフェイスでの PPP カプセル化の設定：例](#)」 (P.347)
- 「[マルチリンク PPP 設定の確認](#)」 (P.348)

POS インターフェイスでの PPP カプセル化の設定：例

次に、POS インターフェイスを作成し、PPP カプセル化を設定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# encapsulation ppp
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# ppp pap sent-username P1_CRS-8 password xxxx
RP/0/RP0/CPU0:router(config-if)# ppp authentication chap pap MIS-access
RP/0/RP0/CPU0:router(config-if)# ppp chap password encrypted xxxx
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

次に、最初の認証が失敗した後に 2 回リトライできる（認証が失敗した場合に全部で 3 回リトライできる）ように POS インターフェイス 0/3/0/1 を設定する例を示します。

```
RP/0/RP0/CPU0:router# configuration
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/1
RP/0/RP0/CPU0:router(config-if)# ppp max-bad-auth 3
```

シリアル インターフェイスでの PPP カプセル化の設定 : 例

次に、PPP MS-CHAP をカプセル化したシリアル インターフェイスを作成して設定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface serial 0/3/0/0:0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# encapsulation ppp
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# ppp authentication ms-chap MIS-access
RP/0/RP0/CPU0:router(config-if)# ppp ms-chap password encrypted xxxx
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

MLPPP の設定 : 例

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# controller t3 0/1/0/0
RP/0/0/CPU0:router# mode t1
RP/0/0/CPU0:router(config-t3)# clock source internal
RP/0/0/CPU0:router(config-t3)# exit
RP/0/0/CPU0:router(config)# controller t1 0/1/0/0/0
RP/0/0/CPU0:router(config-t1)# channel-group 0
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 7-12
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1)# exit
RP/0/0/CPU0:router(config)# controller mgmtmultilink 0/1/0/0
RP/0/0/CPU0:router(config-mgmtmultilink)# bundle 20
RP/0/0/CPU0:router(config-t3)# commit
RP/0/0/CPU0:router(config-t3)# exit

RP/0/0/CPU0:router(config)# interface multilink 0/1/0/0/1
RP/0/0/CPU0:router(config-if)# ipv4 address 80.170.0.1/24
RP/0/0/CPU0:router(config-if)# multilink fragment-size 128
RP/0/0/CPU0:router(config-if)# keepalive disable
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config)# interface serial 0/1/0/0/1:0
RP/0/0/CPU0:router(config-if)# encapsulation ppp
RP/0/0/CPU0:router(config-if)# group 1
RP/0/0/CPU0:router(config-t3)# commit
RP/0/0/CPU0:router(config-t3)# exit

RP/0/0/CPU0:router(config)# interface multilink 0/1/0/0/1
RP/0/0/CPU0:router(config-if)# multilink
RP/0/0/CPU0:router(config-if-multilink)# ppp multilink minimum-active links 12
RP/0/0/CPU0:router(config-if-multilink)# multilink interleave
RP/0/0/CPU0:router(config-if-multilink)# no shutdown
RP/0/0/CPU0:router(config-t3)# commit
```

マルチリンク PPP 設定の確認

次のコマンドを使用して、マルチリンク設定を確認し、トラブルシューティングを行うことができます。

- 「[show multilink interfaces : 例](#)」 (P.348)
- 「[show ppp interfaces multilink : 例](#)」 (P.349)
- 「[show ppp interface serial : 例](#)」 (P.349)
- 「[show imds interface multilink : 例](#)」 (P.349)

show multilink interfaces : 例

```
RP/0/0/CPU0:Router# show multilink interfaces multilink 0/3/1/0/301
```

```
Multilink0/3/1/0/301 is up, line protocol is up
```

```
Fragmentation: disabled
Member Links: 2 active, 0 inactive
- Serial0/3/1/0/0:0: ACTIVE
- Serial0/3/1/0/1:0: ACTIVE
```

```
RRP/0/0/CPU0:Router# show multilink interfaces
```

```
Multilink0/3/1/0/301 is up, line protocol is up
```

```
Fragmentation: disabled
Member Links: 2 active, 0 inactive
- Serial0/3/1/0/0:0: ACTIVE
- Serial0/3/1/0/1:0: ACTIVE
```

```
Multilink0/3/1/0/302 is up, line protocol is up
```

```
Fragmentation: disabled
Member Links: 2 active, 0 inactive
- Serial0/3/1/1/1:0: ACTIVE
- Serial0/3/1/1/0:0: ACTIVE
```

```
Serial0/3/1/0/0:0 is up, line protocol is up
Multilink group id: 301
Member status: ACTIVE
```

```
Serial0/3/1/1/0:0 is up, line protocol is up
Multilink group id: 302
Member status: ACTIVE
```

```
Serial0/3/1/0/1:0 is up, line protocol is up
Multilink group id: 301
Member status: ACTIVE
```

```
Serial0/3/1/1/1:0 is up, line protocol is up
Multilink group id: 302
Member status: ACTIVE
```

show ppp interfaces multilink : 例

```
RP/0/0/CPU0:Router# show ppp interfaces multilink 0/3/1/0/1

Multilink 0/3/1/0/1 is up, line protocol is up
LCP: Open
  Keepalives disabled
IPCP: Open
  Local IPv4 address: 1.1.1.2
  Peer IPv4 address: 1.1.1.1
Multilink
  Member Links: 2 active, 1 inactive (min-active 1)
    - Serial0/3/1/0/0:0: ACTIVE
    - Serial0/3/1/0/1:0: ACTIVE
    - Serial0/3/1/0/2:0: INACTIVE : LCP has not been negotiated
```

show ppp interface serial : 例

```
RP/0/0/CPU0:Router# show ppp interface Serial 0/3/1/0/0:0

Serial 0/3/1/0/0:0 is up, line protocol is up
LCP: Open
  Keepalives disabled
  Local MRU: 1500 bytes
  Peer MRU: 1500 bytes
  Local Bundle MRRU: 1596 bytes
  Peer Bundle MRRU: 1500 bytes
  Local Endpoint Discriminator: 1b61950e3e9ce8172c8289df0000003900000001
  Peer Endpoint Discriminator: 7d046cd8390a4519087aefb90000003900000001
Authentication
  Of Peer: <None>
  Of Us: <None>
Multilink
  Multilink group id: 1
  Member status: ACTIVE
```

show imds interface multilink : 例

```
RP/0/0/CPU0:Router# show imds interface Multilink 0/3/1/0/1

IMDS INTERFACE DATA (Node 0x0)

Multilink0_3_1_0_1 (0x04001200)
-----
flags: 0x0001002f   type: 55 (IFT_MULTILINK)   encap: 52 (ppp)
state: 3 (up)      mtu: 1600   protocol count: 3
control parent: 0x04000800   data parent: 0x00000000
  protocol          capsulation          state          mtu
-----
12 (ipv4)
      26 (ipv4)        3 (up)        1500
      47 (ipcp)       3 (up)        1500
16 (ppp_ctrl)
      53 (ppp_ctrl)   3 (up)        1500
0 (Unknown)
      139 (c_shim)    3 (up)        1600
      52 (ppp)        3 (up)        1504
      56 (queue_fifo) 3 (up)        1600
      60 (txm_nopull) 3 (up)        1600
```

その他の参考資料

ここでは、PPP カプセル化に関する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システム ブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
この機能によりサポートされた新規 MIB または改訂 MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
RFC-1661	<i>The Point-to-Point Protocol (PPP)</i>
RFC- 1994	<i>PPP Challenge Handshake Authentication Protocol (CHAP)</i>

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの物理インターフェイスのプリコンフィギュレーション

ここでは、Cisco IOS XR ソフトウェアをサポートするルータでの物理インターフェイスのプリコンフィギュレーションについて説明します。

プリコンフィギュレーションは、Cisco CRS-1 ルータでは次のインターフェイスでサポートされています。

- 管理イーサネット
- ギガビット イーサネット
- 10 ギガビット イーサネット
- Packet-over-SONET/SDH (POS)
- Spatial Reuse Protocol (SRP)

プリコンフィギュレーションは、Cisco XR 12000 シリーズ ルータでは次のインターフェイスでサポートされています。

- 管理イーサネット
- ギガビット イーサネット
- 10 ギガビット イーサネット
- ファスト イーサネット
- POS
- シリアル
- ATM

プリコンフィギュレーションによって、モジュラ サービス カードをルータへの装着前に設定できます。カードを装着すると、ただちに設定されます。

プリコンフィギュレーション情報は、通常の方法で設定されたインターフェイスの場合とは異なり、別のシステム データベース ツリー ルート プロセッサ (RP) 上のプリコンフィギュレーション ディレクトリ) に作成されます。

検証機能が動作するのはモジュラ サービス カード上に限られるため、モジュラ サービス カードが存在していなければ検証できないプリコンフィギュレーション データもあります。このようなプリコンフィギュレーション データは、モジュラ サービス カードを装着し、検証機能が起動したときに検証されます。設定がプリコンフィギュレーション領域からアクティブ領域にコピーされるときにエラーが検出されると、設定は拒否されます。



(注)

プリコンフィギュレーションを実行できるのは物理インターフェイスだけです。

Cisco IOS XR ソフトウェアでの物理インターフェイスのプリコンフィギュレーション機能の履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータに POS のプリコンフィギュレーション機能が追加されました。
リリース 3.0	Cisco CRS-1 ルータにイーサネットのプリコンフィギュレーション機能が追加されました。
リリース 3.2	Cisco XR 12000 シリーズ ルータのサポートが追加されました。
リリース 3.3.0	Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータに管理イーサネットのプリコンフィギュレーション機能が追加されました。 Cisco XR 12000 シリーズ ルータにファストイーサネットおよびシリアルインターフェイスのプリコンフィギュレーション機能が追加されました。 Cisco CRS-1 ルータ上で、4 ポート OC-192c/STM-64c POS/DPT PLIM について SRP インターフェイスのプリコンフィギュレーション機能が追加されました。
リリース 3.4.0	このモジュールの設定手順が拡張されました。 Cisco XR 12000 シリーズ ルータに ATM インターフェイスのプリコンフィギュレーション機能が追加されました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- ・「物理インターフェイスのプリコンフィギュレーションを行うための前提条件」(P.354)
- ・「物理インターフェイスのプリコンフィギュレーションに関する情報」(P.355)
- ・「物理インターフェイスのプリコンフィギュレーションを行う方法」(P.357)
- ・「物理インターフェイスのプリコンフィギュレーション例」(P.358)
- ・「関連情報」(P.359)
- ・「その他の参考資料」(P.359)

物理インターフェイスのプリコンフィギュレーションを行うための前提条件

物理インターフェイスのプリコンフィギュレーションを行う前に、次の条件が満たされていることを確認してください。

- ・ この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンドタスク ID は、各コマンドリファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

- プリコンフィギュレーション ドライバおよびファイルがインストールされている必要があります。プリコンフィギュレーション ドライバがインストールされていなくても物理インターフェイスのプリコンフィギュレーションを行える場合もありますが、ルータ上で有効なインターフェイス名の文字列を提供するインターフェイス定義ファイルを設定するには、プリコンフィギュレーション ファイルが必要です。

物理インターフェイスのプリコンフィギュレーションに関する情報

インターフェイスのプリコンフィギュレーションを行うには、次の概念を理解しておく必要があります。

- 「物理インターフェイスのプリコンフィギュレーションの概要」(P.355)
- 「インターフェイスのプリコンフィギュレーションを行う利点」(P.356)
- 「インターフェイス プリコンフィギュレーション コマンドの使用法」(P.356)
- 「アクティブ/スタンバイ RP および仮想インターフェイスの設定」(P.356)

物理インターフェイスのプリコンフィギュレーションの概要

プリコンフィギュレーションは、インターフェイスがシステムに存在しないうちにインターフェイスを設定する作業です。プリコンフィギュレーションされたインターフェイスは、位置（ラック/スロット/モジュール）が一致するインターフェイスが実際にルータに装着されるまで検証または適用されません。適切なモジュラ サービス カードが装着され、インターフェイスが作成されると、事前に作成された設定情報が検証され、問題がなければ、ただちにルータの実行コンフィギュレーションに適用されます。



(注) 適切なモジュラ サービス カードを装着するときには、適切な **show** コマンドを使用してプリコンフィギュレーションの内容を検証してください。

プリコンフィギュレーション済みの状態にあるインターフェイスを表示するには、**show run** コマンドを使用します。



(注) カードを装着し、インターフェイスをアップ状態にするときに、想定される設定と実際にプリコンフィギュレーションされたインターフェイスを比較できるように、サイト プランニング ガイドにプリコンフィギュレーション情報を記入することをお勧めします。



ヒント

プリコンフィギュレーションを実行コンフィギュレーション ファイルに保存するには、**commit best-effort** コマンドを使用します。**commit best-effort** コマンドは、ターゲット コンフィギュレーションと実行コンフィギュレーションを結合し、有効な設定だけをコミットします（ベストエフォート）。セマンティック エラーにより一部の設定が適用されないこともあります。その場合でも有効な設定はアップ状態になります。

インターフェイスのプリコンフィギュレーションを行う利点

プリコンフィギュレーションによって、新しいカードをシステムに追加するときのダウンタイムが短縮されます。プリコンフィギュレーションを行うと、新しいモジュラ サービス カードが即座に設定され、カードのブートアップ中も動作します。

プリコンフィギュレーションを行うもう 1 つの利点は、モジュラ サービス カードの交換時に、カードを取り外した後でも、以前の設定を表示し、変更できることです。

インターフェイス プリコンフィギュレーション コマンドの使用法

システムにまだ存在しないインターフェイスのプリコンフィギュレーションを行うには、グローバル コンフィギュレーション モードで **interface preconfigure** コマンドを使用します。

interface preconfigure コマンドによって、ルータはインターフェイス コンフィギュレーション モードに移行します。ユーザは、使用可能なすべてのコマンドを追加できます。プリコンフィギュレーションされたインターフェイス用に登録された検証機能により、設定が検証されます。ユーザが **end** コマンドを入力するか、それに対応する **exit** コマンドやグローバル コンフィギュレーション コマンドを入力すると、プリコンフィギュレーションが完了します。



(注)

モジュラ サービス カードを装着しなければ検証できない設定もあります。



(注)

新たにプリコンフィギュレーションされたインターフェイスには **no shutdown** コマンドを入力しないでください。このコマンドの **no** 形式は既存の設定を削除するものであり、この場合は既存の設定が存在しないからです。

ユーザがプリコンフィギュレーション時に指定する名前は、作成するインターフェイスの名前と一致する必要があります。インターフェイス名が一致しない場合、インターフェイスの作成時にプリコンフィギュレーションを適用できません。インターフェイス名は、ルータがサポートし、対応するドライバがインストール済みのインターフェイス タイプから始めます。ただし、スロット、ポート、サブインターフェイス番号、およびチャネル インターフェイス番号の情報は検証できません。



(注)

すでに存在し、設定されているインターフェイス名（または e0/3/0/0 のような省略形）は指定できません。

アクティブ/スタンバイ RP および仮想インターフェイスの設定

スタンバイ RP は、必要時に使用可能になり、アクティブ RP から作業を引き継げる状態になります。スタンバイ RP がアクティブ RP となり、アクティブ RP の役割を引き継ぐ必要のある状況を次に示します。

- ウォッチドッグによる障害検出
- スタンバイ RP に対する管理上の引き継ぎ命令
- シャーシからのアクティブ RP の取り外し

セカンダリ RP がシャーシに搭載されていなかった場合、プライマリ RP の稼動中にセカンダリ RP を搭載すると、自動的にスタンバイ RP になります。シャーシからスタンバイ RP を取り外しても、RP の冗長性が失われるだけで、システムに影響はありません。

スイッチオーバー後、すべての仮想インターフェイスはスタンバイ（新たにアクティブになった）RP に存在します。仮想インターフェイスのステートと設定は変更されず、スイッチオーバー時にインターフェイス経由の転送（トンネルの場合）が失われることはありません。Cisco CRS-1 ルータは、ホスト RP のスイッチオーバーを通じて、トンネル上で無停止転送（NSF）を使用します。



(注)

スタンバイ インターフェイス設定の維持されることを保証するために、設定は特に必要ありません。

物理インターフェイスのプリコンフィギュレーションを行う方法

ここでは、インターフェイスの最も基本的なプリコンフィギュレーションについてのみ説明します。

手順の概要

1. **configure**
2. **interface preconfigure type interface-path-id**
3. **ipv4 address ip-address subnet-mask**
4. 追加のインターフェイス パラメータを設定します。
5. **end**
または
commit
6. **exit**
7. **exit**
8. **show running-config**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface preconfigure type interface-path-id 例： RP/0/RP0/CPU0:router(config)# interface preconfigure MgmtEth 0/RP0/CPU0/0	インターフェイスに対するインターフェイス プリコンフィギュレーション モードを開始します。 <i>type</i> は、設定するインターフェイスのタイプに置き換えてください。 <i>interface-path-id</i> は、 <i>rack/slot/module/port</i> 表記のインターフェイス ID に置き換えてください。
ステップ 3	ipv4 address ip-address subnet-mask 例： RP/0/RP0/CPU0:router(config-if-pre)# ip address 192.168.255.255/32	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。

コマンドまたはアクション	目的
ステップ4 -	追加のインターフェイス パラメータを設定します。詳細については、設定するインターフェイスのタイプに対応する、このマニュアルの設定モジュールを参照してください。
ステップ5 <code>exit</code>	インターフェイス プリコンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ6 <code>exit</code>	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ7 <code>end</code> または <code>commit best-effort</code> 例: RP/0/RP0/CPU0:router(config-if-pre)# end または RP/0/RP0/CPU0:router(config-if-pre)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、次に示す変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit best-effort コマンドを使用します。commit best-effort コマンドは、ターゲット コンフィギュレーション と実行コンフィギュレーション を結合し、有効な変更だけをコミットします (ベストエフォート)。セマンティック エラーにより、一部の設定変更がコミットされないことがあります。
ステップ8 <code>show running-config</code> 例: RP/0/RP0/CPU0:router# show running-config	(任意) 現在ルータで使用されている設定情報を表示します。

物理インターフェイスのプリコンフィギュレーション例

ここでは、次の例について説明します。

[「インターフェイスのプリコンフィギュレーション：例」 \(P.358\)](#)

インターフェイスのプリコンフィギュレーション：例

次に、基本的なイーサネット インターフェイスのプリコンフィギュレーション例を示します。


```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface preconfigure TenGigE 0/1/0/0.1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.255.255/32
RP/0/RP0/CPU0:router(config-if)# commit
```

関連情報

- 管理イーサネット インターフェイスの設定の詳細については、『*Cisco IOS XR Getting Started Guide*』の「*Configuring General Router Features*」モジュールを参照してください。管理イーサネット インターフェイスの詳細設定については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの管理イーサネット インターフェイスの高度な設定と変更](#)」モジュールを参照してください。
- ギガビット イーサネット インターフェイスおよびファスト イーサネット インターフェイスの設定方法については、このマニュアルの「[Cisco IOS XR ソフトウェアでのイーサネット インターフェイスの設定](#)」モジュールを参照してください。
- POS インターフェイスの設定方法については、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- シリアル インターフェイスの設定方法については、このマニュアルの「[Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定](#)」モジュールを参照してください。
- ATM インターフェイスの設定方法については、このマニュアルの「[Cisco IOS XR ソフトウェアでの ATM インターフェイスの設定](#)」モジュールを参照してください。

その他の参考資料

ここでは、物理インターフェイスのプリコンフィギュレーションに関連する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『 <i>Cisco IOS XR Master Commands List</i> 』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『 <i>Cisco IOS XR Interface and Hardware Component Command Reference</i> 』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『 <i>Cisco IOS XR Getting Started Guide</i> 』
ユーザ グループとタスク ID に関する情報	『 <i>Cisco IOS XR System Security Configuration Guide</i> 』の「 <i>Configuring AAA Services on Cisco IOS XR Software</i> 」モジュール
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『 <i>Cisco Craft Works Interface Configuration Guide</i> 』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータ上のシリアル インターフェイスについて説明します。シリアル インターフェイスについて設定する前に、そのインターフェイスと関連付けられたクリア チャネル T3/E3 コントローラまたはチャネライズド T1/E1 コントローラ（DS0 チャネル）を設定する必要があります。

シリアル コントローラ インターフェイス設定の機能履歴

リリース	変更点
リリース 3.3.0	<p>Cisco XR 12000 シリーズ ルータにこの機能が追加されました。</p> <p>次のハードウェアについて、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p> <ul style="list-style-type: none">• Cisco XR 12000 SIP-401• Cisco XR 12000 SIP-501• Cisco XR 12000 SIP-601 <p>次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p> <ul style="list-style-type: none">• 2 ポートおよび 4 ポート チャネライズド T3 シリアル SPA• 2 ポートおよび 4 ポート T3/E3 シリアル SPA
リリース 3.4.0	<p>次の機能のサポートが導入されました。</p> <ul style="list-style-type: none">• 相手先固定接続（PVC）とのサブインターフェイス• 次のハードウェア上のシリアル メイン インターフェイスおよび PVC でのフレームリレー カプセル化<ul style="list-style-type: none">– 8 ポート チャネライズド T1/E1 シリアル SPA– 2 ポートおよび 4 ポート チャネライズド T3 シリアル SPA– 2 ポートおよび 4 ポート T3/E3 シリアル SPA– 1 ポート チャネライズド OC-3 SPA– 1 ポート チャネライズド OC-12 SPA– 1 ポート チャネライズド OC-48 SPA– 1 ポート チャネライズド OC-12/STM-4 ISE ラインカード

リリース 3.4.1	<p>Cisco CRS-1 ルータにこの機能が追加されました。</p> <p>次のハードウェアについて、Cisco CRS-1 ルータでのサポートが追加されました。</p> <ul style="list-style-type: none"> • Cisco CRS-1 SIP-800 • 2 ポートおよび 4 ポート T3/E3 シリアル SPA <p>マルチリンク PPP が Cisco XR 12000 シリーズ ルータ上のシリアル インターフェイスでサポートされました。</p>
リリース 3.5.0	<p>次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。</p> <ul style="list-style-type: none"> • 1 ポート チャネライズド OC-12/DS0 SPA • 1 ポート チャネライズド OC-48/DS3 SPA
リリース 3.6.0	変更ありません。
リリース 3.7.0	Cisco XR 12000 シリーズ ルータ上で、1 ポート チャネライズド OC-48/DS3 ラインカードのサポートが追加されました。
リリース 3.8.0	<p>Cisco XR 12000 シリーズ ルータ上で、レイヤ 2 サブインターフェイス ファイルおよび次のラインカードのサービス品質 (QoS) のサポートが追加されました。</p> <ul style="list-style-type: none"> • 1 ポート チャネライズド OC-12/DS0 ラインカード • 4 ポート チャネライズド OC-12/DS3 ラインカード

この章の構成

- 「シリアル インターフェイスを設定するための前提事項」 (P.362)
- 「シリアル インターフェイスに関する情報」 (P.363)
- 「シリアル インターフェイスの概念」 (P.366)
- 「シリアル インターフェイスの設定方法」 (P.372)
- 「シリアル インターフェイスの設定例」 (P.390)
- 「その他の参考資料」 (P.393)

シリアル インターフェイスを設定するための前提事項

シリアル インターフェイスを設定する前に、次のタスクと条件を満たしていることを確認します。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 使用しているハードウェアは、T3/E3 または T1/E1 コントローラおよびシリアル インターフェイスをサポートする必要があります。次のハードウェアが、T3/E3 コントローラおよびシリアル インターフェイスをサポートしていることを確認します（次の一覧のすべてのハードウェアは Cisco XR 12000 シリーズ ルータでサポートされます。2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA だけが Cisco CRS-1 ルータ上でサポートされます）。
 - 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA
 - 2 ポートおよび 4 ポート チャネライズド T3 SPA
 - 4 ポート チャネライズド OC-12/DS3 ラインカード
 - 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード
- 次のハードウェアが、T1/E1 コントローラおよび DS0 チャネルをサポートしていることを確認します。
- 2 ポートおよび 4 ポート チャネライズド T3 SPA
 - 4 ポート チャネライズド OC-12/DS3 ラインカード
 - 1 ポート チャネライズド OC-12/DS0 SPA およびラインカード
 - 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード
- このマニュアルの「[Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよび チャネライズド T3 コントローラの設定](#)」モジュールの説明に従って、設定するシリアル インターフェイスと関連付けるクリア チャネル T3/E3 コントローラまたはチャネライズド T3-to-T1/E1 コントローラを設定済みです。

シリアル インターフェイスに関する情報

Cisco XR 12000 シリーズ ルータは、次のハードウェアでシリアル インターフェイスをサポートしています。

- 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA
- 2 ポートおよび 4 ポート チャネライズド T3 SPA
- 4 ポート チャネライズド OC-12/DS3 ラインカード
- 1 ポート チャネライズド OC-12/DS0 SPA およびラインカード
- 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード

Cisco CRS-1 ルータは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA でシリアル インターフェイスをサポートしています。



(注) 2 ポートおよび 4 ポート チャネライズド T3 SPA は、クリア チャネル モードで実行できます。または、28 T1 コントローラか 21 E1 コントローラにチャネライズドできます。

チャネライズド T3-to-T1/E1 コントローラの場合、ユーザが T1/E1 コントローラの各 DS0 チャネル グループを設定すると、シリアル インターフェイスが自動的に作成されます。

シリアル コントローラ インターフェイスを設定するには、次の概念を理解しておく必要があります。

- 「[概要：クリア チャネル SPA 上のシリアル インターフェイスの設定](#)」(P.364)
- 「[概要：チャネライズド SPA 上のシリアル インターフェイスの設定](#)」(P.365)
- 「[シリアル インターフェイス コンフィギュレーションのデフォルト設定](#)」(P.371)
- 「[シリアル インターフェイスの表記方法](#)」(P.371)

- 「PPP カプセル化」 (P.366)
- 「Cisco HDLC カプセル化」 (P.366)
- 「キープアライブ タイマー」 (P.368)

概要 : クリア チャネル SPA 上のシリアル インターフェイスの設定

表 12 は、Cisco XR 12000 シリーズ ルータおよび Cisco CRS-1 ルータでサポートされる 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA 上に T3 シリアル インターフェイスを設定するために必要なタスクの概要です。

表 12 概要 : クリア チャネル SPA 上の T3 シリアル インターフェイスの設定

ステップ	タスク	モジュール	項
1.	必要に応じて、 hw-module subslot コマンドを使用し、SPA のシリアル モードを T3 に設定します。 (注) デフォルトで、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA は T3 モードで実行するように設定されています。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「カード タイプの設定」
2.	T3 コントローラを設定します。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「カード タイプの設定」
3.	ステップ 2 で設定した T3 コントローラに関連付けるシリアル インターフェイスを設定します。	「Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定」	「シリアル インターフェイスの設定方法」

表 13 は、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA 上に E3 シリアル インターフェイスを設定するために必要なタスクの概要です。

表 13 概要 : クリア チャネル SPA 上の E3 シリアル インターフェイスの設定

ステップ	タスク	モジュール	項
1.	hw-module subslot コマンドを使用し、SPA のシリアル モードを E3 に設定します。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「カード タイプの設定」
2.	E3 コントローラを設定します。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「カード タイプの設定」
3.	ステップ 2 で設定した E3 コントローラに関連付けるシリアル インターフェイスを設定します。	「Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定」	「シリアル インターフェイスの設定方法」

概要：チャネライズド SPA 上のシリアル インターフェイスの設定

表 14 は、Cisco XR 12000 シリーズ ルータでサポートされる次の SPA およびラインカード上に、T1 シリアル インターフェイスを設定するために必要な概要です。

- 2 ポートおよび 4 ポート チャネライズド T3 SPA
- 4 ポート チャネライズド OC-12/DS3 ラインカード
- 1 ポート チャネライズド OC-12/DS0 SPA およびラインカード
- 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード

表 14 概要：T1 DS0 チャネル上のシリアル インターフェイスの設定

ステップ	タスク	モジュール	項
1.	T3 コントローラ パラメータを設定し、SPA モードを T3 に設定します。 28 T1 コントローラが自動的に作成されます。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「カード タイプの設定」
2.	ステップ 1 で作成した T1 コントローラ上に、DS0 チャネル グループを作成し、設定します。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「T1 コントローラの設定」
3.	ステップ 2 で作成したチャネル グループと関連付けられたシリアル インターフェイスを設定します。	「Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定」	「シリアル インターフェイスの設定方法」

表 15 は、Cisco XR 12000 シリーズ ルータでサポートされる次の SPA およびラインカード上に、E1 シリアル インターフェイスを設定するために必要な概要です。

- 2 ポートおよび 4 ポート チャネライズド T3 SPA
- 4 ポート チャネライズド OC-12/DS3 ラインカード
- 1 ポート チャネライズド OC-12/DS0 SPA およびラインカード
- 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード

表 15 概要：E1 DS0 チャネル上のシリアル インターフェイスの設定

ステップ	タスク	モジュール	項
1.	E3 コントローラ パラメータを設定し、SPA モードを T3 に設定します。 21 E1 コントローラが自動的に作成されます。	「Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「チャネル化された T3 コントローラの設定」

表 15 概要 : E1 DS0 チャンネル上のシリアル インターフェイスの設定 (続き)

ステップ	タスク	モジュール	項
2.	ステップ 1 で作成した E1 コントローラ上に、DS0 チャンネル グループを作成し、設定します。	「Cisco IOS XR ソフトウェアでのクリア チャンネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定」	「E1 コントローラの設定」
3.	ステップ 2 で作成したチャンネル グループと関連付けられたシリアル インターフェイスを設定します。	「Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定」	「シリアル インターフェイスの設定方法」

シリアル インターフェイスの概念

シリアル コントローラ インターフェイスを設定するには、次の概念を理解しておく必要があります。

- [「Cisco HDLC カプセル化」 \(P.366\)](#)
- [「PPP カプセル化」 \(P.366\)](#)
- [「キープアライブ タイマー」 \(P.368\)](#)
- [「フレームリレー上の Layer 2 Tunnel Protocol Version 3 ベースのレイヤ 2 VPN」 \(P.370\)](#)

Cisco XR 12000 シリーズ ルータ上の単一のシリアル インターフェイスは、PPP、Cisco HDLC、またはフレームリレーのカプセル化を使用して、単一のインターフェイス上でデータを伝送します。

Cisco HDLC カプセル化

Cisco ハイレベル データリンク コントロール (HDLC) は、HDLC を使用して同期シリアル リンク上でデータを送信するシスコ独自のプロトコルです。また、Cisco HDLC は、シリアル リンク キープアライブを維持するために、Serial Line Address Resolution Protocol (SLARP) と呼ばれる単純な制御プロトコルも提供します。HDLC は、Cisco IOS XR ソフトウェアにおけるシリアル インターフェイスのデフォルト カプセル化タイプです。Cisco HDLC は、開放型システム間相互接続 (OSI) スタックのレイヤ 2 (データ リンク) におけるデータ カプセル化のデフォルトであり、効率的なパケット記述およびエラー制御を実現します。



(注) Cisco HDLC は、シリアル インターフェイスのデフォルト カプセル化タイプです。

Cisco HDLC では、[「キープアライブ タイマー」 \(P.368\)](#) で説明するように、キープアライブを使用してリンク ステータスをモニタします。



(注) キープアライブ タイマーを設定した後で、ピアに送信される SLARP パケットの情報を表示するには、`debug chdlc slarp packet` コマンドを使用します。

PPP カプセル化

PPP は、同期シリアル リンクでのデータ送信に使用される標準プロトコルです。PPP は、リンク プロパティのネゴシエーションを行うリンク制御プロトコル (LCP) も提供します。LCP は、エコー要求および応答を使用して、リンクを継続的に使用できるかどうかをモニタします。



(注) インターフェイスに PPP カプセル化が設定されている場合、ECHOREQ パケットを送信し、ECHOREP 応答を受信しなかった回数が 5 回に達すると、リンク ダウンが宣言され、完全な LCP ネゴシエーションが再度開始されます。

PPP は、リンク上で動作するデータ プロトコルのプロパティをネゴシエーションするプロトコルとして、以下のネットワーク制御プロトコル (NCP) を提供します。

- IP プロパティのネゴシエーションを行う IP コントロール プロトコル (IPCP)
- MPLS プロパティのネゴシエーションを行うマルチプロトコル ラベル スイッチング コントロール プロセッサ (MPLSCP)
- CDP プロパティのネゴシエーションを行うシスコ検出プロトコル コントロール プロセッサ (CDPCP)
- IP Version 6 (IPv6) プロパティのネゴシエーションを行う IPv6CP
- OSI プロパティのネゴシエーションを行う開放型システム間相互接続コントロール プロセッサ (OSICP)

PPP では、「[キープアライブ タイマー](#)」(P.368) で説明するように、キープアライブを使用してリンク ステートをモニタします。

PPP は、データ トラフィックの伝送を許可する前にリモート装置にアイデンティティの証明を要求する、以下の認証プロトコルをサポートします。

- チャレンジ ハンドシェイク 認証プロトコル (CHAP) : CHAP 認証では、リモート装置にチャレンジ メッセージが送信されます。リモート装置は共有秘密鍵でチャレンジ値を暗号化し、暗号化された値と名前を応答メッセージでローカル ルータに返します。ローカル ルータは、リモート装置の名前をローカル ユーザ名データベースまたはリモート セキュリティ サーバ データベースに保存されている対応する秘密鍵と照合し、保存されている秘密鍵を使用することで元のチャレンジ メッセージを暗号化して、暗号化された値と一致することを確認します。
- マイクロソフト チャレンジ ハンドシェイク 認証プロトコル (MS-CHAP) : MS-CHAP は Microsoft バージョンの CHAP です。標準バージョンの CHAP と同様、MS-CHAP も PPP 認証に使用されます。この場合、Microsoft Windows NT または Microsoft Windows 95 を使用しているパーソナル コンピュータとネットワーク アクセス サーバとして動作するシスコのルータまたはアクセス サーバ間で認証が行われます。
- パスワード認証プロトコル (PAP) : PAP 認証では、リモート装置が名前とパスワードを送信する必要があり、それらがローカル ユーザ名データベースまたはリモート セキュリティ サーバ データベース内の対応するエントリと照合されます。



(注) PPP 認証プロトコルのイネーブル化および設定の詳細については、このマニュアルの「[Cisco IOS XR ソフトウェアでの PPP の設定](#)」モジュールを参照してください。

シリアル インターフェイスで CHAP、MS-CHAP、および PAP をイネーブルにするには、インターフェイス コンフィギュレーション モードで **ppp authentication** コマンドを使用します。



(注) PPP 認証をイネーブル化またはディセーブル化しても、リモート装置に対して自身の認証を行うローカル ルータの動作には影響しません。

マルチリンク PPP

マルチリンク ポイントツーポイント プロトコル (MLPPP) は、Cisco XR 12000 シリーズ ルータの 1 ポート チャネライズド OC-12/DS0 SPA およびラインカードでサポートされます。MLPPP を使用すると、複数の物理リンクを 1 つの論理リンクにまとめることができます。MLPPP の実装によって、複数の PPP シリアル インターフェイスが 1 つのマルチリンク インターフェイスにまとめられます。MLPPP は、複数の PPP リンク間におけるデータグラムの分割、再構成、順序付けを行います。MLPPP は 2 ポートおよび 4 ポート チャネライズド T3 SPA でサポートされます。

MLPPP には、PPP シリアル インターフェイスでサポートされている機能と同じ機能（ただし、QoS を除きます）があります。加えて、次の機能も提供します。

- フラグメント サイズ (128、256、512 バイト)
- 長いシーケンス番号 (24 ビット)
- 失われたフラグメントの検出タイムアウト時間 (80 ミリ秒)
- 最小アクティブ リンク設定オプション
- マルチリンク インターフェイスでの LCP エコー要求/応答サポート
- Full T1 および E1 のフレーム化されたリンクとフレーム化されていないリンク

シリアル インターフェイスで MLPPP を設定する方法の詳細については、このマニュアルの「Cisco IOS XR ソフトウェアでの PPP の設定」モジュールを参照してください。

キープアライブ タイマー

シスコのキープアライブはリンク ステートのモニタリングに役立ちます。キープアライブ タイマーの値によって決定される間隔で定期的にキープアライブがピアとの間で送受信されます。ピアから適切なキープアライブ応答を受信しなかったリンクは、ダウン状態に移行します。ピアから適切なキープアライブ応答があった場合、またはキープアライブがディセーブルの場合、リンクはアップ状態に移行します。



(注)

keepalive コマンドは、HDLC カプセル化または PPP カプセル化を使用するシリアル インターフェイスに適用されます。このコマンドはフレームリレー カプセル化を使用するシリアル インターフェイスには適用されません。

カプセル化タイプごとに、ピアから無視されたキープアライブが一定回数に達すると、シリアル インターフェイスがダウン状態に移行します。HDLC カプセル化の場合、キープアライブが 3 回無視されると、インターフェイスがダウン状態に移行します。PPP カプセル化の場合、キープアライブが 5 回無視されると、インターフェイスがダウン状態に移行します。ECHOREQ パケットは、LCP ネゴシエーションが完了したとき (LCP のオープン時など) にだけ送信されます。

リンク制御プロトコル (LCP) がピアに ECHOREQ パケットを送信する間隔 (秒数) を設定するには、インターフェイス コンフィギュレーション モードで **keepalive** コマンドを使用します。

デフォルトのキープアライブ インターバルは 10 秒です。

システムをデフォルトのキープアライブ インターバルに戻すには、**no keepalive** コマンドを使用します。

キープアライブ タイマーをディセーブルにするには、**keepalive disable** コマンドを使用します。



(注)

最小限の中断による再起動 (MDR) のアップグレードを実行する前に、Cisco XR 12000 シリーズ ルータ上でキープアライブを無効にするか、Cisco CRS-1 ルータ上のキープアライブ インターバルを 10 秒以上に設定することをお勧めします (プラットフォーム固有の推奨事項が必要な理由は、

Cisco CRS-1 ルータが、現在 L2 スプーフィングをサポートしているためです。L2 スプーフィングは、MDR 中の初期段階で開始されるプロセスです。Cisco XR 12000 シリーズ ルータでは L2 スプーフィングをサポートしていません。

ピア上の LCP は、ECHOREQ パケットを受信すると、ピアでキープアライブがイネーブルであるかどうかにかかわらず、エコー応答 (ECHOREP) パケットで応答します。

キープアライブは 2 つのピア間で独立しています。一方のピアでキープアライブをイネーブルに設定し、もう一方でディセーブルに設定することもできます。キープアライブがローカルでディセーブルに設定されていても、LCP は受信した ECHOREQ パケットに対して ECHOREP パケットで応答します。同様に、キープアライブ インターバルがそれぞれのピアで異なっても LCP には影響しません。



(注) キープアライブ タイマーを設定した後で、ピアに送信される SLARP パケットの情報を表示するには、`debug chdlc slarp packet` コマンドと他の Cisco HDLC `debug` コマンドを使用します。

フレームリレーのカプセル化

フレームリレー カプセル化をシリアル インターフェイスでイネーブルにする場合、インターフェイス設定は階層的で、次の要素から構成されます。

1. シリアル メイン インターフェイスは物理インターフェイスとポートから構成されます。シリアル インターフェイスが Cisco HDLC カプセル化および PPP カプセル化を使用する接続をサポートしていない場合は、シリアル メイン インターフェイス下に PVC を持つサブインターフェイスを設定する必要があります。フレームリレー接続は PVC だけでサポートされます。
2. シリアル サブインターフェイスはシリアル メイン インターフェイス下に設定されます。シリアル サブインターフェイスは、その下にシリアルを設定しなければトラフィックをアクティブに伝送しません。レイヤ 3 設定は、通常はサブインターフェイス上で行われます。
3. ポイントツーポイント PVC はシリアル サブインターフェイス下に設定します。PVC は、メイン インターフェイスの直下には設定できません。ポイントツーポイント PVC は、各サブインターフェイスに 1 つだけ設定できます。PVC は定義済みの回線バスを使用し、そのバスが中断されるとエラーになります。いずれかの設定から回線が削除されるまで、PVC はアクティブなままです。シリアル PVC 上の接続はフレームリレー カプセル化だけをサポートします。



(注) 親インターフェイスの管理ステートによって、サブインターフェイスとその PVC のステートが決まります。親インターフェイスまたはサブインターフェイスの管理ステートが変わると、その親インターフェイスまたはサブインターフェイス下に設定された子 PVC の管理ステートも変更されます。

シリアル インターフェイスでフレームリレー カプセル化を設定するには、`encapsulation frame-relay` コマンドを使用します。

フレームリレー インターフェイスは、次の 2 種類のカプセル化されたフレームをサポートします。

- Cisco (デフォルト)
- IETF

PVC に Cisco カプセル化または IETF カプセル化を設定するには、PVC コンフィギュレーション モードで `encap` コマンドを使用します。PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン シリアル インターフェイスのカプセル化タイプを継承します。



(注) MPLS に設定されたシリアル メイン インターフェイスには、Cisco カプセル化を設定する必要があります。IETF カプセル化は、MPLS ではサポートされません。

インターフェイスにフレームリレー カプセル化を設定する前に、そのインターフェイスから以前のレイヤ 3 設定がすべて削除されていることを確認する必要があります。たとえば、メイン インターフェイスに IP アドレスが設定されていないことが必要です。IP アドレスが設定されている場合、メイン インターフェイス上のフレームリレー設定は無効になります。

フレームリレー インターフェイス上の LMI

ローカル管理インターフェイス (LMI) プロトコルは、PVC の追加、削除、およびステータスをモニタリングします。また、フレームリレー UNI インターフェイスを構成するリンクの完全性も検証します。デフォルトでは、すべての PVC で **cisco** LMI がイネーブルになります。ただし、このマニュアルの「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」モジュールの「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」で説明するように、デフォルトの LMI タイプを ANSI または Q.933 に変更できます。

LMI タイプが **cisco** (デフォルトの LMI タイプ) の場合、単一のインターフェイスでサポートできる PVC の最大数は、メイン インターフェイスの MTU サイズに関連します。次の式を使用して、カードまたは SPA でサポートされる PVC の最大数を計算します。

$$(MTU - 13) / 8 = PVC \text{ の最大数}$$



(注) シリアル インターフェイスの場合、**mtu** コマンドのデフォルト設定は 1504 バイトです。したがって、**cisco** LMI で設定された 1 つのシリアル インターフェイスでサポートされる PVC のデフォルトの最大数は 186 です。

フレームリレー上の Layer 2 Tunnel Protocol Version 3 ベースのレイヤ 2 VPN

Layer 2 Tunnel Protocol Version 3 (L2TPv3) 機能は、レイヤ 2 Virtual Private Network (VPN; パーチャルプライベートネットワーク) を使用して IP コア ネットワーク上のレイヤ 2 ペイロードをトンネリングするための L2TP プロトコルを定義します。

L2TPv3 は、レイヤ 2 プロトコルを転送するために使用されるトンネリング プロトコルです。さまざまな設定で操作できます。また、パケット スイッチド ネットワーク上のさまざまなレイヤ 2 プロトコルおよび接続をトンネリングできます。

L2TPv3 を設定する前に、L2TPv3 疑似接続をホストする 2 つの接続回路 (AC) 間に接続を設定する必要があります。Cisco IOS XR ソフトウェアは、2 つの AC が結合されているポイントツーポイント、エンドツーエンドのサービスをサポートします。

ここでは、フレームリレー カプセル化を使用するシリアル インターフェイスにレイヤ 2 AC を設定する方法について説明します。



(注) シリアル インターフェイスは DLCI モード レイヤ 2 AC だけをサポートします。レイヤ 2 ポート モード AC はシリアル インターフェイスではサポートされません。

ネットワーク内の L2TPv3 の詳細については、『Cisco IOS XR MPLS Configuration Guide』の「Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software」モジュールを参照してください。L2VPN の設定の詳細については、『Cisco IOS XR MPLS Configuration Guide』の「Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software」モジュールを参照してください。

シリアル インターフェイス コンフィギュレーションのデフォルト設定

T3/E3 SPA でインターフェイスをイネーブルにし、追加のコンフィギュレーション コマンドを適用しない場合、デフォルトのインターフェイス設定は表 16 のようになります。これらのデフォルト設定はコンフィギュレーションで変更できます。

表 16 シリアル インターフェイスのデフォルト設定

パラメータ	コンフィギュレーション ファイルのエントリ	デフォルト設定
キープアライブ	<code>keepalive [disable]</code> <code>no keepalive</code>	10 秒のキープアライブ
カプセル化	<code>encapsulation [hdlc ppp frame-relay [IETF]]</code>	hdlc
最大伝送ユニット (MTU)	<code>mtu bytes</code>	1504 バイト
巡回冗長検査 (CRC)	<code>crc [16 32]</code>	16
シリアル インターフェイス上のデータ ストリームの反転	<code>invert</code>	データ ストリームは反転しません。
ペイロード スクランプリング (暗号化)	<code>scramble</code>	スクランプリングはディセーブルです。
パケット間に挿入される HDLC フラグ シーケンスの数	<code>transmit-delay</code>	デフォルトは 0 (ディセーブル) です。



(注) デフォルト設定は、`show running-config` コマンドの出力には含まれません。

シリアル インターフェイスの表記方法

クリア チャネル SPA 上のシリアル インターフェイスの表記方法は、`rack/slot/module/port` です。次に例を示します。

```
interface serial 0/0/1/2
```

チャネライズド SPA 上の T1、E1、および DS0 インターフェイスの表記方法は、`rack/slot/module/port/channel-num:channel-group-number` です。次に例を示します。

```
interface serial 0/0/1/2/4:3
```

シリアル インターフェイス下にサブインターフェイスと PVC を設定すると、ルータでは、シリアル インターフェイス アドレスの末尾にサブインターフェイス番号が含まれます。この場合の表記方法は `rack/slot/module/port[/channel-num:channel-group-number].subinterface` です。次に例を示します。

```
interface serial 0/0/1/2.1
interface serial 0/0/1/2/4:3.1
```



(注) 値の間のスラッシュは、表記の一部として必要です。

シリアル インターフェイスの表記方法の構文は次のようになります。

- *rack* : ラックのシャーシ番号。
- *slot* : モジュラ サービス カードまたはラインカードの物理スロット番号。
- *module* : モジュール番号。共有ポート アダプタ (SPA) は、そのサブスロット番号から参照されます。
- *port* : コントローラの物理ポート番号。
- *channel-num* : T1 または E1 のチャンネル番号。T1 チャンネルの範囲は 0 ~ 23、E1 チャンネルの範囲は 0 ~ 31 です。
- *channel-group-number* : タイムスロット番号。T1 タイムスロットの範囲は 1 ~ 24、E1 タイムスロットの範囲は 1 ~ 31 です。*channel-group-number* の前には、スラッシュではなくコロンを付けます。
- *subinterface* : サブインターフェイス番号。

有効なインターフェイスの選択肢一覧を表示するには、**serial** キーワードに続けて疑問符 (?) のオンライン ヘルプ機能を使用します。

シリアル インターフェイスの設定方法

チャネライズドまたはクリア チャンネル T3/E3 コントローラを設定した後は、「[Cisco IOS XR ソフトウェアでのクリア チャンネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定](#)」モジュールの説明に従って、そのコントローラに関連付けるシリアル インターフェイスを設定できます。次のタスクでは、シリアル インターフェイスを設定する方法について説明します。

- 「[シリアル インターフェイスの始動](#)」 (P.372)
- 「[オプションのシリアル インターフェイス パラメータの設定](#)」 (P.375)
- 「[PVC を持つポイントツーポイント シリアル サブインターフェイスの作成](#)」 (P.378)
- 「[シリアル インターフェイスでのキープアライブ インターバルの変更](#)」 (P.383)

シリアル インターフェイスの始動

ここでは、シリアル インターフェイスの始動に使用するコマンドについて説明します。

前提条件

Cisco XR 12000 シリーズ ルータには、1 つ以上の SIP、および 1 つ以上の SPA またはラインカードがインストールされ、Cisco IOS XR ソフトウェアを実行している必要があります。

- Cisco XR 12000 SIP-401
- Cisco XR 12000 SIP-501
- Cisco XR 12000 SIP-601
- 2 ポートおよび 4 ポート T3/E3 シリアル SPA
- 2 ポートおよび 4 ポート チャネライズド T3 シリアル SPA

- 4 ポート チャネライズド OC-12/DS3 ラインカード
- 1 ポート チャネライズド OC-12/DS0 SPA およびラインカード
- 1 ポート チャネライズド OC-48/DS3 SPA およびラインカード

Cisco CRS-1 ルータには、次の SIP および SPA がインストールされ、Cisco IOS XR ソフトウェアを実行している必要があります。

- Cisco CRS-1 SIP-800
- 2 ポートおよび 4 ポート T3/E3 シリアル SPA

制約事項

シリアル インターフェイスがアクティブになるためには、シリアル 接続の両端の設定が一致している必要があります。

手順の概要

1. **show interfaces**
2. **configure**
3. **interface serial *interface-path-id***
4. **ipv4 address *ip-address***
5. **no shutdown**
6. **end**
または
commit
7. **exit**
8. **exit**
9. 接続の他端でインターフェイスを始動するために、ステップ 1 ~ 8 を繰り返します。
10. **show ipv4 interface brief**
11. **show interfaces serial *interface-path-id***

詳細手順

	コマンドまたはアクション	目的
ステップ 1	show interfaces 例： RP/0/0/CPU0:router# show interfaces	(任意) 設定されているインターフェイスを表示します。 • このコマンドを使用して、ルータが PLIM カードを認識しているかどうかを確認します。
ステップ 2	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。

シリアル インターフェイスの設定方法

コマンドまたはアクション	目的
ステップ 3 interface serial interface-path-id 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0	シリアル インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4 ipv4 address ip-address 例: RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.2.1 255.255.255.224	インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。 (注) このインターフェイスにフレームリレー カプセル化を設定する場合は、このステップを省略してください。フレームリレーの場合、IP アドレスとサブネット マスクはサブインターフェイスに設定します。
ステップ 5 no shutdown 例: RP/0/0/CPU0:router (config-if)# no shutdown	shutdown 設定を削除します。 (注) shutdown 設定を削除することにより、インターフェイスでの強制的な管理上の停止が排除されるため、インターフェイスはアップ状態またはダウン状態に移行することができます (親 SONET レイヤが管理上の停止状態に設定されていないことを前提とします)。
ステップ 6 end または commit 例: RP/0/0/CPU0:router (config-if)# end または RP/0/0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 exit 例: RP/0/0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 8 exit 例: RP/0/0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。

	コマンドまたはアクション	目的
ステップ 9	<pre>show interfaces configure interface serial interface-path-id no shut exit exit commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# show interfaces RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface serial 0/1/0/1 RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.2.2 255.255.255.224 RP/0/0/CPU0:router (config-if)# no shutdown RP/0/0/CPU0:router (config-if)# commit RP/0/0/CPU0:router (config-if)# exit RP/0/0/CPU0:router (config)# exit</pre>	<p>接続の他端でインターフェイスを始動するために、ステップ 1 ~ 8 を繰り返します。</p> <p>(注) シリアル接続の両端で設定が一致している必要があります。</p>
ステップ 10	<pre>show ipv4 interface brief</pre> <p>例:</p> <pre>RP/0/0/CPU0:router # show ipv4 interface brief</pre>	<p>インターフェイスがアクティブであり、適切に設定されていることを確認します。</p> <p>シリアル インターフェイスが適切に始動されていると、show ipv4 interface brief コマンドの出力結果で、そのインターフェイスの [Status] フィールドに [Up] と表示されます。</p>
ステップ 11	<pre>show interfaces serial interface-path-id</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# show interfaces serial 0/1/0/0</pre>	<p>(任意) インターフェイスの設定を表示します。</p>

次に行う作業

始動したシリアル インターフェイスのデフォルト設定を変更するには、「[オプションのシリアル インターフェイス パラメータの設定](#)」(P.375) を参照してください。

オプションのシリアル インターフェイス パラメータの設定

ここでは、シリアル インターフェイスのデフォルト設定の変更には使用できるコマンドについて説明します。

前提条件

シリアル インターフェイスのデフォルト設定を変更する前に、シリアル インターフェイスを始動して、「[シリアル インターフェイスの始動](#)」(P.372) で説明するように shutdown 設定を削除することをお勧めします。

制約事項

シリアル インターフェイスがアクティブになるためには、シリアル 接続の両端の設定が一致している必要があります。

手順の概要

1. **configure**
2. **interface serial** *interface-path-id*
3. **encapsulation** [**hdlc** | **ppp** | **frame-relay** [**IETF**]]
4. **serial**
5. **crc length**
6. **invert**
7. **scramble**
8. **transmit-delay** *hdlc-flags*
9. **end**
または
commit
10. **exit**
11. **exit**
12. **exit**
13. **show interfaces serial** [*interface-path-id*]

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial <i>interface-path-id</i> 例： RP/0/RP0/CPU0:router(config)# interface serial 0/1/0/0	シリアル インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	encapsulation [hdlc ppp frame-relay [IETF]] 例： RP/0/RP0/CPU0:router(config-if)# encapsulation hdlc	(任意) HDLC や PPP、フレームリレーなどのインターフェイス カプセル化パラメータおよび詳細を設定します。 (注) デフォルトのカプセル化は hdlc です。
ステップ 4	serial 例： RP/0/0/CPU0:router(config-if)# serial RP/0/0/CPU0:ios(config-if-serial)#	(任意) シリアル サブモードを開始し、シリアル パラメータを設定します。

	コマンドまたはアクション	目的
ステップ 5	crc length 例: RP/0/0/CPU0:ios(config-if-serial)# crc 32	(任意) インターフェイスの巡回冗長検査 (CRC) の長さを指定します。16 ビットの CRC モードを指定するには 16 キーワード、 32 ビットの CRC モードを指定するには 32 キーワードを入力します。 (注) デフォルトの CRC の長さは 16 です。
ステップ 6	invert 例: RP/0/0/CPU0:ios(config-if-serial)# inverts	(任意) データ ストリームを反転します。
ステップ 7	scramble 例: RP/0/0/CPU0:ios(config-if-serial)# scramble	(任意) インターフェイス上でペイロード スクランプリングをイネーブルにします。 (注) インターフェイス上のペイロード スクランプリングはディセーブルです。
ステップ 8	transmit-delay hdlc-flags 例: RP/0/0/CPU0:ios(config-if-serial)# transmit-delay 10	(任意) インターフェイス上の送信遅延を指定します。指定できる値は 0 ~ 128 です。 (注) 送信遅延はデフォルトでディセーブルです (送信遅延は 0 に設定されます)。
ステップ 9	end または commit 例: RP/0/RP0/CPU0:router (config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 10	exit 例: RP/0/0/CPU0:router(config-if-serial)# exit RP/0/0/CPU0:router(config-if)#	シリアル コンフィギュレーションモードを終了します。
ステップ 11	exit 例: RP/0/RP0/CPU0:router (config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 12 <code>exit</code> 例: RP/0/RP0/CPU0:router (config)# exit	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
ステップ 13 <code>show interfaces serial [interface-path-id]</code> 例: RP/0/RP0/CPU0:router# show interface serial 0/1/0/0	(任意) 指定したシリアル インターフェイスの一般情報を表示します。

次に行う作業

- 始動したシリアル インターフェイス上に PVC を持つポイントツーポイント フレームリレー サブインターフェイスを作成するには、「[PVC を持つポイントツーポイント シリアル サブインターフェイスの作成](#)」(P.378) を参照してください。
- PPP カプセル化がイネーブルであるシリアル インターフェイスに PPP 認証を設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの PPP の設定](#)」モジュールを参照してください。
- デフォルトのキープアライブ設定を変更するには、「[シリアル インターフェイスでのキープアライブ インターバルの変更](#)」(P.383) を参照してください。
- フレームリレー カプセル化がイネーブルであるシリアル インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」モジュールの「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」を参照してください。

PVC を持つポイントツーポイント シリアル サブインターフェイスの作成

ここに記載する手順では、ポイントツーポイント シリアル サブインターフェイスを作成し、そのシリアル サブインターフェイスに PVC を設定します。



(注)

サブインターフェイスおよび PVC の作成は、フレームリレー カプセル化だけが設定されたインターフェイスでサポートされます。

前提条件

シリアル インターフェイスでサブインターフェイスを作成する前に、「[シリアル インターフェイスの始動](#)」(P.372) で説明するように、フレームリレー カプセル化が設定されたメイン シリアル インターフェイスを始動する必要があります。

制約事項

PVC は、各ポイントツーポイント シリアル サブインターフェイスに 1 つだけ設定できます。

手順の概要

1. configure

2. **interface serial interface-path-id.subinterface point-to-point**
3. **ipv4 address ipv4_address/prefix**
4. **pvc dlci**
5. **end**
または
commit
6. 接続の他端でシリアル サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 5 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial interface-path-id.subinterface point-to-point 例: RP/0/0/CPU0:router (config)# interface serial 0/1/0/0.1	シリアル サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	ipv4 address ipv4_address/prefix 例: RP/0/0/CPU0:router (config-subif)#ipv4 address 10.46.8.6/24	サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。
ステップ 4	pvc dlci 例: RP/0/0/CPU0:router (config-subif)# pvc 20	シリアル PVC を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。 <i>dlci</i> を 16 から 1007 の範囲の PVC ID に置き換えます。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

コマンドまたはアクション	目的
<p>ステップ 5</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router (config-subif)# end または RP/0/0/CPU0:router(config-subif)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6</p> <pre>configure interface serial interface-path-id pvc dlcid commit</pre> <p>例:</p> <pre>RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface serial 0/1/0/1.1 RP/0/0/CPU0:router (config-subif)#ipv4 address 10.46.8.5/24 RP/0/0/CPU0:router (config-subif)# pvc 20 RP/0/0/CPU0:router (config-fr-vc)# commit</pre>	<p>接続の他端でシリアル サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 5 を繰り返します。</p> <p>(注) DLCI (PVC ID) は、サブインターフェイス接続の両端で一致している必要があります。</p> <p>(注) 接続の他端のサブインターフェイスに IP アドレスおよびサブネット マスクを割り当てるときには、接続の両端のアドレスが同じサブネットに属している必要があることに注意してください。</p>

次に行う作業

- オプションの PVC パラメータを設定するには、「[オプションのシリアルインターフェイスパラメータの設定](#)」(P.375) を参照してください。
- フレームリレー カプセル化がイネーブルであるシリアル インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルの「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」モジュールの「[インターフェイスでのデフォルト フレームリレー設定の変更](#)」を参照してください。
- レイヤ 3 QOS サービス ポリシーを PVC サブモードの PVC に付加するには、該当する Cisco IOS XR ソフトウェアのコンフィギュレーション ガイドを参照してください。

オプションの PVC パラメータの設定

ここでは、シリアル PVC でのデフォルト設定の変更に使用できるコマンドについて説明します。

前提条件

PVC のデフォルト設定を変更する前に、「[PVC を持つポイントツーポイント シリアル サブインターフェイスの作成](#)」(P.378) で説明するようにシリアル サブインターフェイスで PVC を作成する必要があります。

制約事項

- 接続がアクティブになるためには、DLCI (PVI ID) が PVC の両端で一致している必要があります。
- PVC DLCI を変更するには、PVC を削除し、新しい DLCI を設定して PVC を追加し直す必要があります。

手順の概要

1. **configure**
2. **interface serial** *interface-path-id.subinterface*
3. **pvc** *dlci*
4. **encap** [**cisco** | **ietf**]
5. **service-policy** {**input** | **output**} *policy-map*
6. **end**
または
commit
7. 接続の他端で PVC を設定するために、ステップ 1 ~ 6 を繰り返します。
8. **show frame-relay pvc** *dlci-number*
9. **show policy-map interface pos** *interface-path-id.subinterface* {**input** | **output**}
or
show policy-map type qos interface pos *interface-path-id.subinterface* {**input** | **output**}

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial <i>interface-path-id.subinterface</i> 例: RP/0/0/CPU0:router (config)# interface serial 0/1/0/0.1	シリアル サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc <i>dlci</i> 例: RP/0/0/CPU0:router (config-subif)# pvc 20	PVC に対するサブインターフェイス コンフィギュレーション モードを開始します。

シリアル インターフェイスの設定方法

コマンドまたはアクション	目的
ステップ 4 <code>encap [cisco ietf]</code> 例: RP/0/0/CPU0:router (config-fr-vc)# encap ietf	(任意) フレームリレー PVC のカプセル化を設定します。 (注) PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン シリアル インターフェイスのカプセル化タイプを継承します。
ステップ 5 <code>service-policy {input output} policy-map</code> 例: RP/0/0/CPU0:router (config-fr-vc)# service-policy output policy1	ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。
ステップ 6 <code>end</code> または commit 例: RP/0/0/CPU0:router (config-fr-vc)# end または RP/0/0/CPU0:router (config-fr-vc)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>configure</code> interface serial interface-path-id.subinterface pvc dlcí encap [cisco ietf] commit 例: RP/0/0/CPU0:router# configure RP/0/0/CPU0:router (config)# interface serial 0/1/0/1.1 RP/0/0/CPU0:router (config-subif)# pvc 20 RP/0/0/CPU0:router (config-fr-vc)# encap cisco RP/0/0/CPU0:router (config-fr-vc)# commit	接続の他端でシリアル サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 6 を繰り返します。 (注) サブインターフェイス接続の両端で設定が一致している必要があります。

コマンドまたはアクション	目的
ステップ 8 <code>show frame-relay pvc dlci-number</code> 例: RP/0/RP0/CPU0:router# show frame-relay pvc 20	(任意) 指定したシリアル インターフェイスの設定を検証します。
ステップ 9 <code>show policy-map interface serial interface-path-id.subinterface {input output}</code> または <code>show policy-map type qos interface serial interface-path-id.subinterface {input output}</code> 例: RP/0/RP0/CPU0:router# show policy-map interface serial 0/1/0/0.1 output または RP/0/RP0/CPU0:router# show policy-map type qos interface serial 0/1/0/0.1 output	(任意) サブインターフェイスに付加された入力ポリシーおよび出力ポリシーの統計情報と設定を表示します。

次に行う作業

フレームリレー カプセル化がイネーブルであるシリアル インターフェイスのデフォルトのフレームリレー設定を変更するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのフレームリレーの設定](#)」モジュールの「[インターフェイスでのデフォルトフレームリレー設定の変更](#)」を参照してください。

シリアル インターフェイスでのキープアライブ インターバルの変更

Cisco HDLC カプセル化または PPP カプセル化がイネーブルであるシリアル インターフェイスのキープアライブ インターバルを変更するには、次の作業を行います。



(注) シリアル インターフェイスで Cisco HDLC カプセル化または PPP カプセル化をイネーブルにした場合、キープアライブ インターバルはデフォルトで 10 秒に設定されます。デフォルトのキープアライブ インターバルを変更する手順は、次のとおりです。



(注) Cisco HDLC は、シリアル インターフェイスにおいてデフォルトでイネーブルになります。

前提条件

キープアライブ タイマーの設定を変更する前に、インターフェイスで Cisco HDLC カプセル化または PPP カプセル化がイネーブルになっていることを確認する必要があります。インターフェイスで Cisco HDLC カプセル化または PPP カプセル化をイネーブルにするには、「[オプションのシリアル インターフェイス パラメータの設定](#)」(P.375) で説明するように **encapsulation** コマンドを使用します。

制約事項

MDR のアップグレードを実行する前に、Cisco XR 12000 シリーズ ルータ上でキープアライブを無効にするか、Cisco CRS-1 ルータ上でキープアライブ インターバルを 10 秒以上に設定することをお勧めします（プラットフォーム固有の推奨事項が必要な理由は、Cisco CRS-1 ルータが、現在 L2 スプーフィングをサポートしているためです。L2 スプーフィングは、MDR 中の初期段階で開始されるプロセスです。Cisco XR 12000 シリーズ ルータでは L2 スプーフィングをサポートしていません）。

手順の概要

1. **configure**
2. **interface serial** *interface-path-id*
3. **keepalive** {*seconds* | **disable**}
4. **end**
または
commit
5. **show interfaces** *type interface-path-id*

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial <i>interface-path-id</i> 例： RP/0/RP0/CPU0:router(config)# interface serial 0/1/0/0	シリアル インターフェイス名と <i>rack/slot/module/port</i> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	keepalive { <i>seconds</i> disable }	リンク制御プロトコル (LCP) がピアに ECHOREQ を送信する頻度 (秒) を指定します。デフォルトのキープアライブ インターバルは 10 秒です。 システムをデフォルトのキープアライブ インターバルに戻すには、 no keepalive コマンドを使用します。 キープアライブ タイマーをディセーブルにするには、 keepalive disable コマンドを使用します。

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <ul style="list-style-type: none"> • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 <code>show interfaces serial interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show interfaces serial 0/1/0/0</code></p>	<p>(任意) インターフェイスの設定を確認します。</p>

レイヤ 2 接続回路 (AC) の設定方法

レイヤ 2 接続回路 (AC) の設定作業について、次の手順で説明します。

- [PVC を持つシリアル レイヤ 2 サブインターフェイスの作成](#)
- [オプションのシリアル レイヤ 2 PVC パラメータの設定](#)



(注) レイヤ 2 スイッチングのためのインターフェイスの設定後は、**ipv4 address** などのルーティング コマンドは使用できません。インターフェイスにルーティング コマンドを設定すると、**l2transport** コマンドが拒否されます。

PVC を持つシリアル レイヤ 2 サブインターフェイスの作成

ここに記載する手順では、PVC を持つレイヤ 2 サブインターフェイスを作成します。

前提条件

シリアル インターフェイスでサブインターフェイスを作成する前に、「[シリアル インターフェイスの始動](#)」(P.372) で説明するようにシリアル インターフェイスを始動する必要があります。

制約事項

各シリアル サブインターフェイスで設定できる PVC は 1 つだけです。

手順の概要

1. **configure**
2. **interface serial *interface-path-id.subinterface* l2transport**
3. **pvc *vpi/vci***
4. **end**
または
commit
5. AC の他端でシリアル サブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 4 を繰り返します。

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial <i>interface-path-id.subinterface</i> l2transport 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0.1 l2transport	サブインターフェイスを作成して、そのサブインターフェイスに対するシリアル サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc <i>vpi/vci</i> 例: RP/0/0/CPU0:router(config-if)# pvc 5/20	シリアル PVC を作成して、シリアル レイヤ 2 転送 PVC コンフィギュレーション モードを開始します。 (注) 各サブインターフェイスに設定できる PVC は 1 つだけです。

コマンドまたはアクション	目的
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router (config-fr-vc) # <code>end</code> または RP/0/0/CPU0:router (config-fr-vc) # <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 <ul style="list-style-type: none"> • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5 AC の他端でシリアルサブインターフェイスおよび関連付けられている PVC を始動するために、ステップ 1 ~ 4 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) AC の両端で設定が一致している必要があります。</p>

次に行う作業

- オプションの PVC パラメータを設定するには、「[オプションのシリアルレイヤ 2 PVC パラメータの設定](#)」(P.387) を参照してください。
- 作成した AC にポイントツーポイント疑似接続 XConnect を設定するには、『*Cisco IOS XR MPLS Configuration Guide*』の「*Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software*」モジュールを参照してください。
- L2VPN を設定するには、『*Cisco IOS XR MPLS Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software*」モジュールを参照してください。

オプションのシリアルレイヤ 2 PVC パラメータの設定

ここでは、シリアルレイヤ 2 PVC でのデフォルト設定の変更に使用できるコマンドについて説明します。

前提条件

PVC のデフォルト設定を変更する前に、「[PVC を持つシリアルレイヤ 2 サブインターフェイスの作成](#)」(P.385) で説明するようにレイヤ 2 サブインターフェイスで PVC を作成する必要があります。

制約事項

PVC の両端での設定が、アクティブにする接続に合っている必要があります。

手順の概要

1. **configure**
2. **interface serial *interface-path-id.subinterface* l2transport**
3. **pvc *dci***
4. **encap [cisco | ietf]**
5. **service-policy {input | output} *policy-map***
6. **fragment end-to-end *fragment-size***
7. **end**
または
commit
8. AC の他端で PVC を設定するために、ステップ 1 ~ 7 を繰り返します。
9. **show policy-map interface serial *interface-path-id.subinterface* {input | output}**
or
show policy-map type qos interface serial *interface-path-id.subinterface* {input | output}

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface serial <i>interface-path-id.subinterface</i> l2transport 例: RP/0/0/CPU0:router(config)# interface serial 0/1/0/0.1 l2transport	レイヤ 2 シリアル サブインターフェイスに対するシリアル サブインターフェイス コンフィギュレーション モードを開始します。
ステップ 3	pvc <i>dci</i> 例: RP/0/0/CPU0:router(config-if)# pvc 100	指定した PVC に対するシリアル フレームリレー PVC コンフィギュレーション モードを開始します。
ステップ 4	encap {cisco ietf} 例: RP/0/0/CPU0:router(config-fr-vc)# encapsulation aal5	フレームリレー PVC のカプセル化を設定します。

コマンドまたはアクション	目的
<p>ステップ 5 <code>fragment end-to-end fragment-size</code></p> <p>例: RP/0/0/CPU0:router(config-fr-vc)# fragment end-to-end 100</p>	<p>インターフェイスでフレームリレー フレームのフラグメンテーションをイネーブルにします。</p> <p><i>fragment-size</i> を、発信元フレームリレー フレームのペイロード バイト数に置き換えます。これが各フラグメントのバイト数になります。この数値には、元のフレームのフレームリレー ヘッダーは含まれません。</p> <p>有効な値は 16 ~ 1600 です。デフォルト値は 53 です。</p>
<p>ステップ 6 <code>service-policy {input output} policy-map</code></p> <p>例: RP/0/0/CPU0:router (config-subif)# service-policy output policy1</p>	<p>ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。</p>
<p>ステップ 7 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-serial-l2transport-pvc)# end または RP/0/0/CPU0:router(config-serial-l2transport-pvc)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 <p>設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。</p>
<p>ステップ 8 AC の他端で PVC を設定するために、ステップ 1 ~ 7 を繰り返します。</p>	<p>AC を始動します。</p> <p>(注) 接続の両端で設定が一致している必要があります。</p>
<p>ステップ 9 <code>show policy-map interface serial interface-path-id.subinterface {input output}</code> または <code>show policy-map type qos interface serial interface-path-id.subinterface {input output}</code></p> <p>例: RP/0/0/CPU0:router# show policy-map interface pos 0/1/0/0.1 output または RP/0/0/CPU0:router# show policy-map type qos interface pos 0/1/0/0.1 output</p>	<p>(任意) サブインターフェイスに付加された入力ポリシー および出力ポリシーの統計情報と設定を表示します。</p>

次に行う作業

- 作成した AC にポイントツーポイント疑似接続 XConnect を設定するには、『Cisco IOS XR MPLS Configuration Guide』の「Layer 2 Tunnel Protocol Version 3 on Cisco IOS XR Software」モジュールを参照してください。
- L2VPN を設定するには、『Cisco IOS XR MPLS Configuration Guide』の「Implementing MPLS Layer 2 VPNs on Cisco IOS XR Software」モジュールを参照してください。

シリアル インターフェイスの設定例

ここでは、次の設定例について説明します。

- 「シリアル インターフェイスの始動と Cisco HDLC カプセル化の設定 : 例」(P.390)
- 「シリアル インターフェイスでのフレームリレー カプセル化の設定 : 例」(P.391)
- 「シリアル インターフェイスでの PPP カプセル化の設定 : 例」(P.392)

シリアル インターフェイスの始動と Cisco HDLC カプセル化の設定 : 例

次に、Cisco HDLC カプセル化を設定した基本的なシリアル インターフェイスの始動例を示します。

```
RP/0/0/CPU0:Router#config
RP/0/0/CPU0:Router(config)# interface serial 0/3/0/0:0
RP/0/0/CPU0:Router(config-if)# ipv4 address 192.0.2.2 255.255.255.252
RP/0/0/CPU0:Router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

次に、キープアライブ メッセージの間隔を 10 秒に設定する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0:0
RP/0/0/CPU0:router(config-if)# keepalive 10
RP/0/0/CPU0:router(config-if)# commit
```

次に、オプションのシリアル インターフェイス パラメータを変更する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0:0
RP/0/0/CPU0:Router(config-if)# serial
RP/0/0/CPU0:Router(config-if-serial)# crc 16
RP/0/0/CPU0:Router(config-if-serial)# invert
RP/0/0/CPU0:Router(config-if-serial)# scramble
RP/0/0/CPU0:Router(config-if-serial)# transmit-delay 3
RP/0/0/CPU0:Router(config-if-serial)# commit
```

次は、**show interfaces serial** コマンドの出力例です。

```
RP/0/0/CPU0:Router# show interfaces serial 0/0/3/0/5:23
Serial0/0/3/0/5:23 is down, line protocol is down
  Hardware is Serial network interface(s)
  Internet address is Unknown
  MTU 1504 bytes, BW 64 Kbit
    reliability 143/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set, keepalive set (10 sec)
  Last clearing of "show interface" counters 18:11:15
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
```



```
2764 packets input, 2816 bytes, 3046 total input drops
0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
3046 input errors, 1 CRC, 0 frame, 0 overrun, 2764 ignored, 281 abort
2764 packets output, 60804 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

シリアル インターフェイスでのフレームリレー カプセル化の設定 : 例

次に、ルータ 1 上に、フレームリレー カプセル化を設定したクリア チャネル SPA 上および PVC を設定したシリアル サブインターフェイス上にシリアル インターフェイスを作成する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/1/0/0
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# frame-relay intf-type dce
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/1/0/0.1 point-to-point
RP/0/0/CPU0:router (config-subif)# ipv4 address 10.20.3.1/24
RP/0/0/CPU0:router (config-subif)# pvc 16
RP/0/0/CPU0:router (config-fr-vc)# encapsulation ietf
RP/0/0/CPU0:router (config-fr-vc)# commit
RP/0/0/CPU0:router (config-fr-vc)# exit
RP/0/0/CPU0:router (config-subif)# exit
RP/0/0/CPU0:router (config)# exit

RP/0/0/CPU0:router# show interface serial 0/1/0/0
Wed Oct  8 04:14:39.946 PST DST
Serial0/1/0/0 is up, line protocol is up
  Interface state transitions: 5
  Hardware is Serial network interface(s)
  Internet address is 10.20.3.1/24
  MTU 4474 bytes, BW 44210 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation FRAME-RELAY, crc 16,
  Scrambling is disabled, Invert data is disabled
  LMI enq sent  0, LMI stat recvd 0, LMI upd recvd 0
  LMI enq recvd 880, LMI stat sent  880, LMI upd sent  0 , DCE LMI up
  LMI DLCI 1023 LMI type is CISCO frame relay DCE
  Last clearing of "show interface" counters 02:23:04
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    858 packets input, 11154 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
    Received 0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    858 packets output, 12226 bytes, 0 total output drops
    0 output errors, 0 underruns, 0 applique, 0 resets
    0 output buffer failures, 0 output buffers swapped out
```

次に、ルータ 1 に接続しているルータ 2 上に、フレームリレー カプセル化を設定したクリア チャネル SPA 上および PVC を設定したシリアル サブインターフェイス上にシリアル インターフェイスを作成する例を示します。

```

RP/0/0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface serial 0/1/0/1
RP/0/0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface serial 0/1/0/1.1 point-to-point
RP/0/0/CPU0:router (config-subif)#ipv4 address 10.20.3.2/24
RP/0/0/CPU0:router (config-subif)# pvc 16
RP/0/0/CPU0:router (config-fr-vc)# encapsulation ietf
RP/0/0/CPU0:router (config-fr-vc)# commit
RP/0/0/CPU0:router(config-fr-vc)# exit
RP/0/0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# exit

RP/0/RP0/CPU0:router# show interface serial 0/1/0/1
Wed Oct  8 04:13:45.046 PST DST
Serial0/1/0/1 is up, line protocol is up
  Interface state transitions: 7
  Hardware is Serial network interface(s)
  Internet address is Unknown
  MTU 4474 bytes, BW 44210 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation FRAME-RELAY, crc 16,
  Scrambling is disabled, Invert data is disabled
  LMI enq sent 1110, LMI stat recvd 875, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
  LMI DLCI 1023 LMI type is CISCO frame relay DTE
  Last clearing of "show interface" counters 02:22:09
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    853 packets input, 12153 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
    Received 0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    853 packets output, 11089 bytes, 0 total output drops
    0 output errors, 0 underruns, 0 applique, 0 resets
    0 output buffer failures, 0 output buffers swapped out

```

シリアル インターフェイスでの PPP カプセル化の設定 : 例

次に、シリアル インターフェイスを作成し、PPP カプセル化を設定する例を示します。

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface serial 0/3/0/0/0:0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RP0/CPU0:router(config-if)# encapsulation ppp
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# ppp authentication chap MIS-access
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes

```

次に、最初の認証が失敗した後に 2 回リトライできる（認証が失敗した場合に全部で 3 回リトライできる）ようにシリアル インターフェイス 0/3/0/0/0 を設定する例を示します。

```

RP/0/RP0/CPU0:router# configuration
RP/0/RP0/CPU0:router(config)# interface serial 0/3/0/0/0:0
RP/0/RP0/CPU0:router(config-if)# encapsulation ppp
RP/0/RP0/CPU0:router(config-if)# ppp authentication chap
RP/0/RP0/CPU0:router(config-if)# ppp max-bad-auth 3

```

```
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

その他の参考資料

ここでは、T3/E3 および T1/E1 コントローラおよびシリアル インターフェイスに関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システムブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および『Cisco IOS XR System Security Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
FRF.1.2	PVC User-to-Network Interface (UNI) Implementation Agreement - July 2000
ANSI T1.617 Annex D	-
ITU Q.933 Annex A	-

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC 1294	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 1315	<i>Management Information Base for Frame Relay DTEs</i>
RFC 1490	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 1586	<i>Guidelines for Running OSPF Over Frame Relay Networks</i>
RFC 1604	<i>Definitions of Managed Objects for Frame Relay Service</i>
RFC 2115	<i>Management Information Base for Frame Relay DTEs Using SMIPv2</i>
RFC 2390	<i>Inverse Address Resolution Protocol</i>
RFC 2427	<i>Multiprotocol Interconnect Over Frame Relay</i>
RFC 2954	<i>Definitions of Managed Objects for Frame Relay Service</i>

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの SRP インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアを実行するルータで、サポートされる Cisco Dynamic Packet Transport (DPT) インターフェイスにスペース再利用プロトコル (SRP) を設定する方法について説明します。

SRP はシスコが開発した MAC 層プロトコルです。Cisco DPT 製品と連携して使用されます。DPT 製品によって、スケーラブルなインターネット サービス、信頼性の高い IP 対応光トランスポート、およびネットワーク操作の簡略化を実現できます。これらのソリューションで、信頼性の高い光パケットリング インフラストラクチャ全体にわたって、IP サービスを拡張および展開することができます。



(注) 以降、SRP という用語は、DPT 製品に関連する機能を説明するときに使用します。

Cisco IOS XR ソフトウェアの SRP インターフェイス設定の機能履歴

リリース	変更点
リリース 3.2.2	この機能は Cisco CRS-1 ルータで導入されました。また、4 ポート OC-192c/STM-64c POS/DPT PLIM でのみサポートされます。
リリース 3.4.0	このコマンドは、16 ポート OC-48c/STM-16c POS/DPT PLIM で初めてサポートされました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	この機能のサポートは、次の共有ポート アダプタ (SPA) の Cisco CRS-1 ルータに追加されました。 <ul style="list-style-type: none">• 1 ポート OC-192/STM-64 POS/RPR SPA XFP 光ファイバ• 4 ポート OC-48/STM-16 POS/RPR SPA• 2 ポート OC-48/STM-16 POS/RPR SPA

この章の構成

- 「SRP インターフェイスを設定するための前提条件」(P.396)
- 「SRP インターフェイスの設定に関する情報」(P.396)
- 「SRP インターフェイスの設定方法」(P.397)

- 「SRP インターフェイスの設定例」(P.420)
- 「その他の参考資料」(P.421)

SRP インターフェイスを設定するための前提条件

SRP インターフェイスを設定する前に、次の条件を満たしていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『Cisco IOS XR Task ID Reference Guide』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

- 新しい SRP インターフェイス設定に割り当てるインターフェイスの IP アドレスを調べておく必要があります。
- 使用しているハードウェアが SRP をサポートしていることを確認します。現在、SRP は次の PLIM および SPA でサポートされます。
 - 4 ポート OC-192c/STM-64c POS/DPT PLIM
 - 16 ポート OC-48c/STM-16c POS/DPT PLIM
 - 1 ポート OC-192/STM-64 POS/RPR SPA XFP 光ファイバ
 - 4 ポート OC-48/STM-16 POS/RPR SPA
 - 2 ポート OC-48/STM-16 POS/RPR SPA

SRP インターフェイスの設定に関する情報

SRP のパケット宛先削除という特性があるため、帯域幅空間を再利用することができます。従来のテクノロジーでは発信元での削除処理を取り入れています。この方法では、発信元がパケットを削除するまで、パケットはリング全体を通過し続けます。発信元ノードと宛先ノードがリング上で隣接している場合でも、発信元に戻って削除されるまで、リング全体を通過し続けます。SRP では、宛先ノードがパケットを読み取った後にパケットを削除することで、空いている帯域幅をより効率的に使用できるようになります。その結果、SRP リング上で他のノードが使用できる帯域幅が増えます。

SRP リングは、外側リングと内側リングという反対方向に回転する 2 つのファイバから構成されます。2 つのリングは、データ パケットおよび制御パケットを伝送するために同時に使用されます。SRP では、明示的な制御パケットと、データ パケット内に含まれる制御情報の両方を使用しています（制御パケットは、キープアライブ、保護切り替え、帯域幅制御の伝播などのタスクを処理します）。制御パケットは、対応するデータ パケットとは反対方向で伝播します。それによって、宛先まで最短のパスでデータが伝送されます。二重光ファイバリングを使用することで、パケットの生存率を高めることができます。ノードの失敗時やファイバの切断時には、データは代替のリングで伝送されます。

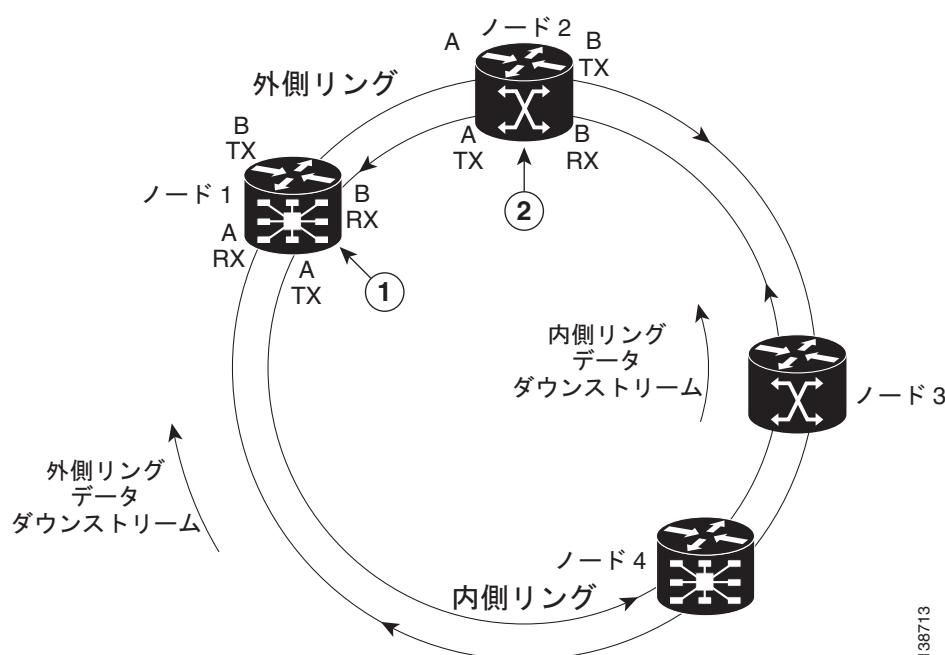
SRP リングはメディアに依存しないため、SONET/SDH、波長分割多重 (WDM)、ダーク ファイバなど、多様な基盤テクノロジーで運用できます。組み込みのファイバ転送インフラストラクチャを問わず、SRP リングを実行できるというこの機能によって、パケットが最適化された転送が可能なパスが実現し、広帯域幅の IP ネットワークを構築できます。図 9 は、Cisco CRS-1 ルータ および Cisco XR 12000 シリーズ ルータ で作成した SRP リングです。

2つのリングを区別するために、一方を「内側」リング、もう一方を「外側」リングと呼びます。SRPは、ある方向（ダウンストリーム）でデータパケットを送信し、他のファイバ上の反対方向（アップストリーム）で対応する制御パケットを送信するという処理方法です。この処理方法によって両方のファイバを同時に使用することで、パケットの転送に帯域幅を最大限に利用し、また、適応的な帯域幅使用および自己回復のための制御信号の伝播を加速します。

図 9 に示すように、SRP ノードは、SRP サイド A を使用して外側リングのデータを受信し（RX）、内側リングのデータを伝送します（TX）。また、SRP サイド B を使用して内側リングのデータを受信し（RX）、外側リングのデータを伝送します（TX）。あるノード上のサイド A は、隣接する SRP ノード上のサイド B に接続します。

SRP インターフェイスを設定するコマンドについては、『Cisco IOS XR Interface and Hardware Component Command Reference』を参照してください。

図 9 SRP リングの例



1 Cisco CRS-1 ルータ

2 Cisco XR 12000 シリーズ ルータ

SRP インターフェイスの設定方法

ここでは、次の手順について説明します。

- 「PortPLIM ポートでの SRP のイネーブル化」 (P.398) (SRP 対応の PLIM を使用する場合に必要です)
- 「OC-48/STM-16 SPA ポートでの SRP のイネーブル化」 (P.400) (SRP 対応の OC-48/STM-16 SPA を使用する場合に必要です)
- 「OC-192/STM-64 SPA ポートでの SRP のイネーブル化」 (P.403) (SRP 対応の OC-192/STM-64 SPA を使用する場合に必要です)
- 「基本的な SRP 設定の作成」 (P.405) (必須)
- 「インテリジェント保護スイッチング (IPS) の設定」 (P.407) (任意)

- 「SRP によるモジュラ サービス品質コマンドライン インターフェイス (MQC) の設定」(P.410) (任意)
- 「リングへのノードの追加」(P.414) (任意)

PortPLIM ポートでの SRP のイネーブル化

PLIM ポートで SRP の使用をイネーブルにするには、このタスクを実行する必要があります。デフォルトで、POS/DPT PLIM は POS のみをサポートします。

制約事項

4 ポート OC-192c/STM-64c POS/DPT PLIM では、各ポート ペア (0 と 1、または 2 と 3) を同一に設定する必要があります。ポート 0 を SRP 用に設定し、ポート 1 を SRP 用に設定しない場合、この設定は機能しません。

16 ポート OC-48c/STM-16c POS/DPT PLIM では、4 ポートの各グループを同一に設定する必要があります。単一の SRP インターフェイスとしてポート 0 と 1 を使用する場合、0、1、2、および 3 という 4 つのポートすべてを SRP 用に設定する必要があります。同様に、設定が正常に機能するためには、ポート 4 ~ 7、8 ~ 11、12 ~ 15 も、SRP 用または POS 用として同一に設定する必要があります。

手順の概要

1. **configure**
2. **hw-module port *port-number-1* srp location *instance***
3. **hw-module port *port-number-2* srp location *instance***
4. **hw-module port *port-number-3* srp location *instance***
5. **hw-module port *port-number-4* srp location *instance***
6. **end**
または
commit
7. **hw-module location *node-id* reload**

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>hw-module port port-number-1 srp location instance</code> 例: RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/5/cpu0	最初のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。まず、低い数値のポートを偶数にする必要があります (たとえば、0 や 2)。 (注) 16 ポート OC-48c/STM-16c POS/DPT PLIM では、4 つの連続するポートのグループを同様に設定する必要があります (ポート 0 ~ 3、4 ~ 7、8 ~ 11、12 ~ 15)。
ステップ 3 <code>hw-module port port-number-2 srp location instance</code> 例: RP/0/RP0/CPU0:router(config)# hw-module port 1 srp location 0/5/cpu0	2 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。2 番目の高い数値のポートは奇数にする必要があります。 (注) 16 ポート OC-48c/STM-16c POS/DPT PLIM では、4 つの連続するポートのグループを同様に設定する必要があります (ポート 0 ~ 3、4 ~ 7、8 ~ 11、12 ~ 15)。
ステップ 4 <code>hw-module port port-number-3 srp location instance</code> 例: RP/0/RP0/CPU0:router(config)# hw-module port 2 srp location 0/5/cpu0	16 ポート OC-48c/STM-16c POS/DPT PLIM の場合、3 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。まず、低い数値のポートを偶数にする必要があります (たとえば、0 や 2)。 (注) 16 ポート OC-48c/STM-16c POS/DPT PLIM では、4 つの連続するポートのグループを同様に設定する必要があります (ポート 0 ~ 3、4 ~ 7、8 ~ 11、12 ~ 15)。
ステップ 5 <code>hw-module port port-number-4 srp location instance</code> 例: RP/0/RP0/CPU0:router(config)# hw-module port 3 srp location 0/5/cpu0	16 ポート OC-48c/STM-16c POS/DPT PLIM の場合、4 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。2 番目の高い数値のポートは奇数にする必要があります。 (注) 16 ポート OC-48c/STM-16c POS/DPT PLIM では、4 つの連続するポートのグループを同様に設定する必要があります (ポート 0 ~ 3、4 ~ 7、8 ~ 11、12 ~ 15)。

コマンドまたはアクション	目的
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config)# <code>end</code> または RP/0/RP0/CPU0:router(config)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7 <code>hw-module location node-id reload</code></p> <p>例: RP/0/RP0/CPU0:router# <code>hw-module location 0/5/cpu0 reload</code></p>	<p>PLIM をリロードし、hw-module port コマンドを有効にします。</p>



(注)

この設定の変更内容を反映し、SRP インターフェイスを作成するには、PLIM をリロードする必要があります。

この手順を完了すると、次の SRP インターフェイスを使用して、スロット 5 の PLIM で設定できるようになります。

- 0/5/0/0 (ポート 0/5/0/0 および 0/5/0/1 で構成されます)
- 0/5/0/2 (ポート 0/5/0/2 および 0/5/0/3 で構成されます)

OC-48/STM-16 SPA ポートでの SRP のイネーブル化

OC-48/STM-16 SPA ポートで SRP の使用をイネーブルにするには、このタスクを実行する必要があります。

制約事項

4 ポート OC-48/STM-16 POS/RPR SPA または 2 ポート OC-48/STM-16 POS/RPR SPA 上のすべてのポートは、POS モードまたは SRP モードで動作する必要があります。そのため、SRP を使用する場合、すべての SPA ポートでイネーブルにする必要があります。

手順の概要

1. **configure**
2. **hw-module port *port-number-1* srp location *instance* spa-bay *number***
3. **hw-module port *port-number-2* srp location *instance* spa-bay *number***
4. **hw-module port *port-number-3* srp location *instance* spa-bay *number***
5. **hw-module port *port-number-4* srp location *instance* spa-bay *number***
6. **end**
または
commit
7. **hw-module subslot *subslot-id* reload**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	hw-module port <i>port-number-1</i> srp location <i>instance</i> spa-bay <i>number</i> 例: RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/5/cpu0 spa-bay 2	最初のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。まず、低い数値のポートを偶数にする必要があります (たとえば、0 や 2)。
ステップ 3	hw-module port <i>port-number-2</i> srp location <i>instance</i> spa-bay <i>number</i> 例: RP/0/RP0/CPU0:router(config)# hw-module port 1 srp location 0/5/cpu0 spa-bay 2	2 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。2 番目の高い数値のポートは奇数にする必要があります。
ステップ 4	hw-module port <i>port-number-3</i> srp location <i>instance</i> spa-bay <i>number</i> 例: RP/0/RP0/CPU0:router(config)# hw-module port 2 srp location 0/5/cpu0 spa-bay 2	4 ポート OC-48/STM-16 POS/RPR SPA の場合、3 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。まず、低い数値のポートを偶数にする必要があります (たとえば、0 や 2)。
ステップ 5	hw-module port <i>port-number-4</i> srp location <i>instance</i> spa-bay <i>number</i> 例: RP/0/RP0/CPU0:router(config)# hw-module port 3 srp location 0/5/cpu0 spa-bay 2	4 ポート OC-48/STM-16 POS/RPR SPA の場合、4 番目のポートで SRP 機能をイネーブルにします。 (注) SRP インターフェイスを適切に設定するには、2 つの連続する物理ポートが必要です。2 番目の高い数値のポートは奇数にする必要があります。

コマンドまたはアクション	目的
<p>ステップ 6</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config)# end または RP/0/RP0/CPU0:router(config)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7</p> <pre>hw-module subslot subslot-id reload</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# hw-module subslot 0/5/cpu0 reload</pre>	<p>SPA をリロードし、hw-module port コマンドを有効にします。</p> <p>(注) この設定の変更内容を反映し、SRP インターフェイスを作成するには、SPA をリロードする必要があります。</p>

この手順を完了すると、次の SRP インターフェイスを使用して、スロット 5 の SPA で設定できるようになります。

- 0/5/0/0 (ポート 0/5/0/0 および 0/5/0/1 で構成されます)
- 0/5/0/2 (ポート 0/5/0/2 および 0/5/0/3 で構成されます)



(注)

特定の SPA に関連付けられている OC-48/STM-16 ポートで SRP の使用をディセーブルにするには、この項と同じ手順を実行します。ただし、ステップ 2 ~ 5 では、**no hw-module port port-number srp location instance spa-bay number** コマンドを使用します。

OC-48/STM-16 ポートの設定を誤り、その設定を削除する場合、そのポートについてだけ、**no hw-module port port-number srp location instance spa-bay number** コマンドを指定できます。ただし、この処理を実行するのは、ここで説明した場合にだけ実行することをお勧めします。SRP 設定の一部を削除すると、ルータが不確定な状態のままになる可能性があります。

OC-192/STM-64 SPA ポートでの SRP のイネーブル化

OC-192/STM-64 SPA ポートで SRP の使用をイネーブルにするには、このタスクを実行する必要があります。

このタスクを進める前に、SRP インターフェイスが 2 つの OC-192/STM-64 POS/RPR SPA で構成され、それぞれが個別のベイに設置されていることを確認します。また、各 SPA は個別のプロセスで実行されます。そのため、単一の SRP インターフェイスに 2 つの SPA とプロセスがあります。

手順の概要

1. **configure**
2. **hw-module port *port-number-1* srp location instance *spa-bay number***
3. **hw-module port *port-number-2* srp location instance *spa-bay number***
4. **commit**
5. **hw-module subslot *subslot-id-1* shutdown**
6. **hw-module subslot *subslot-id-2* shutdown**
7. **commit**
8. **no hw-module subslot *subslot-id-1* shutdown**
9. **no hw-module subslot *subslot-id-2* shutdown**
10. **commit**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	hw-module port <i>port-number-1</i> srp location instance <i>spa-bay number</i> 例： RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/5/cpu0 spa-bay 0	最初のベイの SPA で SRP 機能をイネーブルにします。
ステップ 3	hw-module port <i>port-number-2</i> srp location instance <i>spa-bay number</i> 例： RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/5/cpu0 spa-bay 1	2 番目のベイの SPA で SRP 機能をイネーブルにします。
ステップ 4	commit 例： RP/0/RP0/CPU0:router(config)# commit	設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、 commit コマンドを使用します。

SRP インターフェイスの設定方法

コマンドまたはアクション	目的
ステップ 5 <code>hw-module subslot subslot-id shutdown</code> 例: RP/0/RP0/CPU0:router(config)# hw-module subslot 0/5/0 shutdown	(注) この設定の変更内容を反映し、SRP インターフェイスを作成するには、各 SPA をリロードする必要があります。そのために、各 SPA をシャットダウンしてから、起動することをお勧めします。各 SPA をリロードするときに、 hw-module subslot subslot-id reload コマンドは推奨されません。このコマンドを使用すると、SRP インターフェイスを構成する 2 つの SPA およびプロセスに同期の問題が発生する可能性があるためです。 SRP ロケーション 0/5/cpu0 のベイ 0 にある SPA をシャットダウンします。
ステップ 6 <code>hw-module subslot subslot-id shutdown</code> 例: RP/0/RP0/CPU0:router(config)# hw-module subslot 0/5/1 shutdown	(注) この設定の変更内容を反映し、SRP インターフェイスを作成するには、各 SPA をリロードする必要があります。そのために、各 SPA をシャットダウンしてから、起動することをお勧めします。各 SPA をリロードするときに、 hw-module subslot subslot-id reload コマンドは推奨されません。このコマンドを使用すると、SRP インターフェイスを構成する 2 つの SPA およびプロセスに同期の問題が発生する可能性があるためです。 SRP ロケーション 0/5/cpu0 のベイ 1 にある SPA をシャットダウンします。
ステップ 7 <code>commit</code> 例: RP/0/RP0/CPU0:router(config)# commit	設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、 commit コマンドを使用します。
ステップ 8 <code>no hw-module subslot subslot-id shutdown</code> 例: RP/0/RP0/CPU0:router(config)# no hw-module subslot 0/5/0 shutdown	SRP ロケーション 0/5/cpu0 のベイ 0 にある SPA をアップ状態に戻します。
ステップ 9 <code>no hw-module subslot subslot-id shutdown</code> 例: RP/0/RP0/CPU0:router(config)# no hw-module subslot 0/5/1 shutdown	SRP ロケーション 0/5/cpu0 のベイ 1 にある SPA をアップ状態に戻します。
ステップ 10 <code>commit</code> 例: RP/0/RP0/CPU0:router(config)# commit	設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、 commit コマンドを使用します。

この手順を完了すると、ポート 0/5/0 および 0/5/1 で構成される SRP インターフェイス 0/5 を構成できるようになります。



(注) 特定の SRP インターフェイスに関連付けられている OC-192/STM-64 ポートで SRP の使用をディセーブルにするには、この項と同じ手順を実行します。ただし、ステップ 2 と 3 では、**no hw-module port port-number srp location instance spa-bay number** コマンドを使用します。

OC-192/STM-64 ポートの設定を誤り、その設定を削除する場合、そのポートについてだけ、**no hw-module port port-number srp location instance spa-bay number** コマンドを指定できます。ただし、この処理を実行するのは、上記で説明した場合にだけ実行することをお勧めします。SRP 設定の一部を削除すると、ルータが不確定な状態のままになる可能性があります。

基本的な SRP 設定の作成

ここでは、基本的な SRP 設定の作成方法について説明します。設定できるパラメータは多数ありますが、このタスクで最も基本的なパラメータについて説明します。



(注) このタスクを実行する前に、インターフェイス上で SRP を有効にする必要があります。「PortPLIM ポートでの SRP のイネーブル化」(P.398) を参照してください。

手順の概要

1. **show interfaces**
2. **configure**
3. **controller sonet interface-path-id clock source internal**
4. **interface srp interface-path-id**
5. **ipv4 address ip-address mask**
6. **srp topology-timer value**
7. **no shutdown**
8. **end**
または
commit
9. **show interfaces srp interface-path-id**
10. **show running-config**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	show interfaces 例： RP/0/RP0/CPU0:router# show interfaces	(任意) 設定されているインターフェイスを表示します。 • また、このコマンドを使用して、ルータが PLIM カードを認識しているかどうかを確認します。
ステップ 2	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。

SRP インターフェイスの設定方法

コマンドまたはアクション	目的
<p>ステップ 3 <code>controller sonet interface-path-id clock source internal</code></p> <p>例: RP/0/RP0/CPU0:router(config)# controller sonet 0/1/0/0 clock source internal RP/0/RP0/CPU0:router(config)# controller sonet 0/1/0/1 clock source internal</p>	<p>SRP インターフェイスを構成する各ポートで、SONET ポート送信クロック ソースを設定します。コントローラ インスタンスは <code>rack/slot/module/port</code> 表記で指定し、internal キーワードには内部クロックを指定します。</p> <p>(注) スペース再利用プロトコル (SRP) インターフェイスでは、内部クロッキングが必要です。</p> <p>(注) SONET コントローラの設定の詳細については、「Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定」を参照してください。</p>
<p>ステップ 4 <code>interface srp interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router(config)# interface srp 0/1/0/0</p>	<p>SRP インターフェイス名と <code>rack/slot/module/port</code> 表記を指定して、インターフェイス コンフィギュレーション モードを開始します。</p>
<p>ステップ 5 <code>ipv4 address ip-address</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.1 255.255.255.224</p>	<p>インターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p>
<p>ステップ 6 <code>srp topology-timer value</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# srp topology-timer 1</p>	<p>(任意) SRP リング上の現在のノードを識別するために送信するリングのトポロジ ディスカバリ メッセージの頻度を指定します。</p>
<p>ステップ 7 <code>no shutdown</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# no shutdown</p>	<p>shutdown 設定を削除します。</p> <ul style="list-style-type: none"> shutdown 設定を削除すると、インターフェイスに強制された管理上のダウン状態が解除され、アップ状態またはダウン状態に移行できるようになります。

	コマンドまたはアクション	目的
ステップ 8	<pre>end または commit 例： RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 9	<pre>show interfaces srp interface-path-id 例： RP/0/RP0/CPU0:router# show interfaces srp 0/1/0/0</pre>	(任意) SRP インターフェイスの設定を表示します。
ステップ 10	<pre>show running-config 例： RP/0/RP0/CPU0:router# show running-config</pre>	(任意) 現在ルータで使用されている設定情報を表示します。

インテリジェント保護スイッチング (IPS) の設定

SRP インターフェイス上で Intelligent Protection Switching (IPS; インテリジェント保護スイッチング) を設定するには、このタスクを実行します。これはオプションのタスクです。

IPS を使用すると、リンクまたはノードが失敗した後に、IP の自己回復および復元や、パフォーマンスモニタリングを実行できます。SRP IPS には次の 2 つのモードがあります。

- 自動 SRP IPS モードは、SRP リングがイベント、ファイバの切断、またはノードのエラーを検出したときに有効になり、トリガ条件がクリアされるまで有効な状態が続きます。トリガがクリアされると、wait-to-restore (WTR) 値が期限切れになるまで、SRP IPS モードは有効な状態が続きます。
- ユーザ設定の SRP IPS モードは、コマンドを入力してすぐに有効になり、ユーザ コマンドによって削除されるか、より優先順位の高い SRP IPS コマンドで上書きされるまで有効な状態が続きます。ユーザ設定のコマンドを無効にするには、**no srp ips request forced-switch global configuration** コマンドまたは **srp remove manual-switch EXEC** コマンドを使用できます。

ユーザ設定の **forced-switch** では、**srp ips request forced-switch** コマンドを入力することで、指定した範囲の両端に優先順位の高い保護スイッチのラップを追加します。たとえば、**srp ips request forced-switch** コマンドを入力すると、DPT PLIM をルータ スロットから取り外す前、またはイベントに反応して、リングの一方に対するデータ トラフィックを強制することができます。

表 17 は、優先順位順（高い方から低い方へ）の IPS 要求一覧です。

表 17 SRP IPS ユーザ要求の説明

SRP IPS 要求	説明
Forced-switch	ユーザ設定の srp ips request forced-switch コマンドを入力することで、指定した範囲の両端に優先順位の高い保護スイッチのラップを追加します。
Manual-switch	ユーザ設定の srp request manual-switch コマンドを入力することで、指定した範囲の両端に優先順位の低い保護スイッチのラップを追加します。



(注) DPT PLIM を取り外す前に、外すインターフェイスの両側で **srp ips request forced-switch** コマンドを使用できます。

特定の範囲で自動設定またはユーザ設定の保護スイッチが要求された場合、保護要求を受信したノードは、(失敗は一方の可能性があるため) 失敗した範囲上の短いパスと、リングに沿った長いパスの両方を使用して、その範囲のもう一方の終端のノードに対して保護要求を発行します。

保護要求がリングに沿って送信されるとき、保護階層が適用されます。たとえば、優先順位の高い Signal Fail (SF) 要求がリングに送信されると、既存の優先順位の低い要求は上書きされます。イベントまたはユーザ設定のコマンドによって優先順位の低い要求が送信され、優先順位の高い要求がリング上にあった場合、優先順位の低い要求は許可されません。



(注) ただし、個別のファイバリンク上で複数の signal-fail 要求と forced-switch 要求が発生する場合、SRP リング上に複数の要求が同時に存在する可能性があります。また、リングは分岐します。

すべての保護スイッチは両方向で実行され、失敗が一方だけの場合でも、送信範囲の両端でラップを追加し、方向を受信します。

手順の概要

1. **configure**
2. **interface srp interface-path-id**
3. **srp ips wtr-timer seconds**
4. **srp ips timer seconds**
5. **srp ips request forced-switch {a | b}**
6. **end**
または
commit
7. **srp {request | remove} manual-switch {a | b} interface srp interface-path-id**
8. **show srp ips interface srp interface-path-id**

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface srp interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface srp 0/1/0/0	<code>rack/slot/module/port</code> 表記で SRP インターフェイス名を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3 <code>srp ips wtr-timer seconds</code> 例: RP/0/RP0/CPU0:router(config-if)# srp ips wtr-timer 60	(任意) ラップの原因が取り除かれた後もラップを残す時間 (秒) を設定します。
ステップ 4 <code>srp ips timer seconds</code> 例: RP/0/RP0/CPU0:router(config-if)# srp ips timer 60 a	(任意) IPS 要求の送信頻度を指定します。デフォルトは 1 秒です。  (注) IPS タイマー値は、リング上のすべてのノードで同じ値にすることをお勧めします。そのため、あるノードで IPS タイマー値を変更した場合、 srp ips timer コマンドを使用して、そのリング上のすべてのノードを変更する必要があります。
ステップ 5 <code>srp ips request forced-switch {a b}</code> 例: RP/0/RP0/CPU0:router(config-if)# srp ips request forced-switch a	(任意) 指定した範囲の両端に、優先順位の高い保護スイッチのラップを追加します。 (注) このコマンドを実行するとノードがディセーブルになるため、必要な場合にだけ実行してください。

コマンドまたはアクション	目的
<p>ステップ 6 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if)# <code>end</code> または RP/0/RP0/CPU0:router(config-if)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7 <code>srp {request remove} manual-switch {a b}</code> <code>interface srp interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router# <code>srp remove manual-switch a interface srp 0/1/0/0</code></p>	<p>(任意) 指定した範囲の両端に、優先順位の低い保護スリッチのラップを追加または削除します。</p> <p>(注) このコマンドは必要な場合にだけ使用してください。</p>
<p>ステップ 8 <code>show srp ips interface srp interface-path-id</code></p> <p>例: RP/0/RP0/CPU0:router# <code>show srp ips interface srp 0/1/0/0</code></p>	<p>(任意) SRP インターフェイスに IPS 構成を表示します。</p>

SRP によるモジュラ サービス品質コマンドライン インターフェイス (MQC) の設定

Modular Quality of Service Command-line interface (MQC; モジュラ サービス品質コマンドライン インターフェイス) を使用して、サービス品質 (QoS) の分類を設定するには、このタスクを実行します。これはオプションのタスクです。



(注)

MQC の詳細については、『[Configuring Modular Quality of Service Packet Classification on Cisco IOS XR Software](#)』および『[Cisco IOS XR Modular Quality of Service Command Reference](#)』を参照してください。

手順の概要

1. configure

2. **class-map match-any** *access-group-name*
3. **match mpls experimental topmost** *exp-value*
4. **exit**
5. **class-map match-any** *access-group-name*
6. **match precedence** *precedence-value*
7. **exit**
8. **policy-map** *policy-name*
9. **class** *class-name*
10. **police cir** *kbps*
11. **set cos** *cos-value*
12. **priority**
13. **exit**
14. **class** *class-name*
15. **priority**
16. **set cos** *cos-value*
17. **exit**
18. **exit**
19. **interface srp** *interface-path-id*
20. **service-policy output** *policy-map*
21. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	class-map match-any <i>class-map-name</i> 例： RP/0/RP0/CPU0:router(config)# class-map match-any voice	クラス マップ コンフィギュレーションモードを開始します。 <ul style="list-style-type: none"> • 指定した名前のクラスとパケットを照合するために使用するクラス マップを作成します。 • match-any を指定すると、トラフィック クラスで受信したトラフィックの場合、一致基準の 1 つに必ず一致し、そのトラフィック クラスの一部と分類されます。

SRP インターフェイスの設定方法

	コマンドまたはアクション	目的
ステップ 3	<pre>match mpls experimental topmost exp-value</pre> <p>例: RP/0/RP0/CPU0:router(config-cmap)# match mpls experimental topmost 4</p>	<p>クラス マップを設定し、最上位のマルチプロトコル ラベル スイッチング (MPLS) ラベルの 3 ビット experimental (EXP) フィールドが、EXP フィールド値に対して検査されるようにします。</p> <ul style="list-style-type: none"> EXP 値の引数は、0 ~ 7 に正確に一致する値と指定されます。
ステップ 4	<pre>exit</pre> <p>例: RP/0/RP0/CPU0:router(config-cmap)# exit</p>	現在のサブモードを終了します。
ステップ 5	<pre>class-map match-any class-map-name</pre> <p>例: RP/0/RP0/CPU0:router(config)# class-map match-any ctrl</p>	<p>クラス マップ コンフィギュレーションモードを開始します。</p> <ul style="list-style-type: none"> 指定した名前のクラスとパケットを照合するために使用するクラス マップを作成します。 match-any を指定すると、トラフィック クラスで受信したトラフィックの場合、一致基準の 1 つに必ず一致し、そのトラフィック クラスの一部と分類されます。
ステップ 6	<pre>match precedence precedence-value</pre> <p>例: RP/0/RP0/CPU0:router(config-cmap)# match precedence internet</p>	<p>(任意) IP precedence 値を一致基準として確認します。</p> <ul style="list-style-type: none"> 有効値の範囲は 0 ~ 63 です。 数値の代わりに予約されたキーワードを指定することもできます。
ステップ 7	<pre>exit</pre> <p>例: RP/0/RP0/CPU0:router(config-cmap)# exit</p>	現在のサブモードを終了します。
ステップ 8	<pre>policy-map policy-name</pre> <p>例: RP/0/RP0/CPU0:router(config)# policy-map srp-policy</p>	<p>ポリシー マップ コンフィギュレーションモードを開始します。</p> <ul style="list-style-type: none"> サービス ポリシーを指定するために、1 つまたは複数のインターフェイスに付加できるポリシー マップを作成または変更します。
ステップ 9	<pre>class class-name</pre> <p>例: RP/0/RP0/CPU0:router(config-pmap)# class voice</p>	作成または変更するポリシーを含むクラス名を指定します。
ステップ 10	<pre>police cir kbps</pre> <p>例: RP/0/RP0/CPU0:router(config-pmap-c)# police cir 2000000</p>	<p>トラフィック ポリシングを設定します。</p> <p>(注) 2000000 は、インターフェイス ライン レートの 10% を表します。</p>
ステップ 11	<pre>set cos cos-value</pre> <p>例: RP/0/RP0/CPU0:router(config-pmap-c)# set cos 4</p>	発信パケットのレイヤ 2 Class of Service (CoS; サービス クラス) 値を設定します。

コマンドまたはアクション	目的
ステップ 12 <code>priority</code> 例: RP/0/RP0/CPU0:router(config-pmap-c)# priority	ポリシー マップに属するトラフィックのクラスに優先順位を指定します。 (注) <code>set cos</code> コマンドを使用し、さらに 2 以上の <code>cos</code> 値を指定している場合にだけ、 <code>priority</code> コマンドを使用してください。
ステップ 13 <code>exit</code> 例: RP/0/RP0/CPU0:router(config-pmap-c)# exit	現在のサブモードを終了します。
ステップ 14 <code>class class-name</code> 例: RP/0/RP0/CPU0:router(config-pmap)# class ctrl	作成または変更するポリシーを含むクラス名を指定します。
ステップ 15 <code>priority</code> 例: RP/0/RP0/CPU0:router(config-pmap-c)# priority	ポリシー マップに属するトラフィックのクラスに優先順位を指定します。 (注) <code>set cos</code> コマンドを使用し、さらに 2 以上の <code>cos</code> 値を指定している場合にだけ、 <code>priority</code> コマンドを使用してください。
ステップ 16 <code>set cos cos-value</code> 例: RP/0/RP0/CPU0:router(config-pmap-c)# set cos 6	発信パケットのレイヤ 2 CoS 値を設定します。
ステップ 17 <code>exit</code> 例: RP/0/RP0/CPU0:router(config-pmap-c)# exit	現在のサブモードを終了します。
ステップ 18 <code>exit</code> 例: RP/0/RP0/CPU0:router(config-pmap)# exit	現在のサブモードを終了します。
ステップ 19 <code>interface srp interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface srp 0/1/0/0	<code>rack/slot/module/port</code> 表記で SRP インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 20 <code>service-policy output policy-map</code> 例: RP/0/RP0/CPU0:router(config-if)# service-policy output srp-policy	インターフェイスのサービス ポリシーとして使用する入力インターフェイスまたは出力インターフェイスにポリシー マップを付加します。 <ul style="list-style-type: none"> • トラフィック ポリシーは、そのインターフェイスから送信されるすべてのトラフィックを評価します。

コマンドまたはアクション	目的
<p>ステップ 21</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

リングへのノードの追加

ここでは、ノードを追加するファイバ上のエリアから、**forced-switch** ラップを挿入する Cisco IOS XR コマンドを使用して、既存の SRP リングにノードを追加する方法について説明します。こうすることで、データトラフィックの損失を最小限に抑えることができます。

この例の目的として、5 番目のノードが 4 ノードリングに追加されます。ノード 5 はノード 1 とノード 4 の間に追加されます。図 10 と図 11 は、単一の DPT PLIM を使用した物理構成図です。図 12 と図 13 は、論理構成図です。

手順の概要

1. **configure**
2. **interface srp interface-path-id**
3. **srp ips request forced-switch {a | b}**
4. **end**
または
commit
5. **interface srp interface-path-id**
6. **no srp ips request forced-switch {a | b}**
7. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<pre>configure</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router1# configure</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<pre>interface srp interface-path-id</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router1(config)# interface srp 0/1/0/0</pre>	rack/slot/module/port 表記でノード 1 の SRP インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	<pre>srp ips request forced-switch {a b}</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router1(config-if)# srp ips request forced-switch a</pre>	<p>(任意) 指定した範囲の両端に、優先順位の高い保護スイッチのラップを追加します。このステップで、切断されるファイバ上でノード 1 からの発信トラフィックが停止し、サイド A 上のノード 1 の横にラップが作成されます。</p> <p>(注) srp ips request forced-switch コマンドを使用しない場合、ステップ 5 を実行するとすぐに、信号エラーがノード 1 とノード 4 で検出され、それらのノード間のエラーを避けて、2 つの信号エラーのラップが自動的に挿入されます。データの損失を最小限に抑えるために、srp ips request forced-switch コマンドを使用することをお勧めします。</p>
ステップ 4	<pre>end</pre> <p>または</p> <pre>commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router1(config-if)# end</pre> <p>または</p> <pre>RP/0/RP0/CPU0:router1(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 5	ノード 1 からノード 4 に接続する光ファイバケーブルを取り外します。	

SRP インターフェイスの設定方法

コマンドまたはアクション	目的
ステップ 6 ケーブルを接続して新しいノードを追加する操作で、受信 (RX) および送信 (TX) のケーブル接続の関係を確認します。	図 13 を参照してください。
ステップ 7 <code>interface srp interface-path-id</code> 例: RP/0/RP0/CPU0:router1(config)# interface srp 0/1/0/0	<i>rack/slot/module/port</i> 表記でノード 1 の SRP インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8 <code>no srp ips request forced-switch {a b}</code> 例: RP/0/RP0/CPU0:router1(config-if)# no srp ips request forced-switch a	指定した範囲の両端に、優先順位の高い保護スイッチのラップを削除します。この操作で、またノード 1 からトラフィックが送信されるようになります (図 13 を参照)。 (注) ステップ 3 を実行した場合、 <code>no srp ips request forced-switch</code> コマンドを使用してラップを削除する必要があります。ステップ 3 を実行しなかった場合、WTR タイマーが期限切れになったときに自動的に削除されます。
ステップ 9 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router1(config-if)# end または RP/0/RP0/CPU0:router1(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、<code>commit</code> コマンドを使用します。
ステップ 10 <code>show srp ips</code> 例: RP/0/RP0/CPU0:router5# show srp ips	ラップが消去され、新しいノードがリングに追加されたことを確認します (図 13 を参照)。
ステップ 11 <code>show srp errors</code> 例: RP/0/RP0/CPU0:router5# show srp errors	新しいリングの設定に問題がないことを確認します。エラーがある場合、LED のステータスを確認し、問題の内容を判断します。

図 10 SRP リング上の 4 ルータ (各ルータが 2 つの物理ポートを使用)

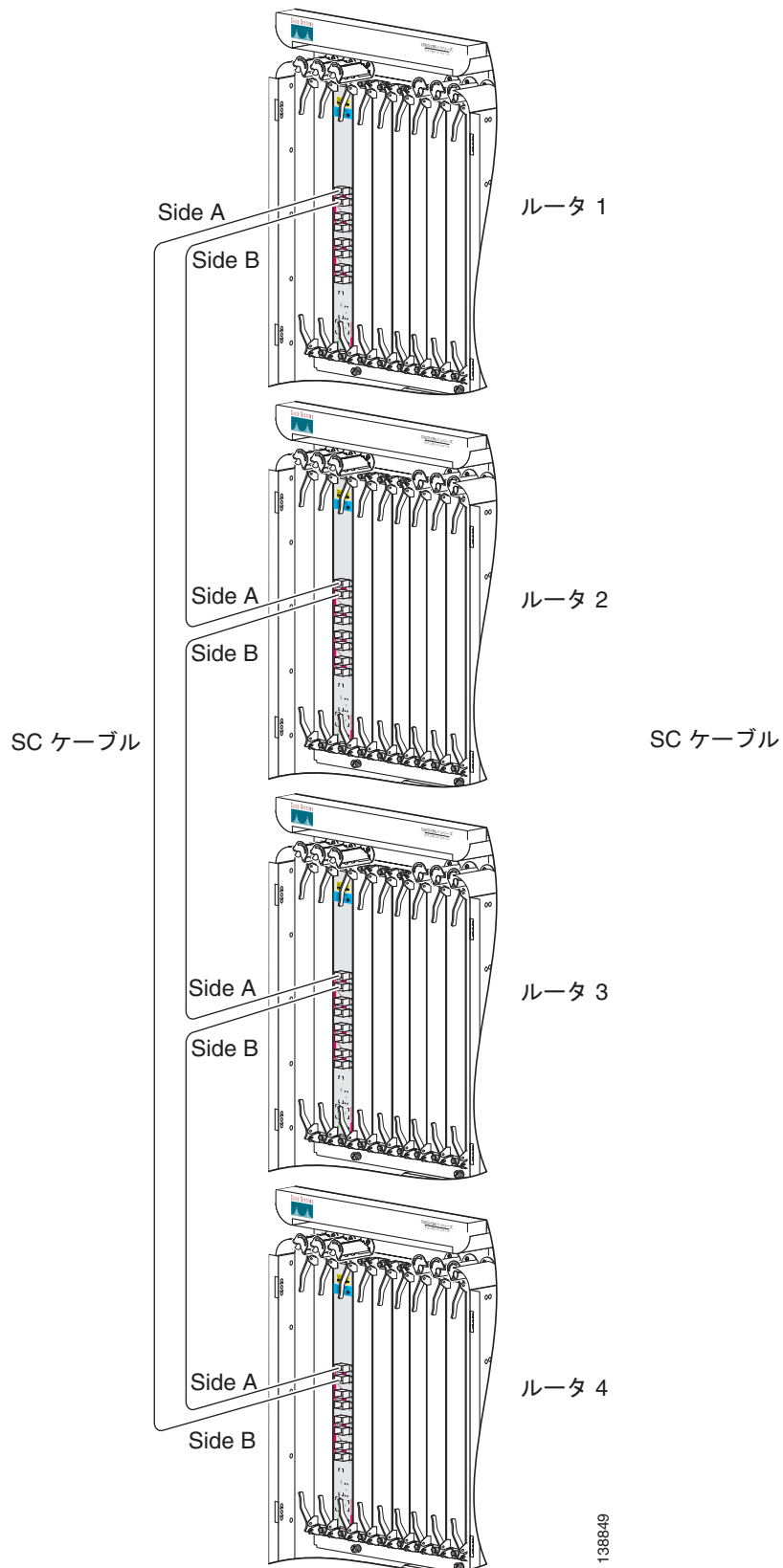


図 11 SRP リングへのルータの追加 (各ルータが 2 つの物理ポートを使用)

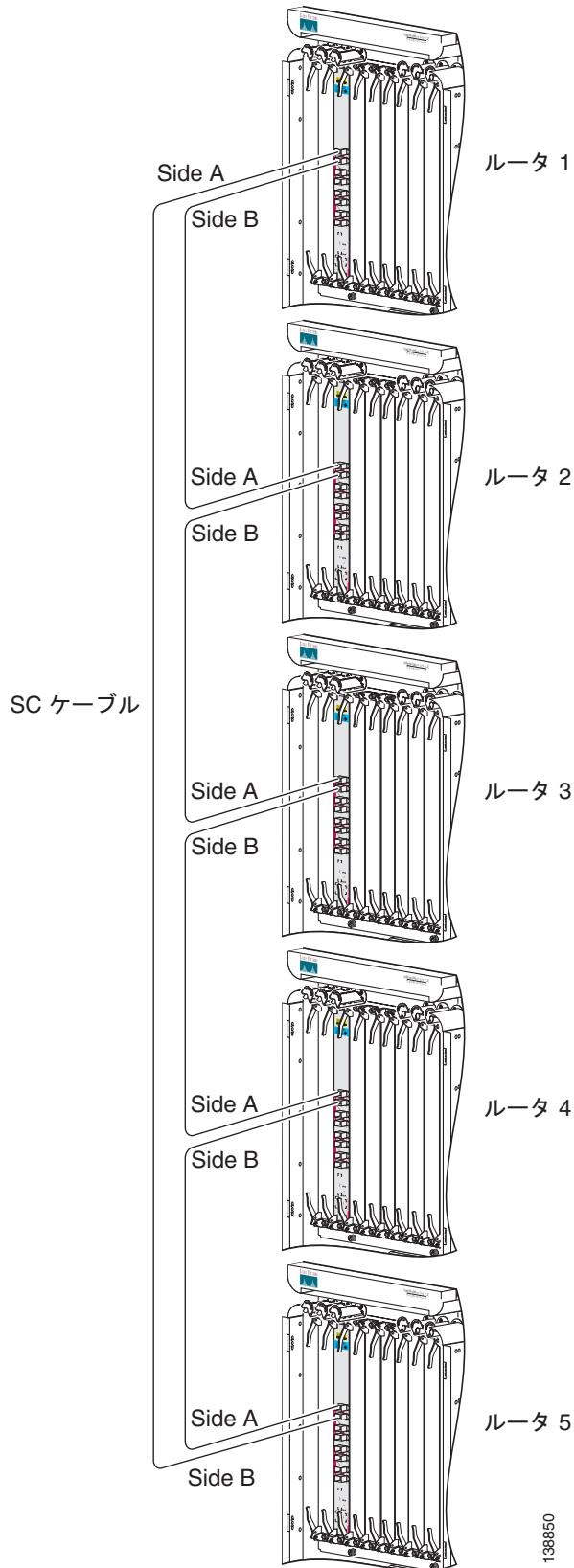
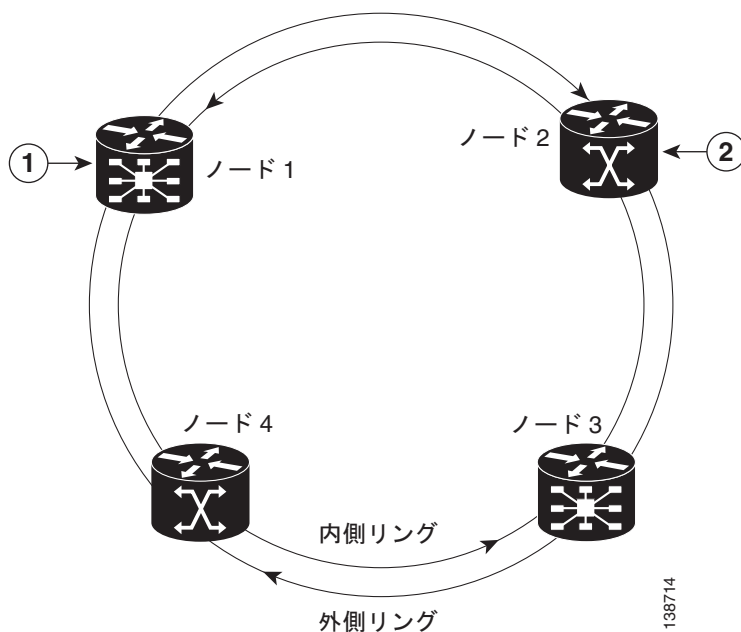


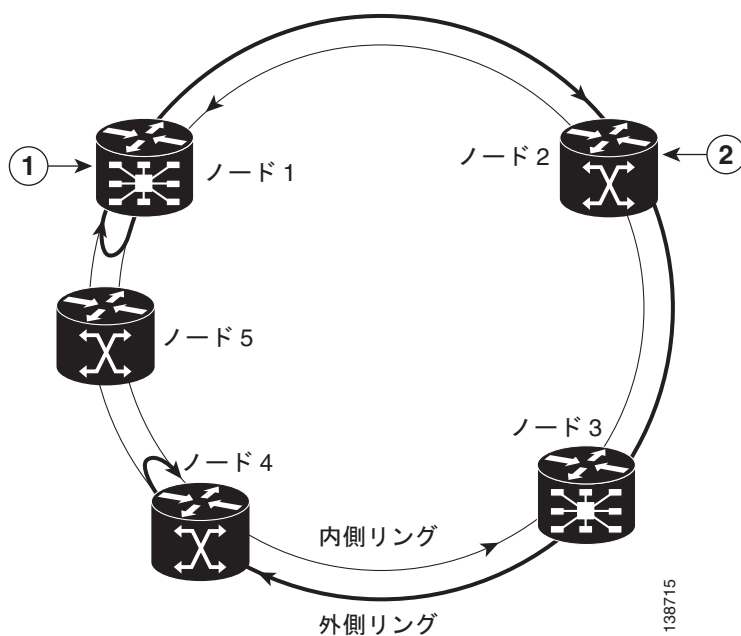
図 12 4 ノードの SRP リング トポロジ



1 Cisco CRS-1 ルータ

2 Cisco XR 12000 シリーズ ルータ

図 13 ラップしたリングに 5 番目のノードを追加した SRP リング トポロジ



1 Cisco CRS-1 ルータ

2 Cisco XR 12000 シリーズ ルータ

SRP インターフェイスの設定例

ここでは、次の設定例について説明します。

- 「SRP のイネーブル化：例」(P.420)
- 「基本的な SRP の設定：例」(P.420)
- 「SRP によるモジュラ QoS の設定：例」(P.421)

SRP のイネーブル化：例

次に、PLIM ポートで SRP をイネーブルにする例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/3/CPU0
RP/0/RP0/CPU0:router(config)# hw-module port 1 srp location 0/3/CPU0
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# end
RP/0/RP0/CPU0:router# hw-module node 0/3/CPU0 reload
```

<Wait for LC to be reloaded, and interface created. Or can use 'preconfigure'...>

次に、4 個の OC-48/STM-16 SPA ポートで SRP をイネーブルにする例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/3/CPU0 spa-bay 2
RP/0/RP0/CPU0:router(config)# hw-module port 1 srp location 0/3/CPU0 spa-bay 2
RP/0/RP0/CPU0:router(config)# hw-module port 2 srp location 0/3/CPU0 spa-bay 2
RP/0/RP0/CPU0:router(config)# hw-module port 3 srp location 0/3/CPU0 spa-bay 2
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# end
RP/0/RP0/CPU0:router# hw-module subslot 0/3/CPU0 reload
```

<Wait for LC to be reloaded, and interface created. Or can use 'preconfigure'...>

次に、1 つの SRP インターフェイスを構成する 2 つの OC-192/STM-64 SPA ポートで SRP をイネーブルにする例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/3/CPU0 spa-bay 0
RP/0/RP0/CPU0:router(config)# hw-module port 0 srp location 0/3/CPU0 spa-bay 1
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# hw-module subslot 0/3/0 shutdown
RP/0/RP0/CPU0:router(config)# hw-module subslot 0/3/1 shutdown
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# no hw-module subslot 0/3/0 shutdown
RP/0/RP0/CPU0:router(config)# no hw-module subslot 0/3/1 shutdown
RP/0/RP0/CPU0:router(config)# commit
```

基本的な SRP の設定：例

次に、SRP の基本的なインターフェイス設定を指定する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# controller SONET 0/3/0/0 clock source internal
RP/0/RP0/CPU0:router(config)# controller SONET 0/3/0/1 clock source internal
RP/0/RP0/CPU0:router(config)# interface SRP 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
```

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router(config)# end
```

SRP によるモジュラ QoS の設定 : 例

次に、2つの QoS クラスを設定する例を示します。1つは音声トラフィック用で、MPLS EXP ビット値は 4 です。もう 1つは制御トラフィック用で、IP precedence 値は 6 です。どちらのクラスのトラフィックも、SRP のハイ プライオリティ キューに送信され、優先順位の高い SRP (4 と 6) でマーキングされます。

```
Last configuration change at 04:56:06 UTC Tue Sep 06 2005 by lab
!
hostname router
class-map match-any ctrl
  match precedence internet
!
class-map match-any voice
  match mpls experimental topmost 4
!
policy-map srp-policy
  class voice
    police cir 2000000
    set cos 4
    priority
  !
  class ctrl
    priority
    set cos 6
  !
!
interface SRP0/7/0/0
  description "Connected to 3-nodes ring"
  service-policy output srp-policy
  ipv4 address 30.30.30.2 255.255.255.0
```

その他の参考資料

ここでは、SRP インターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』

■ その他の参考資料

内容	参照先
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Task ID Reference Guide』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
-	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC-2892	『The Cisco SRP MAC Layer Protocol』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのクリア チャネル SONET コントローラの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータ上でのクリア チャネル SONET コントローラの設定について説明します。SONET コントローラの設定は、Cisco IOS XR ソフトウェアを使用するルータ上で Packet-over-SONET/SDH (POS) を設定するための準備手順です。

SONET では、多重化されたデジタル トラフィックのため光信号と同期フレーム構造を定義できます。これは、米国規格協会 (ANSI) T1.105、ANSI T1.106、および ANSI T1.117 で規定されている、光ネットワークのための速度と形式を定義した規格セットです。



(注)

Cisco CRS-1 ルータは、クリア チャネル SONET コントローラだけをサポートしています。Cisco XR 12000 シリーズ ルータは、1 ポート チャネライズド OC-3 および OC-12 SPA 上で、クリア チャネル SONET コントローラとチャネライズド SONET コントローラの両方をサポートしています。チャネライズド SONET コントローラの設定の詳細については、「[Cisco IOS XR ソフトウェアでのチャネライズド SONET の設定](#)」モジュールを参照してください。

レイヤ 1 SONET コントローラを設定するためのコマンドは、『[Cisco IOS XR Interface and Hardware Component Command Reference](#)』に記載されています。

Cisco IOS XR ソフトウェアでの SONET コントローラの設定機能の履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.0	変更ありません。
リリース 3.2	Cisco XR 12000 シリーズ ルータのサポートが追加されました。
リリース 3.3.0	次のハードウェアについて、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。 <ul style="list-style-type: none">• Cisco XR 12000 SIP-401• Cisco XR 12000 SIP-501• Cisco XR 12000 SIP-601 Cisco XR 12000 シリーズ ルータ上で、2 ポート OC-48 POS/RPR SPA のサポートが追加されました。 Cisco CRS-1 ルータ上で、1 ポート OC-768c/STM-256c POS PLIM のサポートが追加されました。

リリース 3.4.0	次のハードウェアについて、Cisco CRS-1 ルータでのサポートが追加されました。 <ul style="list-style-type: none"> • 2 ポート OC-48/STM-16 POS SPA • 4 ポート OC-48/STM-16 POS SPA
リリース 3.5.0	次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。 <ul style="list-style-type: none"> • 1 ポート チャネライズド OC-3 SPA • 1 ポート チャネライズド OC-12 SPA • 2 ポート OC-12 POS • 4 ポート OC-12 POS • 8 ポート OC-12 POS • 4 ポート OC-3 POS • 8 ポート OC-3 POS
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	次の項で、 delay trigger line コマンドが line delay trigger に更新されました。 <ul style="list-style-type: none"> • クリア チャネル SONET コントローラの設定方法 • Fast Reroute がトリガーされないようにするための hold-off タイマーの設定

この章の構成

- 「クリア チャネル SONET コントローラを設定するための前提条件」 (P.424)
- 「SONET コントローラの設定に関する情報」 (P.425)
- 「クリア チャネル SONET コントローラの設定方法」 (P.428)
- 「SONET コントローラの設定例」 (P.438)
- 「関連情報」 (P.440)
- 「その他の参考資料」 (P.440)

クリア チャネル SONET コントローラを設定するための前提条件

SONET コントローラを設定する前に、次の作業が終了し条件が満たされていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンドタスク ID は、各コマンドリファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR Software System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

- 少なくとも次のいずれかの物理レイヤ インターフェイス モジュール (PLIM) カードがシャーシに搭載されていること。
 - 4 ポート OC-3c/STM-1 POS SPA
 - 8 ポート OC-12c/STM-4 POS SPA
 - 2 ポート OC-48/STM-16 POS SPA (Cisco XR 12000 シリーズ ルータのみ)
 - 4 ポート OC-48/STM-16 POS SPA (Cisco XR 12000 シリーズ ルータのみ)
 - 16 ポート OC-48c/STM-16c POS
 - 4 ポート OC-192c/STM-64c POS
 - 1 ポート OC-192c/STM-64 POS/RPR XFP SPA
 - 1 ポート OC-768c/STM-256c POS PLIM
- SONET コントローラ名とインスタンス ID を、汎用表記 *rack/slot/module/port* で指定する方法を知っていること。SONET コントローラ名とインスタンス ID は、**controller sonet** コマンドが必要です。

SONET コントローラの設定に関する情報

SONET コントローラを設定するには、次の概念について理解する必要があります。

- [「SONET コントローラの概要」 \(P.425\)](#)
- [「SONET コントローラのデフォルト設定値」 \(P.426\)](#)
- [「SONET APS」 \(P.427\)](#)

SONET コントローラの概要

Cisco IOS XR ソフトウェアをサポートするルータでは、ラインカード上の物理ポートはコントローラと呼ばれます。POS インターフェイスまたは SRP インターフェイスを設定する前に、SONET コントローラを設定する必要があります。

物理 SONET ポートを設定するために使用するコマンドは、SONET コントローラ コンフィギュレーション モードにグループ化されています。SONET コントローラ 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 **controller sonet** コマンドを入力します。

ルータは SONET コントローラをレイヤ 1 およびレイヤ 2 の処理で使用します。



(注)

パス UNEQ は、OC-768 カードではサポートされていません。そのため、UNEQ-P アラームと PPLM アラームは、OC-768 インターフェイスで受信される未実装の C2 バイトに対してレポートされません。シスコは、ERDI-P 規格のうち、UNEQ-P コードを除くすべてのエラー コードをサポートしています。

SONET コントローラのデフォルト設定値

表 18 は、SONET コントローラに存在するデフォルト コンフィギュレーション パラメータの一部を説明したものです。

表 18 SONET コントローラのデフォルト設定値

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
SONET コントローラの次のアラームのレポート <ul style="list-style-type: none"> ビット 1 (B1) Bit Error Rate (BER; ビット誤り率) しきい値超過アラート (TCA) エラー ビット 2 (B2) BER TCA エラー 信号障害 BER エラー Section Loss Of Frame (SLOF; セクションフレーム損失) エラー Section Loss Of Signal (SLOS; セクション信号消失) エラー 	enabled	デフォルトでイネーブルになっているアラームのレポートをディセーブルにするには、SONET/SDH コンフィギュレーション モードで no report [b1-tca b2-tca sf-ber slof slos] コマンドを使用します。 Line Alarm Indication Signal (LAIS; ラインのアラーム表示信号)、Line Remote Defect Indication (LRDI; ラインのリモート障害表示)、信号劣化 BER エラーのレポートをイネーブルにするには、SONET/SDH コンフィギュレーション モードで report [lais lrldi sd-ber] コマンドを使用します。
SONET パス コントローラの次のアラームのレポート <ul style="list-style-type: none"> ビット 3 (B3) BER TCA エラー Path Loss Of Pointer (PLOP; パス ポインタ損失) エラー 	enabled	SONET パス コントローラ上で B3 BER TCA または PLOP レポートをディセーブルにするには、SONET/SDH パス コンフィギュレーション サブモードで no report b3-tca コマンドまたは no report plop コマンドを入力します。 Path Alarm Indication Signal (PAIS; パス アラーム検出信号)、Path Payload Mismatch (PPLM; パス ペイロードミスマッチ)、Path Remote Defect Indication (PRDI; パス リモート障害検出)、Path Trace Identity Mismatch (PTIM) エラーのレポートをイネーブルにするには、SONET/SDH パス コンフィギュレーション サブモードで report [pais pplm prdi ptim] コマンドを使用します。

表 18 SONET コントローラのデフォルト設定値 (続き)

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
Synchronous Payload Envelope (SPE; 同期ペイロードエンベロープ) スクランプリング	enabled	SONET コントローラで SPE スクランプリングをディセーブルにするには、SONET コントローラ コンフィギュレーション サブモードで path scrambling disable コマンドを入力します。
キープアライブ タイマー	enabledrep	キープアライブ タイマーをオフにするには、インターフェイス コンフィギュレーション モードで keepalive disable コマンドを入力します。

SONET APS

APS 機能を使用すると、障害時に Packet-over-SONET (POS) インターフェイスのスイッチオーバーが可能になります。この機能は、SONET 機器を交換機器に接続する際に多くの場合必要になります。APS は、SONET ネットワークで、「現用」POS インターフェイスのバックアップとして「保護」POS インターフェイスを使用するメカニズムを指します。現用インターフェイスが障害になると、保護インターフェイスがすぐにトラフィック負荷を引き受けます。現用インターフェイスとその保護インターフェイスは、「APS グループ」を構成します。

Cisco IOS XR ソフトウェアで、SONET APS コンフィギュレーションにより、各冗長回線ペアの現用回線と保護回線が定義されます。現用回線は、プライマリ (優先される) 回線であり、回線が動作可能であるかぎり、通信はその回線上で行われます。現用回線が障害になった場合、APS は保護回線へのスイッチオーバーを開始します。2 つのルータ間で APS が適切に動作するためには、1 つのルータの現用回線が、他のルータでも現用回線になっている必要があります。同じことは、保護回線にも当てはまります。

SONET APS グループで、各接続は双方向または単方向、リバーティプまたは非リバーティプです。同じ信号ペイロードが現用インターフェイスと保護インターフェイスに送信されます。現用インターフェイスと保護インターフェイスは、同じカードの 2 つのポートで終端していても、同じルータ内の別のカードで終端していても、2 台の異なるルータで終端していても構いません。

保護インターフェイスは、劣化、チャネル信号消失、手動介入の際に、アクティブ化するか非アクティブ化するかを現用インターフェイスに対して指示します。現用インターフェイスと保護インターフェイスの間の通信が失われた場合、現用ルータは、保護回線が存在しないかのように、現用インターフェイスの完全な制御を引き受けます。

APS グループでは、それぞれの回線は「チャネル」と呼ばれます。双方向モードでは、受信チャネルと送信チャネルがペアで切り替えられます。単方向モードでは、送信チャネルと受信チャネルは独立して切り替えられます。たとえば、双方向モードで、現用インターフェイスの受信チャネルでチャネル信号消失が発生した場合、受信チャネルと送信チャネルの両方が切り替えられます。

クリア チャネル SONET コントローラの設定方法

ここでは、次の手順について説明します。

- 「クリア チャネル SONET コントローラの設定」 (P.428)
- 「SONET APS の設定」 (P.432)
- 「Fast Reroute がトリガーされないように hold-off タイマーを設定する」 (P.437)

クリア チャネル SONET コントローラの設定

ここでは、POS インターフェイスと SRP インターフェイスを設定するための前提条件として SONET コントローラを設定する方法について説明します。

前提条件

- Cisco IOS XR ソフトウェアが動作するルータに POS ラインカードまたは SPA が搭載されている必要があります。
- ファイバ障害または機器障害から回復できるようにするには、「SONET APS の設定」 (P.432) の説明に従って、ルータ上で SONET APS を設定します。

手順の概要

1. `show version`
2. `show interfaces [pos | srp] interface-path-id`
3. `configure`
4. `controller sonet interface-path-id`
5. `clock source {internal | line}`
6. `line delay trigger value`
7. `line delay clear value`
8. `framing {sdh | sonet}`
9. `loopback {internal | line}`
10. `overhead {j0 | s1s0} byte-value`
11. `path keyword [values]`
12. `end`
または
`commit`
13. `show controllers sonet interface-path-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	show version 例: RP/0/RP0/CPU0:router# show version	(任意) 現在のソフトウェア バージョンを表示します。また、ルータがモジュラ サービス カードを認識していることを確認する場合にも使用できます。
ステップ 2	show interfaces [pos srp] <i>interface-path-id</i> 例: RP/0/RP0/CPU0:router# show interface pos 0/1/0/0	(任意) 設定済みのインターフェイスを表示し、各インターフェイス ポートのステータスを確認します。
ステップ 3	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 4	controller sonet interface-path-id 例: RP/0/RP0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション サブモードを開始し、SONET コントローラ名とインスタンス ID を、 <i>rack/slot/module/port</i> 表記で指定します。
ステップ 5	clock source {internal line} 例: RP/0/RP0/CPU0:router(config-sonet)# clock source internal	SONET ポート送信クロック ソースを設定します。 internal キーワードを指定すると内部クロックが設定され、 line キーワードを指定すると回線から再生したクロックが設定されます。 <ul style="list-style-type: none"> ネットワークからクロッキングを得られる場合は、必ず line キーワードを使用します。internal キーワードは、2 台のルータがバックツージャックで接続されているか、クロッキングが利用できないファイバで接続されている場合に使用します。 デフォルトは line クロックです。 (注) スペース再利用プロトコル (SRP) インターフェイスでは、内部クロッキングが必要です。
ステップ 6	line delay trigger value 例: RP/0/RP0/CPU0:router(config-sonet)# line delay trigger 3000	(任意) SONET 回線遅延トリガー値を設定します。トリガー値の範囲は 0 ~ 60000 ミリ秒で、デフォルトの遅延トリガー値は 0 ミリ秒です。
ステップ 7	line delay clear value 例: RP/0/RP0/CPU0:router(config-sonet)# line delay clear 4000	(任意) SONET 回線遅延トリガー アラームがクリアされるまでの時間を設定します。範囲は 1000 ~ 180000 ミリ秒で、デフォルトは 10 秒です。
ステップ 8	framing {sdh sonet} 例: RP/0/RP0/CPU0:router(config-sonet)# framing sonet	(任意) コントローラ フレーム構成を設定します。同期デジタル ハイアラキー (SDH) フレーム構成の場合は sdh キーワードを、SONET フレーム構成の場合は sonet キーワードを使用します。 SONET フレーム構成 (sonet) がデフォルトです。

■ クリア チャネル SONET コントローラの設定方法

コマンドまたはアクション	目的
ステップ 9 <code>loopback {internal line}</code> 例: RP/0/RP0/CPU0:router(config-sonet)# loopback internal	(任意) ループバック用の SONET コントローラを設定します。 internal キーワードは、内部 (ターミナル) ループバックを選択し、 line キーワードは回線 (ファシリティ) ループバックを選択します。
ステップ 10 <code>overhead {j0 s1s0} byte-value</code> 例: RP/0/RP0/CPU0:router(config-sonet)# overhead s1s0	(任意) コントローラのオーバーヘッドを設定します。 j0 キーワードは STS ID (J0/C1) バイトを指定し、 s1s0 キーワードは H1 バイトのビット s1 および s0 を指定します。 <ul style="list-style-type: none"> • j0 キーワードのデフォルト バイト値は 0xcc、s1s0 キーワードのデフォルト バイト値は 0 です。 • j0 と s1s0 の有効な値の範囲は 0 ~ 255 です。

コマンドまたはアクション	目的
<p>ステップ 11 <code>path keyword [values]</code></p> <p>例 : RP/0/RP0/CPU0:router(config-sonet) # path delay trigger 25</p>	<p>(任意) SONET コントローラ パス値を設定します。</p> <p>キーワードの定義は次のとおりです。</p> <ul style="list-style-type: none"> • ais-shut : シャットダウン時のパス アラーム検出信号 (PAIS) の送信を設定します。 • delay trigger value : SONET パス遅延または遅延トリガー値を設定します。value 引数には、0 ~ 60000 ミリ秒の範囲の数値を指定します。デフォルト値は 0 ミリ秒です。 • overhead [c2 byte-value j1 line] : SONET POH バイト値またはビット値を設定します。STS SPE コンテント (C2) バイトを指定するには c2 キーワードを入力し、byte-value 引数に 0 ~ 255 の範囲の数値を指定します。SONET パストレース (J1) バッファを設定するには、j1 キーワードを入力し、line 引数にパストレース バッファ ID を (ASCII テキストで) 指定します。 • report [b3-tca pais plop pplm prdi ptim] : SONET パスアラーム レポートを設定します。レポートするアラームと、アラームを発生させるビット誤り率 (BER) しきい値を指定します。デフォルトでは、B3 BER しきい値超過アラーム (TCA) とパス ポインタ損失 (PLOP) レポートがイネーブルになっています pais キーワードを指定すると、PAIS レポート ステータスが設定されます。pplm は、パス ペイロード ミスマッチ (PPLM) 障害レポート ステータスを設定します。prdi は、パス リモート障害検出 (PRDI) レポート ステータスを設定します。ptim は、Path Trace Identity Mismatch (PTIM) 障害レポート ステータスを設定します。 <p>SONET/SDH パス コンフィギュレーション サブモードで no report b3-tca コマンドおよび no report plop コマンドを入力すると、それぞれ B3 BER TCA および PLOP レポート ステータスがディセーブルになります。</p> <ul style="list-style-type: none"> • scrambling disable : SPE スクランプリングをディセーブルにします。SPE スクランプリングはデフォルトでイネーブルになっています。 • threshold b3-tca BER : SONET パス BER しきい値を設定します。BER 引数には、3 ~ 9 の範囲の数値を指定します。しきい値は、ビット誤り率を決定するときに、10 の負の指数として解釈されます。たとえば、値 5 は、ビット誤り率が 10 のマイナス 5 乗であることを意味します。デフォルトの BER しきい値は 6 です。 • uneq-shut : シャットダウン時の未実装 (UNEQ) の送信を設定します。

コマンドまたはアクション	目的
<p>ステップ 12</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-sonet)# end または RP/0/RP0/CPU0:router(config-sonet)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 13</p> <pre>show controllers sonet interface-path-id</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show controllers sonet 0/1/0/0</pre>	<p>SONET コントローラの設定を確認します。</p>

次に行う作業

- 関連する POS インターフェイスを設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- 関連する SRP インターフェイスを設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの SRP インターフェイスの設定](#)」モジュールを参照してください。

SONET APS の設定

SONET APS は、SONET 回線層でファイバ（外部）障害または機器（インターフェイスおよび内部）障害からの回復機能を提供します。ここでは、ルータで基本的な自動保護スイッチング（APS）を設定する方法と、**aps group** コマンドを使用して、ルータ上で複数の保護インターフェイスまたは現用インターフェイスを設定する方法について説明します。

コンフィギュレーションを確認する場合や、スイッチオーバーが発生したかどうかを確認するには、**show aps** コマンドを使用します。

前提条件

Cisco IOS XR ソフトウェアが動作するルータに POS ラインカードまたは SPA が搭載されている必要があります。

制約事項

2つのルータ間で APS が適切に動作するためには、1つのルータの現用回線が、他のルータでも現用回線になっている必要があります。同じことは、保護回線にも当てはまります。

手順の概要

1. **configure**
2. **aps group *number***
3. **channel {0 | 1} local sonet *interface***
4. APS グループ内の各チャンネルに対してステップ 3 を繰り返します。
5. **exit**
6. **interface loopback *number***
7. **ipv4 address *ip-address mask***
8. **exit**
9. **interface pos *interface-path-id***
10. **ipv4 address *ip-address mask***
11. **pos crc {16 | 32}**
12. **keepalive {*seconds* | **disable**}**
13. **no shutdown**
14. グループ内の各チャンネルに対してステップ 9 ~ 13 を繰り返します。
15. **exit**
16. **controller sonet *interface-path-id***
17. **ais-shut**
18. **path scrambling disable**
19. **clock source {**internal** | **line**}**
20. グループの各チャンネルに対してステップ 16 ~ 19 を繰り返します。
21. **end**
または
commit
22. **exit**
23. **exit**
24. **show aps**
25. **show aps group [*number*]**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	aps group number 例： RP/0/RP0/CPU0:router(config)# aps group 1	指定した番号を持つ APS グループを追加して、APS グループ コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • aps group コマンドは、グローバル コンフィギュレーション モードで使用します。 • グループを削除するには、このコマンドの no 形式を使用します。たとえば、no aps group number のように入力します。ここで、値の範囲は 1 ~ 255 です。 <p>(注) aps group コマンドを使用するには、aps コマンドの適切なタスク ID に関連付けられたユーザグループのメンバーでなければなりません。</p> <p>(注) aps group コマンドは、設定する保護グループが 1 つだけの場合でも使用します。</p>
ステップ 3	channel {0 1} local sonet interface 例： RP/0/RP0/CPU0:router(config-aps)# channel 0 local SONET 0/0/0/1	APS グループのチャンネルを作成します。 0 は保護チャンネルを指定し、 1 は現用チャンネルを指定します。 (注) 保護チャンネルがローカルな場合、現用チャンネルを割り当てる前に、 channel コマンドを使用して割り当てる必要があります。
ステップ 4	グループ内の各チャンネルに対してステップ 3 を繰り返します	-
ステップ 5	exit	APS グループ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 6	interface loopback number 例： RP/0/RP0/CPU0:router(config)# interface loopback 1	(任意) 2 台のルータによる APS が望ましい場合にループバック インターフェイスを設定し、ループバック インターフェイスのインターフェイス コンフィギュレーション モードを開始します。 (注) この例で、ループバック インターフェイスは相互接続として使用されています。
ステップ 7	ipv4 address ip-address mask 例： RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.0.1 255.255.255.224	ループバック インターフェイスに IPV4 アドレスおよびサブネット マスクを割り当てます。
ステップ 8	exit	ループバック インターフェイスのインターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 9 <code>interface pos interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface POS 0/2/0/0	ステップ 3 で選択したチャネルのインターフェイスを接続し、POS インターフェイスのインターフェイス コンフィギュレーション モードを開始します。
ステップ 10 <code>ipv4 address ip-address mask</code> 例: RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.0.1 255.255.255.224	POS インターフェイスに IPv4 アドレスとサブネット マスクを割り当てます。
ステップ 11 <code>pos crc {16 32}</code> 例: RP/0/RP0/CPU0:router(config-if)# pos crc 32	チャネルの CRC 値を選択します。16 ビットの CRC モードを指定するには 16 キーワード、32 ビットの CRC モードを指定するには 32 キーワードを入力します。 (注) デフォルト CRC は 32 です。
ステップ 12 <code>keepalive {seconds disable}</code> 例: RP/0/RP0/CPU0:router(config-if)# keepalive 3 または RP/0/RP0/CPU0:router(config-if)# keepalive disable	リンク制御プロトコル (LCP) がピアに ECHOREQ を送信する頻度 (秒) を指定します。デフォルトのキープアライブ インターバルは 10 秒です。 システムをデフォルトのキープアライブ インターバルに戻すには、 no keepalive コマンドを使用します。 キープアライブ タイマーをディセーブルにするには、 keepalive disable コマンドを使用します。
ステップ 13 <code>no shutdown</code> 例: RP/0/RP0/CPU0:router(config-if)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> shutdown 設定を削除すると、インターフェイスで強制管理ダウンが削除され、インターフェイスがアップまたはダウン状態に移行できるようになります (親 SONET レイヤが管理ダウンに設定されていないことが前提です)。
ステップ 14 グループ内の各チャネルに対してステップ 9 ~ 13 を繰り返します。	-
ステップ 15 <code>exit</code>	POS インターフェイスのインターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
ステップ 16 <code>controller sonet interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# controller sonet 0/1/0/0	SONET コントローラ コンフィギュレーション モードを開始し、SONET コントローラ名とインスタンス ID を <code>rack/slot/module/port</code> 表記で指定します。
ステップ 17 <code>ais-shut</code> 例: RP/0/RP0/CPU0:router(config-sonet)# ais-shut	シャットダウン時の Alarm Indication Signal (AIS; アラーム表示信号) など、POS パス値を設定します。
ステップ 18 <code>path scrambling disable</code> 例: RP/0/RP0/CPU0:router(config-sonet)# path scrambling disable	(任意) 同期ペイロードエンベロープ (SPE) のスクランプリングをディセーブルにします。 (注) SPE スクリングはデフォルトでイネーブルです。

■ クリア チャネル SONET コントローラの設定方法

コマンドまたはアクション	目的
<p>ステップ 19 <code>clock source {internal line}</code></p> <p>例: RP/0/RP0/CPU0:router(config-sonet)# clock source internal</p>	<p>SONET ポートの TX クロック ソースを設定します。 internal キーワードを指定すると内部クロックが設定され、line キーワードを指定すると回線から再生されたクロックが設定されます。</p> <ul style="list-style-type: none"> クロッキングをネットワークから得る場合には、必ず line キーワードを使用します。internal キーワードは、2 台のルータがバックツーバックで接続されているか、クロッキングが利用できないファイバで接続されている場合に使用します。 デフォルトは回線クロック (line) です。
<p>ステップ 20 グループ内の各チャネルに対してステップ 16 ~ 19 を繰り返します。</p>	-
<p>ステップ 21 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-sonet)# end または RP/0/RP0/CPU0:router(config-sonet)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
<p>ステップ 22 <code>exit</code></p>	SONET コントローラ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。
<p>ステップ 23 <code>exit</code></p>	グローバル コンフィギュレーション モードを終了し、EXEC モードを開始します。
<p>ステップ 24 <code>show aps</code></p> <p>例: RP/0/RP0/CPU0:router# show aps</p>	(任意) 設定済みのすべての SONET APS グループの動作ステータスを表示します。
<p>ステップ 25 <code>show aps group [number]</code></p> <p>例: RP/0/RP0/CPU0:router# show aps group 3</p>	<p>(任意) 設定済みの SONET APS グループの動作ステータスを表示します。</p> <p>(注) 複数のグループを定義する場合は、show aps group コマンドのほうが show aps コマンドよりも有用です。</p>

次に行う作業

- APS グループ内で POS インターフェイスを設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- CORE ネットワークの復元中に FRR がトリガーされないように hold-off タイマーを設定するには、このマニュアルで後述する「[Fast Reroute がトリガーされないように hold-off タイマーを設定する](#)」(P.437) モジュールを参照してください。

Fast Reroute がトリガーされないように hold-off タイマーを設定する

APS がルータ上で設定されている場合、トンネルに対する保護が提供されません。この制限のため、Fast Reroute (FRR) がいまだにマルチプロトコル ラベル スイッチング (MPLS) トラフィック エンジンアリングの保護メカニズムになっています。

APS が SONET コア ネットワーク中で設定されている場合、ルータのダウンストリームに向けてアラームが生成されることがあります。ルータのダウンストリームで FRR が設定されている場合は、SONET レベルで hold-off タイマーを設定し、CORE ネットワークの復元中に FRR がトリガーされないようにしたいことがあります。遅延を設定するにはこの作業を実施します。

前提条件

「[SONET APS の設定](#)」(P.432) に従って SONET APS を設定します。

手順の概要

1. **configure**
2. **controller sonet interface-path-id**
3. **line delay trigger value**
または
path delay trigger value
4. **end**
または
commit

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller sonet interface-path-id 例: RP/0/RP0/CPU0:router(config)# controller sonet 0/6/0/0	SONET コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
<p>ステップ 3 <code>line delay trigger value</code> または <code>path delay trigger value</code></p> <p>例: RP/0/RP0/CPU0:router(config-sonet)# line delay trigger 250 or RP/0/RP0/CPU0:router(config-sonet)# path delay trigger 300</p>	<p>SONET ポート遅延トリガー値をミリ秒単位で設定します。</p> <p>ヒント ステップ 2 とステップ 3 のコマンドは、1 つのコマンド文字列に結合して、グローバル コンフィギュレーション モードから controller sonet r/s/m/p line delay trigger または controller sonet r/s/m/p path delay trigger のように入力できます。</p>
<p>ステップ 4 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-sonet)# end または RP/0/RP0/CPU0:router(config-sonet)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

次に行う作業

- APS グループ内で POS インターフェイスを設定するには、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。

SONET コントローラの設定例

ここでは、次の例について説明します。

- 「[SONET コントローラの設定：例](#)」(P.439)
- 「[SONET APS グループの設定：例](#)」(P.439)

SONET コントローラの設定 : 例

次に、「[クリア チャネル SONET コントローラの設定](#)」(P.428) で概要を示した手順の後で SONET コントローラの設定を行った場合のコマンドと出力の例を示します。この例は、すべてのオプションコマンドの使用方法和、コマンド内でのオプションの一覧を示しています。実際の設定では、これらのコマンドの一部を使用しない場合があります。

```
configure
controller sonet 0/1/0/0
  ais-shut
  clock source internal
  framing sonet
  loopback internal
Loopback is a traffic-effecting operation
  overhead s1s0 1
  path ais-shut
  path delay trigger 0
  path overhead j1 line 11
  path report pais
  path scrambling disable
  path threshold b3-tca 6
  path uneq-shut
  report pais
  threshold b2-tca 4
commit
```

SONET APS グループの設定 : 例

次に、SONET ローカル (1 台のルータ) APS の設定例を示します。

```
aps group 1
  channel 0 local SONET 0/0/0/1
  channel 1 local SONET 0/0/0/2
  signalling sonet
commit
show aps
show aps group 3
```

次に、SONET リモート (2 台のルータ) APS の設定例を示します。

```
RP/0/0/CPU0:router(config)# aps group 1
  channel 0 local SONET 0/0/0/1
  channel 1 remote 172.18.69.123
  signalling sonet
commit
show aps
show aps group 3
```

関連情報

SONET コントローラを設定した後、そのコントローラに関連する POS インターフェイスまたは SRP インターフェイスを設定できます。

- POS インターフェイスを設定するには、このマニュアルの「[Cisco IOS XR ソフトウェアでの POS インターフェイスの設定](#)」モジュールを参照してください。
- SRP インターフェイスを設定するには、このマニュアルの「[Cisco IOS XR ソフトウェアでの SRP インターフェイスの設定](#)」モジュールを参照してください。

その他の参考資料

ここでは、SONET コントローラの設定に関する関連資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『 Cisco IOS XR Master Commands List 』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『 Cisco IOS XR Interface and Hardware Component Command Reference 』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『 Cisco IOS XR Getting Started Guide 』
ユーザ グループとタスク ID に関する情報	『 Cisco IOS XR System Security Configuration Guide 』の「 Configuring AAA Services on Cisco IOS XR Software 」モジュール
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、ルータのインターフェイスとその他のコンポーネントの設定に関する情報	『 Cisco Craft Works Interface Configuration Guide 』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータ上でのクリア チャネル T3/E3 コントローラおよびチャネライズド T3 コントローラの設定について説明します。関連付けられたシリアルインターフェイスを設定する前に、T3/E3 コントローラを設定する必要があります。

T3/E3 コントローラ インターフェイス設定の機能履歴

リリース	変更点
リリース 3.3.0	Cisco XR 12000 シリーズ ルータにこの機能が追加されました。 次の SIP について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。 <ul style="list-style-type: none">• Cisco XR 12000 SIP-401• Cisco XR 12000 SIP-501• Cisco XR 12000 SIP-601 次の SPA について、Cisco XR 12000 シリーズ ルータでのサポートが追加されました。 <ul style="list-style-type: none">• 2 ポートおよび 4 ポート チャネライズド T3 SPA• 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA
リリース 3.4.0	変更ありません。
リリース 3.4.1	4 ポート クリア チャネル T3/E3 SPA の Cisco CRS-1 ルータに、この機能が追加されました。
リリース 3.5.0	Cisco 1 ポート チャネライズド DS0/OC-12 SPA 対応の Cisco XR 12000 シリーズ ルータに、この機能が追加されました。
リリース 3.6.0	8 ポート チャネライズド T1/E1 SPA の Cisco XR 12000 シリーズ ルータに、この機能が追加されました。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「T3/E3 コントローラ設定の前提条件」(P.444)
- 「T3/E3 コントローラおよびシリアル インターフェイスに関する情報」(P.444)
- 「クリア チャネル T3/E3 コントローラおよびチャネル化された T1/E1 コントローラの設定方法」(P.447)
- 「クリア チャネル T3/E3 およびチャネル化した T3 コントローラの設定：例」(P.472)
- 「その他の参考資料」(P.475)

T3/E3 コントローラ設定の前提条件

T3/E3 コントローラを設定する前に、次の作業が終了し条件が満たされていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

- 使用しているハードウェアが T3/E3 コントローラまたはシリアル インターフェイスをサポートしている必要があります。T3/E3 コントローラおよびシリアル インターフェイスをサポートするハードウェアは次のとおりです。
 - 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA
 - 2 ポートおよび 4 ポート チャネライズド T3 SPA



(注) 2 ポートおよび 4 ポート チャネライズド T3 SPA は、クリア チャネル モードで実行できません。または、28 T1 コントローラか 21 E1 コントローラにチャネライズドできます。

- 8 ポート チャネライズド T1/E1 SPA は T1/E1 コントローラおよびシリアル インターフェイスをサポートします。

T3/E3 コントローラおよびシリアル インターフェイスに関する情報

2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA は、シリアル ライン上でのみ、クリア チャネル サービスをサポートします。2 ポートおよび 4 ポート チャネライズド T3 SPA は、クリア チャネル サービスおよびチャネライズド シリアル ラインをサポートします。

コントローラがチャネル化されない場合、このコントローラはクリア チャネル コントローラとなり、関連付けられたシリアル ラインの全帯域幅がシリアル サービスを伝送する単一のチャネル専用となります。



(注) このリリースでは、T3 の T1/E1 へのチャネル化だけがサポートされます。

T3 コントローラがチャネル化されると、より小さい帯域幅の T1 または E1 コントローラに論理的に分割されます。どちらのコントローラに分割されるかは、選択したチャネル化のモードによって決まります。T1 または E1 コントローラのシリアル インターフェイスの帯域幅の合計は、チャネル化された T1 または E1 コントローラを含む T3 コントローラの帯域幅を超過できません。

T3 コントローラをチャネル化すると、T1 または E1 の各コントローラは自動的にさらに DS0 タイムスロットにチャネル化されます。単一の T1 コントローラは 24 DS0 タイムスロットを伝送し、単一の E1 コントローラは 31 DS0 タイムスロットを伝送します。ユーザは、これらの DS0 タイムスロットを個々のチャネル グループに分割できます。各チャネル グループはそれぞれ、単一のシリアル インターフェイスをサポートします。

コントローラがチャネル化され、チャネル グループが作成されると、サービスは関連付けられたシリアル インターフェイスでプロビジョニングされます。

このリリースのチャネル化機能では、次のタイプのチャネルにチャネル化することができます。

- 単一の T3 コントローラを 28 T1 コントローラにチャネル化 (コントローラ サイズ合計は 44210 kbps)。
- 単一の T3 コントローラを 21 E1 コントローラにチャネル化 (コントローラ サイズ合計は 34010 kbps)。
- 単一の T1 コントローラは、最大 1.536 MB をサポートします。
- 単一の E1 コントローラは、最大 2.048 MB をサポートします。



(注) 単一の共有ポート アダプタ (SPA) は、最大 448 チャネル グループをサポートできます。

チャネル化された T3 コントローラおよびその関連付けられたシリアル インターフェイスと設定は、4 段階の手順で行います。

- ステップ 1** T3 コントローラを設定し、コントローラのモードを T1 または E1 に設定します。
- ステップ 2** T1 または E1 コントローラを設定します。
- ステップ 3** チャネル グループを作成し、目的に合わせて DS0 タイムスロットをこれらのチャネル グループに割り当てます。
- ステップ 4** このマニュアルで後述する「Cisco IOS XR ソフトウェアでのシリアル インターフェイスの設定」モジュールの説明に従って、各チャネル グループに関連付けられたシリアル インターフェイスを設定します。

T3 および E3 コントローラのデフォルト設定値

表 19 に、T3 および E3 コントローラのデフォルト設定パラメータを示します。

表 19 T3 および E3 コントローラのデフォルト設定値

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
データ ラインのフレーム タイプ	T3 の場合 : C ビット フ レーム構成 E3 の場合 : G.751	framing { auto-detect c-bit m23 }
各 T3/E3 リンクのクロッキング	internal	clock source { internal line }
ケーブル長	224 フィート	cablelength <i>feet</i>
Maintenance Data Link (MDL; メンテナ ンス データ リンク) メッセージ (T3 のみ)	disable	mdl transmit { idle-signal path test-signal } { disable enable }
E3 ポートの各国用予約ビット (E3 のみ)	enable 、ビット パターン 値は 1	national bits { disable enable }



(注)

シリアル リンクでクロッキングを設定する場合、一方のエンドを **internal** にし、もう一方を **line** にする必要がありま。接続の両エンドに **internal** クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに **line** クロッキングを設定すると、ラインはアップ状態になりません。

T1 および E1 コントローラのデフォルト設定値

表 20 に、T1 および E1 コントローラのデフォルト設定パラメータを示します。

表 20 T1 および E1 コントローラのデフォルト設定値

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
データ ラインのフレーム タイプ	T1 の場合 : 拡張スーパー フレーム (esf) E1 の場合 : CRC-4 エ ラー監視機能 (crc4) 付 きのフレーム構成	T1 の場合 : framing { sf esf } E1 の場合 : framing { crc4 no-crc4 unframed }
検出および T1 イエロー アラームの生成 (T1 のみ)	T1 チャネルでイエロー アラームが検出され、生 成されます。	yellow { detection generation } { disable enable }
各 T1 および E1 リンクのクロッキング	internal	clock source { internal line }

表 20 T1 および E1 コントローラのデフォルト設定値 (続き)

パラメータ	デフォルト値	コンフィギュレーション ファイルのエントリ
ケーブル長 (T1 のみ)	cablelength long コマンドの場合 : <i>db-gain-value: gain26; db-loss-value: 0db</i> cablelength short コマンドの場合 : 533 feet	ケーブル長を 655 フィートよりも長く設定する場合 : cablelength long <i>db-gain-value db-loss-value</i> ケーブル長を 655 フィート以下に設定する場合 : cablelength short <i>length</i>
ANSI T1.403 または AT&T TR54016 についての秒単位のパフォーマンス レポートの T1 チャネルの Facility Data Link (FDL; ファシリティ データ リンク) を通じた伝送 (T1 のみ)	disable	fdl {ansi att} {enable disable}
E1 ポートの各国用予約ビット (E1 のみ)	0 (16 進表記の <i>0x1f</i> に一致します)	national bits <i>bits</i>



(注)

シリアルリンクでクロッキングを設定する場合、一方のエンドを **internal** にし、もう一方を **line** にする必要があります。接続の両エンドに **internal** クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに **line** クロッキングを設定すると、ラインはアップ状態になりません。

クリア チャネル T3/E3 コントローラおよびチャネル化された T1/E1 コントローラの設定方法

T3/E3 コントローラは、Cisco IOS XR ソフトウェア のコンフィギュレーション スペースの物理レイヤのコントロール要素で設定します。このコンフィギュレーションについては、次のタスクで説明します。

- 「カードタイプの設定」 (P.448)
- 「クリア チャネル E3 コントローラの設定」 (P.450)
- 「デフォルトの E3 コントローラ設定の変更」 (P.451)
- 「クリア チャネル T3 コントローラの設定」 (P.454)
- 「チャネル化された T3 コントローラの設定」 (P.455)
- 「デフォルトの T3 コントローラ設定の変更」 (P.457)
- 「T1 コントローラの設定」 (P.460)
- 「E1 コントローラの設定」 (P.463)
- 「BERT の設定」 (P.467)

カードタイプの設定

デフォルトでは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA は T3 モードで起動し、8 ポート チャネライズド T1/E1 SPA は T1 モードで起動します。2 ポートまたは 4 ポートのクリア チャネル T3/E3 SPA を E3 モードで使用するか、または 8 ポート チャネライズド T1/E1 SPA を E1 モードで使用するには、ここに記載されているように **hw-module subslot card type** コマンドのデフォルト設定を変更する必要があります。



(注) **hw-module subslot card type** コマンドを使用すると、SPA 上のすべてのポートが同じタイプに設定されます。



注意

hw-module subslot card type コマンドがコミットされると、SPA は自動的にリセットされます。



(注)

hw-module subslot card type コマンドは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA および 8 ポート チャネライズド T1/E1 SPA のみに適用されます。2 ポートおよび 4 ポート チャネライズド T3 SPA は、T3 モードでのみ実行されます。

前提条件

2 ポートまたは 4 ポート クリア チャネル T3/E3 SPA または 8 ポート チャネライズド T1/E1 SPA 上でインターフェイスを以前に設定したことがある場合、そのカードタイプを変更するには、以前に定義した T3/E3 または T1/E1 コントローラおよびシリアル インターフェイス コンフィギュレーションを削除する必要があります。**no controller [e1 | e3 | t1 | t3]** コマンドと **no interface serial** コマンドを使用して、コントローラおよびインターフェイスのコンフィギュレーションをデフォルトに戻します。

制約事項

このタスクは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA および 8 ポート チャネライズド T1/E1 SPA のみに適用されます。

手順の概要

1. **configure**
2. **hw-module subslot *subslot-id* cardtype {e1 | e3 | t1 | t3}**
3. **end**
または
commit

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/0/CPU0:router# <code>configure</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>hw-module subslot subslot-id cardtype {e1 e3 t1 t3}</code> 例: RP/0/0/CPU0:router(config)# <code>hw-module subslot 0/1/0 cardtype e3</code> または RP/0/0/CPU0:router(config)# <code>hw-module subslot 0/2/0 cardtype e1</code>	SPA のシリアル モードを設定します。 <ul style="list-style-type: none"> • t3 : B3ZS コーディングを使用するネットワークでの 44,210 Kbps の T3 接続を指定します。これがデフォルトの設定です。 • e3 : 主に欧州で使用されているデータ転送レート 34,010 Kbps の広域デジタル転送方式を指定します。 • t1 : 最大 1.536 MB をサポートする 24 DS0 タイムスロットを指定します。 • e1 : 最大 2.048 MB をサポートする 31 DS0 タイムスロットを指定します。
ステップ 3 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config)# <code>end</code> または RP/0/0/CPU0:router(config)# <code>commit</code>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

クリア チャネル E3 コントローラの設定

クリア チャネル モードにある E3 コントローラは、単一シリアル インターフェイスを伝送します。E3 コントローラを設定するには、E3 コンフィギュレーション モードを使用します。

前提条件

E3 をサポートするカードを設定するには、最初に **hw-module subslot cardtype** コマンドを使用する必要があります。

制約事項

- コントローラ タイプに有効でないオプションを設定すると、設定をコミットするときにエラーが表示されます。
- 単一の SPA では、T3 インターフェイスと E3 インターフェイスの併用はサポートされません。
- このタスクは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA のみに適用されます。

手順の概要

1. **configure**
2. **controller e3 interface-path-id**
3. **mode serial**
4. **no shutdown**
5. **end**
または
commit
6. **show controllers e3 interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller e3 interface-path-id 例： RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<i>rack/slot/module/port</i> 表記で E3 コントローラ名を指定し、E3 コンフィギュレーション モードを開始します。
ステップ 3	mode serial 例： RP/0/0/CPU0:router(config-e3)# mode serial	ポートのモードをクリア チャネル シリアルに設定します。 (注) このステップは、2 ポートおよび 4 ポート チャネライズド T3 SPA にのみ必要です。2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA は、デフォルトでシリアル モードで実行されます。

コマンドまたはアクション	目的
ステップ 4 <code>no shutdown</code> 例: RP/0/0/CPU0:router(config-e3)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> • <code>shutdown</code> 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 5 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-e3)# end または RP/0/0/CPU0:router(config-e3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 6 <code>show controllers e3 interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers e3 0/1/0/0	(任意) E3 コントローラに関する情報を表示します。

次に行う作業

- 設定した E3 コントローラ上で実行されているデフォルト設定を、このモジュールで後述する「[デフォルトの E3 コントローラ設定の変更](#)」の説明に従って変更します。
- このモジュールで後述する「[BERT の設定](#)」の説明に従って、その完全性をテストするため、コントローラの Bit Error Rate Test (BERT; ビット誤り率テスト) を設定します。
- このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定](#)」モジュールの説明に従って、関連付けられたシリアルインターフェイスを設定します。

デフォルトの E3 コントローラ設定の変更

ここでは、このモジュールで前述した「[T3 および E3 コントローラのデフォルト設定値](#)」で説明したデフォルトの E3 コントローラ設定を変更する手順について説明します。

前提条件

このモジュールで前述した「クリア チャネル E3 コントローラの設定」の説明に従って、クリア チャネル E3 コントローラを設定する必要があります。

手順の概要

1. **configure**
2. **controller e3 interface-path-id**
3. **clock source {internal | line}**
4. **cablelength feet**
5. **framing {g751 | g832}**
6. **national bits {disable | enable}**
7. **no shutdown**
8. **end**
または
commit
9. **show controllers e3 interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller e3 interface-path-id 例： RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<i>rack/slot/module/port</i> 表記で E3 コントローラ名を指定し、E3 コンフィギュレーション モードを開始します。
ステップ 3	clock source {internal line} 例： RP/0/0/CPU0:router(config-e3)# clock source internal	(任意) 個々の E3 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアル リンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 4	cablelength feet 例： RP/0/0/CPU0:router(config-e3)# cablelength 250	(任意) ルータからネットワーク装置までのケーブルの長さを指定します。 (注) デフォルトのケーブル長は 224 フィートです。

コマンドまたはアクション	目的
ステップ 5 <code>framing {g751 g832}</code> 例: RP/0/0/CPU0:router(config-e3)# framing g832	(任意) E3 ポートのフレーム タイプを指定します。設定可能な E3 フレーム タイプは、G.751 および G.832 です。 (注) E3 のデフォルトのフレーム構成は G.751 です。
ステップ 6 <code>national bits {disable enable}</code> 例: RP/0/0/CPU0:router(config-e3)# national bits enable	(任意) E3 ポートの 0x1F 各国用予約ビットパターンをイネーブルまたはディセーブルにします。 (注) E3 各国用ビットはデフォルトでイネーブルに設定され、ビットパターン値は 1 です。
ステップ 7 <code>no shutdown</code> 例: RP/0/0/CPU0:router(config-e3)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 8 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-e3)# end または RP/0/0/CPU0:router(config-e3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 9 <code>show controllers e3 interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers e3 0/1/0/0	(任意) E3 コントローラに関する情報を表示します。

次に行う作業

- 設定した T3 コントローラ上で実行されているデフォルト設定を、このモジュールで後述する「[デフォルトの T3 コントローラ設定の変更](#)」の説明に従って変更します。
- このモジュールで後述する「[BERT の設定](#)」の説明に従って、その完全性をテストするため、コントローラのビット誤り率テスト (BERT) を設定します。
- このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定](#)」モジュールの説明に従って、関連付けられたシリアルインターフェイスを設定します。

クリア チャネル T3 コントローラの設定

クリア チャネル モードにある T3 コントローラは、単一シリアル インターフェイスを伝送します。T3 コントローラを設定するには、T3 コンフィギュレーション モードを使用します。

前提条件

このモジュールで前述した「[カード タイプの設定](#)」の説明に従って **hw-module subslot cardtype** コマンドを使用し、T3 をサポートするようにカードを設定する必要があります。

制約事項

- このタスクは、2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA のみに適用されます。
- コントローラ タイプに有効でないオプションを設定すると、設定をコミットするときにエラーが表示されます。
- 単一の SPA では、T3 インターフェイスと E3 インターフェイスの併用はサポートされません。

手順の概要

- configure**
- controller t3 interface-path-id**
- mode serial**
- no shutdown**
- end**
または
commit
- show controllers t3 interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller t3 interface-path-id 例： RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	rack/slot/module/port 表記で T3 コントローラ名を指定し、T3 コンフィギュレーション モードを開始します。
ステップ 3	mode serial 例： RP/0/0/CPU0:router(config-t3)# mode serial	ポートのモードをクリア チャネル シリアルに設定します。 (注) このステップは、2 ポートおよび 4 ポート チャネライズド T3 SPA にのみ必要です。2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA は、デフォルトでシリアル モードで実行されます。

コマンドまたはアクション	目的
ステップ 4 <code>no shutdown</code> 例: RP/0/0/CPU0:router(config-t3)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> • <code>shutdown</code> 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 5 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 6 <code>show controllers t3 interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers t3 0/1/0/0	(任意) T3 コントローラに関する情報を表示します。

次に行う作業

- 設定した T3 コントローラ上で実行されているデフォルト設定を、このモジュールで後述する「[デフォルトの T3 コントローラ設定の変更](#)」の説明に従って変更します。
- このモジュールで後述する「[BERT の設定](#)」の説明に従って、その完全性をテストするため、コントローラの Bit Error Rate Test (BERT; ビット誤り率テスト) を設定します。
- このドキュメントで後述する「[Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定](#)」モジュールの説明に従って、関連付けられたシリアルインターフェイスを設定します。

チャネル化された T3 コントローラの設定

2 ポートおよび 4 ポート チャネライズド T3 SPA は、T1、E1、および DS0 へのチャネル化をサポートします。ここでは、単一の T3 コントローラを 28 T1 コントローラまたは 21 E1 コントローラにチャネル化する手順について説明します。T1 または E1 コントローラを作成すると、次の説明に従って、それらのコントローラを DS0 タイムスロットにチャネル化することができます。

- [T1 コントローラの設定](#)
- [E1 コントローラの設定](#)

個々の T1 コントローラは、24 DS0 タイムスロットの合計をサポートします。また、個々の E1 コントローラは、31 DS0 タイムスロットの合計をサポートします。



(注)

コントローラ タイプに有効でないオプションを設定すると、設定をコミットするときにエラーが表示されます。

手順の概要

1. **configure**
2. **controller t3 interface-path-id**
3. **mode [t1 | e1]**
4. **no shutdown**
5. **end**
または
commit
6. **show controllers t3 interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller T3 interface-path-id 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<i>rack/slot/module/port</i> 表記で T3 コントローラ名を指定し、T3 コンフィギュレーション モードを開始します。
ステップ 3	mode t1 例: RP/0/0/CPU0:router(config-t3)# mode t1	チャネライズされたコントローラのモードを T1 に設定し、28 T1 コントローラを作成します。
ステップ 4	no shutdown 例: RP/0/0/CPU0:router(config-t3)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> • shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。

コマンドまたはアクション	目的
<p>ステップ 5 <code>end</code> または <code>commit</code></p> <p>例: RP/0/0/CPU0:router(config-t3)# <code>end</code> または RP/0/0/CPU0:router(config-t3)# <code>commit</code></p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6 <code>show controllers t3 interface-path-id</code></p> <p>例: RP/0/0/CPU0:router# <code>show controllers t3 0/1/0/0</code></p>	<p>(任意) T3 コントローラに関する情報を表示します。</p>

次に行う作業

- 設定した T3 コントローラ上で実行されているデフォルト設定を、このモジュールで後述する「[デフォルトの T3 コントローラ設定の変更](#)」(P.457) の説明に従って変更します。
- T3 コントローラを 28 T1 コントローラにチャネル化したら、このマニュアルで後述する「[T1 コントローラの設定](#)」モジュールの説明に従って T1 コントローラを設定し、それらに DS0 タイムスロットを割り当てます。
- T3 コントローラを 21 E1 コントローラにチャネル化したら、このマニュアルで後述する「[E1 コントローラの設定](#)」モジュールの説明に従って E1 コントローラを設定し、それらに DS0 タイムスロットを割り当てます。

デフォルトの T3 コントローラ設定の変更

ここでは、このモジュールで前述した「[T3 および E3 コントローラのデフォルト設定値](#)」で説明したデフォルトの T3 コントローラ設定を変更する手順について説明します。

前提条件

次の説明に従って、クリア チャネルまたはチャネル化した T3 コントローラを設定する必要があります。

- [クリア チャネル T3 コントローラの設定](#)
- [チャネル化された T3 コントローラの設定](#)

手順の概要

1. `configure`
2. `controller t3 interface-path-id`
3. `clock source {internal | line}`
4. `cablelength feet`
5. `framing {auto-detect | c-bit | m23}`
6. `mdl transmit {idle-signal | path | test-signal} {disable | enable}`
7. `mdl string {eic | fi | fic | gen-number | lic | port-number | unit} string`
8. `no shutdown`
9. `end`
または
`commit`
10. `show controllers t3 interface-path-id`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller T3 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	<code>rack/slot/module/port</code> 表記で T3 コントローラ名を指定し、T3 コンフィギュレーション モードを開始します。
ステップ 3	<code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-t3)# clock source internal	(任意) T3 ポートのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアル リンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 4	<code>cablelength feet</code> 例: RP/0/0/CPU0:router(config-t3)# cablelength 250	(任意) ルータからネットワーク装置までのケーブルの長さを指定します。 (注) デフォルトのケーブル長は 224 フィートです。

コマンドまたはアクション	目的
ステップ 5 framing { <i>auto-detect</i> <i>c-bit</i> <i>m23</i> } 例: RP/0/0/CPU0:router(config-t3)# framing c-bit	(任意) T3 ポートのフレーム タイプを指定します。 (注) T3 のデフォルトのフレーム タイプは C-bit です。
ステップ 6 mdl transmit { <i>idle-signal</i> <i>path</i> <i>test-signal</i> } { <i>disable</i> <i>enable</i> } 例: RP/0/0/CPU0:router(config-t3)# mdl transmit path enable	(任意) T3 ポートのメンテナンス データ リンク (MDL) メッセージをイネーブルにします。 (注) MDL メッセージは、T3 フレーム構成が C-bit パリティである場合にのみサポートされます。 (注) MDL メッセージはデフォルトで表示されます。
ステップ 7 mdl string { <i>eic</i> <i>fi</i> <i>fic</i> <i>gen-number</i> <i>lic</i> <i>port-number</i> <i>unit</i> } <i>string</i> 例: RP/0/0/CPU0:router(config-t3)# mdl fi facility identification code	(任意) MDL メッセージで送信される文字列の値を指定します。
ステップ 8 no shutdown 例: RP/0/0/CPU0:router(config-t3)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 9 end または commit 例: RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットをを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 10 show controllers t3 <i>interface-path-id</i> 例: RP/0/0/CPU0:router# show controllers t3 0/1/0/0	(任意) T3 コントローラに関する情報を表示します。

次に行う作業

- クリア チャネル T3 コントローラを設定したら、次の作業を行います。
 - このモジュールで後述する「BERT の設定」(P.467) の説明に従って、その完全性をテストするため、コントローラのビット誤り率テスト (BERT) を設定します。
 - このマニュアルで後述する「Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定」モジュールの説明に従って、関連付けられたシリアル インターフェイスを設定します。
- T3 コントローラを 28 T1 コントローラにチャネル化したら、このマニュアルで後述する「T1 コントローラの設定」モジュールの説明に従って T1 コントローラを設定し、それらに DS0 タイムスロットを割り当てます。
- T3 コントローラを 21 E1 コントローラにチャネル化したら、このマニュアルで後述する「E1 コントローラの設定」モジュールの説明に従って E1 コントローラを設定し、それらに DS0 タイムスロットを割り当てます。

T1 コントローラの設定

ここでは、個々の T1 コントローラを設定し、それを 24 の個別の DS0 タイムスロットにチャネル化する手順について説明します。

前提条件

- 2 ポートまたは 4 ポートのチャネル化した T3 SPA が必要です。またはルータに 8 ポート チャネライズド T1/E1 SPA がインストールされている必要があります。
- 2 ポートまたは 4 ポートのチャネル化した T3 SPA がある場合、このモジュールの「チャネル化された T3 コントローラの設定」(P.455) の説明に従って、チャネル化した T3 コントローラを T1 モードで実行するように設定する必要があります。

制約事項

コントローラ タイプに有効でないオプションを設定すると、設定をコミットするときにエラーが表示されます。

手順の概要

1. `show controllers t1 interface-path-id`
2. `configure`
3. `controller t1 interface-path-id`
4. `framing {sf | esf}`
5. `yellow {detection | generation} {disable | enable}`
6. `clock source {internal | line}`
7. `fdl {ansi | att} {enable | disable}`
8. `no shutdown`
9. `channel-group channel-group-number`
10. `timeslots range`

11. `speed kbps`

12. `exit`

13. ステップ 9 ~ 12 を繰り返し、タイムスロットをチャネル グループに割り当てます。各コントローラには、最大 24 のタイムスロットを設定できます。

14. `exit`

15. ステップ 2 ~ 14 を繰り返し、さらなるチャネル グループをコントローラに割り当てます。

16. `end`

または
`commit`

詳細手順

ステップ 1	<code>show controllers t1 interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers t3 0/1/0/0	(任意) ステップ 3 で作成した T1 コントローラに関する情報を表示します。
ステップ 2	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>controller t1 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t1 0/3/0/0/0	T1 コンフィギュレーション モードを開始します。
ステップ 4	<code>framing {sf esf}</code> 例: RP/0/0/CPU0:router(config-t1)# framing esf	(任意) T1 データ ラインのフレーム タイプを指定します。 <ul style="list-style-type: none"> <code>sf</code>: スーパーフレーム <code>esf</code>: 拡張スーパーフレーム (注) T1 のデフォルトのフレーム タイプは拡張スーパーフレーム (<code>esf</code>) です。
ステップ 5	<code>yellow {detection generation} {disable enable}</code> 例: RP/0/0/CPU0:router(config-t1e1)# yellow detection enable	(任意) T1 でのイエロー アラームの検出と生成をイネーブ ルまたはディセーブルにします。 (注) デフォルトでは、T1 チャネルでイエロー アラーム が検出され、生成されます。
ステップ 6	<code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-t1e1)# clock source internal	(任意) 個々の T1 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは <code>internal</code> です。 (注) シリアルリンクでクロッキングを設定する場合、 一方のエンドを <code>internal</code> にし、もう一方を <code>line</code> に する必要があります。接続の両エンドに <code>internal</code> クロッキングを設定すると、フレーム同期のずれ が生じます。接続の両エンドに <code>line</code> クロッキング を設定すると、ラインはアップ状態になりません。

■ クリア チャネル T3/E3 コントローラおよびチャネル化された T1/E1 コントローラの設定方法

ステップ 7	<pre>fdl {ansi att} {enable disable}</pre> <p>例 : RP/0/0/CPU0:router(config-t1e1)# fdl ansi enable</p>	<p>Facility Data Link (FDL; ファシリティ データ リンク) を介した ANSI T1.403 または AT&T TR54016 についての秒単位のパフォーマンス レポートの伝送をイネーブルにします。</p> <p>(注) FDL ansi および att はデフォルトでディセーブルに設定されています。</p>
ステップ 8	<pre>no shutdown</pre> <p>例 : RP/0/0/CPU0:router(config-t1e1)# no shutdown</p>	<p>shutdown 設定を削除します。</p> <ul style="list-style-type: none"> shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 9	<pre>channel-group channel-group-number</pre> <p>例 : RP/0/0/CPU0:router(config-t1) # channel-group 0</p>	<p>T1 チャネル グループを作成し、そのチャネル グループのチャネル グループ コンフィギュレーション モードを開始します。</p>
ステップ 10	<pre>timeslots range</pre> <p>例 : RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 7-12</p>	<p>1 つまたは複数の DS0 タイムスロットをチャネル グループに関連付け、関連付けたシリアル サブインターフェイスをそのチャネル グループに作成します。</p> <ul style="list-style-type: none"> 範囲は 1 ~ 24 タイムスロットです。 24 タイムスロットすべてを単一のチャネル グループに割り当てることも、タイムスロットを複数のチャネル グループに分割することもできます。 <p>(注) 個々の T1 コントローラは、24 DS0 タイムスロットの合計をサポートします。</p>
ステップ 11	<pre>speed kbps</pre> <p>例 : RP/0/0/CPU0:router(config-t1e1-channel_group)# speed 64</p>	<p>(任意) DS0 の速度を Kbps 単位で指定します。有効値は 56 と 64 です。</p> <p>(注) デフォルトの速度は 64 kbps です。</p>
ステップ 12	<pre>exit</pre> <p>例 : RP/0/0/CPU0:router(config-t1-channel_group)# exit</p>	<p>チャネル グループ コンフィギュレーション モードを終了します。</p>
ステップ 13	<p>ステップ 9 ~ 12 を繰り返し、タイムスロットをチャネル グループに割り当てます。各コントローラには、最大 24 のタイムスロットを設定できます。</p>	<p>-</p>
ステップ 14	<pre>exit</pre> <p>例 : RP/0/0/CPU0:router(config-t1)# exit</p>	<p>T1 コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。</p>

ステップ 15	Repeat Step 2 through Step 14 to assign more channel groups to a controller as desired.	-
ステップ 16	<pre>end または commit 例： RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

次に行う作業

- このモジュールの「BERT の設定」(P.467) の説明に従って、その完全性をテストするため、コントローラの Bit Error Rate Test (BERT; ビット誤り率テスト) を設定します。
- このマニュアルで後述する「Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定」モジュールの説明に従って、関連付けられたシリアルインターフェイスを設定します。

E1 コントローラの設定

ここでは、個々の E1 コントローラを設定し、それを 31 の個別の DS0 タイムスロットにチャネル化する手順について説明します。

前提条件

- 2 ポートまたは 4 ポートのチャネル化した T3 SPA が必要です。またはルータに 8 ポート チャネライズド T1/E1 SPA がインストールされている必要があります。
- 2 ポートまたは 4 ポートのチャネル化した T3 SPA がある場合、このモジュールの「チャネル化された T3 コントローラの設定」の説明に従って、チャネル化した T3 コントローラを E1 モードで実行するように設定する必要があります。

制約事項

コントローラ タイプに有効でないオプションを設定すると、設定をコミットするときにエラーが表示されます。

手順の概要

1. `show controllers e1 interface-path-id`
2. `configure`
3. `controller e1 interface-path-id`
4. `clock source {internal | line}`
5. `framing {crc4 | no-crc4 | unframed}`
6. `national bits bits`
7. `no shutdown`
8. `channel-group channel-group-number`
9. `timeslots range`
10. `speed kbps`
11. `exit`
12. ステップ 8 ~ 11 を繰り返し、タイムスロットをチャネル グループに割り当てます。各コントローラには、最大 24 のタイムスロットを設定できます。
13. `exit`
14. ステップ 2 ~ 13 を繰り返し、目的に合わせて、さらなるチャネル グループをコントローラに割り当てます。
15. `end`
または
`commit`

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>show controllers e1 interface-path-id</code> 例: RP/0/0/CPU0:router# show controllers e1 0/1/0/0	(任意) E1 コントローラに関する情報を表示します。
ステップ 2	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>controller e1 interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller e1 0/3/0/0/0	E1 コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 4 <code>clock source {internal line}</code> 例: RP/0/0/CPU0:router(config-e1)# clock source internal	(任意) 個々の E1 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアルリンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 5 <code>framing {crc4 no-crc4 unframed}</code> 例: RP/0/0/CPU0:router(config-e1)# framing unframed	(任意) E1 データ ラインのフレーム タイプを指定します。E1 に有効なフレーム タイプは次のとおりです。 <ul style="list-style-type: none"> • crc4 : CRC-4 エラー監視機能付きのフレーム構成 • no-crc4 : CRC-4 エラー監視機能なしのフレーム構成 • unframed : フレーム化されていない E1 (注) E1 のデフォルトのフレーム タイプは crc4 です。
ステップ 6 <code>national bits bits</code> 例: RP/0/0/CPU0:router(config-e1)# national bits 10	(任意) E1 ポートの各国用予約ビットを指定します。範囲は 0 ~ 31 です。 (注) デフォルトのビット パターンは 0 です。これは 16 進表記の <i>0x1f</i> に一致します。
ステップ 7 <code>no shutdown</code> 例: RP/0/0/CPU0:router(config-e1)# no shutdown	shutdown 設定を削除します。 <ul style="list-style-type: none"> • shutdown 設定を削除すると、コントロールに強制された管理上のダウンが解除され、コントローラをアップ状態またはダウン状態に移行できるようになります。
ステップ 8 <code>channel-group channel-group-number</code> 例: RP/0/0/CPU0:router(config-e1)# channel-group 0	E1 チャネル グループを作成し、そのチャネル グループのチャネル グループ コンフィギュレーション モードを開始します。
ステップ 9 <code>timeslots range</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# timeslots 1-16	1 つまたは複数のタイムスロットをチャネル グループに関連付け、関連付けたシリアル サブインターフェイスをそのチャネル グループに作成します。 <ul style="list-style-type: none"> • 範囲は 1 ~ 31 タイムスロットです。 • 31 タイムスロットすべてを単一のチャネル グループに割り当てることも、タイムスロットを複数のチャネル グループに分割することもできます。 (注) 各 E1 コントローラは、31 DS0 タイムスロットの合計をサポートします。
ステップ 10 <code>speed kbps</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# speed 100	(任意) DS0 の速度を Kbps 単位で指定します。有効値は 56 と 64 です。 (注) デフォルトの速度は 64 kbps です。

コマンドまたはアクション	目的
ステップ 11 <code>exit</code> 例: RP/0/0/CPU0:router(config-e1-channel_group)# exit	チャンネル グループ コンフィギュレーション モードを終了 します。
ステップ 12 ステップ 8 ~ 11 を繰り返し、タイムスロットをチャ ネル グループに割り当てます。	-
ステップ 13 <code>exit</code> 例: RP/0/0/CPU0:router(config-e1)# exit	E1 コンフィギュレーション モードを終了します。
ステップ 14 ステップ 2 ~ 13 を繰り返し、目的に合わせて、さら なるチャンネル グループをコントローラに割り当てま す。	-
ステップ 15 <code>end</code> または <code>commit</code> 例: RP/0/0/CPU0:router(config-e3)# end または RP/0/0/CPU0:router(config-e3)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求める プロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュ レーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、設定変更をコミットせずにコン フィギュレーションセッションが終了し、ルータ が EXEC モードに戻ります。 - cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われ ず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保 存し、コンフィギュレーションセッションを継続する には、commit コマンドを使用します。

次に行う作業

- このモジュールの「BERT の設定」(P.467) の説明に従って、その完全性をテストするため、コン
トローラの Bit Error Rate Test (BERT; ビット誤り率テスト) を設定します。
- このマニュアルで後述する「Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定」
モジュールの説明に従って、関連付けられたシリアルインターフェイスを設定します。

BERT の設定

ビット誤り率テスト (BERT) は、各 T3/E3 または T1/E1 コントローラ、および DS0 チャネル グループでサポートされています。これは、フレーム化されていない T3/E3 または T1/E1 信号でのみ行われ、一度に 1 つのポート上でのみ実行されます。個々のチャネル グループでもサポートされます。

BERT の結果を参照するには、EXEC モードで **show controllers t1** コマンドまたは **show controllers t3** コマンドを使用します。BERT の結果には、次の情報が含まれます。

- 選択したテスト パターンのタイプ
- テストのステータス
- 選択したインターバル
- BER テストの残り時間
- ビット エラーの合計
- 受信したビット数の合計

BERT はデータ挿入型です。テストの実行中、正規のデータはラインにフローされません。BERT の進行中、ラインはアラーム状態に置かれ、BERT が完了すると正常状態に復元されます。

T3/E3 および T1/E1 コントローラでの BERT の設定

ここでは、T3/E3 ライン、T1/E1 ライン、または個々のチャネル グループでビット誤り率テスト (BERT) のパターンをイネーブルにする手順について説明します。

前提条件

クリア チャネル T3/E3 コントローラを設定するか、T3 の T1/E1 コントローラへのチャネル化を行う必要があります。

手順の概要

1. **configure**
2. **controller [t3 | e3 | t1 | e1] interface-path-id**
3. **bert pattern pattern**
4. **bert interval time**
5. **bert error [number]**
6. **end**
または
commit
7. **exit**
8. **exit**
9. **bert [t3 | e3 | t1 | e1] interface-path-id [channel-group channel-group-number] [error] start**
10. **bert [t3 | e3 | t1 | e1] interface-path-id [channel-group channel-group-number] stop**
11. **show controllers [t3 | e3 | t1 | e1] interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller [t3 e3 t1 e1]</code> <code>interface-path-id</code> 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	コントローラ名とインスタンス ID を <code>rack/slot/module/port</code> 表記で指定し、T3、E3、T1、または E1 コントローラ コンフィギュレーション モードを開始します。
ステップ 3	<code>bert pattern pattern</code> 例: RP/0/0/CPU0:router(config-t3)# bert pattern 2^15	コントローラで特定のビット誤り率テスト (BERT) のパターンをイネーブルにします。すべてのコントローラおよびチャネル グループに有効なパターンには、 0s 、 1s 、 2^15 、 2^20 、 2^20-QRSS 、 2^23 、 alt-0-1 があります。T1 および E1 コントローラに有効なパターンには 1in8 、 3in24 、 55Daly 、 55Octet があります。チャネル グループに有効なパターンには 2^11 、 2^9 、 ds0-1 、 ds0-2 、 ds0-3 、 ds0-4 があります。 (注) BER テストを開始するには、EXEC モードで <code>bert</code> コマンドを使用する必要があります。
ステップ 4	<code>bert interval time</code> 例: RP/0/0/CPU0:router(config-t3)# bert pattern 2^15	(任意) T3/E3 または T1/E1 ラインでのビット誤り率テスト (BERT) のパターンを指定します。インターバルの値は 1 ~ 14400 の範囲で指定できます。
ステップ 5	<code>bert error [number]</code> 例: RP/0/0/CPU0:router(config-t3)# bert error 10	ビット ストリームに追加する BERT エラーの数を指定します。範囲は 1 ~ 255 です。

コマンドまたはアクション	目的
<p>ステップ 6</p> <pre>end または commit</pre> <p>例: RP/0/0/CPU0:router(config-t3)# end または RP/0/0/CPU0:router(config-t3)# commit </p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 7</p> <pre>exit</pre> <p>例: RP/0/0/CPU0:router(config-t3)# exit </p>	<p>T3/E3 または T1/E1 コントローラ コンフィギュレーションモードを終了します。</p>
<p>ステップ 8</p> <pre>exit</pre> <p>例: RP/0/0/CPU0:router(config)# exit </p>	<p>グローバル コンフィギュレーション モードを終了します。</p>
<p>ステップ 9</p> <pre>bert [t3 e3 t1 e1] interface-path-id [channel-group channel-group-number] [error] start</pre> <p>例: RP/0/0/CPU0:router# bert t3 0/3/0/0 start RP/0/0/CPU0:router# bert t3 0/3/0/0 error </p>	<p>指定した T3/E3 または T1/E1 コントローラで、設定した BERT テストを開始します。</p> <p>(注) オプションの error キーワードを指定して、実行中の BERT ストリームにエラーを挿入することもできます。</p>
<p>ステップ 10</p> <pre>bert [t3 e3 t1 e1] interface-path-id [channel-group channel-group-number] stop</pre> <p>例: RP/0/0/CPU0:router# bert t3 0/3/0/0 stop </p>	<p>指定した T3/E3 または T1/E1 コントローラで、設定した BERT テストを停止します。</p>
<p>ステップ 11</p> <pre>show controllers [t3 e3 t1 e1] interface-path-id</pre> <p>例: RP/0/0/CPU0:router# show controllers t3 0/3/0/0 </p>	<p>設定した BERT の結果を表示します。</p>

次に行う作業

このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定](#)」モジュールの説明に従って、テストしたコントローラに関連付けられたシリアルインターフェイスを設定します。

DS0 チャネル グループでの BERT の設定

ここでは、個々の DS0 チャネル グループでビット誤り率テスト (BERT) のパターンをイネーブルにする手順について説明します。

前提条件

クリア チャネル T3/E3 コントローラを設定するか、T3 の T1/E1 コントローラへのチャネライズを行う必要があります。

手順の概要

1. **configure**
2. **controller [t1| e1] interface-path-id**
3. **channel-group channel-group-number**
4. **bert pattern pattern**
5. **bert interval time**
6. **end**
または
commit
7. **exit**
8. **exit**
9. **exit**
10. **bert [t1| e1] interface-path-id [channel-group channel-group-number][error] start**
11. **bert [t1| e1] interface-path-id [channel-group channel-group-number] stop**
12. **show controllers [t1| e1] interface-path-id**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例: RP/0/0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	controller [t1 e1] interface-path-id 例: RP/0/0/CPU0:router(config)# controller t3 0/1/0/0	コントローラ名とインスタンス ID を <i>rack/slot/module/port</i> 表記で指定し、T1 または E1 コントローラ コンフィギュレーション モードを開始します。

コマンドまたはアクション	目的
ステップ 3 <code>channel-group channel-group-number</code> 例: RP/0/0/CPU0:router(config-t1)# channel-group 1 RP/0/0/CPU0:router(config-t1-channel_group)#	特定のチャンネルグループのチャンネルグループ コンフィギュレーション モードを開始します。 <i>channel-group-number</i> を、BERT を設定するチャンネルグループを指す番号に置き換えます。
ステップ 4 <code>bert pattern pattern</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# bert pattern 2^15	T3 ラインで特定のビット誤り率テスト (BERT) のパターンをイネーブルにします。すべてのコントローラおよびチャンネルグループに有効なパターンには、 0s 、 1s 、 2^15 、 2^20 、 2^20-QRSS 、 2^23 、 alt-0-1 があります。T1 および E1 コントローラに有効なパターンには 1in8 、 3in24 、 55Daly 、 55Octet があります。チャンネルグループに有効なパターンには 2^11 、 2^9 、 ds0-1 、 ds0-2 、 ds0-3 、 ds0-4 があります。 (注) BER テストを開始するには、EXEC モードで bert コマンドを使用する必要があります。
ステップ 5 <code>bert interval time</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# bert interval 5	(任意) T3/E3 または T1/E1 ラインでのビット誤り率テスト (BERT) パターンの時間を分単位で指定します。インターバルの値は 1 ~ 14400 の範囲で指定できます。
ステップ 6 <code>end</code> または commit 例: RP/0/0/CPU0:router(config-t1-channel_group)# end または RP/0/0/CPU0:router(config-t1-channel_group)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>exit</code> 例: RP/0/0/CPU0:router(config-t1-channel_group)# exit	チャンネルグループ コンフィギュレーション モードを終了します。

■ クリア チャネル T3/E3 およびチャネル化した T3 コントローラの設定 : 例

	コマンドまたはアクション	目的
ステップ 8	exit 例: RP/0/0/CPU0:router(config-t1)# exit	T1 または E1 コンフィギュレーション モードを終了します。
ステップ 9	exit 例: RP/0/0/CPU0:router(config)# exit	グローバル コンフィギュレーション モードを終了します。
ステップ 10	bert [t1 e1] interface-path-id [channel-group channel-group-number] [error] start 例: RP/0/0/CPU0:router# bert t1 0/3/0/0/0 start RP/0/0/CPU0:router# bert t1 0/3/0/0/0 error	指定したチャネル グループで、設定した BERT テストを開始します。 (注) オプションの error キーワードを指定して、実行中の BERT ストリームにエラーを挿入することもできます。
ステップ 11	bert [t1 e1] interface-path-id [channel-group channel-group-number] stop 例: RP/0/0/CPU0:router# bert t1 0/3/0/0/0 stop	指定したチャネル グループで、設定した BERT テストを停止します。
ステップ 12	show controllers [t1 e3] interface-path-id 例: RP/0/0/CPU0:router# show controllers t3 0/3/0/0	設定した BERT の結果を表示します。

次に行う作業

このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのシリアルインターフェイスの設定](#)」モジュールの説明に従って、テストしたコントローラに関連付けられたシリアル インターフェイスを設定します。

クリア チャネル T3/E3 およびチャネル化した T3 コントローラの設定 : 例

ここでは、次の例について説明します。

クリア チャネル T3 コントローラの設定 : 例

次に、クリア チャネル T3 コントローラの設定例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)#controller T3 0/3/2/0
RP/0/0/CPU0:router(config-t3)#clock source internal
RP/0/0/CPU0:router(config-t3)#mode serial
RP/0/0/CPU0:router(config-t3)#cablelength 4
RP/0/0/CPU0:router(config-t3)#framing c-bit
RP/0/0/CPU0:router(config-t3)#commit
```

```
RP/0/0/CPU0:router (config-t3)#
```

T3 コントローラでのチャネル化した T1 コントローラの設定 : 例

次に、28 T1 コントローラがチャネル化されている T3 コントローラの設定例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router (config)# controller T3 0/3/0/0
RP/0/0/CPU0:router (config-t3)# mode t1
RP/0/0/CPU0:router (config-t3)# framing m23
RP/0/0/CPU0:router (config-t3)# cablelength 11
RP/0/0/CPU0:router (config-t3)# clock source line
RP/0/0/CPU0:router (config-t3)#commit
RP/0/0/CPU0:router (config-t3)#exit
RP/0/0/CPU0:router (config)# exit
RP/0/0/CPU0:router#show controllers T1 ?

0/3/0/0/0   T1 Interface Instance
0/3/0/0/1   T1 Interface Instance
0/3/0/0/10  T1 Interface Instance
0/3/0/0/11  T1 Interface Instance
0/3/0/0/12  T1 Interface Instance
0/3/0/0/13  T1 Interface Instance
0/3/0/0/14  T1 Interface Instance
0/3/0/0/15  T1 Interface Instance
0/3/0/0/16  T1 Interface Instance
0/3/0/0/17  T1 Interface Instance
0/3/0/0/18  T1 Interface Instance
0/3/0/0/19  T1 Interface Instance
0/3/0/0/2   T1 Interface Instance
0/3/0/0/20  T1 Interface Instance
0/3/0/0/21  T1 Interface Instance
0/3/0/0/22  T1 Interface Instance
0/3/0/0/23  T1 Interface Instance
0/3/0/0/24  T1 Interface Instance
0/3/0/0/25  T1 Interface Instance
0/3/0/0/26  T1 Interface Instance
0/3/0/0/27  T1 Interface Instance
0/3/0/0/3   T1 Interface Instance
0/3/0/0/4   T1 Interface Instance
0/3/0/0/5   T1 Interface Instance
--More--
RP/0/0/CPU0:router (config)#
RP/0/0/CPU0:router (config)# controller t1 0/3/0/0/0
RP/0/0/CPU0:router (config-t1)# channel-group 0
RP/0/0/CPU0:router (config-t1-channel_group)# timeslots 1-24
RP/0/0/CPU0:router (config-t1-channel_group)# exit
RP/0/0/CPU0:router (config-t1)# exit
RP/0/0/CPU0:router (config)# controller t1 0/3/0/0/1
RP/0/0/CPU0:router (config-t1)# channel-group 0
RP/0/0/CPU0:router (config-t1-channel_group)# timeslots 1-24
RP/0/0/CPU0:router (config-t1-channel_group)# exit
RP/0/0/CPU0:router (config-t1)# exit
RP/0/0/CPU0:router (config)# controller t1 0/3/0/0/2
RP/0/0/CPU0:router (config-t1)# channel-group 0
RP/0/0/CPU0:router (config-t1-channel_group)# timeslots 1-12
RP/0/0/CPU0:router (config-t1-channel_group)# exit
RP/0/0/CPU0:router (config-t1)# channel-group 1
RP/0/0/CPU0:router (config-t1-channel_group)# timeslots 13-24
RP/0/0/CPU0:router (config-t1-channel_group)# exit
RP/0/0/CPU0:router (config-t1)# exit
```

■ クリア チャネル T3/E3 およびチャネル化した T3 コントローラの設定 : 例

```

RP/0/0/CPU0:router(config)# controller t1 0/3/0/0/3
RP/0/0/CPU0:router(config-t1)# channel-group 0
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 1-6
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1)# channel-group 1
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 7-12
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1)# channel-group 2
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 13-18
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1)# channel-group 3
RP/0/0/CPU0:router(config-t1-channel_group)# timeslots 19-24
RP/0/0/CPU0:router(config-t1-channel_group)# exit
RP/0/0/CPU0:router(config-t1-channel_group)#commit

```

T3 コントローラでの BERT の設定 : 例

次に、T3 コントローラで BERT を設定し、BERT の結果を表示する例を示します。

```

RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# controller t3 0/3/0/1
RP/0/0/CPU0:router(config-t3)# bert pattern 0s

Run bert from exec mode for the bert config to take effect

RP/0/0/CPU0:router(config-t3)#exit
RP/0/0/CPU0:router(config)# exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]
RP/0/0/CPU0:router#bert t3 0/3/0/1 start

```

```

RP/0/0/CPU0:router# bert t3 0/3/0/1 stop

```

```

RP/0/0/CPU0:router# show controllers t3 0/3/0/1

T30/3/0/1 is up
No alarms detected.
MDL transmission is disabled
EIC: , LIC: , FIC: , UNIT:
Path FI:
Idle Signal PORT_NO:
Test Signal GEN_NO:
FEAC code received: No code is being received
Framing is C-BIT Parity, Line Code is B3ZS, Clock Source is Internal
Data in current interval (108 seconds elapsed):
  0 Line Code Violations, 0 P-bit Coding Violation
  0 C-bit Coding Violation, 0 P-bit Err Secs
  0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
  0 Unavailable Secs, 0 Line Errored Secs
  0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
Data in Interval 1:
  0 Line Code Violations, 0 P-bit Coding Violation
  0 C-bit Coding Violation, 0 P-bit Err Secs
  0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
  0 Unavailable Secs, 0 Line Errored Secs
  0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
Data in Interval 2:
  0 Line Code Violations, 0 P-bit Coding Violation
  0 C-bit Coding Violation, 0 P-bit Err Secs

```

```

0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
0 Unavailable Secs, 0 Line Errored Secs
0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
Data in Interval 3:
0 Line Code Violations, 0 P-bit Coding Violation
0 C-bit Coding Violation, 0 P-bit Err Secs
0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
0 Unavailable Secs, 0 Line Errored Secs
0 C-bit Errored Secs, 0 C-bit Severely Errored Secs

```

その他の参考資料

ここでは、T3/E3 および T1/E1 コントローラに関する参考資料について説明します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システムブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
この機能によりサポートされた新規 MIB または改訂 MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでのトンネル インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータで Tunnel-IPSec インターフェイスを設定する方法について説明します。トンネル インターフェイスは、別のトランスポート プロトコル内に任意のパケットのカプセル化を提供する仮想インターフェイスです。保護されていない公開ルートでも、Tunnel-IPSec インターフェイスによってセキュアな通信が可能になります。

仮想インターフェイスは、ルータ内部の論理パケット スイッチング エンティティです。仮想インターフェイスは、グローバル スコープを持ちますが、関連付けられた位置を持ちません。Cisco IOS XR ソフトウェアでは、物理インターフェイスを識別するために *rack/slot/module/port* 表記を使用していますが、インターフェイス名で仮想インターフェイスを識別した後は、グローバルに一意な数字による ID を使用します。この数字による ID は、たとえば、Loopback 0、Loopback 1、Null99999 です。Loopback 0 と Null 0 を同時に使用できるように、各仮想インターフェイス タイプの ID は一意です。

仮想インターフェイスのコントロール プレーンは、アクティブ RP 上に存在します。設定とコントロール プレーンは、スタンバイ RP 上にミラーリングされ、スイッチオーバーが発生した場合には、仮想インターフェイスがそれまでのスタンバイ RP に移り、このスタンバイ RP が新たにアクティブ RP となります。



(注) 親インターフェイスに応じて、サブインターフェイスは物理インターフェイスまたは仮想インターフェイスになります。

仮想トンネルは、任意の RP または DRP で設定されますが、作成および操作は RP からだけ実行されます。



(注) トンネルには、モジュラ サービス カードとの 1 対 1 の関連付けはありません。

Cisco IOS XR ソフトウェアのトンネル インターフェイス設定機能の履歴

リリース	変更点
リリース 2.0	Cisco CRS-1 ルータにこの機能が追加されました。
リリース 3.0	変更ありません。
リリース 3.2	変更ありません。
リリース 3.3.0	変更ありません。
リリース 3.4.0	変更ありません。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。

リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「トンネル インターフェイスを設定するための前提事項」 (P.478)
- 「トンネル インターフェイスの設定に関する情報」 (P.478)
- 「トンネル インターフェイスの設定方法」 (P.480)
- 「トンネル インターフェイスの設定例」 (P.482)
- 「関連情報」 (P.483)
- 「その他の参考資料」 (P.483)

トンネル インターフェイスを設定するための前提事項

この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンド タスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンド タスク ID は、各コマンド リファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。

トンネル インターフェイスの設定に関する情報

トンネル インターフェイスを設定するには、次の概念を理解しておく必要があります。

- 「トンネル インターフェイスの概要」 (P.478)
- 「仮想インターフェイスの命名規則」 (P.479)
- 「Tunnel-IPSec の概要」 (P.479)
- 「Tunnel-IPSec の命名規則」 (P.480)
- 「クリプト プロファイル セット」 (P.480)
- 「トンネル インターフェイスの設定方法」 (P.480)

トンネル インターフェイスの概要

トンネリングによって、トランスポート プロトコル内の任意のパケットをカプセル化できるようになります。この機能は、設定用の単純なインターフェイスを用意するために、仮想インターフェイスとして実装されます。トンネル インターフェイスは特定の「パッセンジャ」プロトコルや「トランスポート」プロトコルに関連付けられません。トンネル インターフェイスは、任意の標準のポイントツーポイント

イント カプセル化スキームを実装するために必要なサービスを提供できるように設計されたアーキテクチャです。サポートされるトンネルはポイントツーポイントリンクなので、リンクごとに個別のトンネルを設定する必要があります。

トンネル インターフェイスを設定するには、必須の手順が 3 つあります。

1. トンネル インターフェイスを設定します。 **interface tunnel-ipsec ID** を指定します。
2. トンネルの発信元を設定します。 **tunnel source {ip-address | interface-id}** を指定します。
3. トンネルの宛先を設定します。 **tunnel destination {ip-address | tunnel-id}** を指定します。

仮想インターフェイスの命名規則

仮想インターフェイス名では、インターフェイスのラック、スロット、モジュール、およびポートを識別するために、*rack/slot/module/port* という物理インターフェイスの表記方法を使用しません。仮想インターフェイスは、物理的なインターフェイスやサブインターフェイスには関連付けられないためです。

仮想インターフェイスでは、仮想インターフェイス タイプごとに、グローバルに一意的な数字による ID を使用します。

仮想インターフェイスの表記例：

Interface	IP-Address	Status	Protocol
Loopback0	10.9.0.0	Up	Up
Loopback10	10.7.0.0	Up	Up
Tunnel-TE5000	172.18.189.38	Down	Down
Null10	10.8.0.0	Up	Up

Tunnel-IPSec の概要

IP Security (IPSec; IP セキュリティ) は、インターネット上のプライベート通信のセキュリティを確保するためのオープン スタンドアロッドのフレームワークです。IPSec は、パブリック ネットワークやセキュアではないネットワークでデータを送信する必要がある、仮想プライベート ネットワーク (VPN) やファイアウォールなどのアプリケーションをサポートするために使用できます。ルータ IPSec プロトコルスイートには、IP レイヤにプライバシー、完全性、および認証サービスを提供するために使用できる一連の標準が用意されています。また、IPSec プロトコルスイートには、ネットワーク レイヤ セキュリティの主要な管理要件をサポートする暗号化技術も含まれます。

IPSec を使用すると、Secure Shell (SSH; セキュア シェル) または Secure Socket Layer (SSL; セキュア ソケット レイヤ) を使用する必要はありません。使用すると、同じデータの暗号化と復号化が 2 回実行され、不要なオーバーヘッドが生じます。IPSec デーモンは、RP と DRP の両方で実行されます。IPSec はルータのオプションの機能です。IPSec は、セキュアなトランスポートが必要なアプリケーションが複数あるユーザに適しています。クライアント側では、Cisco VPN 3000 Client や他のサードパーティ製 IPSec クライアント ソフトウェアを使用して、IPSec VPN を構築できます。



(注)

IPSec トンネルはコントロールプレーンに存在するため、トンネルを始動または終了する必要はありません。IPSec トンネルへの送信は、RP または DRP からローカルで発信されたトラフィックの場合だけであり、Tunnel-IPSec に適用するプロファイルの一部として設定されたアクセス制御リスト (ACL) によって検出されます。

Tunnel-IPSec の命名規則

プロファイルは、インターフェイスの `tunnel-ipsec` のインターフェイス設定サブモードから入力されません。例：

```
interface tunnel-ipsec 30
  profile <profile name>
```

クリプト プロファイル セット

クリプト プロファイル セットを設定し、トンネル インターフェイス（またはクリプト IPSec トランスポート）に適用する必要があります。クリプト IPSec トランスポートの使用の詳細については、「[他の参考資料](#)」(P.483)に記載されているリンクを参照してください。2つの IPSec ピア間で IPSec が正常に動作するには、両方のピアのクリプト プロファイル エントリに互換性のある設定ステートメントを含める必要があります。

2つのピアがセキュリティ アソシエーションの確立を試行するには、相手側ピアのクリプト プロファイル エントリのいずれかと互換性のあるクリプト プロファイル エントリが、各ピアに1つ以上必要です。2つのクリプト プロファイル エントリに互換性があると判断するには、少なくとも次の基準を満たす必要があります。

- 互換性のあるクリプト アクセス リストを含む必要があります。応答側のピアが動的クリプト プロファイルを使用している場合、ローカルのクリプト アクセス リストのエントリは、ピアのクリプト アクセス リストから「許可」される必要があります。
- 各ピアは、相手側ピアを識別する必要があります（ただし、応答側ピアが動的クリプト プロファイルを使用している場合を除きます）。
- 少なくとも1つのトランスフォーム セットが共通している必要があります。



(注)

クリプト プロファイルは共有できません。つまり、複数のインターフェイスに同じプロファイルは設定できません。

トンネル インターフェイスの設定方法

ここでは、次の手順について説明します。

- 「[Tunnel-IPSec インターフェイスの設定](#)」(P.480) (必須)

Tunnel-IPSec インターフェイスの設定

ここでは、Tunnel-IPSec インターフェイスの設定方法について説明します。

前提条件

profile コマンドを使用するには、クリプト コマンドの適切なタスク ID を含むタスク グループに関連付けられたユーザ グループに属している必要があります。**tunnel destination** コマンドを使用するには、インターフェイス コマンドの適切なタスク ID を含むタスク グループに関連付けられたユーザ グループに属している必要があります。

ユーザ グループとタスク ID の詳細については、『Cisco IOS XR System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。

次のタスクは、Tunnel-IPSec インターフェイスを作成するために必要です。

- IPSec セキュリティ アソシエーションのグローバルな存続期間を設定する
- チェックポイント処理を設定する
- クリプト プロファイルを設定する

前提条件のチェックポイント処理とクリプト プロファイルの設定方法、および IPSec セキュリティ アソシエーションのグローバルな存続期間を設定する方法の詳細については、『Cisco IOS XR System Security Configuration Guide』の「Implementing IPSec Network Security on Cisco IOS XR Software」モジュールを参照してください。

クリプト プロファイルの設定後は、IPSec トラフィックが通過する各トンネル インターフェイスにクリプト プロファイルを適用する必要があります。トンネル インターフェイスにクリプト プロファイル セットを適用すると、ルータは、クリプトで保護されるトラフィックの代理として、接続またはセキュリティ アソシエーションのネゴシエーション中に、クリプト プロファイル セットと照合してすべてのインターフェイスのトラフィックを評価し、指定したポリシーを使用するようになります。

手順の概要

1. **configure**
2. **interface tunnel-ipsec identifier**
3. **profile profile-name**
4. **tunnel source {ip-address | interface-id}**
5. **tunnel destination {ip-address | tunnel-id}**
6. **end**
または
commit
7. **show ip route**

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface tunnel-ipsec identifier 例： RP/0/RP0/CPU0:router(config)# interface tunnel-ipsec 30	クリプト プロファイルを適用する IPSec インターフェイスを特定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	profile profile-name 例： RP/0/RP0/CPU0:router(config-if)# profile user1	IPSec プロセスのトンネルに適用するクリプト プロファイル名を割り当てます。 • 同じクリプト プロファイルは、異なる IPSec モードで共有できません。

■ トンネル インターフェイスの設定例

コマンドまたはアクション	目的
ステップ 4 <code>tunnel source</code> (<i>ip-address</i> <i>interface-id</i>) 例: RP/0/RP0/CPU0:router(config-if)# tunnel source Ethernet0/1/1/2	トンネルの発信元 IP アドレスまたはインターフェイス ID を指定します。 <ul style="list-style-type: none"> このコマンドは、静的プロファイルと動的プロファイルのどちらにも必要です。
ステップ 5 <code>tunnel destination</code> { <i>ip-address</i> <i>tunnel-id</i> } 例: RP/0/RP0/CPU0:router(config-if)# tunnel destination 192.168.164.19	(任意) トンネルの宛先 IP アドレスを指定します。 <ul style="list-style-type: none"> 動的プロファイルの場合、このコマンドは不要です。
ステップ 6 <code>end</code> または <code>commit</code> 例: RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
ステップ 7 <code>show ip route</code> 例: RP/0/RP0/CPU0:router# show ip route	トンネルのフォワーディング情報を表示します。 <ul style="list-style-type: none"> show ip route コマンドで、アドバタイズの内容、および静的ルートと自動ルートが表示されます。

トンネル インターフェイスの設定例

ここでは、次の例について説明します。

[「Tunnel-IPSec : 例」 \(P.482\)](#)

Tunnel-IPSec : 例

次に、プロファイルを作成し、IPSec トンネルに適用するプロセスの例を示します。必要な準備手順についても示します。まずトランスフォーム セットを定義し、プロファイルを作成してから、IPSec トンネルを設定します。

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# crypto ipsec transform-set tset1
RP/0/RP0/CPU0:router(config-transform-set tset1)# transform esp-sha-hmac
RP/0/RP0/CPU0:router(config-transform-set tset1)# end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: yes

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# crypto ipsec profile user1
RP/0/RP0/CPU0:router(config-user1)# match sampleacl transform-set tset1
RP/0/RP0/CPU0:router(config-user1)# set pfs group5
RP/0/RP0/CPU0:router(config-user1)# set type dynamic
RP/0/RP0/CPU0:router(config-user1)# exit

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tunnel-ipsec 30
RP/0/RP0/CPU0:router(config-if)# profile user1
RP/0/RP0/CPU0:router(config-if)# tunnel source MgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:router(config-if)# tunnel destination 192.168.164.19
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: yes

```

関連情報

次に、各トランスポートにクリプトプロファイルを適用する必要があります。トランスポートにクリプトプロファイルセットを適用すると、ルータは、クリプトで保護されるトラフィックの代理として、接続またはセキュリティアソシエーションのネゴシエーション中に、クリプトプロファイルセットと照合してすべてのインターフェイスのトラフィックを評価し、指定したポリシーを使用するようになります。

各トランスポートにクリプトプロファイルを適用する方法の詳細については、『Cisco IOS XR System Security Configuration Guide』の「Implementing IPSec Network Security on Cisco IOS XR Software」モジュールを参照してください。

その他の参考資料

ここでは、トンネルインターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
IPSec とクリプトプロファイルに関する情報	『Cisco IOS XR System Security Configuration Guide』
MPLS-TE 用にトンネルインターフェイスを設定する方法など、MPLS トラフィック エンジニアリングに関する情報	『Cisco IOS XR Multiprotocol Label Switching Configuration Guide』

内容	参照先
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



Cisco IOS XR ソフトウェアでの 802.1Q VLAN インターフェイスの設定

ここでは、Cisco IOS XR ソフトウェアをサポートするルータの 802.1Q VLAN インターフェイスの設定と管理について説明します。

IEEE 802.1Q 仕様は、VLAN メンバーシップ 情報のあるタグ付きイーサネット フレームの標準方式を確立し、ブリッジド LAN インフラストラクチャ内にある VLAN トポロジーの定義、操作、および管理ができる VLAN ブリッジの動作を定義します。

802.1Q 規格では、ブロードキャストおよびマルチキャストのトラフィックが必要以上の帯域を消費しないように、大規模なネットワークを小規模なパーツに分割することで問題に対処することを目的としています。また、内部ネットワークのセグメント間に、より高レベルのセキュリティを実現できます。

Cisco IOS XR ソフトウェアの 802.1Q VLAN インターフェイス設定の機能履歴

リリース	変更点
リリース 3.2	Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータに、この機能が追加されました。
リリース 3.3.0	<ul style="list-style-type: none">バンドルされたイーサネット インターフェイスでの VLAN コマンドのサポートが追加されました。Cisco CRS-1 ルータ SPA での dot1q native vlan コマンドのサポートが追加されました。
リリース 3.4.0	<ul style="list-style-type: none">レイヤ 2 バーチャル プライベート ネットワーク (L2VPN) 機能が最初にサポートされたのは、Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータのイーサネット インターフェイスでした。8 ポート 1 ギガビット イーサネット SPA について、Cisco CRS-1 ルータおよび Cisco XR 12000 シリーズ ルータでのサポートが追加されました。
リリース 3.5.0	変更ありません。
リリース 3.6.0	変更ありません。
リリース 3.7.0	変更ありません。
リリース 3.8.0	変更ありません。

この章の構成

- 「802.1Q VLAN インターフェイス設定の前提条件」 (P.486)
- 「802.1Q VLAN インターフェイスの設定に関する情報」 (P.486)
- 「802.1Q VLAN インターフェイスの設定方法」 (P.489)
- 「VLAN インターフェイスの設定例」 (P.499)
- 「その他の参考資料」 (P.501)

802.1Q VLAN インターフェイス設定の前提条件

802.1Q VLAN インターフェイスを設定する前に、次の条件を満たしていることを確認してください。

- この設定作業を行うには、Cisco IOS XR ソフトウェアのシステム管理者が、対応するコマンドタスク ID を含むタスク グループに関連付けられたユーザ グループにユーザを割り当てる必要があります。すべてのコマンドタスク ID は、各コマンドリファレンスおよび『*Cisco IOS XR Task ID Reference Guide*』に記載されています。

タスク グループの割り当てについてサポートが必要な場合は、システム管理者に連絡してください。ユーザ グループおよびタスク ID の詳細については、『*Cisco IOS XR Software System Security Configuration Guide*』の「*Configuring AAA Services on Cisco IOS XR Software*」モジュールを参照してください。
- ギガビットイーサネット インターフェイス、10 ギガビットイーサネット インターフェイス、ファストイーサネット インターフェイス、またはイーサネットバンドルの設定が完了している必要があります。
 - ギガビットイーサネット、10 ギガビットイーサネット、or ファストイーサネット インターフェイスを設定するには、このマニュアルで前述した「*Cisco IOS XR ソフトウェアでのイーサネット インターフェイスの設定*」モジュールを参照してください。
 - イーサネットバンドルを設定するには、このマニュアルで後述する「*Cisco IOS XR ソフトウェアでのリンクバンドルの設定*」モジュールを参照してください。

802.1Q VLAN インターフェイスの設定に関する情報

802.1Q VLAN インターフェイスを設定するには、次の概念を理解しておく必要があります。

- 「802.1Q VLAN の概要」 (P.487)
- 「802.1Q タグ付きフレーム」 (P.487)
- 「サブインターフェイス」 (P.487)
- 「サブインターフェイス MTU」 (P.487)
- 「ネイティブ VLAN」 (P.488)
- 「イーサネットバンドルでの VLAN サブインターフェイス」 (P.488)

802.1Q VLAN の概要

VLAN とは、実際は異なる LAN セグメント上のデバイスでも、同じセグメントで接続している場合と同様に通信できるように設定された、1 つまたは複数の LAN 上にあるデバイスのグループです。

VLAN は、物理接続ではなく論理接続に基づいているため、ユーザ管理、ホスト管理、帯域割り当て、およびリソースの最適化がとて柔軟です。

IEEE 802.1Q プロトコル規格では、ブロードキャストおよびマルチキャストのトラフィックが必要以上の帯域を消費しないように、大規模なネットワークを小規模なパーツに分割することで問題に対処しています。また、内部ネットワークのセグメント間に、より高レベルのセキュリティを実現できます。

802.1Q仕様は、イーサネットフレームに VLAN メンバーシップ情報を挿入する標準方式を確立します。

Cisco IOS XR ソフトウェアは、ギガビットイーサネット、10 ギガビットイーサネット、およびファストイーサネットインターフェイス上での VLAN サブインターフェイス設定をサポートします。

802.1Q タグ付きフレーム

IEEE 802.1Q タグ ベースの VLAN は、MAC ヘッダーの特別なタグを使用し、ブリッジでのフレームの VLAN メンバーシップを識別できます。このタグは、VLAN およびサービス品質 (QoS) プライオリティの識別に使用されます。VLAN は、手動での入力によってスタティックに作成することも、Generic Attribute Registration Protocol (GARP) VLAN Registration プロトコル (GVRP) を介してダイナミックに作成することもできます。VLAN ID は、フレームを特定の VLAN に関連付けて、スイッチがネットワークでフレームを処理する必要があるという情報を提供します。タグ付きフレームは、タグなしフレームよりも 4 バイト長く、イーサネットフレームの Type および Length フィールドにある 2 バイトの Tag Protocol Identifier (TPID) フィールドと、イーサネットフレームの Source Address フィールドの後ろから始まる 2 バイトの Tag Control Information (TCI) が含まれます。

サブインターフェイス

サブインターフェイスは、ハードウェア インターフェイス上に作成される論理インターフェイスです。これらのソフトウェア定義のインターフェイスにより、単一のハードウェア インターフェイス上でトラフィックを論理チャンネルに分割することができ、また、物理インターフェイス上で帯域幅を効率的に利用することができます。

サブインターフェイスは、インターフェイス名の末尾に拡張を追加することで、他のインターフェイスと区別されます。たとえば、物理インターフェイス TenGigE 0/1/0/0 上のイーサネット サブインターフェイス 23 は、TenGigE 0/1/0/0.23 となります。

サブインターフェイスがトラフィックを渡すことができるようにするには、有効なタグ付きプロトコルのカプセル化と VLAN 識別子の割り当てが必要です。すべてのイーサネット サブインターフェイスは常に、デフォルトで 802.1Q VLAN でカプセル化されます。ただし、VLAN 識別子は明示的に定義する必要があります。

サブインターフェイス MTU

サブインターフェイスの MTU は、物理インターフェイスから継承されます。これには、802.1Q VLAN タグに許可されている追加の 4 バイトも含まれます。

ネイティブ VLAN

各物理ポートには、ネイティブ VLAN が割り当てられていることがあります。PVID パラメータに指定された LAN には、すべてのタグなしフレームが割り当てられています。受信パケットが PVID によるタグ付きであると、そのパケットはタグなしとして扱われます。このため、ネイティブ VLAN に関連付けられた設定は、メイン インターフェイス上で行う必要があります。ネイティブ VLAN では、VLAN 対応のブリッジまたはステーションと、VLAN に対応しないブリッジまたはステーションとの共存は許可されます。

イーサネットバンドルでの VLAN サブインターフェイス

イーサネットバンドルは、1 つ以上のイーサネットポートのグループを集約し、1 つのリンクとして扱うようにしたものです。単一のイーサネットバンドルに複数の VLAN サブインターフェイスを追加することができます。

イーサネットバンドルの設定方法については、このマニュアルで後述する「[Cisco IOS XR ソフトウェアでのリンクバンドルの設定](#)」モジュールを参照してください。イーサネットバンドルに VLAN サブインターフェイスを作成する手順は、物理イーサネットインターフェイスに VLAN サブインターフェイスを作成する手順とまったく同じです。

イーサネットバンドルに VLAN サブインターフェイスを作成するには、このモジュールで後述する「[802.1Q VLAN インターフェイスの設定方法](#)」を参照してください。



(注)

イーサネットバンドルは、Cisco CRS-1 ルータでのみサポートされます。Cisco XR 12000 シリーズ ルータではサポートされません。

VLAN インターフェイスでのレイヤ 2 VPN

レイヤ 2 バーチャルプライベート ネットワーク (L2VPN) 機能によって、サービスプロバイダ (SP) は地理的に離れたカスタマー サイトにもレイヤ 2 サービスを提供できるようになります。詳細は、このマニュアルで前述した「[Cisco IOS XR ソフトウェアでのイーサネットインターフェイスの設定](#)」モジュールにある「[イーサネットインターフェイスでのレイヤ 2 VPN](#)」を参照してください。

VLAN 接続回路 (AC) を設定するための設定モデルは、基本の VLAN の設定に使用するモデルに類似しています。ユーザはまず VLAN サブインターフェイスを作成し、次にサブインターフェイス コンフィギュレーション モードで VLAN を設定します。AC を作成するには、**interface** コマンド文字列に **l2transport** キーワードを含めて、そのインターフェイスがレイヤ 2 インターフェイスであることを指定する必要があります。

VLAN AC は、L2VPN 操作の 3 つのモードをサポートします。

- 基本の Dot1Q AC : AC は、特定の VLAN タグで送受信されるすべてのフレームに対応します。
- Q-in-Q AC : AC は、特定の外部 VLAN タグおよび特定の内部 VLAN タグで送受信されるすべてのフレームに対応します。Q-in-Q は、2 つのタグのスタックを使用する Dot1Q の拡張です。
- Q-in-Any AC : AC は、内部 VLAN タグが L3 終端でない限り、特定の外部 VLAN タグおよび任意の内部 VLAN タグで送受信されるすべてのフレームに対応します。Q-in-Any は、ワイルドカード化を使用して任意の 2 番目のタグに一致させる Q-in-Q の拡張です。



(注) Q-in-Any モードは、基本の Dot1Q モードを変化させたものです。Q-in-Any モードではフレームは基本の Q-in-Q カプセル化が行われていますが、Q-in-Any モードでは内部タグは無関係です。ただし、いくつかの特定の内部 VLAN タグが特定のサービス用に使用される場合を除きます。たとえば、一般的なインターネット アクセスに L3 サービスを提供するために、あるタグが使用されることがあります。

CE-to-PE リンクの各 VLAN は、(VC タイプ 4 または VC タイプ 5 を使用する) 独立した L2VPN 接続として設定できます。VLAN に L2VPN を設定するには、「[VLAN での接続回路の設定](#)」(P.494) を参照してください。

VLAN に L2VPN を設定する場合は、次の事項に注意する必要があります。

- Cisco IOS XR ソフトウェアは LC ごとに 4k AC をサポートします。
- ポイントツーポイント接続では、2 つの AC を同じタイプにするべきではありません。たとえば、ポート モードのイーサネット AC を、Dot1Q イーサネット AC に接続することができます。
- 疑似接続は、VLAN モードまたはポート モードで実行できます。VLAN モードで実行される疑似接続に単一の Dot1Q タグを設定することができますが、ポート モードで実行される疑似接続にタグを設定することはできません。これらの異なるタイプの回路を接続するには、インターワーキングが必要です。この場合のインターワーキングは、タグのポップ、プッシュ、書き換えの形を取ります。レイヤ 2 VPN を使用するメリットは、まったく異なるタイプのメディアを接続するのに必要なインターワーキングを簡素化できることにあります。
- MPLS 疑似接続の両側にある AC は、異なるタイプでもかまいません。この場合、AC の一方または両方のエンドで、疑似接続を行うための適切な変換が行われます。

AC および疑似接続の情報を表示するには、**show interfaces** コマンドを使用します。



(注) L2VPN ネットワークの設定の詳細については、『*Cisco IOS XR Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs*」モジュールを参照してください。

802.1Q VLAN インターフェイスの設定方法

ここでは、次の手順について説明します。

- 「[802.1Q VLAN サブインターフェイスの設定](#)」(P.489)
- 「[ネイティブ VLAN の設定](#)」(P.492)
- 「[VLAN での接続回路の設定](#)」(P.494)
- 「[802.1Q VLAN サブインターフェイスの削除](#)」(P.497)

802.1Q VLAN サブインターフェイスの設定

ここでは、802.1Q VLAN サブインターフェイスの設定手順について説明します。これらのサブインターフェイスを削除するには、このモジュールの「[802.1Q VLAN サブインターフェイスの削除](#)」を参照してください。

手順の概要

1. configure

2. **interface** {GigabitEthernet | TenGigE | fastethernet | Bundle-Ether}
interface-path-id.subinterface
3. **dot1q vlan** *vlan-id*
4. **ipv4 address** *ip-address mask*
5. **exit**
6. ステップ 2 ~ 5 を繰り返し、残りの VLAN サブインターフェイスを定義します。
7. **end**
または
commit
8. **show vlan interface** [{GigabitEthernet | TenGigE | Bundle-Ether | fastethernet}
interface-path-id] [*location instance*]
9. **show vlan trunks** [*brief*] [*location instance*] [{GigabitEthernet | TenGigE | Bundle-Ether |
fastethernet} *interface-path-id*] [*summary*]

詳細手順

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface {GigabitEthernet TenGigE Bundle-Ether fastethernet} <i>interface-path-id.subinterface</i> 例： RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.10	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 <ul style="list-style-type: none"> • <i>interface-path-id</i> 引数を、次のいずれかのインスタンスに置き換えます。 <ul style="list-style-type: none"> – 物理イーサネット インターフェイス インスタンスまたはイーサネットバンドルインスタンス。名前の表記は <i>rack/slot/module/port</i> の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ~ 65535 です。 • <i>subinterface</i> 引数を、サブインターフェイス値に置き換えます。範囲は 0 ~ 4095 です。 • 名前の表記は <i>instance.subinterface</i> の形式で、表記の一部として引数をピリオドで区切る必要があります。
ステップ 3	dot1q vlan <i>vlan-id</i> 例： RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 100	VLAN AC をサブインターフェイスに割り当てます。 <ul style="list-style-type: none"> • <i>vlan-id</i> 引数にはサブインターフェイス ID を指定します。範囲は 1 ~ 4094 です (0 と 4095 は予約されています)。基本の Dot1Q AC を設定するには、次の構文を使用します。 dot1q vlan <i>vlan-id</i> • Q-in-Q AC を設定するには、次の構文を使用します。 dot1q vlan <i>vlan-id vlan</i> <i>vlan-id</i>

コマンドまたはアクション	目的
<p>ステップ 4 <code>ipv4 address ip-address mask</code></p> <p>例: RP/0/RP0/CPU0:router(config-subif)# ipv4 address 178.18.169.23/24</p>	<p>サブインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p> <ul style="list-style-type: none"> • <i>ip-address</i> をインターフェイスのプライマリ IPv4 アドレスに置き換えます。 • <i>mask</i> を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクを指定するには、次の 2 つの方法があります。 <ul style="list-style-type: none"> – ネットワーク マスクには、4 パートのドット付き 10 進アドレスを指定できます。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワーク アドレスに属することを示します。 – ネットワーク マスクはスラッシュ (/) と数字で指定できます。たとえば、/8 は、マスクの先頭 8 ビットが 1 で、アドレスの対応するビットがネットワーク アドレスであることを示します。
<p>ステップ 5 <code>exit</code></p> <p>例: RP/0/RP0/CPU0:router(config-subif)# exit</p>	<p>(任意) サブインターフェイス コンフィギュレーション モードを終了します。</p> <ul style="list-style-type: none"> • <code>exit</code> コマンドは、明示的に指定する必要はありません。
<p>ステップ 6 ステップ 2 ～ 5 を繰り返し、残りの VLAN サブインターフェイスを定義します。</p>	-
<p>ステップ 7 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config)# end または RP/0/RP0/CPU0:router(config)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • <code>end</code> コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – <code>yes</code> と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – <code>no</code> と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – <code>cancel</code> と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーションセッションを継続するには、<code>commit</code> コマンドを使用します。

コマンドまたはアクション	目的
<p>ステップ 8 <code>show vlan interface</code> [<i>type interface-path-id</i>] [<i>location instance</i>]</p> <p>例: RP/0/RP0/CPU0:router# show vlan interface 5</p>	<p>(任意) インターフェイスの設定を表示します。</p> <ul style="list-style-type: none"> 特定のポートの設定を表示するには、location キーワードを使用します。 指定したインターフェイスまたはサブインターフェイスの設定を表示するには、interface キーワードを使用します。
<p>ステップ 9 <code>show vlan trunks</code> [<i>brief</i>] [<i>location instance</i>] [{<i>GigabitEthernet</i> <i>TenGigE</i> <i>Bundle-Ether</i> <i>fastethernet</i>} <i>interface-path-id</i>] [<i>summary</i>]</p> <p>例: RP/0/RP0/CPU0:router# show vlan trunk summary</p>	<p>(任意) 各 VLAN トランク インターフェイスに関するサマリー情報を表示します。</p> <ul style="list-style-type: none"> キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> brief : 簡潔なサマリーを表示します。 summary : 完全なサマリーを表示します。 location : 指定したポート上の VLAN トランク インターフェイスに関する情報を表示します。 interface : 指定したインターフェイスまたはサブインターフェイスに関する情報を表示します。

ネイティブ VLAN の設定

ここでは、インターフェイスでネイティブまたはデフォルトの VLAN を設定する手順について説明します。



(注)

この手順は、Cisco XR 12000 シリーズ ルータには該当しません。

手順の概要

- `configure`
- `interface` {*GigabitEthernet* | *TenGigE* | *fastethernet* | *Bundle-Ether*} *interface-path-id*
- `dot1q native vlan` *number*
- `end`
または
`commit`
- `show vlan trunks` [*brief*] [*location instance*] [{*GigabitEthernet* | *TenGigE* | *Bundle-Ether* | *fastethernet*} *interface-path-id*] [*summary*]

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface {GigabitEthernet TenGigE fastethernet Bundle-Ether} interface-path-id</code> 例: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4	インターフェイス コンフィギュレーション モードを開始し、イーサネット インターフェイスの名称を指定します。 <i>interface-path-id</i> 引数を、次のいずれかのインスタンスに置き換えます。 <ul style="list-style-type: none"> 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は <i>rack/slot/module/port</i> の形式で、表記の一部として値をスラッシュで区切る必要があります。 イーサネット バンドル インスタンス。範囲は 1 ~ 65535 です。
ステップ 3 <code>dot1q native vlan number</code> 例: RP/0/RP0/CPU0:router(config-if)# dot1q native vlan 1	802.1Q トランク インターフェイスに関連付けられた、デフォルトまたはネイティブの VLAN を定義します。 <ul style="list-style-type: none"> <i>number</i> 引数は、トランク インターフェイスの ID です。 範囲は 1 ~ 4094 です (0 と 4095 は予約されています)。

コマンドまたはアクション	目的
<p>ステップ 4</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end または RP/0/RP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 5</p> <pre>show vlan trunks [brief] [location instance] [GigabitEthernet TenGigE Bundle-Ether fastethernet] interface-path-id] [summary]</pre> <p>例:</p> <pre>RP/0/RP0/CPU0:router# show vlan trunk summary</pre>	<p>(任意) 各 VLAN トランク インターフェイスに関するサマリー情報を表示します。</p> <ul style="list-style-type: none"> キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> brief : 簡潔なサマリーを表示します。 summary : 完全なサマリーを表示します。 location : 指定したポート上の VLAN トランク インターフェイスに関する情報を表示します。 interface : 指定したインターフェイスまたはサブインターフェイスに関する情報を表示します。

VLAN での接続回路の設定

VLAN で接続回路を設定するには、次の手順で操作します。

手順の概要

1. **configure**
2. **interface** {**GigabitEthernet** | **TenGigE** | **fastethernet** | **Bundle-Ether**} *interface-path-id.subinterface* **l2transport**
3. **dot1q vlan** *vlan-id* [**vlan** {*vlan id* | **any**}]
4. **l2protocol** {**cdp** | **pvst** | **stp** | **vtp**} {[**tunnel**] **experimental bits** | **drop**}
5. **end**
または
commit

6. show interfaces [GigabitEthernet | TenGigE] interface-path-id

詳細手順

コマンドまたはアクション	目的
ステップ 1 <code>configure</code> 例: RP/0/RP0/CPU0:router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2 <code>interface [GigabitEthernet TenGigE] interface-path-id.subinterface l2transport</code> 例: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0.1 l2transport	サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。 <ul style="list-style-type: none"> • <code>interface-path-id</code> 引数を、次のいずれかのインスタンスに置き換えます。 <ul style="list-style-type: none"> – 物理イーサネット インターフェイス インスタンスまたはイーサネットバンドル インスタンス。名前の表記は <code>rack/slot/module/port</code> の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネットバンドル インスタンス。範囲は 1 ~ 65535 です。 • <code>subinterface</code> 引数を、サブインターフェイス値に置き換えます。範囲は 0 ~ 4095 です。 • 名前の表記は <code>instance.subinterface</code> の形式で、表記の一部として引数をピリオドで区切る必要があります。 (注) コマンド文字列に <code>l2transport</code> キーワードを含める必要があります。そうしないと、AC ではなく、レイヤ 3 サブインターフェイスが作成されます。
ステップ 3 <code>dot1q vlan vlan-id [vlan {vlan-id any}]</code> 例: RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 10 vlan any	VLAN AC をサブインターフェイスに割り当てます。 <ul style="list-style-type: none"> • <code>vlan-id</code> 引数にはサブインターフェイス ID を指定します。範囲は 1 ~ 4094 です (0 と 4095 は予約されています)。基本の Dot1Q AC を設定するには、次の構文を使用します。 <code>dot1q vlan vlan-id</code> • Q-in-Q AC を設定するには、次の構文を使用します。 <code>dot1q vlan vlan-id vlan vlan-id</code> • Q-in-Any AC を設定するには、次の構文を使用します。 <code>dot1q vlan vlan-id vlan any</code>

802.1Q VLAN インターフェイスの設定方法

コマンドまたはアクション	目的
<p>ステップ 4 <code>l2protocol {cdp pvst stp vtp} {[tunnel] experimental bits drop}</code></p> <p>例: RP/0/RP0/CPU0:router(config-subif)# l2protocol pvst tunnel</p>	<p>インターフェイスでレイヤ 2 プロトコルのトンネリングおよびデータ ユニットのパラメータを設定します。</p> <p>使用できるプロトコルは次のとおりです。</p> <ul style="list-style-type: none"> • cdp : CDP のトンネリングおよびデータ ユニットのパラメータ。 • pvst : VLAN スパニング ツリー プロトコルのトンネリングおよびデータ ユニットのパラメータを設定します。 • stp : スパニング ツリー プロトコルのトンネリングおよびデータ ユニットのパラメータ。 • vtp : VLAN Trunk Protocol のトンネリングおよびデータ ユニットのパラメータ。 <p>指定したプロトコルに関連するパケットをトンネル処理するには、tunnel オプションを含めます。</p> <p>指定したプロトコルの EXP ビットを変更するには、experimental bits キーワードの引数を含めます。</p> <p>指定したプロトコルに関連するパケットをドロップするには、drop キーワードを含めます。</p>
<p>ステップ 5 <code>end</code> または <code>commit</code></p> <p>例: RP/0/RP0/CPU0:router(config-if-12)# end または RP/0/RP0/CPU0:router(config-if-12)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: – yes と入力すると、実行コンフィギュレーションファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーションセッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーションセッションが継続されます。 • 設定変更を実行コンフィギュレーションファイルに保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。
<p>ステップ 6 <code>show interfaces [GigabitEthernet TenGigE] interface-path-id.subinterface</code></p> <p>例: RP/0/RP0/CPU0:router# show interfaces TenGigE 0/3/0/0.1</p>	<p>(任意) ルータ上のインターフェイスに関する統計情報を表示します。</p>

次に行う作業

- AC でポイントツーポイントの疑似相互接続を設定する方法については、『*Cisco IOS XR Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2 VPNs*」モジュールを参照してください。
- マルチプロトコル ラベル スイッチング (MPLS) やサービス品質 (QoS) など、レイヤ 3 サービス ポリシーを VLAN に付加する方法については、該当する Cisco IOS XR ソフトウェアのコンフィギュレーション ガイドを参照してください。

802.1Q VLAN サブインターフェイスの削除

ここでは、このモジュールの「[802.1Q VLAN サブインターフェイスの設定](#)」で設定した 802.1Q VLAN サブインターフェイスを削除する手順について説明します。

手順の概要

1. **configure**
2. **no interface** {**GigabitEthernet** | **TenGigE** | **fastethernet** | **Bundle-Ether**} *interface-path-id.subinterface*
3. ステップ 2 を繰り返し、その他の VLAN サブインターフェイスを削除します。
4. **end**
または
commit
5. **show vlan interface** [*type interface-path-id*] [**location instance**]
6. **show vlan trunks** [**brief**] [**location instance**] [{**GigabitEthernet** | **TenGigE** | **Bundle-Ether** | **fastethernet**} *interface-path-id*] [**summary**]

詳細手順

	コマンドまたはアクション	目的
ステップ 1	<p>configure</p> <p>例： RP/0/RP0/CPU0:router# configure</p>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 2	<p>no interface {GigabitEthernet TenGigE fastethernet Bundle-Ether} <i>interface-path-id.subinterface</i></p> <p>例： RP/0/RP0/CPU0:router(config)# no interface TenGigE 0/2/0/4.10</p>	<p>サブインターフェイスを削除すると、そのサブインターフェイスに適用されているすべての設定も自動的に削除されます。</p> <ul style="list-style-type: none"> • <i>interface-path-id</i> 引数を、次のいずれかのインスタンスに置き換えます。 <ul style="list-style-type: none"> – 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前の表記は <i>rack/slot/module/port</i> の形式で、表記の一部として値をスラッシュで区切る必要があります。 – イーサネット バンドル インスタンス。範囲は 1 ～ 65535 です。 • <i>subinterface</i> 引数を、サブインターフェイス値に置き換えます。範囲は 0 ～ 4095 です。 <p>名前の表記は <i>instance.subinterface</i> の形式で、表記の一部として引数をピリオドで区切る必要があります。</p>
ステップ 3	<p>ステップ 2 を繰り返し、その他の VLAN サブインターフェイスを削除します。</p>	-
ステップ 4	<p>end または commit</p> <p>例： RP/0/RP0/CPU0:router(config)# end または RP/0/RP0/CPU0:router(config)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを発行すると、変更のコミットを求めるプロンプトが表示されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、設定変更をコミットせずにコンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 – cancel と入力すると、コンフィギュレーション セッションの終了や設定変更のコミットは行われず、ルータでは現在のコンフィギュレーション セッションが継続されます。 • 設定変更を実行コンフィギュレーション ファイルに保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

コマンドまたはアクション	目的
<p>ステップ 5 <code>show vlan interface</code> <i>[[GigabitEthernet TenGigE Bundle-Ether] interface-path-id location instance]</i></p> <p>例: RP/0/RP0/CPU0:router# show vlan trunk summary</p>	<p>(任意) インターフェイスの設定を表示します。</p> <ul style="list-style-type: none"> • ポートの設定を表示するには、location キーワードを使用します。 • 指定したインターフェイスまたはサブインターフェイスの設定を表示するには、interface キーワードを使用します。
<p>ステップ 6 <code>show vlan trunks</code> <i>[brief] [location instance] [[GigabitEthernet TenGigE Bundle-Ether fastethernet] interface-path-id] [summary]</i></p> <p>例: RP/0/RP0/CPU0:router# show vlan trunk summary</p>	<p>(任意) 各 VLAN トランク インターフェイスに関するサマリー情報を表示します。</p> <ul style="list-style-type: none"> • キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> - brief : 簡潔なサマリーを表示します。 - summary : 完全なサマリーを表示します。 - location : 指定したポート上の VLAN トランク インターフェイスに関する情報を表示します。 - interface : 指定したインターフェイスまたはサブインターフェイスに関する情報を表示します。

VLAN インターフェイスの設定例

ここでは、次の例について説明します。

「VLAN サブインターフェイス : 例」(P.499)

VLAN サブインターフェイス : 例

次に、一度に 3 つの VLAN サブインターフェイスを作成するための包括的な例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 10
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.10.1/24
RP/0/RP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.2
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 20
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.20.1/24
RP/0/RP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.3
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 30
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.30.1/24
RP/0/RP0/CPU0:router(config-subif)# commit
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# exit

RP/0/RP0/CPU0:router# show vlan trunks summary
VLAN trunks: 1,
  1 are 802.1Q (Ether).
Sub-interfaces: 3,
  3 are up.
802.1Q VLANs: 3,
  3 have VLAN Ids.

RP/0/RP0/CPU0:router# show vlan interface
interface          encapsulation  vlan-id  intf-state
Te0/2/0/4.1       802.1Q        10       up
```

VLAN インターフェイスの設定例

```
Te0/2/0/4.2      802.1Q      20 up
Te0/2/0/4.3      802.1Q      30 up
```

```
RP/0/RP0/CPU0:router# show vlan trunks brief
interface      encapsulations      intf-state
Te0/2/0/4      802.1Q (Ether)      up
```

次に、1つのイーサネットバンドルに、一度に2つのVLANサブインターフェイスを作成するための包括的な例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1
RP/0/RP0/CPU0:router(config-if)# ip address 1.0.0.1/24
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1.1
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 10
RP/0/RP0/CPU0:router(config-subif)# ip address 10.0.0.1/24
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1.2
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 20
RP/0/RP0/CPU0:router(config-subif)# ip address 20.0.0.1/24
RP/0/RP0/CPU0:router(config-subif)# exit
```

次に、基本の Dot1Q AC を作成する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 20
RP/0/RP0/CPU0:router(config-subif)# l2protocol pvst tunnel
RP/0/RP0/CPU0:router(config-subif)# commit
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# exit
```

次に、基本の Q-in-Q AC を作成する例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.2
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 20 vlan 10
RP/0/RP0/CPU0:router(config-subif)# l2protocol cdp drop
RP/0/RP0/CPU0:router(config-subif)# commit
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# exit
```

次に、基本の Q-in-Any AC を作成する例を示します。

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface TenGigE 0/2/0/4.3
RP/0/0/CPU0:router(config-subif)# dot1q vlan 30 vlan any
RP/0/0/CPU0:router(config-subif)# l2protocol vtp experimental 7
RP/0/0/CPU0:router(config-subif)# commit
RP/0/0/CPU0:router(config-subif)# exit
RP/0/0/CPU0:router(config)# exit
```

その他の参考資料

ここでは、VLAN インターフェイスの設定に関連する参考資料を示します。

関連資料

内容	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報	『Cisco IOS XR Getting Started Guide』
ユーザ グループとタスク ID に関する情報	『Cisco IOS XR Interface and Hardware Component Command Reference』
リモートの Craft Works Interface (CWI) クライアント管理アプリケーションからの、Cisco CRS-1 ルータ上のインターフェイスとその他のコンポーネントの設定に関する情報	『Cisco Craft Works Interface Configuration Guide』

規格

規格	タイトル
この機能によりサポートされた新規規格または改訂規格はありません。またこの機能による既存規格のサポートに変更はありません。	-

MIB

MIB	MIB リンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	-

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、テクニカル ヒント、ツールへのリンクなど、さまざまな技術的コンテンツを検索可能な形で提供しています。Cisco.com に登録されている場合は、次のページからログインしてさらに多くのコンテンツにアクセスできます。	http://www.cisco.com/techsupport



INDEX

HC	Cisco IOS XR Interface and Hardware Component Configuration Guide
IC	Cisco IOS XR IP Addresses and Services Configuration Guide
MCC	Cisco IOS XR Multicast Configuration Guide
MNC	Cisco IOS XR System Monitoring Configuration Guide
MPC	Cisco IOS XR MPLS Configuration Guide
QC	Cisco IOS XR Modular Quality of Service Configuration Guide
RC	Cisco IOS XR Routing Configuration Guide
SBC	Cisco IOS XR Session Border Controller Configuration Guide
SC	Cisco IOS XR System Security Configuration Guide
SMC	Cisco IOS XR System Management Configuration Guide
VFC	Cisco IOS XR Virtual Firewall Configuration Guide

A

address-family ipv4 コマンド [HC-73, HC-74](#)

ais-shut コマンド [HC-435](#)

aps group コマンド [HC-104, HC-434](#)

aps サブモード

「aps group コマンド」を参照

channel コマンド [HC-104, HC-105, HC-434](#)

interface loopback コマンド [HC-434](#)

area コマンド (BFD) [HC-69, HC-71](#)

atm address-registration コマンド [HC-45](#)

atm ilmi-config disable コマンド [HC-47](#)

atm ilmi-keepalive コマンド [HC-45](#)

ATM インターフェイス

ATM の概要 [HC-3](#)

F4 OAM の概要 [HC-5](#)

ILMI

イネーブル化 [HC-44, HC-45](#)

概要 [HC-5](#)

ディセーブル化 [HC-46, HC-47](#)

OAM パケット [HC-20](#)

VC クラスの付加 [HC-40](#)

VC クラス マッピングの概要 [HC-4](#)

VP トンネル

概要 [HC-4](#)

表示 [HC-21, HC-24](#)

アドレス登録およびコールバック機能 [HC-45](#)

確認

ILMI の設定 [HC-46, HC-48](#)

接続 [HC-24](#)

作成

PVC [HC-14](#)

VP トンネル [HC-20](#)

インターフェイス [HC-14](#)

サブインターフェイス [HC-13, HC-14](#)

始動 [HC-8, HC-54](#)

設定

IP アドレスおよびサブネット マスク [HC-23](#)

OAM AIS/RDI [HC-38](#)

OAM F5 ループバック [HC-38](#)

OAM 管理 [HC-38](#)

PVC [HC-15](#)

VC クラス [HC-37, HC-57](#)

VP トンネル [HC-19, HC-22, HC-56](#)

オプション パラメータ [HC-11, HC-54](#)

トラフィック シェーピング [HC-20, HC-39](#)

前提条件 [HC-2](#)

付加、VC クラス [HC-41, HC-42, HC-44](#)

B

bert pattern コマンド [HC-468, HC-471](#)

BFD

BFD コンフィギュレーション モード [HC-75, HC-76](#)

BGP コンフィギュレーション モード [HC-67](#)

IPv6 [HC-65](#)

OSPFv3 コンフィギュレーション モード [HC-71](#)

- OSPF コンフィギュレーション モード [HC-69](#)
 VLAN バンドル [HC-65](#)
 VPN VRF インスタンスの指定 [HC-73](#)
 イネーブル化
 インターフェイス [HC-68, HC-70](#)
 迅速な検出 [HC-73, HC-74](#)
 スタティック ルート [HC-72](#)
 ネイバ [HC-66](#)
 ローカル装置とピアの間 [HC-68](#)
 エコー モードのディセーブル化 [HC-74, HC-75, HC-77](#)
 概要 [HC-63](#)
 カウンタ
 クリア [HC-77](#)
 表示 [HC-77](#)
 迅速な検出の設定 [HC-69, HC-71](#)
 スタティック ルートの設定 [HC-73](#)
 設定
 BFD 係数 [HC-67](#)
 係数 [HC-69, HC-71](#)
 最小間隔 [HC-67, HC-69, HC-71](#)
 前提条件 [HC-62](#)
 パケット形式 [HC-66](#)
 例 [HC-78](#)
 bfd fast-detect コマンド [HC-68, HC-69, HC-71](#)
 bfd minimum-interval コマンド [HC-67, HC-69, HC-71](#)
 bfd multiplier コマンド [HC-67, HC-69, HC-71](#)
 bfd コマンド [HC-75, HC-76](#)
 Bundle-Ether コマンド [HC-207](#)
 bundle id コマンド [HC-207, HC-211](#)
 bundle コマンド [HC-342](#)
-
- C**
- cablelength コマンド [HC-446, HC-452, HC-458](#)
 cache entries コマンド [HC-269](#)
 cache permanent コマンド [HC-269](#)
 cache timeout コマンド [HC-269](#)
 channel-group コマンド [HC-184, HC-189, HC-340, HC-462](#)
 channel コマンド [HC-104, HC-105, HC-434](#)
 CHAP
 ppp [HC-319, HC-367](#)
 イネーブル化 [HC-320](#)
 拒否 [HC-335](#)
 定義 [HC-290](#)
 パスワードの設定 [HC-325](#)
 class-int コマンド [HC-41](#)
 class-map コマンド [HC-411, HC-412](#)
 class-vc コマンド [HC-44](#)
 class コマンド [HC-412, HC-413](#)
 clear bfd counters packet コマンド [HC-78](#)
 clear flow exporter コマンド [HC-272](#)
 clear flow monitor コマンド [HC-272](#)
 clock source コマンド [HC-446](#)
 commit コマンド [HC-400](#)
 configure コマンド [HC-399](#)
 Control Ethernet Inactive Link Test [HC-139](#)
 Control Ethernet Ping Test [HC-139](#)
 controller e1 コマンド [HC-470](#)
 controller mgmtmultilink コマンド [HC-341](#)
 controller sonet コマンド [HC-406](#)
 crc コマンド [HC-289, HC-371](#)
-
- D**
- debug chdlc slarp packet コマンド [HC-289](#)
 delay trigger コマンド [HC-429](#)
 destination コマンド [HC-264](#)
 dot1q native vlan コマンド [HC-493](#)
 dot1q vlan コマンド [HC-490, HC-495](#)
 dscp コマンド [HC-264](#)
 duplex full コマンド [HC-158](#)
 duplex half コマンド [HC-158](#)
 duplex コマンド [HC-246](#)
 Dynamic Packet Transport (DPT)、Cisco [HC-395](#)

E

- E1 コントローラ
 - カードタイプの設定 [HC-448](#)
 - デフォルト設定値 [HC-445, HC-446](#)
 - フレームタイプ [HC-446, HC-465](#)
- E1 モードから T1 モードへの変更 [HC-448](#)
- E3 コントローラ
 - E3 コンフィギュレーションモード [HC-450](#)
 - カードタイプの設定 [HC-448](#)
 - クロックソース、デフォルト値 [HC-446](#)
 - ケーブル長の設定 [HC-446](#)
 - 設定
 - 各国用予約ビット [HC-446](#)
 - クリアチャンネル [HC-450](#)
 - クリアチャンネルシリアル [HC-450](#)
 - 設定の前提条件 [HC-444](#)
 - フレームタイプ [HC-446, HC-453](#)
- E3 モードから T3 モードへの変更 [HC-448](#)
- echo disable コマンド [HC-75, HC-77](#)
- encapsulation frame relay コマンド [HC-171](#)
- encapsulation コマンド [HC-38, HC-288, HC-371](#)
- encap コマンド (フレームリレー) [HC-170, HC-171, HC-179](#)
- end コマンド [HC-400](#)
- exit コマンド [HC-412, HC-413](#)
- exporter コマンド [HC-269](#)

F

- f4oam disable コマンド [HC-20](#)
- Fabric Diagnostic Test [HC-140](#)
- Fabric Multicast Diagnosis Test [HC-140](#)
- Fabric Ping Test [HC-139](#)
- fdl ansi [HC-447](#)
- fdl コマンド [HC-447, HC-462](#)
- File System Functionality Verification Test [HC-140](#)
- flow-control コマンド [HC-146, HC-155](#)
- flow exporter-map コマンド [HC-263](#)

- flow monitor-map コマンド [HC-263, HC-268, HC-278](#)
- flow コマンド [HC-271](#)
- frame-relay intf-type dce コマンド [HC-172](#)
- frame-relay intf-type dte コマンド [HC-172](#)
- frame-relay intf-type コマンド [HC-171, HC-179](#)
- frame-relay lmi disable コマンド [HC-181](#)
- frame-relay lmi-t391dte コマンド [HC-173](#)
- frame-relay lmi-type コマンド [HC-171, HC-179](#)
- frame-relay lmi コマンド [HC-171](#)
- framing コマンド [HC-446](#)

H

- HDLC [HC-289](#)
- hw-module node reload コマンド [HC-400](#)
- hw-module port コマンド [HC-399](#)
- hw-module subslot cardtype コマンド [HC-448, HC-449](#)

I

- IEEE 802.3ad 規格 [HC-205](#)
- if プリコンフィギュレーションサブモード、ipv4 address コマンド [HC-357](#)
- ILMI、ATM インターフェイスでの [HC-44](#)
- interface Bundle-Ether コマンド [HC-209, HC-215](#)
- interface Bundle-POS コマンド [HC-207](#)
- interface loopback コマンド [HC-434](#)
- interface POS コマンド [HC-223](#)
- interface preconfigure コマンド [HC-357](#)
- interface コマンド [HC-14](#)
- invert コマンド [HC-377](#)
- IPS (Intelligent Protection Switching) [HC-407](#)
- ipv4 address コマンド [HC-14, HC-207](#)
- ipv4 unreachable コマンド [HC-234](#)

K

- keepalive コマンド [HC-288, HC-305, HC-343, HC-384, HC-435](#)

L

l2protocol コマンド [HC-162](#), [HC-496](#)
 l2transport コマンド [HC-162](#)
 L2VPN
 「レイヤ 2 VPN」を参照 [HC-147](#)
 LACP (Link Aggregation Control Protocol) [HC-205](#)
 Layer 2 VPN
 概要 [HC-147](#)
 設定
 接続回路 [HC-161](#)
 レイヤ 2 トランスポート モード [HC-162](#)
 レイヤ 2 プロトコルのトンネリングおよびデータ
 ユニット [HC-162](#)
 LCP (Link Control Protocol) [HC-289](#), [HC-366](#)
 LMI [HC-172](#), [HC-180](#), [HC-181](#)
 lmi-n391dte コマンド [HC-173](#)
 lmi-n392dce コマンド [HC-173](#)
 lmi-n392dte コマンド [HC-173](#)
 lmi-n393dce コマンド [HC-173](#)
 lmi-n393dte コマンド [HC-173](#)
 lmi-t392dce コマンド [HC-173](#)
 Local Management Interface (LMI) [HC-370](#)
 loopback コマンド [HC-430](#)

M

mac accounting コマンド [HC-161](#)
 mac-accounting コマンド [HC-146](#), [HC-147](#)
 mac address コマンド [HC-146](#), [HC-155](#)
 match mpls experimental topmost コマンド [HC-412](#)
 match precedence コマンド [HC-412](#)
 mdl string コマンド [HC-459](#)
 mdl transmit コマンド [HC-446](#), [HC-459](#)
 mode コマンド [HC-450](#)
 MQC (Modular Quality of Service) の設定、SRP による [HC-410](#)
 MS-CHAP 認証
 ppp [HC-290](#), [HC-320](#), [HC-328](#), [HC-336](#)

イネーブル化 [HC-320](#), [HC-321](#)

ディセーブル化 [HC-336](#)

パスワードの設定 [HC-327](#)

表示 [HC-322](#)

mtu コマンド [HC-146](#), [HC-147](#), [HC-289](#), [HC-371](#)

multilink fragment-size コマンド [HC-185](#), [HC-343](#)

multilink interleave コマンド [HC-345](#)

multilink コマンド [HC-345](#)

Multiprotocol Label Switching Control Processor
(MPLSCP) [HC-289](#), [HC-367](#)

N

national bits コマンド [HC-446](#), [HC-447](#), [HC-453](#),
[HC-465](#)

negotiation auto コマンド [HC-147](#), [HC-156](#), [HC-158](#)

neighbor コマンド (BFD) [HC-67](#)

NetFlow

IPv4 の宛先ベース アカウンティング [HC-261](#),
[HC-277](#), [HC-281](#)

エクスポート マップ [HC-257](#)

概要 [HC-253](#), [HC-255](#)

サンブラ マップ [HC-256](#)

サンブラ マップ コンフィギュレーション サブモード
[HC-259](#)

制約事項 [HC-255](#)

設定 [HC-262](#)

前提条件 [HC-254](#)

フロー エクスポート マップ コンフィギュレーション
サブモード [HC-258](#)

フロー エクスポート マップ バージョン コンフィギュ
レーション サブモード [HC-258](#), [HC-259](#)

モニタ マップ [HC-256](#)

Network Control Protocol (NCP) [HC-289](#), [HC-367](#)

no interface コマンド [HC-498](#)

Nonstop Forwarding [HC-207](#)

no shutdown コマンド
(警告) [HC-356](#)

null0 サブモード、ipv4 unreachable コマンド [HC-234](#)

O

oam ais-rdi コマンド [HC-38](#)
 oam-pvc manage コマンド [HC-38](#)
 oam retry コマンド [HC-38](#)
 options コマンド [HC-264](#)
 overhead コマンド [HC-430](#)

P

PAP 認証

ppp [HC-290, HC-323](#)
 イネーブル化 [HC-320, HC-321, HC-323, HC-324](#)
 拒否 [HC-333](#)
 定義 [HC-319](#)
 ディセーブル化 [HC-332](#)
 表示 [HC-322](#)
 path scrambling コマンド [HC-435](#)
 path コマンド [HC-431, HC-438](#)
 ping コマンド [HC-24](#)
 police コマンド [HC-412](#)
 policy-map コマンド [HC-412](#)
 pos crc コマンド [HC-99](#)
 POS (Packet-over-SONET)
 「POS インターフェイス」を参照
 POS インターフェイス
 HDLC カプセル化
 概要 [HC-289](#)
 説明 [HC-285](#)
 PPP カプセル化
 概要 [HC-289](#)
 説明 [HC-285](#)
 PVCを持つサブインターフェイスの作成 [HC-299](#)
 インターフェイス コンフィギュレーション モード
 interface multilink コマンド [HC-343, HC-345](#)
 interface pos コマンド [HC-98, HC-295, HC-297](#)
 interface コマンド [HC-344](#)
 始動 [HC-294](#)

設定

CRC 値 [HC-99, HC-289, HC-297](#)
 MTU [HC-99, HC-289, HC-297, HC-312](#)
 PPP 認証 [HC-290](#)
 インターフェイス カプセル化 [HC-98, HC-297, HC-344](#)
 オプション パラメータ [HC-296](#)
 カプセル化タイプ [HC-288](#)
 キープアライブ タイマー [HC-288, HC-304, HC-305, HC-343, HC-435](#)

デフォルト設定

CRC [HC-289](#)
 mtu [HC-289](#)
 カプセル化 [HC-288](#)
 キープアライブ [HC-288](#)
 フレーム リレー カプセル化 [HC-285](#)

PPP

CHAP

イネーブル化 [HC-320, HC-321](#)
 拒否 [HC-335](#)
 認証 [HC-319](#)
 パスワードの設定 [HC-325, HC-326](#)
 表示 [HC-322](#)

MS-CHAP

ppp [HC-320, HC-367](#)
 イネーブル化 [HC-321](#)
 認証 [HC-320](#)
 ディセーブル化 [HC-336](#)
 パスワードの設定 [HC-327, HC-328](#)
 表示 [HC-322](#)

PAP

イネーブル化 [HC-320, HC-321, HC-323, HC-324](#)
 拒否 [HC-333](#)
 ディセーブル化 [HC-332](#)
 認証 [HC-323, HC-367](#)
 表示 [HC-322](#)

POS インターフェイス [HC-285, HC-289](#)

POS の設定例 [HC-346](#)

インターフェイス、表示 [HC-332](#)

概要 [HC-318](#)

シリアル インターフェイス [HC-366](#)

シリアル、設定例 [HC-347](#)

前提条件 [HC-318](#)

デフォルト設定の変更 [HC-329, HC-330](#)

ppp authentication コマンド [HC-290, HC-319, HC-321, HC-367](#)

ppp chap password コマンド [HC-326](#)

ppp chap refuse コマンド [HC-335](#)

ppp max-bad-auth コマンド [HC-330](#)

ppp max-configure コマンド [HC-330](#)

ppp max-failure コマンド [HC-330](#)

ppp max-terminate コマンド [HC-331](#)

ppp ms-chap password コマンド [HC-328](#)

ppp ms-chap refuse コマンド [HC-336](#)

ppp multilink minimum-active links コマンド [HC-345](#)

ppp pap refuse コマンド [HC-333](#)

ppp pap sent-username コマンド [HC-323, HC-324](#)

ppp timeout authentication コマンド [HC-331](#)

ppp timeout retry コマンド [HC-331](#)

priority コマンド [HC-413](#)

profile コマンド [HC-481](#)

PVC

ATM インターフェイス [HC-15](#)

ATM サブインターフェイス [HC-13, HC-43](#)

POS サブインターフェイス [HC-299, HC-301](#)

VP トンネル [HC-22](#)

pvc コマンド [HC-14](#)

R

random コマンド [HC-266](#)

record ipv4 コマンド [HC-269, HC-278](#)

remote-as コマンド (BFD) [HC-67](#)

RommonRevision Test [HC-140](#)

router bgp コマンド [HC-67](#)

router ospfv3 コマンド (BFD) [HC-71](#)

router ospf コマンド (BFD) [HC-69](#)

router static コマンド (BFD) [HC-73](#)

RP、プリコンフィギュレーション ディレクトリ [HC-353](#)

S

sampler-map コマンド [HC-266](#)

scramble コマンド [HC-377](#)

Scratch Register Test [HC-140](#)

Self-Ping over Fabric Test [HC-139](#)

service-policy コマンド [HC-413](#)

set cos コマンド [HC-412, HC-413](#)

SF (Signal Fail) [HC-408](#)

shape コマンド [HC-20](#)

show aps コマンド [HC-106, HC-436](#)

show atm ilmi-status コマンド [HC-46, HC-48](#)

show atm vp-tunnel interface atm コマンド [HC-21](#)

show atm vp-tunnel コマンド [HC-24](#)

show bfd counters コマンド [HC-78](#)

show bundle Bundle-Ether コマンド [HC-213, HC-219](#)

show bundle Bundle-POS コマンド [HC-224](#)

show controller sonet コマンド [HC-432](#)

show controllers コマンド [HC-451](#)

show flow exporter-map コマンド [HC-265, HC-279](#)

show flow monitor map コマンド [HC-270](#)

show frame-relay lmi コマンド [HC-172](#)

show interfaces コマンド [HC-154, HC-390](#)

show ip route コマンド [HC-482](#)

show lacp bundle Bundle-Ether コマンド [HC-213](#)

show lacp bundle Bundle-POS コマンド [HC-224](#)

show mac accounting コマンド [HC-161](#)

show ppp interfaces コマンド [HC-322, HC-332](#)

show running-config コマンド [HC-407](#)

show sampler-map コマンド [HC-267](#)

show srp ips interface srp コマンド [HC-410](#)

show version コマンド [HC-154](#)

show vlan interface コマンド [HC-499](#)

show vlan trunks コマンド [HC-220](#)

show vlan コマンド [HC-492](#)

SLARP (Serial Line Address Resolution Protocol) [HC-289](#), [HC-291](#), [HC-366](#), [HC-369](#)

SONET APS (SONET Automatic Protection Switching) [HC-432](#)

SONET (Synchronous Optical Network)

APS [HC-432](#)

Fast Reroute (FFR) [HC-437](#)

説明 [HC-423](#)

SONET コントローラ

設定 [HC-428](#)

フレーム タイプ [HC-429](#)

sonet サブモード

ais-shut コマンド [HC-435](#)

clock source コマンド [HC-406](#), [HC-429](#), [HC-436](#)

「controller sonet コマンド」を参照

delay trigger コマンド [HC-429](#), [HC-438](#)

framing コマンド [HC-429](#)

loopback コマンド [HC-430](#)

overhead コマンド [HC-430](#)

path scrambling コマンド [HC-435](#)

path コマンド [HC-431](#), [HC-438](#)

source コマンド [HC-264](#)

Spatial Reuse Protocol (SRP) [HC-395](#)

MQC の設定 [HC-410](#)

PLIM でのイネーブル化 [HC-398](#)

SPA でのイネーブル化 [HC-400](#), [HC-403](#)

基本設定 [HC-405](#)

定義 [HC-395](#)

リングへのノードの追加 [HC-414](#)

speed コマンド [HC-147](#), [HC-158](#)

srp interface srp request manual-switch コマンド [HC-410](#)

srp ips request forced-switch コマンド [HC-409](#), [HC-415](#), [HC-416](#)

srp ips timer コマンド [HC-409](#)

srp ips wtr-timer コマンド [HC-409](#)

srp topology-timer コマンド [HC-406](#)

SSH (Secure Shell) [HC-479](#)

SSL (Secure Socket Layer) [HC-479](#)

T

T1 コントローラ

ANSI T1.403 または AT&T TR54016 パフォーマンスレポート [HC-447](#), [HC-462](#)

BERT の設定 [HC-470](#)

DS0 タイムスロットの関連付け [HC-184](#), [HC-190](#), [HC-341](#), [HC-462](#)

T1 コンフィギュレーション モード [HC-184](#), [HC-189](#), [HC-340](#), [HC-461](#)

T1 チャネル グループの作成 [HC-184](#), [HC-189](#), [HC-340](#), [HC-462](#)

イエロー アラーム [HC-446](#)

カード タイプの設定 [HC-448](#)

クロック ソース [HC-446](#), [HC-461](#)

デフォルト設定値 [HC-445](#), [HC-446](#)

フレーム タイプ [HC-446](#), [HC-461](#)

T1 モードから E1 モードへの変更 [HC-448](#)

T3 コントローラ

カード タイプの設定 [HC-448](#)

クロック ソース

設定 [HC-340](#), [HC-452](#), [HC-458](#)

デフォルト値 [HC-446](#)

ケーブル長の設定 [HC-446](#)

設定 [HC-454](#)

BERT [HC-468](#)

FRF.12 エンドツーエンド フラグメンテーション [HC-189](#)

MDL メッセージ [HC-446](#)

クリア チャネル E3 コントローラ [HC-450](#)

クリア チャネル T3 コントローラ [HC-454](#)

チャネライズド T3 コントローラ [HC-456](#)

マルチリンク フレームリレー バンドル インターフェイス [HC-183](#)

設定の前提条件 [HC-444](#)

フレーム タイプ [HC-446](#), [HC-459](#)

変更

デフォルト E3 コントローラ [HC-452](#)

デフォルト T3 コントローラ [HC-458](#)

T3 モードから E3 モードへの変更 [HC-448](#)

template コマンド [HC-264](#)

timeslots コマンド [HC-184](#), [HC-190](#), [HC-341](#), [HC-462](#)
 transmit-delay コマンド [HC-377](#)
 transport udp コマンド [HC-264](#)
 tunnel destination コマンド [HC-482](#)
 tunnel source コマンド [HC-482](#)

ATM インターフェイス [HC-56](#)
 作成 [HC-19](#)
 設定 [HC-19](#), [HC-51](#)
 例 [HC-58](#)
 vrf コマンド (BFD) [HC-73](#)

V

vc-class コマンド [HC-38](#)

VC クラス

ATM PVC [HC-42](#)
 ATM インターフェイス [HC-37](#), [HC-38](#), [HC-40](#)
 ATM サブインターフェイス [HC-41](#)
 作成 [HC-57](#)
 設定 [HC-57](#)

version v9 コマンド [HC-264](#)

VLAN

802.1Q フレーム タギング [HC-487](#)
 MTU の継承 [HC-487](#)
 VLAN AC の割り当て [HC-490](#)
 VLAN サブインターフェイスの削除 [HC-497](#)
 概要 [HC-487](#)
 サブインターフェイスの概要 [HC-487](#)
 設定
 IP アドレスおよびサブネットマスク [HC-491](#)
 サブインターフェイス [HC-489](#)
 ネイティブ VLAN [HC-492](#), [HC-493](#)
 バンドル [HC-488](#)
 ネイティブ VLAN の説明 [HC-488](#)
 表示
 VLAN インターフェイス [HC-492](#), [HC-499](#)
 VLAN トランク [HC-492](#), [HC-499](#)
 レイヤ 2 VPN
 接続回路の設定 [HC-494](#), [HC-495](#)
 レイヤ 2 プロトコルのトンネリングおよびデータ
 ユニットの設定 [HC-496](#)
 レイヤ 2 VPN サポート [HC-488](#)

vp-tunnel コマンド [HC-20](#)

VP トンネル

W

wait-to-restore (WTR) [HC-407](#)

Y

yellow コマンド [HC-446](#), [HC-461](#)

い

イーサネット インターフェイス

VLAN

802.1Q フレーム タギング [HC-487](#)
 MTU の継承 [HC-487](#)
 VLAN AC の割り当て [HC-490](#)
 VLAN インターフェイスの表示 [HC-492](#)
 VLAN トランクの表示 [HC-492](#)
 概要 [HC-487](#)
 サブインターフェイスの概要 [HC-487](#)
 サブインターフェイスの削除 [HC-497](#)
 サブインターフェイスの設定 [HC-489](#)
 接続回路の設定 [HC-495](#)
 ネイティブ VLAN の設定 [HC-492](#), [HC-493](#)
 ネイティブ VLAN の説明 [HC-488](#)
 レイヤ 2 VPN 接続回路の設定 [HC-494](#)
 レイヤ 2 VPN、設定 [HC-496](#)

イーネーブル化

フロー制御 [HC-155](#)
 レイヤ 2 トランスポート モード [HC-162](#)

ギガビットイーサネット規格 [HC-148](#)

IEEE 802.3ab 1000BASE-T ギガビットイーサ
 ネット [HC-148](#)

IEEE 802.3ae 10 Gbps イーサネット [HC-149](#)

- IEEE 802.3z 1000 Mbps ギガビット イーサネット
ト **HC-149**
- IEEE 802.3 物理イーサネット インフラストラク
チャ **HC-148**
- 設定
- IP アドレスおよびサブネットマスク **HC-155**
 - MAC アカウンティング **HC-146, HC-161**
 - MAC アドレス **HC-146, HC-155**
 - MTU **HC-146, HC-155**
 - 接続回路 **HC-161**
 - フロー制御 **HC-146**
- デフォルト設定
- MAC アカウンティング **HC-146**
 - MAC アドレス **HC-146**
 - mtu **HC-146**
 - フロー制御 **HC-146**
- 表示
- MAC アカウンティングの統計情報 **HC-161**
 - イーサネット インターフェイス **HC-156**
- レイヤ 2 VPN
- VLAN サポート **HC-488**
 - 概要 **HC-147**
 - レイヤ 2 VPN ポート **HC-161**
 - レイヤ 2 プロトコルのトンネリングおよびデータ
ユニットの設定 **HC-162**
- インターフェイス
- トンネル **HC-478**
 - リンク バンドル **HC-199**
 - QoS **HC-206**
 - 設定 **HC-208**
 - 前提条件 **HC-200, HC-201**
 - リンク スイッチオーバー **HC-207**
- インターフェイス サブモード
- bundle id コマンド **HC-211, HC-212, HC-218, HC-223**
 - controller sonet コマンド **HC-435**
 - duplex コマンド **HC-246**
 - 「interface preconfigure コマンド」を参照
 - interface コマンド **HC-435**
 - 「interface コマンド」を参照
 - ipv4 address コマンド **HC-186**
 - mtu コマンド **HC-245**
 - no shutdown コマンド **HC-211**
 - pos crc コマンド **HC-435**
 - show srp ips interface srp コマンド **HC-410**
 - speed コマンド **HC-248**
 - srp interface srp request manual-switch コマン
ド **HC-410**
 - srp ips request forced-switch コマンド **HC-409, HC-415, HC-416**
 - srp ips timer コマンド **HC-409**
 - srp ips wtr-timer コマンド **HC-409**
 - srp topology-timer コマンド **HC-406**
 - tunnel destination コマンド **HC-482**
 - tunnel source コマンド **HC-482**
- インターフェイス サブモード、service-policy コマン
ド **HC-413**
-
- え
- エクスポータ マップ **HC-257**
-
- か
- カード タイプ
- E1 モードへの変更 **HC-448**
 - E3 モードへの変更 **HC-448**
 - T1 モードへの変更 **HC-448**
 - T3 モードへの変更 **HC-448**
- 回線エミュレーション **HC-7**
- 仮想インターフェイス
- アクティブ / スタンバイ RP **HC-231, HC-356**
 - スイッチオーバー **HC-229**
 - ヌル インターフェイスの定義 **HC-231**
 - 命名規則 **HC-229, HC-479**
- 管理イーサネット インターフェイスの設定 **HC-242**

き

キープアライブ タイマー

説明 [HC-290](#)

モニタ

POS リンク ステート [HC-290](#)

SLARP パケット [HC-289](#)

く

クラス マップ サブモード

match mpls experimental topmost コマンド [HC-412](#)

match precedence コマンド [HC-412](#)

クリア チャネル [HC-58](#)

クリア チャネル ATM [HC-51, HC-58](#)

さ

サンブラ マップ [HC-256](#)

サンブラ マップ コンフィギュレーション サブモード [HC-259](#)

し

シリアル インターフェイス

PPP カプセル化 [HC-366](#)

設定

CRC [HC-377](#)

IP アドレスおよびサブネット マスク [HC-374](#)

インターフェイス カプセル化 [HC-376](#)

キープアライブ タイマー [HC-384](#)

送信遅延 [HC-377](#)

前提条件 [HC-362](#)

データ ストリームの反転 [HC-377](#)

デフォルト設定

CRC [HC-371](#)

mtu [HC-371](#)

カプセル化 [HC-371](#)

キープアライブ [HC-371](#)

ペイロード スクランブリング、イネーブル化 [HC-377](#)

リンク ステート [HC-366, HC-367, HC-368](#)

診断 [HC-137](#)

す

スイッチオーバー [HC-207](#)

せ

制御パケット [HC-396](#)

ち

チャネライズド ATM [HC-48, HC-58](#)

チャネライズド SONET の設定 [HC-92, HC-96](#)

て

テスト

Control Ethernet Inactive Link [HC-139](#)

Control Ethernet Ping [HC-139](#)

Fabric Diagnostic [HC-140](#)

Fabric Multicast Diagnosis [HC-140](#)

Fabric Ping [HC-139](#)

File System Functionality Verification Test [HC-140](#)

RommonRevision [HC-140](#)

Scratch Register [HC-140](#)

Self-Ping over Fabric [HC-139](#)

デフォルト設定

MAC アドレス (管理イーサネット) [HC-243](#)

速度 (管理イーサネット) [HC-243](#)

フロー制御、管理イーサネット [HC-243](#)

と

トラフィック フィルタリング [HC-231](#)

トランスペアレント スイッチオーバー [HC-242](#)

ぬ

ヌル インターフェイス

- 設定 [HC-231](#)
- 表示 [HC-231](#)
- 命名規則 [HC-229](#)

は

波長分割多重 (WDM) [HC-396](#)

ふ

ファスト イーサネット インターフェイス

- shutdown 設定の削除 [HC-158](#)
- オートネゴシエーション [HC-147, HC-158](#)
- 設定
 - IP アドレスおよびサブネットマスク [HC-158](#)
 - MAC アカウンティング [HC-147](#)
 - MTU [HC-147, HC-158](#)
 - インターフェイス速度 [HC-158](#)
 - デュプレックス操作 [HC-147, HC-158](#)

デフォルト設定

- auto-negotiation [HC-147](#)
- MAC アカウンティング [HC-147](#)
- mtu [HC-147](#)
- インターフェイス速度 [HC-147](#)
- デュプレックス操作 [HC-147](#)

プリコンフィギュレーション

- ディレクトリ [HC-353](#)
- 物理インターフェイスに対する制約事項 [HC-353](#)
- 命名規則 [HC-356](#)
- 利点 [HC-356](#)

フレームリレー

LMI

- イネーブル化 [HC-171](#)
- 概要 [HC-172](#)
- 設定 [HC-171, HC-179](#)
- ディセーブル化 [HC-171, HC-180, HC-181](#)

ポーリング [HC-173](#)POS インターフェイス [HC-285](#)PVC [HC-171](#)概要 [HC-170](#)シリアル インターフェイス [HC-369](#)

設定

- PVC カプセル化 [HC-179](#)
- サポートのタイプ [HC-171, HC-179](#)

設定例 [HC-193](#)前提条件 [HC-170](#)デフォルト設定 [HC-171](#)デフォルト設定の変更 [HC-178](#)フロー エクスポート マップ コンフィギュレーション サブモード [HC-258](#)フロー エクスポート マップ バージョン コンフィギュレーション サブモード [HC-258, HC-259](#)

ほ

ポリシー マップ サブモード

- class コマンド [HC-412, HC-413](#)
- police コマンド [HC-412](#)
- policy-map コマンド [HC-412](#)
- priority コマンド [HC-413](#)
- set cos コマンド [HC-412, HC-413](#)

ま

マルチリンク フレームリレー バンドル インターフェイス [HC-186](#)

め

命名規則

- 事前設定 [HC-356](#)
- ヌル インターフェイス [HC-229](#)
- ループバック [HC-229](#)

も

モニタ マップ [HC-256](#)

り

リンク スイッチオーバー [HC-207](#)

リング (外側および内側) [HC-396](#)

リンク バンドル、VLAN バンドルの設定 [HC-488](#)

る

ループバックの命名規則 [HC-229](#)