



## Cisco NCS 560 シリーズルータ（IOS XR リリース 7.0.x）システム管理コンフィギュレーションガイド

初版：2019年8月30日

### シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター  
0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

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.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned 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. (1721R)

© 2019 Cisco Systems, Inc. All rights reserved.



## 目次

---

### 第 1 章

#### フレキシブルコマンドライン インターフェイスの設定 1

フレキシブル CLI 設定グループ 1

フレキシブル設定の制約事項 1

設定グループの構成 3

    シンプルな設定グループ：例 4

    異なる場所に適用された設定グループ：例 5

設定グループの構成の確認 6

設定グループの正規表現 7

    正規表現を使用した設定例 14

        正規表現を使用した設定グループ：例 14

        正規表現を使用した設定グループの継承：例 16

        レイヤ 2 トランスポート設定グループ：例 18

        設定グループの優先順位：例 18

        設定グループへの変更は自動的に継承される：例 19

---

### 第 2 章

#### 管理性の設定 21

XML の管理機能について 21

管理機能の設定方法 22

    XML エージェントの設定 22

管理機能の設定例 23

    XML エージェントでの VRF のイネーブル化：例 23

---

### 第 3 章

#### オブジェクト トラッキングの設定 25

    オブジェクト トラッキングの設定 25

オブジェクトトラッキングの実装の前提条件	25
オブジェクトトラッキングについて	26
オブジェクトトラッキングの実装方法	26
インターフェイスのラインプロトコルステートのトラッキング	26
IP ルートの到達可能性のトラッキング	28
オブジェクトリストに基づくトラッキングの設定	30
オブジェクトリストに基づくトラッキングの設定：しきい値の割合	32
オブジェクトリストに基づくトラッキングの設定：しきい値の重み	34
IPSLA の到達可能性のトラッキング	36
オブジェクトトラッキングの設定例	37

---

**第 4 章**

<b>物理端末および仮想端末の設定</b>	<b>41</b>
物理端末と仮想端末を実装するための前提条件	41
物理端末および仮想端末の実装について	41
ラインテンプレート	41
ラインテンプレート コンフィギュレーションモード	42
ラインテンプレート ガイドライン	42
端末の識別	43
VTY プール	43
Cisco IOS XR ソフトウェアでの物理および仮想端末の実装方法	44
テンプレートの変更	44
VTY プールの作成および変更	45
端末および端末セッションのモニタリング	47
物理および仮想端末の実装の設定例	49

---

**第 5 章**

<b>簡易ネットワーク管理プロトコルの設定</b>	<b>53</b>
SNMP の実装の前提条件	53
Cisco IOS XR ソフトウェアでの SNMP の使用に関する制約事項	53
SNMP の実装について	54
SNMP 機能の概要	54
SNMP マネージャ	54

SNMP エージェント	54
MIB	54
SNMP バージョン	55
SNMPv1、SNMPv2c、および SNMPv3 の比較	56
SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル	57
SNMPv3 の利点	58
SNMPv3 のコスト	59
ユーザベースのセキュリティ モデル	59
View-Based Access Control Model	60
SNMP の IP precedence および DSCP サポート	60
サブスクリバセッションでのセッション MIB のサポート	61
SNMP 通知	61
セッションタイプ	62
Cisco IOS XR ソフトウェアでの SNMP の実装方法	62
SNMPv3 の設定	62
SNMPv3 の設定：例	64
SNMP トラップ通知の設定	69
トラップ通知の設定：例	70
SNMP エージェントの連絡先、場所、およびシリアル番号の設定	71
SNMP エージェント パケットの最大サイズの定義	72
通知操作値の変更	72
IP precedence および DSCP 値の設定	73
SNMP トラフィックの IP precedence 値の設定：例	74
SNMP トラフィックの IP DSCP 値の設定：例	74
SNMP コンテキスト マッピングの表示	74
パケット損失のモニタリング	75
維持する MIB データの設定	76
インターフェイスのサブセットに対する linkUp および linkDown トラップの設定	77
第 6 章	
定期的な MIB データの収集および転送の設定	81
定期的な MIB データの収集および転送の前提条件	81

定期的な MIB データの収集および転送に関する情報	81
SNMP のオブジェクトとインスタンス	81
バルク統計情報オブジェクトリスト	82
バルク統計情報スキーマ	82
バルク統計情報転送オプション	82
定期的な MIB データの収集および転送の利点	83
定期的な MIB データの収集および転送の設定方法	83
バルク統計情報オブジェクトリストの設定	83
バルク統計情報スキーマの設定	84
バルク統計情報転送オプションの設定	86
定期的な MIB データの収集および転送：例	90

---

**第 7 章****Cisco Discovery Protocol の設定 91**

CDP の実装の前提条件	91
CDP の実装について	91
CDP の実装方法：Cisco IOS XR ソフトウェア	93
CDP のイネーブル化	93
CDP デフォルト設定の変更	94
CDP のモニタリング	95
CDP の実装の設定例	96

---

**第 8 章****スマート ライセンスを使用したライセンスの設定 99**

スマート ライセンスとは	99
スマート ライセンスの仕組み	100
スマート ライセンスの導入オプション	101
Call Home について	103
サポート対象の柔軟な消費モデル ライセンス	103
スマート ライセンスを使用したライセンスの設定	104
デバイスの登録とアクティブ化	104
スマート ライセンス設定の確認	105
スマート ライセンス登録の更新	107

スマート ライセンスの登録解除	108
スマート ライセンスのワークフロー	109
ライセンス、製品インスタンス、および登録トークン	109
仮想アカウント	110
コンプライアンス レポート	110





# 第 1 章

## フレキシブルコマンドラインインターフェイスの設定

このモジュールでは、フレキシブル コマンドライン インターフェイス (CLI) の設定グループを構成および使用する方法について説明します。

- [フレキシブル CLI 設定グループ \(1 ページ\)](#)
- [フレキシブル設定の制約事項 \(1 ページ\)](#)
- [設定グループの構成 \(3 ページ\)](#)
- [設定グループの構成の確認 \(6 ページ\)](#)
- [設定グループの正規表現 \(7 ページ\)](#)

### フレキシブル CLI 設定グループ

フレキシブル コマンドライン インターフェイス (CLI) の設定グループは、設定グループ内に一連の構成ステートメントを定義し、このグループをルータ設定ツリーの複数の階層レベルに適用することにより、反復構成を最小限に抑える機能を提供します。

フレキシブル CLI 設定グループは、階層内でグループが適用されている場所に基づいて、設定ツリーの複数のサブモードで一致が確認される正規表現を使用します。設定サブモードで一致が見つかった場合、そのグループに定義されている対応する構成が、一致するサブモード内で継承されます。

フレキシブル CLI 設定グループには、自動継承機能もあります。自動継承とは、CLI 設定グループに対して行われた変更が、その階層レベルで適用グループを持つ一致するサブモードの構成に自動的に適用されることを意味します。これにより、フレキシブル CLI 設定グループを適用した場所に応じて、構成の変更または追加を一度行い、複数の場所に自動的に適用することができます。

### フレキシブル設定の制約事項

フレキシブル設定グループを使用する場合は、次の制約事項に注意してください。

- フレキシブル CLI 設定グループは管理構成ではサポートされず、対応する適用グループは管理構成ではサポートされません。
- 設定グループ内の事前設定されたインターフェイスの使用はサポートされていません。
- 設定グループをサポートしているイメージからサポートしていないイメージにダウンロードすることはサポートされていません。
- アクセスリスト、QoS およびルート ポリシー設定は、設定グループの使用をサポートしていません。次のような構成は有効ではありません。

```
group g-not-supported
  ipv4 access-list ...
  !
  ipv6 access-list ...
  !
  ethernet-service access-list ...
  !
  class-map ...
  !
  policy-map ...
  !
  route-policy ...
  !
end-group
```

ただし、次の例に示すような構成を参照できます。

```
group g-reference-ok
  router bgp 6500
  neighbor 7::7
    remote-as 65000
    bfd fast-detect
    update-source Loopback300
    graceful-restart disable
    address-family ipv6 unicast
      route-policy test1 in
      route-policy test2 out
      soft-reconfiguration inbound always
    !
  !
  interface Bundle-Ether1005
    bandwidth 10000000
    mtu 9188
    service-policy output input_1
    load-interval 30
  !
end-group
```

- 一部の正規表現はグループ内ではサポートされていません。たとえば、‘?’、‘|’ および ‘\$’ はグループ内ではサポートされていません。また、/d や /w などの文字はサポートされていません。
  - 正規表現内で複数の一致式を表す選択演算子「|」はサポートされていません。たとえば、次の式はサポートされていません。

Gig.\*|Gig.\*\.\* : ギガビットイーサネット インターフェイスまたはギガビットイーサネット サブインターフェイスでの照合時。

Gig.\*0/0/0/[1-5]|Gig.\*0/0/0/[10-20] : Gig.\*0/0/0/[1-5] または Gig.\*0/0/0/[10-20] での照合時。

'TenGigE.\*|HundredGigE.\* : TenGigE.\* または HundredGigE.\* での照合時。

- **location** キーワードのノード識別子を必要とするコマンドはサポートされていません。たとえば、次の構成はサポートされません。

```
lpts pifib hardware police location 0/RP0/CPU0
```

- 同じ構成に対する設定グループ内の重複する正規表現はサポートされていません。次に例を示します。

```
group G-INTERFACE
interface 'gig.*a.*'
    mtu 1500
!
interface 'gig.*e.* '
    mtu 2000
!
end-group

interface gigabitethernet0/0/0/* ---- where * is 0 to 31
    apply-group G-INTERFACE
```

interface GigabitEthernet0/0/0/\* 設定が mtu 1500 または mtu 2000 を継承するかどうかを判別できないため、この設定は許可されません。設定グループの両方の式は、GigabitEthernet0/0/0/\* と一致します。

- 1 つの apply-group コマンドでは、最大 8 つの設定グループが許可されます。

## 設定グループの構成

設定グループには、ルータ設定ツリーの複数の階層レベルで使用できる一連の構成ステートメントが含まれています。設定グループで正規表現を使用すると、複数のインスタンスに適用できる汎用コマンドを作成できます。

設定グループを作成して使用するには、このタスクを使用します。



(注) フレキシブル CLI 設定は、XML インターフェイスでは利用できません。

手順

### ステップ 1 configure

**ステップ 2 group group-name**

例：

```
RP/0/RP0/cpu 0: router(config)# group g-interf
```

設定グループの名前を指定し、グループを定義するためのグループ コンフィギュレーション モードを開始します。group-name 引数は最大 32 文字で、特殊文字は使用できません。

**ステップ 3** グローバル コンフィギュレーション モードから開始して、設定コマンドを入力します。インターフェイス名やその他の変数インスタンスには正規表現を使用します。

例：

```
RP/0/RP0/cpu 0: router(config)# group g-interf
RP/0/RP0/cpu 0: router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-if)# mtu 1500
```

この設定グループに含める構成ステートメントを指定します。

正規表現の使用に関する詳細については、[正規表現を使用した設定グループの継承：例（16 ページ）](#) を参照してください。この例は、すべてのギガビットイーサネットインターフェイスに適用されます。

**ステップ 4 end-group**

例：

```
RP/0/RP0/cpu 0: router(config-GRP-if)# end-group
```

設定グループの設定を完了し、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 5 apply-group**

例：

```
RP/0/RP0/cpu 0: router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/cpu 0: router(config-if)# apply-group g-interf
```

グループが適用されている場所に適用可能なルータ構成に設定グループの構成を追加します。グループは複数の場所に適用することができ、その影響は場所とコンテキストによって異なります。

グループ g-interf からの MTU 値は、インターフェイス TenGigE0/11/0/0 に適用されます。このグループがグローバル コンフィギュレーション モードで適用される場合、この MTU 値は、MTU 値が設定されていないすべてのギガビットイーサネットインターフェイスによって継承されます。

## シンプルな設定グループ：例

次に、設定グループを使用してグローバル設定をシステムに追加する例を示します。

```
RP/0/RP0/cpu 0: router(config)# group g-logging
RP/0/RP0/cpu 0: router(config-GRP)# logging trap notifications
RP/0/RP0/cpu 0: router(config-GRP)# logging console debugging
RP/0/RP0/cpu 0: router(config-GRP)# logging monitor debugging
RP/0/RP0/cpu 0: router(config-GRP)# logging buffered 10000000
RP/0/RP0/cpu 0: router(config-GRP)# end-group

RP/0/RP0/cpu 0: router(config)# apply-group g-logging
```

この設定がコミットされると、g-logging 設定グループに含まれるすべてのコマンドがコミットされます。

## 異なる場所に適用された設定グループ：例

設定グループは異なる場所に適用でき、その影響は適用されるコンテキストによって異なります。次の設定グループを考えてみましょう。

```
RP/0/RP0/cpu 0: router(config)# group g-interfaces
RP/0/RP0/cpu 0: router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-if)# mtu 1500
RP/0/RP0/cpu 0: router(config-GRP-if)# exit
RP/0/RP0/cpu 0: router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-if)# mtu 1000
RP/0/RP0/cpu 0: router(config-GRP-if)# exit
RP/0/RP0/cpu 0: router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-if)# mtu 2000
RP/0/RP0/cpu 0: router(config-GRP-if)# end-group
```

このグループはギガビットイーサネットインターフェイスに適用でき、各インスタンスでは、適用可能な MTU が適用されます。たとえば、次の例では、ギガビットイーサネットインターフェイスの MTU が 1000 に設定されています。

```
RP/0/RP0/cpu 0: router(config)# interface TenGigE0/11/0/0
RP/0/RP0/cpu 0: router(config-if)# apply-group g-interfaces
RP/0/RP0/cpu 0: router(config-if)# ipv4 address 2.2.2.2 255.255.255.0
```

次の例では、ギガビットイーサネットインターフェイスの MTU が 1500 に設定されています。

```
RP/0/RP0/cpu 0: router(config)# interface TenGigE0/11/0/0
RP/0/RP0/cpu 0: router(config-if)# apply-group g-interfaces
RP/0/RP0/cpu 0: router(config-if)# ipv4 address 3.3.3.3 255.255.255.0
```

どちらの場合も同じ設定グループが使用されていますが、適用可能な構成ステートメントのみが使用されています。

## 設定グループの構成の確認

このタスクを使用して、設定グループを使用したルータの設定を確認します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>show running-config group [group-name]</b> 例 :  <pre>RP/0/RP0/cpu 0: router# show running-config group  group g-int-ge  interface 'GigabitEthernet.*'    mtu 1000    negotiation auto ! end-group</pre>	特定のまたはすべての構成済み設定グループの内容を表示します。
ステップ 2	<b>show running-config</b> 例 :  <pre>RP/0/RP0/cpu 0: router# show running-config  group G-INTERFACE-MTU  interface 'GigabitEthernet.*'    mtu 1500 ! end-group  interface interface TenGigE0/11/0/0  apply-group G-INTERFACE-MTU ! interface interface TenGigE0/11/0/1  apply-group G-INTERFACE-MTU  mtu 2000 !</pre>	実行コンフィギュレーションを表示します。適用されているグループが表示されます。これらの設定グループが実際の構成に影響を与えるかどうかに関しては示されません。この例では、グループ G-INTERFACE-MTU は interface TenGigE0/11/0/1 に適用されますが、設定された MTU 値は 1500 ではなく 2000 です。これは、コマンド <b>mtu 2000</b> がインターフェイス上で直接設定されている場合に発生します。実際の構成と設定グループの構成が同じであれば、実際の構成が優先されます。
ステップ 3	<b>show running-config inheritance</b> 例 :  <pre>RP/0/RP0/cpu 0: router# show running-config inheritance . . group G-INTERFACE-MTU  interface 'GigabitEthernet.*'    mtu 1500 ! end-group</pre>	設定グループが適用されている継承された構成を表示します。

	コマンドまたはアクション	目的
	<pre> . . interface interface TenGigE0/11/0/0 ## Inherited from group G-INTERFACE-MTU mtu 1500 ! interface interface TenGigE0/11/0/1 mtu 2000 ! . . </pre>	
ステップ 4	<p><b>show running-config interface x/y/z inheritance detail</b></p> <p>例 :</p> <pre> RP/0/RP0/cpu 0: router# show running-config interface interface TenGigE0/11/0/0 inheritance detail  interface interface TenGigE0/11/0/0 ## Inherited from group G-INTERFACE-MTU mtu 1500 </pre>	特定のコンフィギュレーション コマンドに対する継承された構成を表示します。

## 設定グループの正規表現

正規表現は、設定グループを幅広く適用可能にするために設定グループで使用されます。Portable Operating System Interface for UNIX (POSIX) 1003.2 正規表現は、構成ステートメントの名前でサポートされています。正規表現を区切るには、単一引用符を使用する必要があります。



(注) すべての POSIX 正規表現がサポートされているわけではありません。

### インターフェイス識別子の正規表現

設定グループは、正確なインターフェイス識別子を受け入れません。設定グループに適用可能なインターフェイスのグループを識別するには、正規表現を使用する必要があります。正規表現 `.*` は使用できません。インターフェイス識別子の正規表現は、一義的な単語で始まり、その後正規表現を続けなければなりません。たとえば、ギガビットイーサネットインターフェイスを設定するには、正規表現 `'GigabitEthernet.*'` を使用します。

ルータ設定で使用可能なインターフェイス タイプのリストを表示するには、設定グループプロンプトで **interface ?** と入力します。

```
RP/0/RP0/cpu 0: router(config-GRP)# interface ?
```

```

ATM          'RegExp': ATM Network Interface(s)
BVI          'RegExp': Bridge-Group Virtual Interface
Bundle-Ether 'RegExp': Aggregated Ethernet interface(s)
GigabitEthernet 'RegExp': GigabitEthernet/IEEE 802.3 interface(s)
IMA         'RegExp': ATM Network Interface(s)
Loopback    'RegExp': Loopback interface(s)
MgmtEth     'RegExp': Ethernet/IEEE 802.3 interface(s)
Multilink   'RegExp': Multilink network interface(s)
Null        'RegExp': Null interface
PW-Ether    'RegExp': PWHE Ethernet Interface
PW-IW       'RegExp': PWHE VC11 IP Interworking Interface
Serial      'RegExp': Serial network interface(s)
tunnel-ip   'RegExp': GRE/IPinIP Tunnel Interface(s)
tunnel-mte  'RegExp': MPLS Traffic Engineering P2MP Tunnel interface(s)
tunnel-te   'RegExp': MPLS Traffic Engineering Tunnel interface(s)
tunnel-tp   'RegExp': MPLS Transport Protocol Tunnel interface

```



- (注) インターフェイスのタイプを一意なものにするのに十分な文字数のみを入力するよう要求されますが、フレーズ全体を入力することをお勧めします。正規表現で使用されるすべてのインターフェイスタイプでは、大文字と小文字が区別されます。

サブインターフェイスを指定するには、式の前に\`\`という文字を付けます（バックslashピリオド）。たとえば、すべてのギガビットイーサネットサブインターフェイスを設定するには、`interface 'GigabitEthernet.*\..*'` を使用します。

次の例に示すように、レイヤ2トランスポートインターフェイスまたはポイントツーポイントインターフェイスを指定できます。

```

group g-l2t
  interface 'Gi.*\..*' l2transport
  .
  .
end-group
group g-ptp
  interface 'Gi.*\..*' point-to-point
  .
  .
end-group

```

### OSPF 設定の正規表現

正確なルータプロセス名と OSPF 領域は使用できません。プロセス名または OSPF 領域のグループを指定するには、正規表現を使用する必要があります。OSPF 領域がスカラー値または IP アドレスのいずれかになるように指定するには、次の例のように正規表現 `'.*'` を使用します。

```

group g-ospf
router ospf '.*'
area '.*'
mtu-ignore enable
!
!

```



```

interface \.*'
  bundle port-priority 10
!
interface \.*Ethernet.*'
  bundle port-priority 10
!
end-group

```

この例では、**bundle** コマンドはインターフェイス タイプ **GigabitEthernet** ではサポートされていますが、インターフェイス タイプ **FastEthernet** ではサポートされていません。正規表現 **'.\*'** および **'.\*Ethernet.\*'** は、**GigabitEthernet** タイプと **FastEthernet** タイプの両方に一致します。**bundle** コマンドはこれらの両方のインターフェイス タイプには適用できず、ユニフォーム タイプに解決しないので、システムはこの設定を許可しません。



(注) システムが正規表現からどの構成をすべきかを判別できない場合、その式は有効であると見なされません。



(注) インターフェイス識別子の参照時に正規表現 **'.\*'** は使用できません。インターフェイス識別子の正規表現は、一義的な単語で始まり、その後に正規表現を続けなければなりません。詳細については、この項の「インターフェイス識別子の正規表現」を参照してください。

### 重複する正規表現

正規表現は、設定グループ内の構成ステートメントの名前で使用されます。これにより、一致する名前に適用された場合に設定による継承が可能になります。正規表現を区切るには、単一引用符が使用されます。同じ構成に対し設定グループ内の重複する正規表現は許可されません。

以下に示す例は、複数の設定グループを作成して適用するプロセスを示しています。

```

RP/0//CPU0:router(config)#group FB_flexi_snmp
RP/0//CPU0:router(config-GRP)# snmp-server vrf \.*'
RP/0//CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 traps version 2c group_1
RP/0//CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 informs version 2c group_1
RP/0//CPU0:router(config-GRP-snmp-vrf)# context group_1

RP/0//CPU0:router(config-GRP-snmp-vrf)#
RP/0//CPU0:router(config-GRP-snmp-vrf)#commit

RP/0//CPU0:router(config-GRP-snmp-vrf)#root
RP/0//CPU0:router(config)#
RP/0//CPU0:router(config)#snmp-server vrf vrf1
RP/0//CPU0:router(config-snmp-vrf)#snmp-server vrf vrf10
RP/0//CPU0:router(config-snmp-vrf)#!
RP/0//CPU0:router(config-snmp-vrf)#snmp-server vrf vrf100
RP/0//CPU0:router(config-snmp-vrf)#
RP/0//CPU0:router(config-snmp-vrf)#commit

RP/0//CPU0:router(config-snmp-vrf)#root

```

```

RP/0//CPU0:router(config)#
RP/0//CPU0:router(config)#apply-group FB_flexi_snmp
RP/0//CPU0:router(config)#do sh running-config group
group FB_flexi_snmp
  snmp-server vrf '.*'
    host 1.1.1.1 traps version 2c group_1
    host 1.1.1.1 informs version 2c group_1
  context group_1
!
end-group
apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
RP/0//CPU0:ios#show running-config inheritance detail

```

```

group FB_flexi_snmp
  snmp-server vrf '.*'
    host 1.1.1.1 traps version 2c group_1
    host 1.1.1.1 informs version 2c group_1
  context group_1
!
end-group
snmp-server vrf vrf1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1
!
snmp-server vrf vrf10
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1
!
snmp-server vrf vrf100
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1

```

次の例は、正規表現を示しています。この例では、`snmp-server vrf '.*'` および `snmp-server vrf '[\w]+'` は 2 つの異なる正規表現です。

```

group FB_flexi_snmp
snmp-server vrf '.*'
host 1.1.1.1 traps version 2c group_1
host 1.1.1.1 informs version 2c group_1
context group_1
!

```

```
snmp-server vrf '[\w]+'
host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
!
end-group
```

この個々の正規表現は、次に示すように3つの式 `snmp-server vrf vrf1`、`snmp-server vrf vrf10` および `snmp-server vrf vrf100` すべてに結合されます。

```
apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
```

設定グループには、正規表現の重複のインスタンスが存在することがあります。このような場合、適用されたときに最も優先順位の高い正規表現がアクティブ化され、継承されます。その正規表現は、最も優先順位の高い辞書式順で最初に来ます。

次の例は、重複する正規表現を使用する方法と、優先順位の高い式がどのように適用されるかを示しています。

```
group FB_flexi_snmp
snmp-server vrf '.*'

host 1.1.1.1 traps version 2c group_1

host 1.1.1.1 informs version 2c group_1

context group_1

!

snmp-server vrf '[\w]+'

host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
!
end-group
```

次に示す式は最も優先順位が高いです。

```
group FB_flexi_snmp
snmp-server vrf '.*'

host 1.1.1.1 traps version 2c group_1

host 1.1.1.1 informs version 2c group_1
```

```
context group_1
```

上記の例は、2つの異なる正規表現 `snmp-server vrf '.*'` と `snmp-server vrf '[\w]+'` を示しています。

次の式は、これら2つの式がどのようにマージされるかを示しています。

```
apply-group FB_flexi_snmp

snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
```

優先順位の低い正規表現の変更は、継承に影響しません。

優先順位が低い（上位ではない）既存の正規表現に加えられた変更は、継承には影響しません。

```
snmp-server vrf '[\w]+'

host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
```

次に示すように、優先順位の高い式が継承されます。

```
group FB_flexi_snmp

snmp-server vrf '.*'

host 1.1.1.1 traps version 2c group_1

host 1.1.1.1 informs version 2c group_1

context group_1
```

### グループの優先順位継承の適用

優先順位によって継承が制御されます。



- (注) Cisco IOS XR リリース 6.3.1 以降では、コミット全体に必要なすべてのグループ定義があれば、フレキシブル CLI 設定グループ定義、**apply-group** および **exclude-group** コマンドを任意の順序で入力できます。

グループの優先順位の継承を適用すると、フレキシブルな設定グループがグループ間の共通の構成ステートメントを処理できるようになります。複数の設定グループに共通の構成ステートメントがある場合、継承の優先順位は、内部グループに存在する構成ステートメントが外部グループに存在する構成ステートメントよりも優先されます。タイブレーカーの場合、優先順位

は正規表現の辞書式順序に従って割り当てられます。ユーザが定義したコマンドの順序は受け入れられません。

たとえば、設定グループ ONE の構成ステートメントは、別のグループよりも優先されます。設定グループ SEVEN の構成ステートメントは、他のグループに存在しない場合にのみ使用されます。設定グループ内では、継承の優先順位は最長一致です。

```

apply-group SIX SEVEN
router ospf 0
apply-group FOUR FIVE
area 0
apply-group THREE
interface TenGigE0/11/0/0
apply-group ONE TWO

!
!
!
```

上記の例は、2つのシナリオを示しています。最も内側のグループ (**apply-group ONE TWO**) が最も優先順位が高くなります。ケース 1

最初のシナリオは、どのグループが優先順位を得るかを示しています。この例では、どのグループが異なる設定グループ (共通のものがない異なるグループ) 間で適用されるかを示しています。グループ 1 (**ONE TWO**) を適用すると、7つのグループすべてがインターフェイス `interfaceTenGigE0/11/0/0` に一致し、適用されます。

ケース 2

ここで、すべてが同じ (共通の) 構成を持つ場合、グループ 1 がアクティブになります。つまり、`apply-group ONE TWO` がアクティブになります。グループ 1 が削除されると、グループ 2 がアクティブになります。

## 正規表現を使用した設定例

### 正規表現を使用した設定グループ : 例

次の例では、正確なインターフェイスに正規表現を使用して、ISIS ルーティングパラメータでギガビットイーサネットインターフェイスを設定するための設定グループの定義を示しています。

```

RP/0/RP0/cpu 0: router(config)# group g-isis-gige
RP/0/RP0/cpu 0: router(config-GRP)# router isis '.*'
RP/0/RP0/cpu 0: router(config-GRP-isis)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-isis-if)# lsp-interval 20
RP/0/RP0/cpu 0: router(config-GRP-isis-if)# hello-interval 40
RP/0/RP0/cpu 0: router(config-GRP-isis-if)# address-family ipv4 unicast
RP/0/RP0/cpu 0: router(config-GRP-isis-if-af)# metric 10
RP/0/RP0/cpu 0: router(config-GRP-isis-if-af)# end-group
RP/0/RP0/cpu 0: router(config)#
```

この設定グループの使用について説明するために、ISIS ルーティング パラメータを使用してこれらのギガビットイーサネットインターフェイスを設定すると仮定します。

```
router isis green
interface TenGigE0/11/0/0
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
  !
!
interface TenGigE0/11/0/1
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
  !
!
interface TenGigE0/11/0/2
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
  !
!
interface TenGigE0/11/0/3
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
  !
!
!
```

設定グループを使用してこれらのインターフェイスを設定するには、3つの方法が考えられます。1つ目は、次に示すように、インターフェイス設定内でグループを適用することです。

```
router isis green
interface TenGigE0/11/0/0
  apply-group g-isis-gige
  !
!
interface TenGigE0/11/0/1
  apply-group g-isis-gige
  !
!
interface TenGigE0/11/0/2
  apply-group g-isis-gige
  !
!
interface TenGigE0/11/0/3
  apply-group g-isis-gige
  !
!
```

この状況では、設定グループを適用するインターフェイスのみが設定を継承します。

## 正規表現を使用した設定グループの継承：例

設定グループを使用してこれらのインターフェイスを設定する2つ目の方法は、次に示すように、**router isis** 設定内に設定グループを適用することです。

```
router isis green
  apply-group g-isis-gige
  interface TenGigE0/11/0/0
  !
  interface TenGigE0/11/0/1
  !
  interface TenGigE0/11/0/2
  !
  interface TenGigE0/11/0/3
  !
  !
```

このようにすると、**ISIS** グリーン設定で設定する他のギガビットイーサネットインターフェイスもこれらの設定を継承します。

設定グループを使用してこれらのインターフェイスを設定する3つ目の方法は、次に示すように、グローバルレベルでグループを適用することです。

```
  apply-group g-isis-gige
router isis green
  interface TenGigE0/11/0/0
  !
  interface TenGigE0/11/0/1
  !
  interface TenGigE0/11/0/2
  !
  interface TenGigE0/11/0/3
  !
  !
```

この例では、**ISIS** 用に設定されたすべてのギガビットイーサネットインターフェイスにグループの設定が適用されます。

## 正規表現を使用した設定グループの継承：例

## 設定グループより優先されるローカル設定

明示的な設定は、設定グループから適用された設定よりも優先されます。たとえば、次の設定がルータ上で実行されているとします。

```
router ospf 100
  packet-size 1000
  !
```

次の設定グループを構成して適用し、設定にコミットします。

```
RP/0/RP0/cpu 0: router(config)# group g-ospf
RP/0/RP0/cpu 0: router(config-GRP)# router ospf '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf)# nsf cisco
```

```
RP/0/RP0/cpu 0: router(config-GRP-ospf)# packet-size 3000
RP/0/RP0/cpu 0: router(config-GRP-ospf)# end-group

RP/0/RP0/cpu 0: router(config)# apply-group g-ospf
```

結果は事実上次の構成となります。

```
router ospf 100
  packet-size 1000
  nsf cisco
```

明示的なローカル設定が優先されるため、packet-size 3000 は設定グループから継承されないことに注意してください。

### 互換性のある構成は継承される

設定グループの構成は、継承されるルータ上の構成と一致する必要があります。構成が一致しない場合は、継承されません。たとえば、次の設定がルータ上で実行されているとします。

```
router ospf 100
  auto-cost disable
!
```

次の設定を構成し、設定にコミットします。

```
RP/0/RP0/cpu 0: router(config)# group g-ospf
RP/0/RP0/cpu 0: router(config-GRP)# router ospf '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf)# area '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf-ar)# packet-size 2000
RP/0/RP0/cpu 0: router(config-GRP-ospf)# end-group

RP/0/RP0/cpu 0: router(config)# apply-group g-ospf

RP/0/RP0/cpu 0: router(config)# router ospf 200
RP/0/RP0/cpu 0: router(config-ospf)# area 1
```

結果は事実上次の構成となります。

```
router ospf 100
  auto-cost disable

router ospf 200
  area 1
  packet-size 2000
```

パケットサイズは、ospf 200 構成によって継承されますが、ospf 100 構成では領域が設定されていないため継承されません。

## レイヤ2トランスポート設定グループ：例

次に、レイヤ2トランスポートサブインターフェイスを使用して設定グループを構成および適用する例を示します。

```
RP/0/RP0/cpu 0: router(config)# group g-l2trans-if
RP/0/RP0/cpu 0: router(config-GRP)# interface 'TenGigE.*\..*' l2transport
RP/0/RP0/cpu 0: router(config-GRP)# mtu 1514
RP/0/RP0/cpu 0: router(config-GRP)# end-group

RP/0/RP0/cpu 0: router(config)# interface TenGigE0/0/0/0.1 l2transport
RP/0/RP0/cpu 0: router(config-if)# apply-group g-l2trans-if
```

この設定がコミットされると、10ギガビットイーサネットインターフェイス0/11/0/0.1は1514 MTU値を継承します。これは、10ギガビットイーサネットインターフェイスの **show running-config inheritance** コマンドで表示される出力です。

```
interface TenGigE0/11/0/0.1 l2transport
## Inherited from group g-l2trans-if
mtu 1514
!
```

## 設定グループの優先順位：例

同様の構成ステートメントが複数の設定グループに含まれている場合、内部の設定モードで適用されるグループが外部モードで適用されるグループよりも優先されます。次の例では、OSPFにさまざまなコスト値を設定している2つの設定グループを示しています。

```
RP/0/RP0/cpu 0: router(config)# group g-ospf2
RP/0/RP0/cpu 0: router(config-GRP)# router ospf '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf)# area '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf-ar)# cost 2
RP/0/RP0/cpu 0: router(config-GRP-ospf-ar)# end-group

RP/0/RP0/cpu 0: router(config)# group g-ospf100
RP/0/RP0/cpu 0: router(config-GRP)# router ospf '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf)# area '.*'
RP/0/RP0/cpu 0: router(config-GRP-ospf-ar)# cost 100
RP/0/RP0/cpu 0: router(config-GRP-ospf-ar)# end-group
```

これらの設定グループを次のように適用する場合、g-ospf2で指定されたコスト2は、グループがより内部の設定モードで適用されるため、OSPFエリア0によって継承されます。この場合、グループ g-ospf100 の設定は無視されます。

```
RP/0/RP0/cpu 0: router(config)# router ospf 0
RP/0/RP0/cpu 0: router(config-ospf)# apply-group g-ospf100
RP/0/RP0/cpu 0: router(config-ospf)# area 0
RP/0/RP0/cpu 0: router(config-ospf-ar)# apply-group g-ospf2
```

## 設定グループへの変更は自動的に継承される：例

ルータ構成にコミットされて適用されている設定グループに変更を加えると、変更はルータ構成によって自動的に継承されます。たとえば、次の構成がコミットされているとします。

```
group g-interface-mtu
  interface 'GigabitEthernet.*'
    mtu 1500
  !
end-group

interface POS0/0/0/0
  apply-group g-interface-mtu
!
```

ここで、次の例のように設定グループを変更します。

```
RP/0/RP0/cpu 0: router(config)# group g-interface-mtu
RP/0/RP0/cpu 0: router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/cpu 0: router(config-GRP-if)# mtu 2000
RP/0/RP0/cpu 0: router(config-GRP-if)# end-group
```

この設定グループがコミットされると、インターフェイス GigabitEthernet0/0/0/0 の MTU 設定が自動的に 2000 に更新されます。

■ 設定グループへの変更は自動的に継承される：例



## 第 2 章

# 管理性の設定

このモジュールでは、Extensible Markup Language (XML) エージェント サービスをイネーブルにするために必要な設定について説明します。XML パーサー インフラストラクチャは、Document Object Model (DOM)、Simple アプリケーションプログラミング インターフェイス (API) for XML (SAX)、および文書型定義 (DTD) の妥当性検査機能を使用した XML ドキュメントの解析と生成を実現します。

- DOM を使用すると、XML ドキュメントをプログラムによって作成、操作、生成できます。
- SAX は、XML タグ用のユーザ定義の関数をサポートします。
- DTD は、定義されたドキュメント タイプの妥当性検査を可能にします。
- [XML の管理機能について \(21 ページ\)](#)
- [管理機能の設定方法 \(22 ページ\)](#)
- [管理機能の設定例 \(23 ページ\)](#)

## XML の管理機能について

Cisco IOS XR Extensible Markup Language (XML) API は、外部管理アプリケーションが使用するルータとのプログラマブルインターフェイスを実現します。このインターフェイスは、XML 形式の要求および応答ストリームを使用するルータ設定とモニタリングのメカニズムを提供します。XML インターフェイスは、管理データ API (MDA) の上に構築されています。これは、Cisco IOS XR コンポーネントが、MDA スキーマ定義ファイルを介してデータ モデルをパブリッシュできるようにするメカニズムを提供します。

Cisco IOS XR ソフトウェアには、専用 TCP 接続、Secure Socket Layer (SSL)、または特定の VPN ルーティングおよび転送 (VRF) インスタンスを使用して XML 経由でルータにアクセスする機能があります。

# 管理機能の設定方法

## XML エージェントの設定

ここでは、XML エージェントの設定方法について説明します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>xml agent [ssl]</b>  例： <pre>RP/0/RP0/cpu 0: router(config)# xml agent</pre>	専用の TCP 接続を介して Extensible Markup Language (XML) 要求をイネーブルにし、XML エージェント コンフィギュレーション モードを開始します。Secure Socket Layer (SSL) 上での XML 要求をイネーブルにするには、 <b>ssl</b> キーワードを使用します。
ステップ 2	<b>iteration on size iteration-size</b>  例： <pre>RP/0/RP0/cpu 0: router(config-xml-agent)# iteration on size 500</pre>	大きい XML エージェントの応答の反復サイズを KB 単位で設定します。デフォルト値は 48 です。
ステップ 3	<b>session timeout timeout</b>  例： <pre>RP/0/RP0/cpu 0: router(config-xml-agent)# session timeout 5</pre>	XML エージェントのアイドルタイムアウトを分単位で設定します。デフォルトでは、タイムアウトは設定されていません。
ステップ 4	<b>throttle {memory size   process-rate tags}</b>  例： <pre>RP/0/RP0/cpu 0: router(config-xml-agent)# throttle memory 300</pre>	XML エージェントの処理能力を設定します。 <ul style="list-style-type: none"> <li>• MB 単位でメモリサイズを指定します。有効値の範囲は、100 ~ 600 です。IOS XR 64 ビットでは、値の範囲は 100 ~ 1024 です。デフォルトは 300 です。</li> <li>• XML エージェントが 1 秒に処理できるタグ数の処理率を指定します。有効値の範囲は、1000 ~ 30000 です。デフォルトでは処理率は抑制されません。</li> </ul>

	コマンドまたはアクション	目的
ステップ 5	<b>vrf { vrfname   default} [ipv4 access-list access-list-name]</b>  例： RP/0/RP0/cpu 0: router(config-xml-agent)# vrf vrf1	指定された VPN ルーティングおよび転送 (VRF) インスタンスでメッセージを送受信するように、専用エージェントまたは SSL エージェントを設定します。

## 管理機能の設定例

### XML エージェントでの VRF のイネーブル化：例

次に、専用 XML エージェントを VRF1、VRF2 およびデフォルト VRF 経由でメッセージを送受信するように設定する例を示します。

```
RP/0/RP0/cpu 0: router(config)# xml agent
RP/0/RP0/cpu 0: router(config-xml-agent)# vrf VRF1
RP/0/RP0/cpu 0: router(config-xml-agent)# vrf VRF2
```

次に、専用エージェントから VRF2 へのアクセスを削除する例を示します。

```
RP/0/RP0/cpu 0: router(config)# xml agent ssl
RP/0/RP0/cpu 0: router(config-xml-ssl)# vrf VRF1
RP/0/RP0/cpu 0: router(config-xml-ssl-vrf)# vrf VRF2

RP/0/RP0/cpu 0: router(config)# xml agent
RP/0/RP0/cpu 0: router(config-xml-agent)# no vrf VRF1
```

次に、XML SSL エージェントを VRF1、VRF2 およびデフォルト VRF 経由でメッセージを送受信するように設定する例を示します。

```
RP/0/RP0/cpu 0: router(config)# xml agent ssl
RP/0/RP0/cpu 0: router(config-xml-agent)# vrf VRF1
RP/0/RP0/cpu 0: router(config-xml-agent)# vrf VRF2
```

次に、専用 XML エージェントから VRF2 へのアクセスを削除する例を示します。

```
RP/0/RP0/cpu 0: router(config)# xml agent ssl
RP/0/RP0/cpu 0: router(config-xml-agent)# no vrf VRF2
```





## 第 3 章

# オブジェクト トラッキングの設定

ここでは、Cisco IOS XR ネットワークでのオブジェクト トラッキングの設定について説明します。このモジュール内に記載されているコマンドの詳細については、「[その他の関連資料](#)」の項を参照してください。設定タスクを実行する手順の中で出現する可能性のあるその他のコマンドについて記載されたマニュアルを検索するには、トピック「[その他の関連資料](#)」の「[テクニカル ドキュメント](#)」の項を参照してください。

- [オブジェクト トラッキングの設定 \(25 ページ\)](#)
- [オブジェクト トラッキングの実装の前提条件 \(25 ページ\)](#)
- [オブジェクト トラッキングについて \(26 ページ\)](#)
- [オブジェクト トラッキングの実装方法 \(26 ページ\)](#)
- [オブジェクト トラッキングの設定例 \(37 ページ\)](#)

## オブジェクト トラッキングの設定

ここでは、Cisco IOS XR ネットワークでのオブジェクト トラッキングの設定について説明します。このモジュール内に記載されているコマンドの詳細については、「[その他の関連資料](#)」の項を参照してください。設定タスクを実行する手順の中で出現する可能性のあるその他のコマンドについて記載されたマニュアルを検索するには、トピック「[その他の関連資料](#)」の「[テクニカル ドキュメント](#)」の項を参照してください。

## オブジェクト トラッキングの実装の前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

## オブジェクトトラッキングについて

オブジェクトトラッキングとは、オブジェクトを追跡して、そのプロパティの変化に基づいて、トラッキング対象オブジェクトとは関係のない別のオブジェクトに対してアクションを実行する仕組みです。

各トラッキング対象オブジェクトは、トラッキングコマンドラインインターフェイス (CLI) で指定された一意の名前で識別されます。Cisco IOS XR が処理し、この名前を使用して特定のオブジェクトを追跡します。

トラッキングプロセスでは、定期的にトラッキング対象オブジェクトをポーリングして、ステータスのアップ、ダウンなどの変化をユーザの指定により即時または時間をおいてレポートします。

リストを使った方法で複数のオブジェクトを追跡することもできます。リストはオブジェクトの組み合わせにブール論理式を使った柔軟なメソッドです。リストでは次の演算を使用します。

- **ブールAND関数**：トラッキング対象リストにブールAND関数を指定した場合、サブセット内に定義された各オブジェクトはアップステートでなければならないため、トラッキング対象オブジェクトもアップステートになります。
- **ブールOR関数**：トラッキング対象リストにブールOR関数を指定した場合、サブセット内に定義されたオブジェクトのうち少なくとも1つがアップステートでなければならないため、トラッキング対象オブジェクトもアップステートであることを意味します。

## オブジェクトトラッキングの実装方法

ここでは、さまざまなオブジェクトトラッキングの手順を説明します。

### インターフェイスのラインプロトコルステートのトラッキング

インターフェイスのラインプロトコルステートをトラッキングするには、グローバルコンフィギュレーションモードで次の作業を実行します。

インターフェイスのラインプロトコルがアップしている場合は、トラッキング対象オブジェクトはアップ状態と見なされます。

トラッキング対象オブジェクトの設定後、そのステータスがトラッキング対象になっているインターフェイスを関連付けたり、トラッキングオブジェクトがインターフェイスをポーリングしてステータスを取得するまで待機する秒数を指定したりすることができます。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code>	

	コマンドまたはアクション	目的
ステップ 2	<b>track track-name</b> 例 : RP/0/RP0/cpu 0: router(config)# track track1	トラック コンフィギュレーション モードを開始します。 • <b>track-name</b> : トラッキングの対象となるオブジェクト名を指定します。
ステップ 3	<b>type line-protocol state</b> 例 : RP/0/RP0/cpu 0: router(config-track)# type line-protocol state	インターフェイスのラインプロトコルに基づいてトラッキングを作成します。
ステップ 4	<b>interface type interface-path-id</b> 例 : RP/0/RP0/cpu 0: router(config-track-line-prot)# interface atm 0/2/0/0.1	プロトコルステートをトラッキングするインターフェイスを指定します。 • <b>type</b> : インターフェイスタイプを指定します。詳細については、疑問符 (?) オンラインヘルプ機能を使用します。 • <b>interface-path-id</b> : 物理インターフェイスまたは仮想インターフェイスを識別します。 (注) ルータに現在設定されている可能性があるすべてのインターフェイスのリストを表示するには、 <b>show interfaces</b> コマンドを使用します。 (注) 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は、常にアップステートであり、そのためトラッキングできません。
ステップ 5	<b>exit</b> 例 : RP/0/RP0/cpu 0: router(config-track-line-prot)# exit	トラック ラインプロトコル コンフィギュレーション モードを終了します。
ステップ 6	(任意) <b>delay {up seconds down seconds}</b> 例 : RP/0/RP0/cpu 0: router(config-track)# delay up 10	オブジェクトがアップかダウンかのトラッキング間に発生可能な遅延をスケジューリングします。

	コマンドまたはアクション	目的
ステップ 7	<p>次のいずれかのコマンドを使用します。</p> <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• <b>commit</b></li> </ul> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track)# end</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-track)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li>• <b>end</b> コマンドを実行すると、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。</li> </ul> <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> <li>• <b>yes</b> と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。</li> <li>• <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。</li> <li>• <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> <li>• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## IP ルートの到達可能性のトラッキング

ホストまたはネットワークがリモートサイトでダウン状態になった場合、ルーティングプロトコルはルータに通知し、ルーティングテーブルはそれに応じて更新されます。ルーティングプロセスは、ルーティングアップデートによってルートの状態が変わった場合にトラッキングプロセスに通知するように設定されます。

ルーティングテーブルエントリがルートに存在し、そのルートがアクセス可能であると、トラッキング対象オブジェクトはアップ状態にあると見なされます。

手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>track track-name</b>  例：  RP/0/RP0/cpu 0: router(config)# track track1	トラック コンフィギュレーション モードを開始します。  • <i>track-name</i> : トラッキングの対象となるオブジェクト名を指定します。
ステップ 3	<b>type route reachability</b>  例：  RP/0/RP0/cpu 0: router(config-track)# type route reachability vrf internet	ルーティングアップデートによってルートの状態が変わった場合にトラッキングプロセスに通知するようにルーティングプロセスを設定します。
ステップ 4	次のいずれかのコマンドを使用します。  • <b>vrf vrf-table-name</b> • <b>route ipv4 IP-prefix/mask</b>  例：  RP/0/RP0/cpu 0: router(config-track-route)# vrf vrf-table-4  または  RP/0/RP0/cpu 0: router(config-track-route)# route ipv4 10.56.8.10/16	トラッキングする IP ルートのタイプを設定します。これは、ルータのタイプによって次のいずれかで構成可能です。  • <i>vrf-table-name</i> : VRF テーブル名。 • <i>IP-prefix/mask</i> : ネットワークとサブネットマスクからなる IP プレフィックス (例: 10.56.8.10/16) 。
ステップ 5	<b>exit</b>  例：  RP/0/RP0/cpu 0: router(config-track-line-prot)# exit	トラック ラインプロトコル コンフィギュレーション モードを終了します。
ステップ 6	(任意) <b>delay {up seconds down seconds}</b>  例：  RP/0/RP0/cpu 0: router(config-track)# delay up 10	オブジェクトがアップかダウンかのトラッキング間に発生可能な遅延をスケジューリングします。
ステップ 7	<b>commit</b>	

## オブジェクトリストに基づくトラッキングの設定

グローバル コンフィギュレーション モードでこのタスクを実行し、ブール式を使用してリストの状態を判断して、トラッキング対象オブジェクトリスト（ここではインターフェイスまたはプレフィックスのリスト）を作成します。

トラッキング対象リストには1つまたは複数のオブジェクトが含まれます。ブール式では、AND または OR 演算子を使用して2種類の演算を実行できます。たとえば、AND 演算子を使用して2つのインターフェイスをトラッキングする場合、アップは両方のインターフェイスがアップ状態であることを意味し、ダウンはいずれか一方のインターフェイスがダウン状態であることを意味します。



(注) トラッキング対象リストにオブジェクトを追加するには、そのオブジェクトが存在している必要があります。

NOT 演算子は、1つまたは複数のオブジェクトに指定し、そのオブジェクトの状態を否定します。

トラッキング対象オブジェクトを設定したら、状態をトラッキングするインターフェイスを関連付ける必要があります。オプションとして、トラッキングオブジェクトがインターフェイスをポーリングしてその状態を取得するまでの待機時間を秒数で指定できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>track track-name</b>  例：  RP/0/RP0/cpu 0: router(config)# track track1	トラック コンフィギュレーション モードを開始します。  • <b>track-name</b> : トラッキングの対象となるオブジェクト名を指定します。
ステップ 3	<b>type list boolean { and   or }</b>  例：  RP/0/RP0/cpu 0: router(config-track-list)# type list boolean and	ブール リスト オブジェクトを設定し、トラッキング リスト コンフィギュレーション モードを開始します。  • <b>boolean</b> : トラッキング対象リストのステータスがブール式に基づいて決まることを指定します。  • <b>and</b> : リストについて、すべてのオブジェクトがアップの場合はアップ、ダウンのオブジェクトが1つ以上ある場合はダウンになるように指定します。たとえば2つのインターフェイスをトラッキングする場合、

	コマンドまたはアクション	目的
		<p>アップは両方のインターフェイスがアップ状態であることを意味し、ダウンはいずれか一方のインターフェイスがダウン状態であることを意味します。</p> <ul style="list-style-type: none"> <li>• <b>or</b> : 少なくとも1つのオブジェクトがアップであればリストがアップになるように指定します。たとえば2つのインターフェイスをトラッキングする場合、アップはいずれか一方のインターフェイスがアップ状態であることを意味し、ダウンは両方のインターフェイスがダウン状態であることを意味します。</li> </ul>
ステップ 4	<p><b>object object-name [ not ]</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-track-list)# object 3 not</pre>	<p>リストによるトラッキングの対象となるオブジェクトを指定します。</p> <ul style="list-style-type: none"> <li>• <i>object-name</i> : トラッキングするオブジェクトの名前。</li> <li>• <b>not</b> : オブジェクトの状態を否定します。</li> </ul>
ステップ 5	<p><b>exit</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-track-line-prot)# exit</pre>	<p>トラック ラインプロトコル 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 6	<p>(任意) <b>delay {up seconds down seconds}</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-track)# delay up 10</pre>	<p>オブジェクトがアップかダウンかのトラッキング間に発生可能な遅延をスケジューリングします。</p>
ステップ 7	<p>次のいずれかのコマンドを使用します。</p> <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• <b>commit</b></li> </ul> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-track)# end</pre> <p>または</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li>• <b>end</b> コマンドを実行すると、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。</li> </ul> <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>

	コマンドまたはアクション	目的
	RP/0/RP0/cpu 0: router(config-track)# commit	<ul style="list-style-type: none"> <li>• <b>yes</b> と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。</li> <li>• <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。</li> <li>• <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> <li>• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## オブジェクトリストに基づくトラッキングの設定：しきい値の割合

グローバル コンフィギュレーション モードでこのタスクを実行し、しきい値の割合を使用してリストの状態を判断して、トラッキング対象オブジェクトリスト（ここではインターフェイスまたはプレフィックスのリスト）を作成します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>track track-name</b> 例： RP/0/RP0/cpu 0: router(config)# track track1	トラック 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"> <li>• <b>track-name</b>：トラッキングの対象となるオブジェクト名を指定します。</li> </ul>

	コマンドまたはアクション	目的
ステップ 3	<p><b>type list threshold percentage</b></p> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track-list)# type list threshold percentage</pre>	<p>トラッキングのタイプにしきい値の割合リストを設定します。</p>
ステップ 4	<p><b>object object-name</b></p> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 1 RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 2 RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 3 RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 4</pre>	<p>トラック タイプ track1 のメンバーに object 1、object 2、object 3 および object 4 を設定します。</p>
ステップ 5	<p><b>threshold percentage up percentage down percentage</b></p> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track-list-threshold)# threshold percentage up 50 down 33</pre>	<p>リストがそれぞれアップ状態またはダウン状態であると見なされるために、アップ状態またはダウン状態である必要があるオブジェクトの割合を設定します。</p> <p>たとえば、object 1、object 2、および object 3 がアップ状態にあり、object 4 がダウン状態にある場合、リストはアップ状態にあると見なされます。</p>
ステップ 6	<p>次のいずれかのコマンドを使用します。</p> <ul style="list-style-type: none"> <li>• end</li> <li>• commit</li> </ul> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track)# end</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-track)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li>• end コマンドを実行すると、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。</li> </ul> <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> <li>• 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て、ルータが EXEC モードに戻ります。</li> </ul>

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> <li>• <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。</li> <li>• <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> <li>• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## オブジェクトリストに基づくトラッキングの設定：しきい値の重み

グローバル コンフィギュレーション モードでこのタスクを実行し、しきい値の重みを使用してリストの状態を判断して、トラッキング対象オブジェクトリスト（ここではインターフェイスまたはプレフィックスのリスト）を作成します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>track track-name</b> 例： <pre>RP/0/RP0/cpu 0: router(config)# track track1</pre>	トラック 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"> <li>• <b>track-name</b>：トラッキングの対象となるオブジェクト名を指定します。</li> </ul>
ステップ 3	<b>type list threshold weight</b> 例： <pre>RP/0/RP0/cpu 0: router(config-track-list)# type list threshold weight</pre>	トラッキングのタイプにしきい値の重みリストを設定します。

	コマンドまたはアクション	目的
ステップ 4	<p><b>object</b> <i>object-name</i> <b>weight</b> <i>weight</i></p> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 1 weight 10 RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 2 weight 5 RP/0/RP0/cpu 0: router(config-track-list-threshold)# object 3 weight 3</pre>	<p>track t1 のメンバーに object 1、object 2 および object 3 を設定し、それぞれに重み 10、5 および 3 を設定します。</p>
ステップ 5	<p><b>threshold</b> <b>weight up</b> <i>weight</i> <b>down</b> <i>weight</i></p> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track-list-threshold)# threshold weight up 10 down 5</pre>	<p>リストがそれぞれアップ状態またはダウン状態であると見なされるために、アップ状態またはダウン状態である必要があるオブジェクトの重みの範囲を設定します。この例では、object 1 および 2 がアップ状態にあり、累積の重みは 15 である（10～5 の範囲内ではない）ため、リストはダウン状態と見なされます。</p>
ステップ 6	<p>次のいずれかのコマンドを使用します。</p> <ul style="list-style-type: none"> <li>• end</li> <li>• commit</li> </ul> <p>例：</p> <pre>RP/0/RP0/cpu 0: router(config-track)# end</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-track)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li>• <b>end</b> コマンドを実行すると、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。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> <li>• <b>yes</b> と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。</li> <li>• <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。</li> <li>• <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュ</li> </ul> </li> </ul>

	コマンドまたはアクション	目的
		<p>レシジョンセッションは終了せず、設定変更もコミットされません。</p> <ul style="list-style-type: none"> <li>• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## IPSLA の到達可能性のトラッキング

IP サービス レベル契約 (SLA) 動作の戻りコードのトラッキングをイネーブルにするには、このタスクを使用します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b> 例： RP/0/RP0/cpu 0: router# <b>configure</b>	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	<b>track track-name</b> 例： RP/0/RP0/cpu 0: router(config)# <b>track t1</b>	トラック コンフィギュレーションモードを開始します。
ステップ 3	<b>type rtr ipsla-no reachability</b> 例： RP/0/RP0/cpu 0: router(config-track)# <b>type rtr 100 reachability</b>	到達可能性をトラッキングする IP SLA 動作 ID を指定します。 <i>ipsla-no</i> の有効値は、1 ~ 2048 の範囲です。
ステップ 4	<b>commit</b>	

### IPSLA トラッキングの設定：例

次に、IPSLA のトラッキング設定の例を示します。

```
RP/0/RP0/cpu 0: router(config)# track track1
RP/0/RP0/cpu 0: router(config-track)# type rtr 1 reachability
RP/0/RP0/cpu 0: router(config-track)# delay up 5
RP/0/RP0/cpu 0: router(config-track)# delay down 10
```

## オブジェクトトラッキングの設定例

インターフェイスがアップ状態かダウン状態かのトラッキング：実行コンフィギュレーションの例

```
track connection100
  type list boolean and
  object object3 not
  delay up 10
  !
interface service-ipsec 23
  line-protocol track connection100
  !
```

インターフェイスのラインプロトコルステートのトラッキング：実行コンフィギュレーションの例

この例では、トラフィックはインターフェイス service-ipsec1 から到着し、インターフェイス TenGigE0/11/0/3 を経由して終了します。

```
track IPsec1
  type line-protocol state
  interface TenGigE0/11/0/3
  !
interface service-ipsec 1
  ipv4 address 70.0.0.1 255.255.255.0
  profile vrfl_profile_ipsec
  line-protocol track IPsec1
  tunnel source 80.0.0.1
  tunnel destination 80.0.0.2
  service-location preferred-active 0/0/1
  !
```

次に、前述の例を実行した後の **show track** コマンドの出力例を示します。

```
RP/0/RP0/cpu 0: router# show run track

Track IPsec1
Interface GigabitEthernet0_0_0_3 line-protocol
!
  Line protocol is UP
  1 change, last change 10:37:32 UTC Thu Sep 20 2007
  Tracked by:
  service-ipsec1
  !
```

**IP ルートの到達可能性のトラッキング：実行コンフィギュレーションの例**

この例では、インターフェイス `service-ipsec1` から到着したトラフィックの宛先がネットワーク `7.0.0.0/24` にあります。このトラッキング手順は、ルーティングプロトコルプレフィックスの状態に従い、ルーティングテーブルに変更があったときに信号を送ります。

```
track PREFIX1
  type route reachability
    route ipv4 7.0.0.0/24
    !
  interface service-ipsec 1
  vrf 1
  ipv4 address 70.0.0.2 255.255.255.0
  profile vrf_1_ipsec
  line-protocol track PREFIX1
  tunnel source 80.0.0.2
  tunnel destination 80.0.0.1
  service-location preferred-active 0/2/0
```

**オブジェクトのリストに基づいたトラックの構築：実行コンフィギュレーションの例**

この例では、インターフェイス `service-ipsec1` から到着するトラフィックが、インターフェイス `TenGigE0/11/0/3` およびインターフェイス `ATM0/2/0/0.1` を介して終了します。トラフィックの宛先はネットワーク `7.0.0.0/24` です。

いずれかのインターフェイスまたはリモートネットワークがダウンした場合は、トラフィックフローが停止される必要があります。これを行うには、ブール AND 式を使用します。

```
track C1
  type route reachability
    route ipv4 3.3.3.3/32
    !
  !
track C2
  type route reachability
    route ipv4 1.2.3.4/32
    !
  !
track C3
  type route reachability
    route ipv4 10.0.20.2/32
    !
  !
track C4
  type route reachability
    route ipv4 10.0.20.0/24
    !
  !
track OBJ
  type list boolean and
    object C1
    object C2
  !
```

```
!  
track OBJ2  
  type list boolean or  
    object C1  
    object C2  
!
```

### IPSLA ベースのオブジェクトトラッキングの設定：コンフィギュレーションの例

次に、ACL と IPSLA 設定を含む IPSLA ベースのオブジェクトトラッキングの設定例を示します。

ACL の設定：

```
RP/0/RP0/cpu 0: router(config)# ipv4 access-list abf-track  
RP/0/RP0/cpu 0: router(config-ipv4-acl)# 10 permit any nexthop track track1 1.2.3.4
```

オブジェクトトラッキングの設定：

```
RP/0/RP0/cpu 0: router(config)# track track1  
RP/0/RP0/cpu 0: router(config-track)# type rtr 1 reachability  
RP/0/RP0/cpu 0: router(config-track)# delay up 5  
RP/0/RP0/cpu 0: router(config-track)# delay down 10
```

IPSLA の設定：

```
RP/0/RP0/cpu 0: router(config)# ipsla  
RP/0/RP0/cpu 0: router(config-ipsla)# operation 1  
RP/0/RP0/cpu 0: router(config-ipsla-op)# type icmp echo  
RP/0/RP0/cpu 0: router(config-ipsla-icmp-echo)# source address 2.3.4.5  
RP/0/RP0/cpu 0: router(config-ipsla-icmp-echo)# destination address 1.2.3.4  
RP/0/RP0/cpu 0: router(config-ipsla-icmp-echo)# frequency 60  
RP/0/RP0/cpu 0: router(config-ipsla-icmp-echo)# exit  
RP/0/RP0/cpu 0: router(config-ipsla-op)# exit  
RP/0/RP0/cpu 0: router(config-ipsla)# schedule operation 1  
RP/0/RP0/cpu 0: router(config-ipsla-sched)# start-time now  
RP/0/RP0/cpu 0: router(config-ipsla-sched)# life forever
```





## 第 4 章

# 物理端末および仮想端末の設定

ラインテンプレートは、物理端末回線および仮想端末回線（VTY）を介した着信および送信転送の標準属性の設定を定義します。VTY プールを使用して、さまざまな仮想端末回線にテンプレートの設定を適用します。

ここでは、Cisco IOS XR ネットワークでの物理端末および仮想端末の実装に必要なタスクについて説明します。

- [物理端末と仮想端末を実装するための前提条件](#)（41 ページ）
- [物理端末および仮想端末の実装について](#)（41 ページ）
- [Cisco IOS XR ソフトウェアでの物理および仮想端末の実装方法](#)（44 ページ）
- [物理および仮想端末の実装の設定例](#)（49 ページ）

## 物理端末と仮想端末を実装するための前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

## 物理端末および仮想端末の実装について

物理端末および仮想端末を実装するには、この項の内容を理解しておく必要があります。

### ラインテンプレート

Cisco IOS XR ソフトウェアでは次のラインテンプレートが使用できます。

- デフォルトラインテンプレート：物理および仮想端末回線に適用されます。
- コンソールラインテンプレート：コンソール回線に適用されます。
- ユーザ定義ラインテンプレート：仮想端末回線の範囲に適用できます。

## ラインテンプレート コンフィギュレーション モード

ラインテンプレートの属性の変更は、ラインテンプレート コンフィギュレーション モードで行います。ラインテンプレート コンフィギュレーション モードに移行するには、XR コンフィギュレーション モードから **line** コマンドを実行し、変更するテンプレートを指定します。これらのラインテンプレートは、**line** コマンドを使用して設定できます。

- **console** : コンソール テンプレート
- **default** : デフォルト テンプレート
- **template** : ユーザ定義 テンプレート

**line** コマンドでテンプレートを指定すると、ルータは指定されたラインの端末属性を設定できるラインテンプレート コンフィギュレーション モードを開始します。次に、コンソールの属性を指定する例を示します。

```
RP/0/RP0/cpu 0: router(config)# line console  
RP/0/RP0/cpu 0: router(config-line)#
```

ラインテンプレート コンフィギュレーション モードで、すべての使用可能なオプションを表示するには、オンラインヘルプ機能 (?) を使用します。次に、便利なオプションの一部を示します。

- **absolute-timeout** : ライン切断のタイムアウト値を指定します。
- **escape-character** : ラインのエスケープ文字を変更します。
- **exec-timeout** : EXEC タイムアウトを指定します。
- **length** : 画面に表示する行数を設定します。
- **session-limit** : 許容される発信接続の数を指定します。
- **session-timeout** : 入力トラフィックがない場合に接続を切断するインターバルを指定します。
- **timestamp** : 各コマンドの前にタイムスタンプを表示します。
- **width** : 表示端末の幅を指定します。

## ラインテンプレート ガイドライン

コンソールテンプレートの変更およびユーザ定義テンプレートの設定について、次のガイドラインが適用されます。

- ルータ上の物理端末回線（コンソールポート）のテンプレートは、ラインテンプレート コンフィギュレーション モードから変更します。コンソールテンプレートでラインテンプレート コンフィギュレーション モードを開始するには、XR コンフィギュレーション モードから **line console** コマンドを使用します。

- 仮想回線のテンプレートは、**line template-name** コマンドでユーザ定義テンプレートを設定し、ラインテンプレートコンフィギュレーションからユーザ定義テンプレートの端末属性を設定し、**vtypool** コマンドを使用して複数の仮想端末回線にテンプレートを適用することによって変更します。



- (注) VTY プールを作成または変更する前に、XR コンフィギュレーション モードで **telnet server** コマンドを使用して telnet サーバをイネーブルにします。詳細については、『Cisco IOS XR IP Addresses and Services Configuration Guide』および『Cisco IOS XR IP Addresses and Services Command Reference』を参照してください。

## 端末の識別

コンソールポート用の物理端末回線は、各コンソールポートが存在するアクティブまたはスタンバイルートプロセッサ (RP) 上での位置 (*rack/slot/module* の形式で表される) によって識別されます。仮想端末の場合、物理的な位置は適用できません。Cisco IOS XR ソフトウェアは、VTY 接続が確立された順序に従って VTY ID を VTY に割り当てます。

## VTY プール

各仮想ラインは、共通のラインテンプレートコンフィギュレーションを使用する接続プールのメンバーです。複数の VTY プールが存在する場合があります。それぞれ、VTY プールに設定されているとおりに、定義された数の VTY が含まれます。Cisco IOS XR ソフトウェアは、デフォルトで次の VTY プールをサポートします。

- デフォルトの VTY プール：デフォルトの VTY プールは、5 つの VTY (VTY 0 ~ 4) で構成され、それぞれデフォルトラインテンプレートを参照します。
- デフォルトの障害マネージャプール：デフォルトの障害マネージャプールは、6 つの VTY (VTY 100 ~ 105) で構成され、それぞれデフォルトラインテンプレートを参照します。

デフォルトの VTY プールおよびデフォルトの障害マネージャプールのほかに、デフォルトテンプレートまたはユーザ定義テンプレートを参照できる、ユーザ定義の VTY プールを設定することもできます。

VTY プールを設定する際は、次のガイドラインに従ってください。

- デフォルト VTY プールの VTY の範囲は、VTY 0 から開始し、5 つ以上の VTY を含む必要があります。
- 0 ~ 99 の範囲の VTY は、デフォルトの VTY プールを参照できます。
- 5 ~ 99 の範囲の VTY は、ユーザ定義の VTY プールを参照できます。
- 100 以上の範囲の VTY は、障害マネージャの VTY プール用に予約されています。
- 障害マネージャ VTY プールの VTY の範囲は、VTY 100 から開始し、6 つ以上の VTY を含む必要があります。

- 1 つの VTY がメンバになることができる VTY プールは 1 つだけです。別のプールにすでに含まれる VTY を含めると、VTY プールの設定は失敗します。
- VTY プールを設定するときにアクティブな VTY プールからアクティブな VTY を削除しようとする、その VTY プールの設定は失敗します。

## Cisco IOS XR ソフトウェアでの物理および仮想端末の実装方法

### テンプレートの変更

ここでは、コンソールラインテンプレートとデフォルトラインテンプレートの端末属性を変更する方法について説明します。設定した端末属性によって、指定したテンプレートのテンプレート設定が変更されます。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>line {console   default}</b> 例 : <pre>RP/0/RP0/cpu 0: router(config)# line console</pre> または <pre>RP/0/RP0/cpu 0: router(config)# line default</pre>	指定された回線テンプレートの回線テンプレート コンフィギュレーション モードが開始されます。 <ul style="list-style-type: none"> <li>• <b>console</b> : コンソールテンプレートのラインテンプレート コンフィギュレーション モードが開始されます。</li> <li>• <b>default</b> : デフォルトラインテンプレートのラインテンプレート コンフィギュレーション モードが開始されます。</li> </ul>
ステップ 3	ラインテンプレート コンフィギュレーション モードでコマンドを使用して、特定のテンプレートの端末属性を設定します。	—
ステップ 4	次のいずれかのコマンドを使用します。 <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• <b>commit</b></li> </ul> 例 :	設定変更を保存します。 <ul style="list-style-type: none"> <li>• <b>end</b> コマンドを実行すると、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。</li> </ul>

	コマンドまたはアクション	目的
	<pre>RP/0/RP0/cpu 0: router(config-line)# end</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-line)# commit</pre>	<p>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</p> <ul style="list-style-type: none"> <li>• <b>yes</b> と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。</li> <li>• <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータがEXECモードに戻ります。変更はコミットされません。</li> <li>• <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> <li>• 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## VTY プールの作成および変更

このタスクでは、VTY プールを作成および変更する方法について説明します。

VTY プールを参照するようにデフォルトのラインテンプレートを設定する場合は、ステップ3～ステップ5 (**line template** および **exit** コマンド) を省略できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	

	コマンドまたはアクション	目的
ステップ 2	<b>telnet {ipv4   ipv6} server max-servers limit</b> 例 :  <pre>RP/0/RP0/CPU0:router(config)# telnet ipv4 server max-servers 10</pre>	許可できる Telnet サーバの数を指定します。最大で 100 台までの Telnet サーバを許可できます。  (注) デフォルトでは、Telnet サーバは許可されていません。Telnet サーバを使用できるようにするには、このコマンドを設定する必要があります。
ステップ 3	<b>line template template-name</b> 例 :  <pre>RP/0/RP0/CPU0:router(config)# line template 1</pre>	ユーザ定義のテンプレートのライン テンプレート 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	ライン テンプレート コンフィギュレーション モードでコマンドを使用して、特定のライン テンプレートの端末属性設定を設定します。	—
ステップ 5	<b>exit</b> 例 :  <pre>RP/0/RP0/CPU0:router(config-line)# exit</pre>	ライン テンプレート 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 モードに戻ります。
ステップ 6	<b>vty-pool {default   pool-name   eem} first-vty last-vty [line-template {default   template-name}]</b> 例 :  <pre>RP/0/RP0/CPU0:router(config)#vty-pool default 0 5 line-template default</pre> または  <pre>RP/0/RP0/CPU0:router(config)#vty-pool pool1 5 50 line-template template1</pre> または  <pre>RP/0/RP0/CPU0:router#vty-pool eem 100 105 line-template template1</pre> <pre>RP/0/RP0/CPU0:router(config)#vty-pool default 0 5 line-template template1</pre>	VTY プールを作成または変更します。  <ul style="list-style-type: none"> <li>• <b>line-template</b> キーワードを使用してライン テンプレートを指定しないと、VTY プールがデフォルトのライン テンプレートになります。</li> <li>• <b>default</b> : デフォルトの VTY プールを設定します。             <ul style="list-style-type: none"> <li>• デフォルトの VTY プールは、VTY 0 から開始し、5 つ以上の VTY (VTY 0 ~ 4) を含む必要があります。</li> <li>• デフォルトの VTY プールを構成する VTY の範囲を大きくすることによって、デフォルトの VTY プールのサイズを変更できます。</li> </ul> </li> </ul>

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> <li>• <i>pool-name</i> : ユーザ定義の VTY プールを作成します。 <ul style="list-style-type: none"> <li>• ユーザ定義のプールは、少なくとも VTY 5 から開始する必要があります。ただし、デフォルトの VTY プールのサイズが変更されたかどうかにもよります。</li> <li>• デフォルトの VTY プールの VTY の範囲のサイズが変更された場合、デフォルト ライン テンプレートがない最初の範囲の値を使用します。たとえば、デフォルトの VTY プールの VTY の範囲が 10 個の VTY が含まれるよう変更されている場合は (VTY 0 ~ 9)、ユーザ定義の VTY プールの範囲の値は VTY 10 から始まるようにします。</li> </ul> </li> <li>• <i>eem</i> : Embedded Event Manager のプールを設定します。 <ul style="list-style-type: none"> <li>• デフォルトの Embedded Event Manager の VTY プールは、VTY 100 から開始し、6 つ以上の VTY (VTY 100 ~ 105) を含む必要があります。</li> </ul> </li> <li>• <i>line-template template-name</i> : ユーザ定義のテンプレートを参照する VTY プールを設定します。</li> </ul>
ステップ 7	commit	

## 端末および端末セッションのモニタリング

このタスクでは、物理回線および端末回線に使用可能な **show EXEC** コマンドを使用して、端末と端末セッションをモニタする方法について説明します。



(注) コマンドは任意の順序で入力できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<p>(任意) <b>show line</b> [<b>aux location node-id</b>   <b>console location node-id</b>   <b>vtty number</b>]</p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show line</pre>	<p>端末回線の端末パラメータを表示します。</p> <ul style="list-style-type: none"> <li>• <b>show line aux location node-id</b> EXEC コマンドを指定すると、補助回線の端末パラメータが表示されます。</li> <li>• <b>show line console location node-id</b> EXEC コマンドを指定すると、コンソールの端末パラメータが表示されます。 <ul style="list-style-type: none"> <li>• <b>location node-id</b> キーワードおよび引数については、それぞれの補助回線またはコンソールポートが存在するルートプロセッサ (RP) の場所を入力します。</li> <li>• <b>node-id</b> 引数は、<i>rack/slot/module</i> の形式で入力します。</li> </ul> </li> <li>• <b>show line vty number</b> EXEC コマンドを指定すると、指定した VTY の端末パラメータが表示されます。</li> </ul>
ステップ 2	<p>(任意) <b>show terminal</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show terminal</pre>	<p>現在の端末回線の端末属性設定を表示します。</p>
ステップ 3	<p>(任意) <b>show users</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show users</pre>	<p>ルータのアクティブ回線に関する情報を表示します。</p>

## 物理および仮想端末の実装の設定例

### コンソール テンプレートの変更 : 例

この設定例は、コンソールラインテンプレートの端末属性の設定を変更する方法を示します。

```
line console
  exec-timeout 0 0
  escape-character 0x5a
  session-limit 10
  disconnect-character 0x59
  session-timeout 100
  transport input telnet
  transport output telnet
```

この設定例では、次の端末属性がコンソールラインテンプレートに適用されます。

- 端末セッションの EXEC タイムアウトは 0 分、0 秒に設定されます。EXEC タイムアウトを 0 分、0 秒に設定すると、EXEC タイムアウト機能がディセーブルになります。したがって、端末セッションの EXEC セッションがタイムアウトになることはありません。
- エスケープ文字は 0x5a の 16 進数値に設定されます (0x5a の 16 進数値は「Z」の文字に変換されます)。
- 発信端末セッションのセッション制限は、10 接続に設定されます。
- 切断文字は 0x59 の 16 進数値に設定されます (0x59 の 16 進文字は「Y」の文字に変換されます)。
- 発信端末セッションのセッションタイムアウトは 100 分 (1 時間 40 分) に設定されます。
- 着信端末セッションに許可されるトランスポート プロトコルは、Telnet です。
- 発信端末セッションに許可されるトランスポート プロトコルは、Telnet です。

コンソールラインテンプレートの端末属性がコンソールに適用されたことを確認するには、**show line** コマンドを使用します。

```
RP/0/RP0/cpu 0: router:router# show line console location 0/RP0/CPU0
Tue Nov 24 03:10:24.656 UTC
Tty          Speed      Overruns      Acc I/O
*con0/RP0/CPU0      9600      0/0          -/-

Line "con0_RP1_CPU0", Location "0/RP1/CPU0", Type "Console"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600, "No" Parity, 2 stopbits, 8 databits
Template: console
Capabilities: Timestamp Enabled
Allowed transports are telnet.
```

### デフォルト テンプレートの変更 : 例

次の設定例では、デフォルト ラインテンプレートの端末設定を上書きする例を示します。

```
line default
  exec-timeout 0 0
  width 512
  length 512
```

次の例では、次の端末属性はデフォルト ラインテンプレートのデフォルト 端末属性の設定を上書きします。

- 端末セッションの EXEC タイムアウトは 0 分、0 秒に設定されます。EXEC タイムアウトを 0 分、0 秒に設定すると、EXEC タイムアウト機能をディセーブルにします。したがって、端末セッションの EXEC セッションは一切タイムアウトしません（デフォルト ラインテンプレートのデフォルトの EXEC タイムアウトは 10 分です）。
- デフォルト テンプレートを参照する端末の端末画面幅が 512 文字に設定されます（デフォルト ラインテンプレートのデフォルトの幅は 80 文字です）。
- デフォルト テンプレートを参照する端末に一度に表示する長さ、つまり行数は 512 行に設定されます（デフォルト ラインテンプレートのデフォルトの長さは 24 行です）。

### デフォルト VTY プールを参照するユーザ定義テンプレートの設定 : 例

この設定例では、VTY 用のユーザ定義ラインテンプレート（この例では test という名前）を設定し、デフォルト VTY プールを参照するようにラインテンプレート テストを設定する方法を示します。

```
line template test
  exec-timeout 100 0
  width 100
  length 100
  exit
vty-pool default 0 4 line-template test
```

### ユーザ定義の VTY プールを参照するユーザ定義テンプレートの設定 : 例

この設定例は、VTY のユーザ定義のラインテンプレート（この例では test2 という名前）を設定し、ユーザ定義の VTY プール（この例では pool1 という名前）を参照するラインテンプレート テストを設定する方法を示します。

```
line template test2
  exec-timeout 0 0
  session-limit 10
  session-timeout 100
  transport input all
  transport output all
```

```
exit
vty-pool pool1 5 50 line-template test2
```

### 障害マネージャの VTY プールを参照するユーザ定義テンプレートの設定：例

この設定例では、VTY のユーザ定義のラインテンプレート（この例では test3 という名前）を設定し、障害マネージャの VTY プールを参照するラインテンプレートテストを設定する方法を示します。

```
line template test3
width 110
length 100
session-timeout 100
exit
vty-pool eem 100 106 line-template test3
```





## 第 5 章

# 簡易ネットワーク管理プロトコルの設定

簡易ネットワーク管理プロトコル (SNMP) は、アプリケーション層プロトコルであり、SNMP マネージャと SNMP エージェントとの通信に使用されるメッセージフォーマットを提供します。SNMP では、ネットワーク内のデバイスのモニタリングと管理に使用する標準フレームワークと共通言語が提供されます。

ここでは、Cisco IOS XR ネットワーク上において SNMP の実装に必要な作業について説明します。

- [SNMP の実装の前提条件 \(53 ページ\)](#)
- [Cisco IOS XR ソフトウェアでの SNMP の使用に関する制約事項 \(53 ページ\)](#)
- [SNMP の実装について \(54 ページ\)](#)
- [サブスクリバセッションでのセッション MIB のサポート \(61 ページ\)](#)
- [Cisco IOS XR ソフトウェアでの SNMP の実装方法 \(62 ページ\)](#)

## SNMP の実装の前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

## Cisco IOS XR ソフトウェアでの SNMP の使用に関する制約事項

SNMP 出力は、32 ビット幅しかありません。そのため、 $2^{32}$  を超える情報は表示できません。 $2^{32}$  は 4.29 ギガビットになります。



(注) 10 ギガビット インターフェイスは  $2^{32}$  を超えているため、インターフェイスに関する速度情報を表示しようとすると、結果が連結形式で表示される場合があります。

10 ギガビットを超えるインターフェイスの正しい速度を表示するには、ifHighSpeed を使用できます。

## SNMP の実装について

SNMP を実装するには、この項の内容を理解しておく必要があります。

### SNMP 機能の概要

SNMP フレームワークは 3 つの部分で構成されます。

- SNMP マネージャ
- SNMP エージェント
- 管理情報ベース (MIB)

### SNMP マネージャ

SNMP マネージャは、SNMP を使用するネットワークホストのアクティビティを制御およびモニタするために使用されるシステムです。最も一般的な管理システムは、ネットワーク管理システム (NMS) と呼ばれます。NMS という用語は、ネットワーク管理に使用する専用デバイスを意味する場合と、このようなデバイス上で使用するアプリケーションを意味する場合があります。さまざまなネットワーク管理アプリケーションが SNMP とともに使用可能です。簡単なコマンドラインアプリケーションから機能が豊富なグラフィカルユーザインターフェイス (CiscoWorks 2000 製品ラインなど) まで、このような機能は多岐にわたっています。

### SNMP エージェント

SNMP エージェントは、管理対象デバイスの内部で動作するソフトウェアコンポーネントであり、デバイスのデータを保持し、必要に応じて管理システムにそれらのデータを報告します。エージェントおよび MIB は、ルータに常駐します。SNMP エージェントをイネーブルにするには、マネージャとエージェントの関係を定義する必要があります。

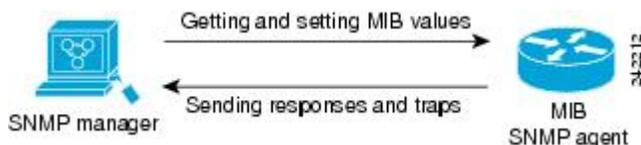
### MIB

管理情報ベース (MIB) は、ネットワーク管理情報用の仮想情報ストレージ領域であり、管理対象オブジェクトの集合で構成されます。MIB 内には、MIB モジュールで定義された関連オブジェクトの集合体があります。MIB モジュールは、STD 58、RFC 2578、RFC 2579、および RFC 2580 の定義に従って、SNMP MIB モジュール言語で記述されます。なお、個々の MIB モジュールも MIB と呼ばれます。たとえば、インターフェイスグループ MIB (IF-MIB) はシステム上の MIB 内の MIB モジュールです。

SNMP エージェントには、SNMP マネージャが Get 操作や Set 操作を通じて値を要求したり変更したりできる MIB 変数が含まれています。マネージャでは、エージェントからの値の取得またはエージェントへの値の保存が可能です。エージェントは、デバイスパラメータやネットワークデータの保存場所である MIB から値を収集します。エージェントは、マネージャのデータ取得要求やデータ設定要求にも応答できます。

次の図に、SNMP マネージャと SNMP エージェントの間の通信の関係を示します。マネージャは、MIB 値の取得および設定の要求をエージェントに送信できます。エージェントはこれらの要求に応答できます。このやりとりとは別に、エージェント側からは、任意の通知（トラップ）をマネージャに送信して、ネットワークの状況をマネージャに通知できます。

図 1: SNMP エージェントと SNMP マネージャの間の通信



### IP-MIB のサポート

RFC4293 IP-MIB は、IPv4 と IPv6 の統計情報を個別に提供するように特別に設計されました。RFC 4293 で定義されている **ipIfStatsTable** には、インターフェイス固有の統計情報がリストされています。ipIfStatsTable の IPv6 統計情報のサポートは以前に追加されていますが、IP-MIB の IOS-XR 実装では、以前のリリースの場合 RFC4293 に従い IPv4 統計情報をサポートしていませんでした。

リリース 6.3.2 以降から、IP-MIB の IOS-XR 実装では、RFC4293 に従い IPv4 統計情報がサポートされています。これにより、インターフェイスごとに IPV4 と IPv6 の統計情報を個別に収集することができます。ipIfStatsTable は、2つのサブ ID アドレス タイプ (IPv4 または IPv6) とインターフェイス ifindex[1] によってインデックス付けされます。IPv4 および IPv6 への IP-MIB サポートの実装は、読みやすさと保守性を向上させるためにリリース 6.3.2 から分離されています。

IPv4 統計情報について ipIfStatsTable に追加された OID のリストは次のとおりです。

- ipIfStatsInReceives
- ipIfStatsHCInReceives
- ipIfStatsInOctets
- ipIfStatsHCInOctets
- ipIfStatsOutTransmits
- ipIfStatsHCOutTransmits
- ipIfStatsOutOctets
- ipIfStatsHCOutOctets
- ipIfStatsDiscontinuityTime

IPv4 統計情報用に追加された新しい OID のリストについては、「[SNMP OID Navigator](#)」を参照してください。

## SNMP バージョン

Cisco IOS XR ソフトウェアでは、次のバージョンの SNMP がサポートされています。

- 簡易ネットワーク管理プロトコルバージョン 1 (SNMPv1)
- 簡易ネットワーク管理プロトコルバージョン 2c (SNMPv2c)
- 簡易ネットワーク管理プロトコルバージョン 3 (SNMPv3)

SNMPv1 および SNMPv2c はどちらも、コミュニティベース形式のセキュリティを使用します。エージェントの MIB にアクセスできるマネージャのコミュニティが、IP アドレス アクセス コントロール リスト および パスワード によって定義されます。

SNMPv2c サポートには、バルク取得メカニズム、および管理ステーションに対するより詳細なエラーメッセージ報告が含まれています。バルク取得メカニズムは、テーブルおよび大量の情報の取得をサポートして、必要なラウンドトリップの回数を最小化します。SNMPv2c ではエラー処理のサポートが改善されました。たとえば、異なる種類のエラー条件が区別されるように、エラーコードが拡張されました。SNMPv1 では、これらの条件は単一のエラーコードを使用して報告されていました。エラーリターンコードでエラータイプが報告されるようになりました。no such object exceptions、no such instance exceptions、および end of MIB view exceptions の 3 種類の例外も報告されます。

SNMPv3 は、セキュリティモデルです。セキュリティモデルは、ユーザおよびユーザが属するグループに合わせて設定される認証方式です。セキュリティレベルとは、セキュリティモデル内で許可されるセキュリティのレベルです。セキュリティモデルとセキュリティレベルの組み合わせによって、SNMP パケットの処理時に採用されるセキュリティメカニズムが決まります。SNMPv3 で使用可能なセキュリティレベルのリストについては、[SNMPv1、SNMPv2、SNMPv3 のセキュリティモデルおよびセキュリティレベル \(57 ページ\)](#) を参照してください。SNMPv3 機能は、RFC 3411 ~ 3418 をサポートします。

SNMP エージェントは、管理ステーションでサポートされる SNMP のバージョンを使用するように設定する必要があります。エージェントは複数のマネージャと通信できます。このため、1 つの管理ステーションとは SNMPv1 プロトコルを使用して通信し、1 つの管理ステーションとは SNMPv2c プロトコルを使用して通信し、もう 1 つの管理ステーションとは SNMPv3 を使用して通信することがサポートされるように、Cisco IOS-XR ソフトウェアを設定できます。

## SNMPv1、SNMPv2c、および SNMPv3 の比較

SNMP v1、v2c、および v3 はすべて次の動作をサポートします。

- **get-request** : 特定の変数から値を取得します。
- **get-next-request** : 指定した変数の次の値を取得します。この動作はテーブル内からの変数取得によく使用されます。この動作では、SNMP マネージャに正確な変数名を認識させる必要はありません。SNMP マネージャは、必要な変数を MIB 内で順番に検索していきます。
- **get-response** : NMS によって送信された get-request、get-next-request、および set-request に応答する動作です。
- **set-request** : 特定の変数に値を保存する動作です。

- trap : 何らかのイベントが発生したときに、SNMP エージェントによって SNMP マネージャに送信される非送信請求メッセージです。

次の表では、SNMP v1、v2c、および v3 でサポートされるその他の主要な SNMP 機能を示します。

表 1: SNMPv1、v2c、および v3 機能のサポート

機能	SNMP v1	SNMP v2c	SNMP v3
Get-Bulk 動作	なし	あり	あり
Inform 動作	なし	あり (Cisco IOS XR ソフトウェアではなし)	あり (Cisco IOS XR ソフトウェアではなし)
64 ビット カウンタ	なし	あり	あり
テキストの表記法	なし	あり	あり
認証	なし	なし	あり
プライバシー (暗号化)	なし	なし	あり
認証およびアクセス コントロール (ビュー)	なし	なし	あり

## SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル

セキュリティ レベルは、SNMP メッセージを開示から保護する必要があるかどうか、およびメッセージを認証するかどうか判断します。セキュリティ モデル内のさまざまなセキュリティ レベルは、次のとおりです。

- noAuthNoPriv : 認証または暗号化を実行しないセキュリティ レベル。
- authNoPriv : 認証は実行するが、暗号化を実行しないセキュリティ レベル。
- authPriv : 認証と暗号化両方を実行するセキュリティ レベル。

SNMPv1、SNMPv2c、および SNMPv3 の 3 つのセキュリティ モデルを使用できます。セキュリティ モデルとセキュリティ レベルの組み合わせにより、SNMP メッセージの処理中に適用されるセキュリティ メカニズムが決まります。

次の表で、セキュリティ モデルとセキュリティ レベルの組み合わせについて説明します。

表 2: SNMP セキュリティ モデルおよびセキュリティ レベル

モデル	レベル	認証	暗号化	結果
v1	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。

モデル	レベル	認証	暗号化	結果
v2c	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。
v3	noAuthNoPriv	ユーザ名	未対応	ユーザ名の照合を使用して認証します。
v3	authNoPriv	HMAC-MD5 または HMAC-SHA	未対応	HMAC <sup>1</sup> -MD5 <sup>2</sup> アルゴリズムまたは HMAC-SHA <sup>3</sup> に基づいて認証します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	DES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。DES <sup>4</sup> 56 ビット暗号化、および CBC <sup>5</sup> DES (DES-56) 標準に基づいた認証を提供します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	3DES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。168 ビットの 3DES <sup>6</sup> レベルの暗号化を提供します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	AES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。128 ビットの AES <sup>7</sup> レベルの暗号化を提供します。

<sup>1</sup> Hash-Based Message Authentication Code

<sup>2</sup> メッセージ ダイジェスト 5

<sup>3</sup> セキュア ハッシュ アルゴリズム

<sup>4</sup> データ暗号規格

<sup>5</sup> 暗号ブロック連鎖

<sup>6</sup> Triple Data Encryption Standard

<sup>7</sup> Advanced Encryption Standard

3DES および AES 暗号化規格を使用するため、セキュリティパッケージ (k9sec) がインストールされている必要があります。ソフトウェアパッケージのインストールの詳細については、『*Upgrading and Managing Cisco IOS XR Software*』を参照してください。

## SNMPv3 の利点

SNMPv3 は、認証、暗号化、およびアクセスコントロールを提供することで、デバイスへの安全なアクセスを実現します。これらのセキュリティの利点が追加されたことより、次のセキュリティ上の脅威に対して SNMP がセキュリティ保護されます。

- マスカレード：SNMP ユーザが別の SNMP ユーザのアイデンティティを装って、その SNMP ユーザが許可されていない管理操作を実行する脅威。

- メッセージストリームの改変：メッセージが悪意を持って並べ替え、遅延、または再生されて（サブネットワークサービスの通常の操作によって発生するよりも大きい程度に）、SNMP が不正な管理操作を実行するようになる脅威。
- 暴露：SNMP エンジン間でのやり取りが傍受される可能性がある脅威。ローカルポリシーの問題としてこの脅威から保護が必要な場合があります。

さらに、SNMPv3 では、SNMP 管理対象オブジェクト上のプロトコル操作に対するアクセス制御も提供されます。

## SNMPv3 のコスト

SNMPv3 の認証および暗号化は、MIB オブジェクトに対する SNMP 操作の実行時の応答時間をわずかに増加させる要因となります。このコストは、SNMPv3 がもたらすセキュリティ上の利点からすれば、無視できる程度のものであります。

次の表に、セキュリティ モデルとセキュリティ レベルのさまざまな組み合わせを応答時間の短い順に示します。

表 3: 応答時間の短い順

セキュリティ モデル	セキュリティ レベル
SNMPv2c	noAuthNoPriv
SNMPv3	noAuthNoPriv
SNMPv3	authNoPriv
SNMPv3	authPriv

## ユーザベースのセキュリティ モデル

SNMPv3 ユーザベース セキュリティ モデル (USM) は SNMP メッセージレベルセキュリティを参照し、次のサービスを提供します。

- メッセージの完全性：メッセージが不正な方法で変更または破壊されず、データシーケンスが悪意なく起こり得る範囲を超えて変更されていないことを保証します。
- メッセージ発信元の認証：受信データを発信したユーザのアイデンティティが確認されたことを保証します。
- メッセージの機密性：情報が使用不可であること、または不正なユーザ、エンティティ、またはプロセスに開示されないことを保証します。

SNMPv3 は、設定済みユーザによる管理動作のみを許可し、SNMP メッセージを暗号化します。

USM では、次の 2 つの認証プロトコルが使用されます。

- HMAC-MD5-96 認証プロトコル
- HMAC-SHA-96 認証プロトコル

USM は、メッセージ暗号化用のプライバシー プロトコルとして暗号ブロック連鎖 (CBC)-DES (DES-56) を使用します。

## View-Based Access Control Model

SNMP ユーザは、View-Based Access Control Model (VACM) を使用して、SNMP オブジェクトに対する読み取りアクセス、書き込みアクセス、または通知アクセスを指定することにより、SNMP 管理対象オブジェクトへのアクセスを制御できます。これは、ビューによって制限されているオブジェクトへのアクセスを防止します。これらのアクセス ポリシーは、**snmp-server group** コマンドでユーザ グループを構成するときに設定できます。

### MIB ビュー

セキュリティ上の理由から、一部のグループのアクセスを、管理ドメイン内の一部の管理情報のみに限定できることが頻繁に重要になります。この機能を実現するために、管理オブジェクトへのアクセスは、MIB ビューによって制御されます。このビューには、表示可能な管理対象オブジェクト タイプ (およびオプションとしてオブジェクト タイプの特定のインスタンス) のセットが含まれます。

### アクセス ポリシー

アクセスポリシーによって、グループのアクセス権限が決定します。アクセス権限には、次の 3 種類があります。

- 読み取りビュー アクセス：オブジェクト読み取り時に、グループに許可されているオブジェクト インスタンスのセット。
- 書き込みビュー アクセス：オブジェクト書き込み時に、グループに許可されているオブジェクト インスタンスのセット。
- 通知ビューアクセス：オブジェクトの通知での送信時に、グループに許可されているオブジェクト インスタンスのセット。

## SNMP の IP precedence および DSCP サポート

SNMP による IP precedence および差分化サービスコードポイント (DSCP; DiffServ コードポイント) のサポートでは、SNMP トラフィックに特定した QoS を提供します。ユーザがプライオリティの設定を変更することができるため、ルータで生成した SNMP トラフィックを特定の QoS クラスに割り当てます。IP precedence または IP DSCP のコードポイント値は、パケットを重み付けランダム早期検出 (WRED) でどのように処理するかを決定するのに使用します。

ルータで生成された SNMP トラフィックに IP precedence または IP DSCP が設定されると、同じルータの種類異なる SNMP トラフィックに異なる QoS クラスを割り当てられなくなります。

IP precedence 値は、IP ヘッダーの ToS (タイプオブ サービス) バイトの最初の 3 ビットです。IP DSCP コードポイント値は、差分化サービス (DiffServ フィールド) バイトの最初の 6 ビットです。最大 8 つの異なる IP precedence マーキングまたは 64 の異なる IP DSCP マーキングを設定できます。

# サブスクリバセッションでのセッション MIB のサポート

SNMP モニタリングでは、すべてのタイプのサブスクリバに関する情報が必要です。CISCO-SUBSCRIBER-SESSION-MIB は、サブスクリバごとのデータと集約サブスクリバ (PPPoE) データをモデル化するために定義されます。設定されたしきい値を超える集約セッション数に関する通知 (トラップ) をサポートする必要があります。CISCO-SUBSCRIBER-SESSION-MIB の汎用 MIB データ コレクタ マネージャ (DCM) のサポートにより、データ収集が高速化し、並列データの処理も向上します。

## SNMP 通知

SNMP の重要な機能の 1 つは、SNMP エージェントから通知を生成できることです。これらの通知では、要求を SNMP マネージャから送信する必要はありません。Cisco IOS XR ソフトウェアでは、任意 (非同期) の通知は、トラップとしてのみ生成できます。トラップは、ネットワーク上のある状態を SNMP マネージャに通知するメッセージです。通知は、不正なユーザ認証、再起動、接続の切断、隣接ルータとの接続の切断、その他の重要なイベントを表示します。



(注) インフォーム要求 (インフォーム操作) は Cisco IOS XR ソフトウェア ではサポートされています。

トラップの信頼性はインフォームより低くなります。受信側はトラップを受信しても確認応答を送信しないからです。送信側は、トラップが受信されたかどうかを判断できません。インフォーム要求を受信する SNMP マネージャは、SNMP 応答プロトコルデータユニット (PDU) でメッセージの受信を確認応答します。マネージャがインフォーム要求を受信しなかった場合、応答は返されません。送信側が応答を受信しない場合、インフォーム要求を再び送信できます。このため、インフォームの方が目的の宛先に到達する確実性が高くなります。

ただし、インフォームはルータやネットワークのリソースをより多く消費するので、多くの場合、トラップの方が好んで使用されます。送信と同時に廃棄されるトラップと異なり、インフォーム要求は応答を受信するまで、または要求がタイムアウトになるまで、メモリ内に保持する必要があります。またトラップが一度だけ送信されるのに対し、インフォームは数回再試行されることがあります。再送信の回数が増えるとトラフィックが増加し、ネットワークのオーバーヘッドが高くなる原因にもなります。このように、トラップとインフォーム要求の間には、信頼性とリソースのトレードオフの関係があります。

### 図 2: SNMP マネージャで受信したトラップ

この図では、エージェント ルータは SNMP マネージャにトラップを送信します。マネージャはトラップを受信しますが、エージェントに確認応答を送信しません。エージェントには、トラップが宛先に到達したことを知る方法がありません。

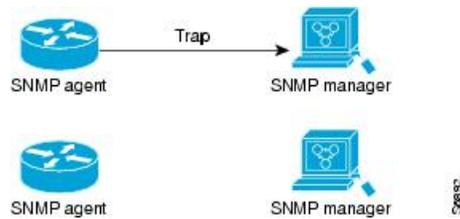
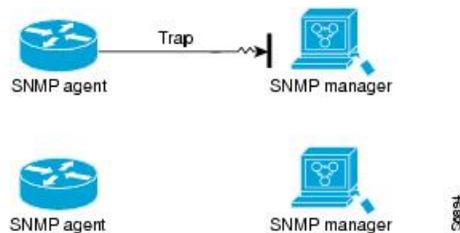


図 3: SNMP マネージャで受信されなかったトラップ

次の図では、エージェントがマネージャにトラップを送信しますが、トラップはマネージャに届きません。トラップが宛先に到達しなかったことをエージェントが確認する方法がないため、トラップは再度送信されません。そのため、マネージャはこのトラップを受信できません。



## セッションタイプ

サポートされているセッションタイプは次のとおりです。

- PPPoE
- IP SUB PKT
- IP SUB DHCP

## Cisco IOS XR ソフトウェアでの SNMP の実装方法

ここでは、SNMP の実装方法について説明します。

**snmp-server** コマンドは、デフォルトで、管理イーサネット インターフェイスで SNMP をイネーブルにします。

## SNMPv3 の設定

このタスクでは、ネットワーク管理およびモニタリングに SNMPv3 を設定する方法について説明します。



- (注) 特定のコマンドで SNMPv3 をイネーブルにすることはできません。SNMPv3 は、最初に行う **snmp-server** グローバル コンフィギュレーション コマンド (**config**) によってイネーブルになります。したがって、このタスクでは **snmp-server** コマンドを実行する順序は重要ではありません。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	(任意) <b>snmp-server engineid local engine-id</b>  例 :  RP/0/RP0/cpu 0: router# snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61	ローカル SNMP エンジンの識別番号を指定します。
ステップ 3	(任意) <b>snmp-server vrf vrf-name</b>  例 :  RP/0/RP0/cpu 0: router# snmp-server vrf vrf_a	SNMP の VRF プロパティを設定します。
ステップ 4	<b>snmp-server view view-name oid-tree {included   excluded}</b>  例 :  RP/0/RP0/cpu 0: router# snmp-server view view_name 1.3.6.1.2.1.1.5 included	ビューレコードを作成または変更します。
ステップ 5	<b>snmp-server group name {v1   v2c   v3 {auth   noauth   priv}} [read view] [write view] [notify view] [access-list-name]</b>  例 :  RP/0/RP0/cpu 0: router# snmp-server group group_name v3 noauth read view_name1 write view_name2	新規 SNMP グループ、または SNMP ユーザを SNMP ビューにマッピングするテーブルを設定します。
ステップ 6	<b>snmp-server user username groupname {v1   v2c   v3 [auth {md5   sha} {clear   encrypted} auth-password [priv des56</b>	SNMP グループに新しいユーザを設定します。

	コマンドまたはアクション	目的
	<pre>{clear   encrypted} priv-password]]} [access-list-name]</pre> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# snmp-server user noauthuser group_name v3</pre>	
<b>ステップ 7</b>	<b>commit</b>	
<b>ステップ 8</b>	<p>(任意) <b>show snmp</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp</pre>	SNMP のステータスに関する情報を表示します。
<b>ステップ 9</b>	<p>(任意) <b>show snmp engineid</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp engineid</pre>	ローカル SNMP エンジンに関する情報を表示します。
<b>ステップ 10</b>	<p>(任意) <b>show snmp group</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp group</pre>	ネットワークの各 SNMP グループに関する情報を表示します。
<b>ステップ 11</b>	<p>(任意) <b>show snmp users</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp users</pre>	SNMP ユーザテーブルの各 SNMP ユーザ名に関する情報を表示します。
<b>ステップ 12</b>	<p>(任意) <b>show snmp view</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp view</pre>	関連する MIB ビューファミリー名、ストレージタイプ、ステータスなど、設定されたビューに関する情報を表示します。

## SNMPv3 の設定 : 例

### エンジン ID の設定

次に、ローカル SNMP エンジンの ID を設定する例を示します。

```
snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61
```



(注) エンジン ID が設定されると、SNMP エージェントが再起動します。

### ローカル SNMP エンジンの ID の確認

次に、ローカル SNMP エンジンの ID を確認する例を示します。

```
config
  show snmp engineid

SNMP engineID 00000009000000a1ffffffff
```

### ビューの作成

ビューを作成するには2つの方法があります。

- **snmp-server view** コマンドの **included** キーワードを使用することによって、ビューに MIB ファミリの ASN.1 サブツリーのオブジェクト識別子 (OID) を包含することができます。
- **snmp-server view** コマンドの **excluded** キーワードを使用することによって、ビューから MIB ファミリの ASN.1 サブツリーの OID サブツリーを除外することができます。

次に、sysName (1.3.6.1.2.1.1.5) オブジェクトを含むビューを作成する例を示します。

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 included
```

次に、システム グループのすべての OID を含むビューを作成する例を示します。

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
```

次に、除外されている sysName オブジェクト (1.3.6.1.2.1.1.5) を除く、システム グループのすべての OID を含むビューを作成する例を示します。

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 excluded
```

### 設定したビューの確認

次に、設定したビューの情報を表示する例を示します。

```
RP/0/RP0/cpu 0: router# show snmp view

vldefault 1.3.6.1 - included nonVolatile active
SNMP_VIEW1 1.3.6.1.2.1.1 - included nonVolatile active
SNMP_VIEW1 1.3.6.1.2.1.1.5 - excluded nonVolatile active
```

## グループの作成

通知、読み取り、または書き込みビューを明示的に指定しないと、Cisco IOS XR ソフトウェアでは v1 デフォルト (1.3.6.1) が使用されます。次に、デフォルトビューを使用するグループを作成する例を示します。

```
RP/0/RP0/cpu 0: router# snmp-server group group-name v3 auth
```

次の設定例は、グループに適用されるビューから除外された `sysUpTime` オブジェクト (1.3.6.1.2.1.1.3) を除く、システム内のすべての OID に対する読み取りアクセス権があり、`sysName` オブジェクト (1.3.6.1.2.1.1.5) に対しては書き込みアクセス権しかないグループを作成する例を示します。

```
!
snmp-server view view_name1 1.3.6.1.2.1.1 included
snmp-server view view_name1 1.3.6.1.2.1.1.3 excluded
snmp-server view view_name2 1.3.6.1.2.1.1.5 included
snmp-server group group_name1 v3 auth read view_name1 write view_name2
!
```

## グループの確認

この例では、設定したグループの属性を確認する方法を示します。

```
RP/0/RP0/cpu 0: router# show snmp group

groupname: group_name1                security model:usm
readview  : view_name1                writeview: view_name2
notifyview: vldefault
row status: nonVolatile
```

## ユーザの作成および確認

次の SNMPv3 ビューおよび SNMPv3 グループの設定があるとします。

```
!
snmp-server view view_name 1.3.6.1.2.1.1 included
snmp-server group group_name v3 noauth read view_name write view-name
!
```

次に、システム グループに対する読み取りビュー アクセスおよび書き込みビュー アクセスの権限を持つ `noAuthNoPriv` ユーザを作成する例を示します。

```
config
 snmp-server user noauthuser group_name v3
```



(注) `noAuthNoPriv` ユーザを作成するには、ユーザが `noauth` グループに属している必要があります。

次に、SNMP ユーザに適用する属性を確認する例を示します。

```
RP/0/RP0/cpu 0: router# show snmp user

User name: noauthuser
Engine ID: localSnmpID
storage-type: nonvolatile active
```

次の SNMPv3 ビューおよび SNMPv3 グループの設定があるとします。

```
!
 snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
 snmp-server group SNMP_GROUP1 v3 auth notify SNMP_VIEW1 read SNMP_VIEW1 write SNMP_VIEW1
!
```

次に、システム グループに対する認証（暗号化を含む）、読み取り/書き込みビュー アクセスの権限を持つユーザを作成する例を示します。

```
config
 snmp-server user userv3authpriv SNMP_GROUP1 v3 auth md5 password123 priv aes 128
 password123
```

次の SNMPv3 ビューおよび SNMPv3 グループの設定があるとします。

```
!
 snmp-server view view_name 1.3.6.1.2.1.1 included
 snmp group group_name v3 priv read view_name write view_name
!
```

次に、システム グループに対する読み取りビュー アクセスおよび書き込みビュー アクセスの権限を持つ `authNoPriv` ユーザを作成する例を示します。

```
RP/0/RP0/cpu 0: router# snmp-server user authuser group_name v3 auth md5 clear auth_passwd
```



- (注) グループはセキュリティ レベル **Auth** に設定されているので、このグループにアクセスするには、ユーザが最低でも「**auth**」として設定されている必要があります（「**priv**」ユーザもこのグループにアクセスできます）。このグループに設定された **authNoPriv** ユーザの **authuser** は、ビューにアクセスするために認証パスワードを入力する必要があります。この例では、**auth\_passwd** が認証パスワード文字列として設定されています。**auth\_passwd** パスワード文字列の前に **clear** キーワードが指定されていることに注意してください。**clear** キーワードは、指定されているパスワード文字列が暗号化されていないことを示します。

次に、SNMP ユーザに適用する属性を確認する例を示します。

```
RP/0/RP0/cpu 0: router# show snmp user
```

```
User name: authuser
Engine ID: localSnmpID
storage-type: nonvolatile active
```

次の SNMPv3 ビューおよび SNMPv3 グループの設定があるとします。

```
!
snmp view view_name 1.3.6.1.2.1.1 included
snmp group group_name v3 priv read view_name write view_name
!
```

次に、システム グループへの読み取りビュー アクセスおよび書き込みビュー アクセスの権限を持つ **authPriv** ユーザを作成する例を示します。

```
config
snmp-server user privuser group_name v3 auth md5 clear auth_passwd priv des56 clear
priv_passwd
```



- (注) グループのセキュリティ レベルは **Priv** なので、ユーザがこのグループにアクセスするには、「**priv**」ユーザとして設定される必要があります。この例のユーザ **privuser** は、ビュー内の OID にアクセスするために、認証パスワードとプライバシーパスワードの両方を入力する必要があります。

次に、SNMP ユーザに適用する属性を確認する例を示します。

```
RP/0/RP0/cpu 0: router# show snmp user
```

```
User name: privuser
Engine ID: localSnmpID
storage-type: nonvolatile active
```

## SNMP トラップ通知の設定

ここでは、SNMP トラップ通知を送信するようにルータを設定する方法について説明します。



- (注) [SNMPv3 の設定 \(62 ページ\)](#) タスクで説明した手順をすでに完了している場合は、[SNMPv3 の設定 \(62 ページ\)](#) を省略できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>snmp-server group name {v1v2v3 {auth   noauth   priv}} [readview] writeview] [notifyview] [access-list-name]</b>  例： RP/0/RP0/cpu 0: router# snmp-server group group_name v3 noauth read view_name1 writer view_name2	新規 SNMP グループ、または SNMP ユーザを SNMP ビューにマッピングするテーブルを設定します。
ステップ 3	<b>snmp-server user groupname {v1v2cv3 {auth   md5   sha} {clear   encrypted} auth-password] [priv des56 {clear   access-list-name}]</b>  例： RP/0/RP0/cpu 0: router# snmp-server group group_name v3 noauth read view_name1 writer view_name2	新規 SNMP グループ、または SNMP ユーザを SNMP ビューにマッピングするテーブルを設定します。
ステップ 4	<b>snmp-server user username groupname {v1v2cv3 {auth   md5   sha} {clear   encrypted} auth-password] [priv des56 {clear   access-list-name}]</b>  例： RP/0/RP0/cpu 0: routerconfig# snmp-server user noauthuser group_name v3	新規 SNMP グループ、または SNMP ユーザを SNMP ビューにマッピングするテーブルを設定します。
ステップ 5	<b>[snmp-server host address [traps] [version {1   2c   3 [auth   noauth   priv]}] community-string [udp-port port] [notification-type]</b>  例： RP/0/RP0/cpu 0: router(config)# snmp-server host 12.26.25.61 traps version 3 noauth userV3noauth	SNMP トラップ通知、使用する SNMP のバージョン、通知のセキュリティレベル、通知の受信者（ホスト）を指定します。

	コマンドまたはアクション	目的
ステップ 6	<b>snmp-server traps [notification-type]</b> 例 : RP/0/RP0/cpu 0: router(config)# snmp-server traps bgp	トラップ通知の送信をイネーブルにし、送信するトラップ通知のタイプを指定します。 <ul style="list-style-type: none"> <li>• トラップを <i>notification-type</i> 引数で指定しない場合は、サポートされるすべてのトラップ通知がルータ上でイネーブルになります。ルータで使用可能なトラップ通知を表示するには、<b>snmp-server traps ?</b> コマンドを入力します。</li> </ul>
ステップ 7	<b>commit</b>	
ステップ 8	(任意) <b>show snmp host</b> 例 : RP/0/RP0/cpu 0: router# show snmp host	設定された SNMP 通知の受信者 (ホスト)、ポート番号、セキュリティ モデルに関する情報を表示します。

## トラップ通知の設定 : 例

次に、異なるタイプのトラップを送信するように SNMP エージェントを設定する例を示します。設定には、v2c ユーザ、noAuthNoPriv ユーザ、anauthNoPriv ユーザ、および AuthPriv ユーザが含まれます。



- (注) デフォルトのユーザ データグラム プロトコル (UDP) ポートは 161 です。 **udp-port** キーワードおよび *port* 引数を使用して UDP ポートを指定しないと、設定された SNMP トラップ通知はポート 161 に送信されます。

```
!
snmp-server host 10.50.32.170 version 2c userV2c udp-port 2345
snmp-server host 10.50.32.170 version 3 auth userV3auth udp-port 2345
snmp-server host 10.50.32.170 version 3 priv userV3priv udp-port 2345
snmp-server host 10.50.32.170 version 3 noauth userV3noauth udp-port 2345
snmp-server user userV2c groupV2c v2c
snmp-server user userV3auth groupV3auth v3 auth md5 encrypted 140F0A13
snmp-server user userV3priv groupV3priv v3 auth md5 encrypted 021E1C43 priv des56
encrypted 1110001C
snmp-server user userV3noauth groupV3noauth v3 LROwner
snmp-server view view_name 1.3 included
snmp-server community public RW
snmp-server group groupV2c v2c read view_name
snmp-server group groupV3auth v3 auth read view_name
snmp-server group groupV3priv v3 priv read view_name
snmp-server group groupV3noauth v3 noauth read view_name
!
```

次に、SNMP トラップ通知の受信者ホストの設定、つまり SNMP トラップ通知の受信者を確認する方法を示しています。出力には、次の情報が表示されます。

- 設定された通知ホストの IP アドレス
- SNMP 通知メッセージが送信される UDP ポート
- 設定されたトラップのタイプ
- 設定されたユーザのセキュリティ レベル
- 設定されたセキュリティ モデル

```
config
show snmp host

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3auth security model: v3 auth

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3noauth security model: v3 noauth

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3priv security model: v3 priv

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userv2c security model: v2c
```

## SNMP エージェントの連絡先、場所、およびシリアル番号の設定

このタスクは、SNMP エージェントのシステムの連絡先文字列、システムの場所の文字列、およびシステム シリアル番号を設定する方法について説明します。



(注) このタスクでは **snmp-server** コマンドを実行する順序は重要ではありません。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	(任意) <b>snmp-server contact</b> <i>system-contact-string</i>  例：  RP/0/RP0/cpu 0: router(config)# snmp-server contact Dial System Operator at beeper # 27345	システムの連絡先文字列を設定します。
ステップ 3	(任意) <b>snmp-server location</b> <i>system-location</i>  例：	システムの場所を表す文字列を設定します。

	コマンドまたはアクション	目的
	RP/0/RP0/cpu 0: router(config)# snmp-server location Building 3/Room 214	
ステップ 4	(任意) <b>snmp-server chassis-id serial-number</b>  例 :  RP/0/RP0/cpu 0: router(config)# snmp-server chassis-id 1234456	システムのシリアル番号を設定します。
ステップ 5	<b>commit</b>	

## SNMP エージェントパケットの最大サイズの定義

このタスクでは、SNMP サーバが要求を受信しているか応答を生成しているときに、許可される SNMP パケットの最大サイズを設定する例を示します。



(注) このタスクでは **snmp-server** コマンドを実行する順序は重要ではありません。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	(任意) <b>snmp-server packetsize byte-count</b>  例 :  RP/0/RP0/cpu 0: router(config)# snmp-server packetsize 1024	最大パケットサイズを設定します。
ステップ 3	<b>commit</b>	

## 通知操作値の変更

SNMP 通知がイネーブルになると、送信元インターフェイス、メッセージキューの長さ、または再送信間隔にデフォルト以外の値を指定することができます。

ここでは、トラップ通知用の送信元インターフェイス、各ホストのメッセージキューの長さ、および再送信間隔を指定する方法について説明します。



(注) このタスクでは **snmp-server** コマンドを実行する順序は重要ではありません。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	(任意) <b>snmp-server trap-source type interface-path-id</b>  例：  RP/0/RP0/cpu 0: router(config)# snmp-server trap-source POS 0/0/1/0	トラップ通知の送信元インターフェイスを指定します。
ステップ 3	(任意) <b>snmp-server queue-length length</b>  例：  RP/0/RP0/cpu 0: router(config)# snmp-server queue-length 20	各通知のメッセージキューの長さを設定します。
ステップ 4	(任意) <b>snmp-server trap-timeout seconds</b>  例：  RP/0/RP0/cpu 0: router(config)# snmp-server trap-timeout 20	再送信キューにある通知を再送信する頻度を定義します。
ステップ 5	<b>commit</b>	

## IP precedence および DSCP 値の設定

ここでは、SNMP トラフィックに対して IP precedence または IP DSCP を設定する方法について説明します。

#### 始める前に

SNMP が設定されていること。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	

## SNMP トラフィックの IP precedence 値の設定 : 例

	コマンドまたはアクション	目的
ステップ 2	次のいずれかのコマンドを使用します。 <ul style="list-style-type: none"> <li>• <code>snmp-server ipv4 precedence value</code></li> <li>• <code>snmp-server ipv4 dscp value</code></li> </ul> 例 :  RP/0/RP0/cpu 0: router(config)# <code>snmp-server dscp 24</code>	SNMP トラフィックの IP precedence または IP DSCP 値を設定します。
ステップ 3	<code>commit</code>	

## SNMP トラフィックの IP precedence 値の設定 : 例

次の例に、SNMP IP precedence 値を 7 に設定する方法を示します。

```
configure
snmp-server ipv4 precedence 7
exit

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

## SNMP トラフィックの IP DSCP 値の設定 : 例

次の例に、SNMP トラフィックの IP DSCP 値を 45 に設定する方法を示します。

```
configure
snmp-server ipv4 dscp 45
exit

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

## SNMP コンテキスト マッピングの表示

SNMP エージェントは、クライアント機能により作成された SNMP コンテキストに基づいてクエリーを提供します。コンテキスト マッピング テーブルが存在します。コンテキスト マッピング テーブルの各エントリには、コンテキスト名、コンテキストを作成した機能の名前、および機能の特定のインスタンスの名前が含まれます。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>show snmp context-mapping</b> 例 :  RP/0/RP0/cpu 0: router# show snmp context-mapping	SNMP コンテキスト マッピング テーブルを表示します。

## パケット損失のモニタリング

パケット損失が指定したしきい値を超えたときの SNMP トラップの生成を設定することにより、パケット損失をモニタすることが可能です。このタスクで説明する設定は、EVENT-MIB の MIB テーブルのエントリの作成をイネーブルにします。これは、その後 SNMP GET 操作を使用してパケット損失をモニタできます。

## 始める前に



- (注) このタスクで説明する設定を使用して、EVENT-MIB MIB テーブルに作成されたエントリは、SNMP SET を使用して変更できません。

SNMP SET を使用して作成された、EVENT-MIB MIB テーブルへのエントリは、このタスクで説明する設定を使用して変更できません。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>snmp-server mibs eventmib packet-loss type interface-path-id falling lower-threshold interval sampling-interval rising upper-threshold</b> 例 :  RP/0/RP0/cpu 0: router(config)# snmp-server mibs eventmib packet-loss falling 1 interval 5 rising 2	<p>パケット損失が指定したしきい値を超えたときに、インターフェイスに対して SNMP EVENT-MIB トラップを生成します。最大 100 のインターフェイスをモニタできます。</p> <p><b>falling lower-threshold</b> : 低い方のしきい値を指定します。2つの間隔間のパケット損失がこのしきい値を下回り、以前に <b>mteTriggerRising</b> トラップが生成されていた場合、SNMP の <b>mteTriggerFalling</b> トラップが生成されます。このトラップは、パケット損失が高い方のしきい値を超えて、その後、低い方のしきい値を再度下回るまで生成されません。</p>

	コマンドまたはアクション	目的
		<p><b>interval sampling-interval</b> : パケット損失の統計情報がポーリングされる頻度を指定します。これは、5 ~ 1440 分の 5 の倍数の値です。</p> <p><b>rising upper-threshold</b> : 高い方のしきい値を指定します。2つの間隔間のパケット損失がこのしきい値を超えると、SNMP の <code>mteTriggreRising</code> トラップが生成されます。このトラップは、パケット損失が下限しきい値を下回ってから、上限しきい値を上回るまで生成されません。</p>

## 維持する MIB データの設定

SNMP MIB 定義では、多くの場合、オブジェクトテーブルに任意の 32 ビットのインデックスを定義しています。MIB の実装では、多くの場合、MIB インデックスから内部データ構造へのマッピングを行います。このデータ構造は他のデータセットのキーになります。このような MIB テーブルでは、テーブル内に含まれるデータが、モデル化されている他の要素の識別子となっている場合があります。たとえば、ENTITY-MIB においては、`entPhysicalTable` のエントリは 31 ビットの値である `entPhysicalIndex` によってインデックス化されていますが、このエントリは `entPhysicalName` またはテーブル内の他のオブジェクトの組み合わせによって識別することができます。

一部の MIB テーブルのサイズが原因で、32 ビット MIB インデックスから、ネットワーク管理ステーションがエントリを識別できる他のデータへのすべてのマッピングを検出するには、膨大な処理が必要になります。そのため、プロセスの再開、リスタート、スイッチオーバー、デバイスのリロードを行っても、一部の MIB インデックスが維持される必要が生じます。

ENTITY-MIB の `entPhysicalTable` および CISCO-CLASS-BASED-QOS-MIB は、このような MIB の例であり、インデックス値を維持する必要が生じる場合があります。

また、CISCO-CLASS-BASED-QOS-MIB 統計情報のクエリ実行時のクエリの応答時間や CPU 使用率の問題により、サービスポリシーの統計情報はキャッシュしておくことが望ましいと言えます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<p>(任意) <code>snmp-server entityindex persist</code></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config)# snmp-server entityindex persist</pre>	ENTITY-MIB データの固定ストレージ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 2	(任意) <b>snmp-server mibs cbqosmib persist</b> 例 :  RP/0/RP0/cpu 0: router(config)# <b>snmp-server mibs cbqosmib persist</b>	CISCO-CLASS-BASED-QOS-MIB データの固定ストレージをイネーブルにします。
ステップ 3	(任意) <b>snmp-server cbqosmib cache refresh time time</b> 例 :  RP/0/RP0/cpu 0: router(config)# <b>snmp-server mibs cbqosmib cache refresh time 45</b>	QoS MIB のキャッシュをイネーブルにして、キャッシュのリフレッシュ時間を設定します。
ステップ 4	(任意) <b>snmp-server cbqosmib cache service-policy count count</b> 例 :  RP/0/RP0/cpu 0: router(config)# <b>snmp-server mibs cbqosmib cache service-policy count 50</b>	QoS MIB のキャッシュをイネーブルにして、キャッシュするサービスポリシーの数に制限を設けます。
ステップ 5	<b>snmp-server ifindex persist</b> 例 :  RP/0/RP0/cpu 0: router(config)# <b>snmp-server ifindex persist</b>	すべての簡易ネットワーク管理プロトコル (SNMP) インターフェイスで、ifIndex パーシステンスをグローバルにイネーブルにします。

## インターフェイスのサブセットに対する linkUp および linkDown トラップの設定

トラップを設定するインターフェイスを表すための正規表現を指定することで、同時に多数のインターフェイスに対して linkUp および linkDown トラップをイネーブルまたはディセーブルにすることができます。

### 始める前に

SNMP が設定されていること。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	

	コマンドまたはアクション	目的
ステップ 2	<p><b>snmp-server interface subset <i>subset-number</i> regular-expression <i>expression</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config)# snmp-server interface subset 10 regular-expression "^Gig[a-zA-Z]+[0-9/]+\."</pre> <p>RP/0/RP0/cpu 0: router(config-snmp-if-subset)#</p>	<p>正規表現で識別されたインターフェイスに対し、snmp-server インターフェイスモードを開始します。</p> <p>subset-number 引数は、インターフェイスのセットを識別し、インターフェイスが複数のサブセットに含まれている場合は、そのサブセットのプライオリティも割り当てます。数値が小さいほどプライオリティが高く、そのコンフィギュレーションは数値が大きいインターフェイスサブセットよりも優先されます。</p> <p>expression 引数は二重引用符で囲んで入力する必要があります。</p> <p>正規表現の詳細については、の「<i>Understanding Regular Expressions, Special Characters, and Patterns</i>」モジュールを参照してください。</p>
ステップ 3	<p><b>notification linkupdown disable</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-snmp-if-subset)# notification linkupdown disable</pre>	<p>設定しているすべてのインターフェイスに対して linkUp および linkDown トラップをディセーブルにします。ディセーブルにしたインターフェイスをイネーブルにするには、このコマンドの <b>no</b> 形式を使用します。</p>
ステップ 4	<b>commit</b>	
ステップ 5	<p>(任意) <b>show snmp interface notification subset <i>subset-number</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp interface notification subset 10</pre>	<p>サブセットのプライオリティで識別されたすべてのインターフェイスについて、linkUp および linkDown 通知のステータスを表示します。</p>
ステップ 6	<p>(任意) <b>show snmp interface notification regular-expression <i>expression</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp interface notification regular-expression "^Gig[a-zA-Z]+[0-9/]+\."</pre>	<p>正規表現で識別されたすべてのインターフェイスについて、linkUp および linkDown 通知のステータスを表示します。</p>

	コマンドまたはアクション	目的
ステップ7	<p>(任意) <b>show snmp interface notification type interface-path-id</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router# show snmp interface notification tengige 0/11/0/0.10</pre>	指定されたインターフェイスについて、linkUp および linkDown 通知のステータスを表示します。





## 第 6 章

# 定期的な MIB データの収集および転送の設定

このマニュアルでは、選択された MIB データをルータから指定されたネットワーク管理システム (NMS) に定期的に転送する方法について説明します。定期的な MIB データ収集と転送機能は、バルク統計情報とも呼ばれます。

- [定期的な MIB データの収集および転送の前提条件](#) (81 ページ)
- [定期的な MIB データの収集および転送に関する情報](#) (81 ページ)
- [定期的な MIB データの収集および転送の設定方法](#) (83 ページ)
- [定期的な MIB データの収集および転送：例](#) (90 ページ)

## 定期的な MIB データの収集および転送の前提条件

定期的な MIB データ収集と転送を使用するには、管理情報の簡易ネットワーク管理プロトコル (SNMP) モデルに精通している必要があります。また、ネットワークデバイスでモニタする MIB 情報や、モニタ対象である MIB オブジェクトの OID またはオブジェクト名を知っている必要があります。

## 定期的な MIB データの収集および転送に関する情報

### SNMP のオブジェクトとインスタンス

SNMP 管理情報のタイプ (またはクラス) をオブジェクトと呼びます。管理情報のタイプの特定のインスタンスをオブジェクト インスタンス (または SNMP 変数) と呼びます。バルク統計情報収集を設定するには、バルク統計情報オブジェクト リストを使用してモニタするオブジェクトタイプと、バルク統計情報スキーマを使用して収集するオブジェクトの特定のインスタンスを指定する必要があります。

オブジェクト識別子 (OID) と呼ばれる一連の番号を使用すると、MIB、MIB テーブル、MIB オブジェクト、オブジェクトのインデックスをすべて指定できます。OID は、バルク統計情報

オブジェクトリスト（一般的なオブジェクト用）とバルク統計情報スキーマ（特定のオブジェクトインスタンス用）の両方のバルク統計情報収集の設定に使用されます。

## バルク統計情報オブジェクトリスト

ポーリング対象の MIB オブジェクトをグループ化するには、1 つまたは複数のオブジェクトリストを作成する必要があります。バルク統計情報オブジェクトリストは、同じ MIB インデックスを共有する、ユーザ指定の MIB オブジェクトのセットです。オブジェクトリストは、指定した名前によって識別されます。名前付きのバルク統計情報オブジェクトリストを使用すると、異なるバルク統計情報スキーマで同じ設定を再利用できます。

オブジェクトリストのオブジェクトはすべて、同じ MIB インデックスを共有する必要があります。ただし、オブジェクトが同じ MIB 内に存在したり、同じ MIB テーブルに属する必要はありません。たとえば、ifInOctets と CISCO-IF-EXTENSION-MIB オブジェクトを同じスキーマでグループ化することが可能です。これは、両方のオブジェクトに対して含まれているテーブルが ifIndex によって指標付けされるためです。

## バルク統計情報スキーマ

定期的な MIB データの収集および転送のメカニズムに対するデータの選択には、次の情報を含むスキーマの定義が必要です。

- オブジェクトリストの名前。
- 指定されたオブジェクトリスト内で取得する必要があるオブジェクトのインスタンス（ワイルドカードを使用して定義された特定のインスタンスまたは一連のインスタンス）。
- 指定したインスタンスに対して必要なサンプリングの頻度（ポーリング間隔）。デフォルトのポーリング間隔は 5 分です。

バルク統計情報スキーマも、指定した名前によって識別されます。この名前は、転送オプションを設定する際に使用されます。

## バルク統計情報転送オプション

収集するデータを設定した後、収集した全データを使用して単一の仮想ファイル（VFile またはバルク統計情報ファイル）が作成されます。このファイルは、FTP または TFTP を使用してネットワーク管理ステーションに転送できます。このファイルの転送頻度を指定できます。デフォルトの転送間隔は、30 分に 1 回です。何らかの理由でプライマリネットワーク管理ステーションに転送できない場合に使用されるセカンダリ宛先を設定することもできます。

転送間隔の値は、ローカルのバルク統計情報ファイルの収集期間（収集間隔）でもあります。収集期間が終了すると、そのバルク統計情報ファイルは凍結し、データを格納するためにローカルのバルク統計情報ファイルが新たに作成されます。その後、凍結したバルク統計情報ファイルは指定した宛先に転送されます。

デフォルトでは、ローカルのバルク統計情報ファイルは、ネットワーク管理ステーションに正常に転送された後に削除されます。

## 定期的な MIB データの収集および転送の利点

定期的な MIB データの収集および転送（バルク統計情報機能）では、バルク ファイル MIB（CISCO-BULK-FILE-MIB.my）と同じ機能の多くを使用できますが、重要な利点があります。主な利点は、この機能が CLI を使用して設定でき、外部のモニタリングアプリケーションが不要なことです。

定期的な MIB データの収集および転送では、バルク統計情報ファイルの保存は主に（揮発性または永続的な）ローカルストレージが十分にある中規模からハイエンドのプラットフォームをターゲットとしています。バルク統計情報ファイルをローカルに保存すると、一時的なネットワーク停止時にデータ損失を最小限に抑えられます。

この機能にはバルク ファイル MIB よりも強力なデータ選択機能があるため、異なるテーブルの MIB オブジェクトをデータ グループ（オブジェクト リスト）にグループ化することも可能です。また、この機能はより柔軟性のあるインスタンス選択メカニズムを備えています。このメカニズムでは、アプリケーションは MIB テーブル全体を取得するように制限されていません。

## 定期的な MIB データの収集および転送の設定方法

### バルク統計情報オブジェクト リストの設定

定期的な MIB データの収集および転送のメカニズムを設定する場合の最初の手順は、1つまたは複数のオブジェクト リストを設定することです。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>snmp-server mib bulkstat object-list</b> <i>list-name</i>  例： <pre>snmp-server mib bulkstat object-list ifMib</pre>	SNMP バルク統計情報オブジェクト リストを定義し、バルク統計情報オブジェクト リスト コンフィギュレーション モードを開始します。
ステップ 3	<b>add {oid   object-name}</b>  例： <pre>RP/0/RP0/cpu 0: router(config-bulk-objects)# add 1.3.6.1.2.1.2.2.1.11</pre>	MIB オブジェクトをバルク統計情報オブジェクト リストに追加します。モニタ対象の全オブジェクトがこのリストに追加されるまで、必要に応じて繰り返します。

	コマンドまたはアクション	目的
	<pre>RP/0/RP0/cpu 0: router(config-bulk-objects)# add ifAdminStatus RP/0/RP0/cpu 0: router(config-bulk-objects)# add ifDescr</pre>	<p>(注) バルク統計情報オブジェクトリスト内のオブジェクトはすべて、同じ MIB インデックスによって指標付けされる必要があります。ただし、オブジェクトリスト内のオブジェクトが同じ MIB または MIB テーブルに属する必要はありません。</p> <p>OID ではなくオブジェクト名を指定すると (add コマンドを使用)、<b>show snmp mib object</b> コマンドの出力で示されるマッピングのあるオブジェクト名だけを使用できます。</p>
ステップ 4	<b>commit</b>	

## バルク統計情報スキーマの設定

定期的な MIB データの収集および転送を設定する場合の 2 つめの手順は、1 つまたは複数のスキーマを設定することです。

### 始める前に

スキーマで使用されるバルク統計情報オブジェクト リストを定義する必要があります。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>snmp-server mib bulkstat schema</b> <i>schema-name</i> 例 : <pre>RP/0/RP0/cpu 0: router(config)# snmp-server mib bulkstat schema intE0 RP/0/RP0/cpu 0: router(config-bulk-sc)#</pre>	バルク統計情報スキーマを指定し、バルク統計情報スキーマ モードを開始します。
ステップ 3	<b>object-list</b> <i>list-name</i> 例 :	このスキーマに含めるバルク統計情報オブジェクトリストを指定します。スキーマ

	コマンドまたはアクション	目的
	<pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# object-list ifMib</pre>	<p>マごとにオブジェクトリストを1つだけ指定してください。複数の <b>object-list</b> コマンドが実行されると、より新しいコマンドによって先行のコマンドが上書きされます。</p>
<p><b>ステップ 4</b></p>	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <li>• <b>instance exact</b> { <b>interface</b> <i>interface-id</i> [<b>sub-if</b>]   <b>oid</b> <i>oid</i>}</li> <li>• <b>instance wild</b> { <b>interface</b> <i>interface-id</i> [<b>sub-if</b>]   <b>oid</b> <i>oid</i>}</li> <li>• <b>instance range</b> <b>start</b> <i>oid</i> <b>end</b> <i>oid</i></li> <li>• <b>instance repetition</b> <i>oid</i> <b>max</b> <i>repeat-number</i></li> </ul> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# instance wild oid 1</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# instance exact interface TenGigE 0/1.25</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# instance range start 1 end 2</pre> <p>または</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# instance repetition 1 max 4</pre>	<p>このスキーマにおけるオブジェクトのインスタンス情報を指定します。</p> <ul style="list-style-type: none"> <li>• <b>instance exact</b> コマンドは、指定されたインスタンスが完全な OID であることを示しています (オブジェクトリストに追加されている場合)。</li> <li>• <b>instance wild</b> コマンドは、指定した OID のすべてのサブインデックスがこのスキーマに属することを示しています。 <b>wild</b> キーワードを使用すると、部分的に「ワイルドカードを使用した」インスタンスを指定できます。</li> <li>• <b>instance range</b> コマンドは、データを収集するインスタンスの範囲を示します。</li> <li>• <b>instance repetition</b> コマンドは、MIB オブジェクトの特定の数のインスタンスに対して繰り返すデータ収集を示します。</li> </ul> <p>(注) 1つのスキーマに設定できる <b>instance</b> コマンドは1つだけです。複数の <b>instance</b> コマンドが実行されると、新しいコマンドによって先行のコマンドが上書きされます。</p>
<p><b>ステップ 5</b></p>	<p><b>poll-interval</b> <i>minutes</i></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-sc)# poll-interval 10</pre>	<p>このスキーマで指定されたオブジェクトインスタンスからデータを収集する頻度を分単位で設定します。デフォルトは、5分に1回です。有効範囲は1～20000です。</p>
<p><b>ステップ 6</b></p>	<p><b>commit</b></p>	

## バルク統計情報転送オプションの設定

定期的な MIB データの収集および転送を設定する最後の手順は、転送オプションを設定することです。収集された MIB データは、VFile（仮想ファイル。このマニュアル内ではバルク統計情報ファイルとも呼ばれている）と呼ばれるローカルファイルのようなエンティティに格納されます。このファイルは、ユーザが指定した間隔でリモートのネットワーク管理ステーションに転送できます。

### 始める前に

バルク統計情報オブジェクトリストとバルク統計情報スキーマは、バルク統計情報転送オプションを設定する前に定義する必要があります。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>snmp-server mib bulkstat transfer-id</b> <i>transfer-id</i>  例： RP/0/RP0/cpu 0: router(config)# snmp-server mib bulkstat transfer bulkstat1	転送設定を名前（ <i>transfer-id</i> 引数）で識別し、バルク統計情報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 3	<b>buffer-size</b> <i>bytes</i>  例： RP/0/RP0/cpu 0: router(config-bulk-tr)# buffersize 3072	（任意）バルク統計情報データファイルの最大サイズをバイト単位で指定します。有効範囲は 1024 ~ 2147483647 バイトです。デフォルトのバッファサイズは 2048 バイトです。  （注） 転送間隔時間が切れる前に、バルク統計情報ファイルの最大バッファサイズに到達した場合、追加で受信したすべてのデータが削除されます。この動作を修正するために、ポーリング頻度を減らしたり、バルク統計情報バッファのサイズを増やせます。
ステップ 4	例：	（任意）バルク統計情報データファイル（VFile）の形式を指定します。デフォルトは <code>schemaASCII</code> です。

	コマンドまたはアクション	目的
		<p>(注) 転送を実行できるのは、<b>schemaASCII</b> (cdcSchemaASCII) 形式を使用した場合に限りです。<b>SchemaASCII</b>は、データ値を解析するためのパーサーフレンドリなヒントを含むヒト可読形式です。</p>
ステップ 5	<p><b>schema <i>schema-name</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# schema TenGigE 0/11/0/0 RP/0/RP0/cpu 0: router(config-bulk-tr)# schema TenGigE/0-CAR RP/0/RP0/cpu 0: router(config-bulk-tr)# schema TenGigE 0/11/0/0</pre>	<p>転送するバルク統計情報スキーマを指定します。必要に応じて、このコマンドを繰り返します。複数のスキーマを単一の転送設定に関連付けることができます。収集された全データが単一のバルク データ ファイル (VFile) に保存されます。</p>
ステップ 6	<p><b>transfer-interval <i>minutes</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# transfer-interval 20</pre>	<p>(任意) バルク統計情報ファイルの転送頻度を分単位で指定します。デフォルト値は、30 分に 1 回です。転送間隔は、収集間隔と同じです。</p>
ステップ 7	<p><b>url <i>primary url</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# url primary ftp://user:password@host/folder/bulkstat1</pre>	<p>バルク統計情報データファイルを転送するネットワーク管理システム (ホスト) と転送に使用するプロトコルを指定します。この宛先は、ユニフォームリソースロケータ (URL) として指定されます。FTP または TFTP は、バルク統計情報ファイルの転送に使用できます。</p>
ステップ 8	<p><b>url <i>secondary url</i></b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# url secondary tftp://10.1.0.1/tftpboot/user/bulkstat1</pre>	<p>(任意) プライマリ ロケーションファイルへの転送イベントで使用する、バックアップの転送先とプロトコルを指定します。FTP または TFTP は、バルク統計情報ファイルの転送に使用できます。</p>
ステップ 9	<p><b>retry <i>number</i></b></p> <p>例 :</p>	<p>(任意) 送信の再試行回数を指定します。デフォルト値は 0 (つまり、再試行しない) です。バルク統計情報ファ</p>

	コマンドまたはアクション	目的
	<pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# retry 1</pre>	<p>イルを送信しようとして失敗した場合に、このコマンドを使用してファイルの再送信を試みるように設定できます。</p> <p>1回の再試行に含まれるのは、まず1番目の宛先に試行され、転送に失敗すると、次に2番目の場所に試行される動作です。たとえば、再試行値が1の場合、まずプライマリ URL に試行された後でセカンダリ URL に、そして再びプライマリ URL へその後で再びセカンダリ URL に試行されます。有効範囲は0～100です。</p> <p>すべての再試行に失敗すると、次の通常の転送は、設定された転送間隔の時間が経過した後に実行されます。</p>
ステップ 10	<p><b>retain minutes</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# retain 60</pre>	<p>(任意) 収集間隔と転送試行の設定が完了したら、バルク統計情報ファイルをシステムメモリに保存する期間を分単位で指定します。デフォルト値は0です。ゼロ (0) は、転送が試行された直後にファイルが削除されることを示します。有効範囲は0～20000です。</p> <p>(注) <b>retry</b> コマンドを使用する場合、保持間隔を0よりも大きく設定する必要があります。再試行の間隔は、保持間隔を再試行回数で割ったものです。たとえば、<b>retain 10</b> および <b>retry 2</b> が設定されている場合は、2回の再試行が5分ごとに1回試行されます。したがって、<b>retain 0</b> に設定した場合、再試行は行われません。</p>
ステップ 11	<p><b>enable</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# enable</pre>	<p>この設定でバルク統計情報のデータ収集と転送のプロセスを開始します。</p> <ul style="list-style-type: none"> <li>このアクションを正常に実行させるには、オブジェクト数が0以外</li> </ul>

	コマンドまたはアクション	目的
		<p>のスキーマを最低 1 つは設定する必要があります。</p> <ul style="list-style-type: none"> <li>定期的なデータ収集とファイル転送が開始されるのは、このコマンドが設定されている場合だけです。逆に、<b>no enable</b> コマンドが設定されていると、収集プロセスが停止します。後続の <b>enable</b> では、動作が再び開始されます。</li> <li><b>enable</b> コマンドを使用して収集プロセスが開始されるたびに、新しいバルク統計情報ファイルにデータが収集されます。<b>no enable</b> コマンドを使用すると、収集したデータの転送プロセスがただちに開始されます（つまり、既存のバルク統計情報ファイルが指定した管理ステーションに転送されます）。</li> </ul>
<p>ステップ 12</p>	<p><b>commit minutes</b></p> <p>例 :</p> <pre>RP/0/RP0/cpu 0: router(config-bulk-tr)# retain 60</pre>	<p>転送間隔時間が切れる前に、バルク統計情報ファイルの最大バッファサイズに到達した場合でも、転送動作は開始されますが、ファイルがいっぱいになった後で受信したバルク統計情報のデータで転送される前のものは削除されます。この動作を修正するために、ポーリング頻度を減らしたり、バルク統計情報バッファのサイズを増やせます。</p> <p><b>retain 0</b> に設定した場合、再試行は行われません。これは、再試行の間隔が <b>retain</b> の値を <b>retry</b> の値で割ったものであるためです。たとえば、<b>retain 10</b> および <b>retry 2</b> が設定されている場合は、再試行が 5 分ごとに 1 回行われます。したがって、<b>retry</b> コマンドを設定した場合、<b>retain</b> コマンドにも適切な値を設定する必要があります。</p>

## 定期的な MIB データの収集および転送 : 例

次に、定期的な MIB データ収集および転送を設定する例を示します。

```
snmp-server mib bulkstat object-list cempo
add cempMemPoolName
add cempMemPoolType
!
snmp-server mib bulkstat schema cempWild
object-list cempo
instance wild oid 8695772
poll-interval 1
!
snmp-server mib bulkstat schema cempRepeat
object-list cempo
instance repetition 8695772.1 max 4294967295
poll-interval 1
!
snmp-server mib bulkstat transfer-id cempt1
enable
url primary tftp://223.255.254.254/auto/tftp-sjc-users3/username/dumpdcm
schema cempWild
schema cempRepeat
transfer-interval 2
!
```

次に、バルク統計情報ファイルの内容の例を示します。

```
Schema-def cempt1.cempWild "%u, %s, %s, %d" Epochtime instanceoid
1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempWild: 1339491515, 8695772.1, processor, 2
cempt1.cempWild: 1339491515, 8695772.2, reserved, 11
cempt1.cempWild: 1339491515, 8695772.3, image, 12
cempt1.cempWild: 1339491575, 8695772.1, processor, 2
cempt1.cempWild: 1339491575, 8695772.2, reserved, 11
cempt1.cempWild: 1339491575, 8695772.3, image, 12
Schema-def cempt1.cempRepeat "%u, %s, %s, %d" Epochtime instanceoid
1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempRepeat: 1339491515, 8695772.1, processor, 2
cempt1.cempRepeat: 1339491515, 8695772.2, reserved, 11
cempt1.cempRepeat: 1339491515, 8695772.3, image, 12
cempt1.cempRepeat: 1339491515, 26932192.1, processor, 2
cempt1.cempRepeat: 1339491515, 26932192.2, reserved, 11
cempt1.cempRepeat: 1339491515, 26932192.3, image, 12
cempt1.cempRepeat: 1339491515, 35271015.1, processor, 2
cempt1.cempRepeat: 1339491515, 35271015.2, reserved, 11
cempt1.cempRepeat: 1339491515, 35271015.3, image, 12
cempt1.cempRepeat: 1339491515, 36631989.1, processor, 2
cempt1.cempRepeat: 1339491515, 36631989.2, reserved, 11
cempt1.cempRepeat: 1339491515, 36631989.3, image, 12
cempt1.cempRepeat: 1339491515, 52690955.1, processor, 2
cempt1.cempRepeat: 1339491515, 52690955.2, reserved, 11
cempt1.cempRepeat: 1339491515, 52690955.3, image, 12
```



## 第 7 章

# Cisco Discovery Protocol の設定

*Cisco Discovery Protocol* (CDP) は、ルータ、ブリッジ、アクセス サーバ、コミュニケーションサーバ、スイッチを含め、シスコ製のあらゆる機器で動作する、メディアにもプロトコルにも依存しないプロトコルです。CDP を使用して、デバイスに直接接続しているすべてのシスコの装置の情報を表示することができます。

- [CDP の実装の前提条件 \(91 ページ\)](#)
- [CDP の実装について \(91 ページ\)](#)
- [CDP の実装方法 : Cisco IOS XR ソフトウェア \(93 ページ\)](#)
- [CDP の実装の設定例 \(96 ページ\)](#)

## CDP の実装の前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

## CDP の実装について

CDP の主な用途は、隣接するデバイスのプロトコルアドレスを取得し、そのデバイスのプラットフォームを検出することです。また、CDP を使用して、ルータが使用するインターフェイスの情報を表示できます。CDP はメディアとプロトコルに依存せず、ルータ、ブリッジ、アクセスサーバ、スイッチなど、シスコ製のすべての機器で実行できます。

SNMP と CDP MIB を併用すると、ネットワーク管理アプリケーションは、隣接するデバイスのデバイス タイプと SNMP エージェントアドレスを認識し、それらのデバイスに SNMP クエリーを送信できます。CDP は CISCO-CDP-MIB を使用します。

CDP は、LAN、フレーム リレー、ATM 物理メディアなど、サブネットワーク アクセス プロトコル (SNAP) をサポートするすべてのメディアで実行されます。CDP の動作はデータリンク層上に限定されます。そのため、異なるネットワーク層プロトコルをサポートする2つのシステムが、相互について認識できます。

CDP 用に設定された各デバイスからマルチキャストアドレスに対してメッセージが定期的に送信されます。このメッセージはアドバタイズメントと呼ばれます。各デバイスは、SNMP メッセージを受信できるアドレスを少なくとも1つアドバタイズします。アドバタイズには、存続可能時間（保持時間）や情報も含まれています。これは、受信側のデバイスが CDP 情報を破棄せずに保持する時間の長さを示します。各デバイスは、他のデバイスから送信される定期的な CDP メッセージを待ち受けます。これは、隣接するデバイスについて認識し、メディアに対するインターフェイスがアップまたはダウンした場合を判断するためです。

CDP Version-2 (CDPv2) は、このプロトコルの最新リリースで、より高度なデバイス追跡機能を備えています。たとえば、より高速なエラー追跡が可能なレポートメカニズムなどが含まれるため、コストがかかるダウンタイムを減らすことができます。レポートされるエラーメッセージは、コンソールまたはロギングサーバに送信でき、また接続ポートの一致していないネイティブ VLAN ID (IEEE 802.1Q) インスタンス、および接続デバイス間の一致していないポートデュプレックスステートをカバーできます。

CDPv2 **show** コマンドを実行すると、隣接するデバイスの VLAN トランキンングプロトコル (VTP) 管理ドメインとデュプレックスモード、CDP 関連のカウンタ、および接続ポートの VLAN ID に関する詳細な情報が出力されます。

Type-Length-Value (TLV) フィールドは、CDP アドバタイズメントに埋め込まれる情報ブロックです。次の表に、CDP アドバタイズメントの TLV 定義の概要を示します。

表 4: CDPv2 の Type-Length-Value 定義

TLV	定義
デバイス ID TLV	文字列形式のデバイス名を識別します。
アドレス TLV	受信デバイスと送信デバイス両方のネットワークアドレスリストを含めます。
ポート ID TLV	CDP パッケージが送信されるポートを指定します。
機能 TLV	スイッチなど、デバイスタイプの形式でデバイスの機能を説明します。
バージョン TLV	デバイスが実行しているソフトウェアリリースバージョンに関する情報を含めます。
プラットフォーム TLV	Cisco 4500 など、デバイスのハードウェアプラットフォーム名を記述します。
VTP 管理ドメイン TLV	システムの設定済み VTP 管理ドメイン名の文字列をアドバタイズします。隣接するネットワークノードの VTP ドメインコンフィギュレーションを確認するために、ネットワークオペレータが使用します。

TLV	定義
ネイティブ VLAN TLV	インターフェイス上の非タグ付きパケットに対して想定される VLAN をインターフェイス単位で示します。CDP はインターフェイスのネイティブ VLAN を認識します。この機能を実装するのは、IEEE 802.1Q プロトコルをサポートするインターフェイスの場合だけです。
全二重/半二重 TLV	CDP ブロードキャストインターフェイスのステータス（デュプレックス設定）を示します。ネットワークオペレータが、隣接するネットワーク要素間の接続の問題を診断するときに使用します。

## CDP の実装方法 : Cisco IOS XR ソフトウェア

### CDP のイネーブル化

CDP をイネーブルにするには、まずルータで CDP をグローバルにイネーブルにしてから、インターフェイス単位で CDP をイネーブルにする必要があります。ここでは、ルータ上で CDP をグローバルにイネーブルにし、次にインターフェイスで CDP をイネーブルにする方法について説明します。

#### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>cdp</b> 例 :  RP/0/RP0/cpu 0: router# cdp	CDP をグローバルにイネーブルにします。
ステップ 3	<b>interface type interface-path-id</b> 例 :  RP/0/RP0/cpu 0: router# int TenGigE 0/11/0/0	インターフェイスコンフィギュレーション モードを開始します。
ステップ 4	<b>cdp</b> 例 :  RP/0/RP0/cpu 0: router(config-if)# int TenGigE 0/11/0/0	特定のインターフェイス上で CDP をイネーブルにします。
ステップ 5	<b>commit</b>	

## CDP デフォルト設定の変更

ここでは、デフォルトのバージョン、保持時間の設定、およびタイマーの設定を変更する方法について説明します。



(注) コマンドは任意の順序で入力できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure</b>	
ステップ 2	<b>cdp advertise v1</b> 例 :  RP/0/RP0/cpu 0: router# cdp advertise v1	近接装置との通信にバージョン 1 (CDPv1) だけを使用するように CDP を設定します。 <ul style="list-style-type: none"> <li>デフォルトでは、CDP をイネーブるにすると、ルータから CDPv2 パケットが送信されます。相手先のデバイスで CDPv2 パケットが処理されない場合は、CDPv1 パケットも送受信されます。</li> <li>この例では、ルータが CDPv1 パケットだけを送受信するよう設定されています。</li> </ul>
ステップ 3	<b>cdp holdtime seconds</b> 例 :  RP/0/RP0/cpu 0: router# cdp holdtime 30	ネットワーク デバイスがルータから送信された CDP パケットを受信した後、破棄するまで保持する時間の長さを指定します。 <ul style="list-style-type: none"> <li>デフォルトでは、CDP がイネーブるの場合、受信ネットワーク デバイスは、CDP パケットを廃棄するまでに 180 秒間保持します。</li> </ul> (注) CDP 保持時間は、CDP の送信間隔 ( <b>cdp timer</b> コマンドを使用して設定します) よりも長い秒数に設定する必要があります。 <ul style="list-style-type: none"> <li>この例では、<i>seconds</i> 引数の保持時間の値が 30 に設定されます。</li> </ul>

	コマンドまたはアクション	目的
ステップ 4	<b>cdp timer seconds</b> 例 : <pre>RP/0/RP0/cpu 0: router# cdp timer 20</pre>	CDP アップデート パケットが送信される頻度を指定します。 <ul style="list-style-type: none"> <li>デフォルトでは、CDP がイネーブされている場合、CDP 更新パケットが 60 秒ごとに 1 回の頻度で送信されます。</li> <li>(注) タイマーの設定時間が短いほど、CDP 更新の送信頻度が高くなります。</li> <li>この例では、CDP 更新パケットが 20 秒ごとに 1 回の頻度で送信されるように設定されます。</li> </ul>
ステップ 5	<b>commit</b>	
ステップ 6	(任意) <b>show cdp</b> 例 : <pre>RP/0/RP0/cpu 0: router# show cdp</pre>	グローバルな CDP 情報を表示します。 出力には、ルータで実行中の CDP バージョン、保持時間の設定、およびタイマー設定が表示されます。

## CDP のモニタリング

このタスクでは、CDP をモニタする例を示します。



(注) コマンドは任意の順序で入力できます。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>show cdp entry</b> {*   <i>entry-name</i> } [ <b>protocol</b>   <b>version</b> ] 例 : <pre>RP/0/RP0/CPU0:router# show cdp entry *</pre>	CDP を使用して検出された特定の近接装置またはすべての近接装置に関する情報を表示します。
ステップ 2	<b>show cdp interface</b> [ <i>type interface-path-id</i>   <b>location node-id</b> ] 例 :	CDP をイネーブにしたインターフェイスに関する情報を表示します。

	コマンドまたはアクション	目的
	RP/0/RP0/CPU0:router# show cdp interface pos 0/0/0/1	
ステップ 3	<b>show cdp neighbors</b> [ <i>type interface-path-id</i>   <b>location node-id</b> ] [ <b>detail</b> ]  例 :  RP/0/RP0/CPU0:router# show cdp neighbors	CDP を使用して検出された隣接するデバイスに関する詳細情報を表示します。
ステップ 4	<b>show cdp traffic</b> [ <b>location node-id</b> ]  例 :  RP/0/RP0/CPU0:router# show cdp traffic	CDP を使用してデバイス間で収集されたトラフィックに関する情報を表示します。

## CDP の実装の設定例

### CDP のイネーブル化 : 例

次の例に、CDP をグローバルに設定し、イーサネット インターフェイス TenGigE 0/11/0/0 上で CDP をイネーブルにする方法を示します。

```
cdp
interface 0/11/0/0
cdp
```

### グローバル CDP 設定の変更 : 例

次に、グローバル CDP 設定を変更する例を示します。この例では、タイマー設定は 20 秒、ホールド時間は 30 秒、および隣接デバイスとの通信に使用する CDP のバージョンは CDPv1 に設定されています。

```
cdp timer 20
cdp holdtime 30
cdp advertise v1
```

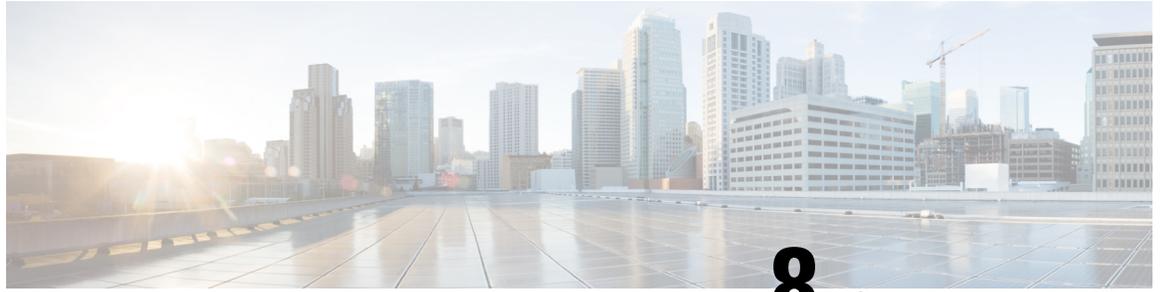
次の例に、**show cdp** コマンドを使用して CDP グローバル設定を確認する方法を示します。

```
RP/0/RP0/cpu 0: router# show cdp

Global CDP information:
Sending CDP packets every 20 seconds
Sending a holdtime value of 30 seconds
```

```
Sending CDPv2 advertisements is not enabled
```





## 第 8 章

# スマート ライセンスを使用したライセンスの設定

- [スマート ライセンスとは \(99 ページ\)](#)
- [スマート ライセンスの仕組み \(100 ページ\)](#)
- [スマート ライセンスの導入オプション \(101 ページ\)](#)
- [Call Home について \(103 ページ\)](#)
- [サポート対象の柔軟な消費モデル ライセンス \(103 ページ\)](#)
- [スマート ライセンスを使用したライセンスの設定 \(104 ページ\)](#)
- [スマート ライセンスのワークフロー \(109 ページ\)](#)
- [ライセンス、製品インスタンス、および登録トークン \(109 ページ\)](#)

## スマート ライセンスとは

スマート ライセンスは、時間のかかる手動のライセンス タスクを自動化できるクラウドベースのソフトウェアライセンス管理ソリューションです。このソリューションを使用すると、ライセンスのステータスとソフトウェアの使用傾向を簡単に追跡できます。

シスコ スマート ライセンスを使うと次の 3 つのコア機能が簡素化されます。

- **購入**：ネットワークにインストールされているソフトウェアを、製品アクティベーションキー (PAK) を指定せずに自動的に登録できます。
- **管理**：ライセンスエンタイトルメントの有効化を自動的に追跡できます。また、すべてのノードにライセンスファイルをインストールする必要はありません。組織構造に合わせたライセンスプール (ライセンスの論理的なグループ) を作成できます。Smart Licensing には、すべての Cisco ソフトウェア ライセンスを 1 つの一元化された Web サイトで管理できる集中型ポータルである Cisco Smart Software Manager が用意されています。Cisco Smart Software Manager に詳細が記載されています。
- **レポート**：スマートライセンスでは、ポータルを使用することで、購入したライセンスとネットワークに展開された製品を統合して表示できます。このデータを使用すると、購入の意思決定を実際の使用状況に基づいてより適切に行うことができます。

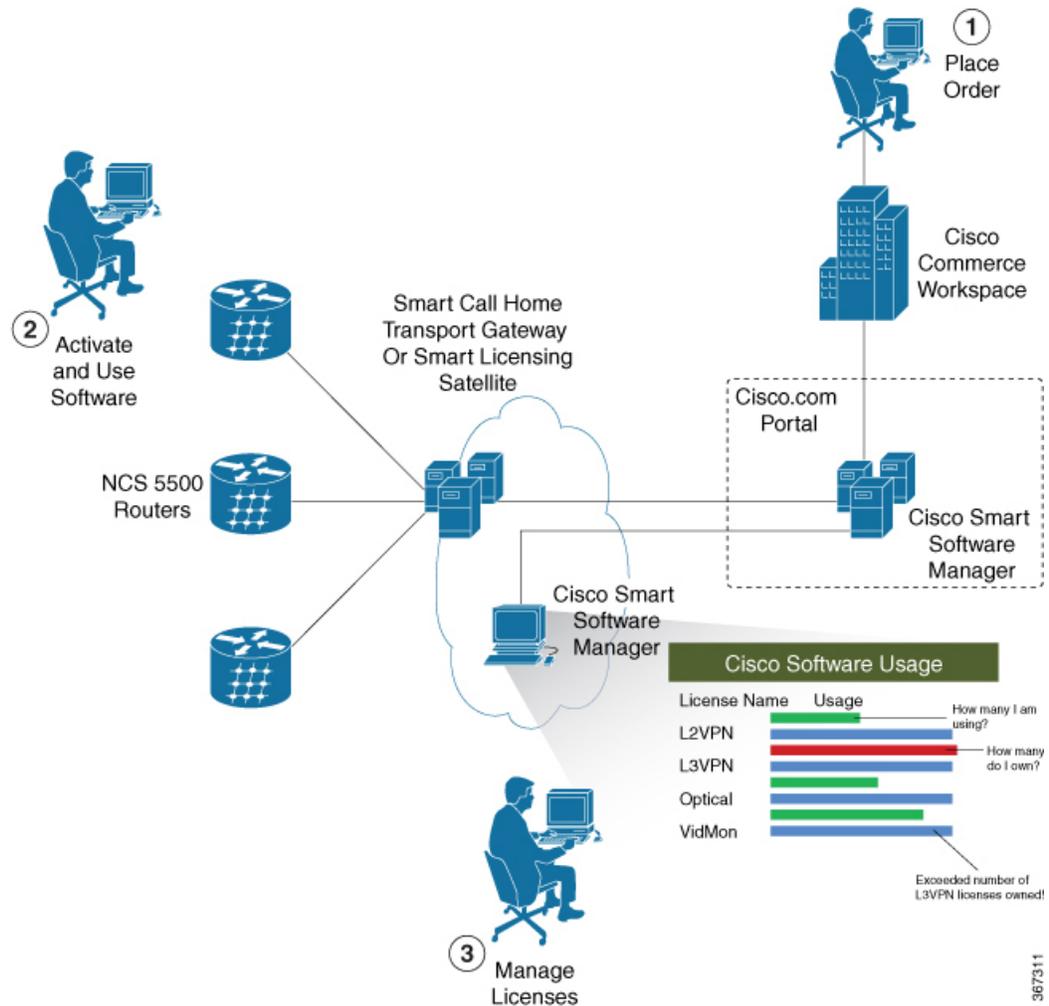


- (注)
- デフォルトでは、スマートライセンスが有効になっています。
  - 非消費モデルのスマートライセンスのみがサポートされています。

## スマートライセンスの仕組み

次の図で示しているように、スマートライセンスの動作モデルを示す3つのステップがあります。

図 4: スマートライセンス : 例



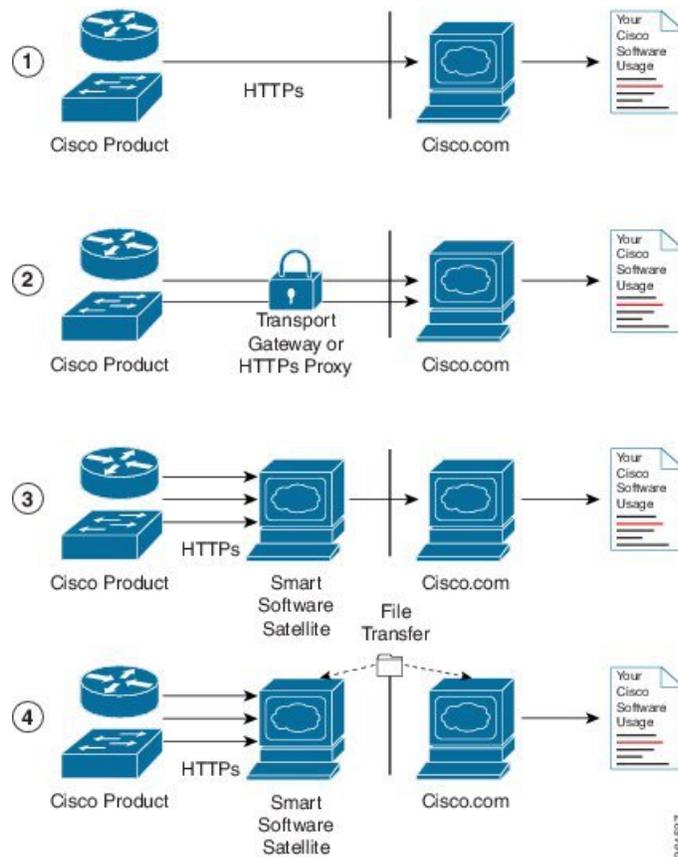
- **スマートライセンスの設定:** スマートライセンスを注文します。ライセンスはCisco.comポータルで管理します。[Smart Software Managerポータルでのスマートライセンスの使用](#)およびアクセスについて規定した利用規約に同意します。

- **スマートライセンスの有効化と使用**：手順に従ってスマートライセンスを有効にします。「スマートライセンスのワークフロー」に手順が記載されています。  
スマートライセンスを有効にしたら、次のいずれかのオプションを使用して通信できます。
  - **Smart Call Home**：Smart Call Home 機能は、スマートライセンスが有効になった後に自動的に設定されます。Smart Call Home は、シスコのライセンス サービスとの通信用メディアとしてスマートライセンスで使用されます。Call Home 機能を使用すると、シスコ製品が定期的に call-home を実行し、ソフトウェアの使用状況情報を監査および調整できるようになります。この情報により、シスコは、インストールベースを効率的に追跡し、それらを継続的に維持および稼働させることができます。また、お客様の手を煩わせることなく、サービスの提供および契約更新に関するサポートをより効果的に行えるようになります。Smart Call Home 機能の詳細については、『[Smart Call Home Deployment Guide](#)』を参照してください。
  - **スマートライセンス サテライト**：スマートライセンス サテライト オプションは、スマートライセンスの使用状況を統合および管理するために使用できるオンプレミス コレクタを提供し、[Cisco.com](#) でのシスコ ライセンス サービスとの通信を促進します。
  - **ライセンスの管理およびレポート**：Smart Software Manager ポータルで、ソフトウェア全体の使用状況に関するレポートを管理および表示できます。

## スマート ライセンスの導入オプション

次の図は、スマート ライセンスの導入に使用できるさまざまなオプションを示しています。

図 5: スマートライセンスの導入オプション



1. **ダイレクトクラウドアクセス**：ダイレクトクラウドアクセスによる導入方法では、シスコ製品からインターネット経由で Cisco.com（シスコライセンス サービス）に使用状況情報を直接送信します。導入に必要な追加コンポーネントはありません。
2. **HTTP プロキシによるダイレクトクラウドアクセス**：HTTP プロキシによるダイレクトクラウドアクセスの導入方法では、シスコ製品から、プロキシサーバ（Smart Call Home Transport Gateway または市販のプロキシ（Apache など）のいずれか）を介して、インターネット経由で <http://www.cisco.com> のシスコライセンス サービスに使用状況情報を送信します。
3. **接続状態のオンプレミス コレクタを介した間接アクセス**：接続状態のオンプレミス コレクタを介した間接アクセスによる導入方法では、シスコ製品から、現地のライセンス認証局として機能するローカルに接続されたコレクタに使用状況情報を送信します。データベースの同期を保つため、周期的に情報が交換されます。
4. **非接続状態のオンプレミス コレクタを介した間接アクセス**：非接続状態のオンプレミス コレクタを介した間接アクセスによる導入方法では、シスコ製品から、現地のライセンス認証局として機能するローカルの接続が解除されたコレクタに使用状況情報を送信します。データベースの同期を保つため、不定期に（月に 1 回など）人による読み取りが可能な情報の交換が実施されます。

オプション 1 と 2 には簡単な導入オプションを、オプション 3 と 4 にはセキュアな環境展開オプションを提供します。オプション 3 および 4 には、スマート ソフトウェア サテライトによるサポートが提供されます。

シスコ製品とシスコ ライセンス サービスとの間の通信は、Smart Call Home ソフトウェアが対応します。

## Call Home について

Call Home では、重要なシステム ポリシーに関する電子メール ベースおよび http/https ベースの通知を提供します。ポケットベル サービスや XML ベースの自動化された解析アプリケーションとの互換性のために、さまざまなメッセージフォーマットが用意されています。この機能を使用して、ネットワーク サポート エンジニアにポケットベルで連絡したり、ネットワーク オペレーションセンターに電子メールを送信したりできます。また、Cisco Smart Call Home サービスを使用して TAC のケースを生成できます。Call Home 機能では、診断情報および環境の障害とイベントに関する情報が含まれるアラート メッセージを配信できます。

Call Home 機能では、複数の受信者（Call Home 宛先プロファイルと呼びます）にアラートを配信できます。各プロファイルには、設定可能なメッセージフォーマットとコンテンツ カテゴリが含まれます。Cisco TAC へアラートを送信するための宛先が事前に定義されていますが、独自の宛先プロファイルを定義することもできます。メッセージを送信するように Call Home を設定すると、適切な CLI show コマンドが実行され、そのコマンドの出力がメッセージに添付されます。Call Home メッセージは次のフォーマットで配信されます。

- 1 または 2 行で障害を説明する、ポケットベルや印刷レポートに適したショートテキストフォーマット。
- 詳細な情報を十分に書式が整えられたメッセージで提供する、ユーザが読むのに適したフルテキストフォーマット。
- Extensible Markup Language (XML) と Adaptive Messaging Language (AML) XML schema definition (XSD) を使用する、コンピュータで読み取り可能な XML フォーマット。AML XSD は Cisco.com Web サイト (<http://www.cisco.com/>) で公開されています。XML フォーマットでは、シスコの TAC との通信が可能になります。

## サポート対象の柔軟な消費モデル ライセンス

スマート ライセンスでは、柔軟な消費ライセンス モデルを使用します。このライセンス モデルは、設定されているポートの容量に基づきます。柔軟な消費モデルライセンスをサポートするシャーシを購入する場合は、柔軟な消費モデルライセンスを設定してライセンス機能を有効にする必要があります。柔軟な消費モデルライセンスは、システムのすべてのポートの使用状況を毎日確認し、結果を Cisco.com の Smart Licensing Manager に報告します。

ハードウェアまたはソフトウェアで柔軟な消費モデルライセンスを有効にするには、グローバル コンフィギュレーション モードでライセンスの **smart flexible-consumption enable** コマンド

を使用します。ハードウェアまたはソフトウェアで柔軟な消費モデルライセンスを無効にするには、グローバルコンフィギュレーションモードで **no license smart flexible-consumption enable** コマンドを使用します。

次の表に、NCS 560 のさまざまな柔軟な消費モデル ライセンスの消費パターンを示します。

表 5: 柔軟な消費ライセンス モデルの使用パターン

柔軟な消費モデル ライセンス	消費パターン
ESS-AC-100G-RTU-1	ルータに固定シャーシがある場合、ライセンスの消費チェックはシャーシで実行されます。
ADV-AC-100G-RTU-1	ルータに固定シャーシがある場合、ライセンスの消費チェックはシャーシで実行されます。

## スマート ライセンスを使用したライセンスの設定

### デバイスの登録とアクティブ化

スマートライセンスのコンポーネントは、<Cisco-8000-image> にパッケージ化されています。Smart Call Home の設定に必要な HTTP クライアントは、<filename> にパッケージ化されています。ここで説明する手順を使用してデバイスを登録およびアクティブ化し、バーチャルアカウントに関連付けます。

デバイスを登録およびアクティブ化するには、次を行う必要があります。

- Cisco Smart Software Manager ポータルから、登録トークンを生成します。
- 登録トークンを使用して、CLI でデバイスを登録します。

#### ポータルから製品登録トークンを生成する

ライセンスを追加する製品を購入しておく必要があります。製品を購入すると、Cisco Smart Software Manager ポータルに対するユーザ名とパスワードが提供されます。このポータルから製品インスタンス登録トークンを生成できます。

1. [スマートソフトウェアライセンシング](#)で Cisco Smart software Manager にログインします。
2. [Inventory] メニューの下で、[General] タブをクリックします。
3. [New Token] をクリックして製品登録トークンを生成します。

デバイスの登録およびアクティブ化に使用される新しいトークン値をコピーして、デバイスをバーチャルアカウントに関連付けます。



(注) このトークンは、290 日間有効です。

### CLI での新しい製品の登録

CLI で、登録トークンを使用してデバイスをアクティブ化します。

```
RP/0/RP0/cpu 0: router#license smart register idtoken token_ID
RP/0/RP0/cpu 0: router#commit
```

登録が成功すると、デバイスでアイデンティティ証明書を受信します。この証明書はデバイスに保存され、それ以降のシスコとのすべての通信で自動的に使用されます。スマートライセンスは、シスコへの登録情報を 290 日ごとに自動的に更新します。登録が失敗した場合、エラーがログに記録されます。また、ライセンスの使用状況データが収集され、毎月レポートが送信されます。必要に応じて、機密情報（ホスト名、ユーザ名、パスワードなど）が使用状況レポートから除外されるように、Smart Call Home 設定を構成できます。

## スマート ライセンス設定の確認

スマートライセンスを有効にした後、**show** コマンドを使用して、デフォルトのスマートライセンス設定を確認できます。問題が検出された場合は、さらに設定を行う前に修正を行ってください。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>show license status</b> 例： <pre>RP/0//CPU0:router#show license status</pre>	スマートライセンスのコンプライアンスステータスを表示します。以下は、表示される可能性があるステータスです。 <ul style="list-style-type: none"> <li>• <b>Waiting</b> : デバイスがライセンス権限付与要求を行った後の初期状態を示します。デバイスはシスコとの通信を確立し、Cisco License Manager に正常に登録されます。</li> <li>• <b>Authorized</b> : デバイスが Cisco License Manager と通信できること、およびライセンス権限付与の要求を開始する権限を持っていることを示します。</li> <li>• <b>Out-Of-Compliance</b> : 1 つ以上のライセンスがコンプライアンス違反であ</li> </ul>

	コマンドまたはアクション	目的
		<p>ることを示します。追加ライセンスを購入する必要があります。</p> <ul style="list-style-type: none"> <li>• <b>Eval Period</b> : スマートライセンスが評価期間が間もなく終了することを示します。Cisco Licensing Manager にデバイスを登録する必要があります。登録しない場合ライセンスが期限切れになります。</li> </ul> <p>(注) スマートライセンスの評価期限切れを警告するメッセージが1時間ごとに繰り返しコンソールに表示されますが、デバイスでは機能上の影響は見られません。これらの繰り返しメッセージを停止するには、新しい登録トークンを使用してデバイスを再度登録する必要があります。</p> <ul style="list-style-type: none"> <li>• <b>Disabled</b> : スマートライセンスが無効になっていることを示します。</li> <li>• <b>Invalid</b> : データベースに存在しないため、シスコが権限付与タグを認識しないことを示します。</li> </ul>
ステップ 2	<b>show license all</b> 例 : <pre>RP/0//CPU0:router#show license all</pre>	使用中のすべての権限を表示します。さらに、関連付けられているライセンス証明書、コンプライアンスステータス、udi、およびその他の詳細が表示されます。
ステップ 3	<b>show license status</b> 例 : <pre>RP/0//CPU0:router#show license status</pre>	使用中のすべての権限のステータスを表示します。
ステップ 4	<b>show license udi</b> 例 : <pre>RP/0//CPU0:router#show license udi</pre>	UDI 情報を表示します。

	コマンドまたはアクション	目的
ステップ 5	<b>show license summary</b> 例： RP/0//CPU0:router#show license summary	使用中のすべての権限の概要を表示します。
ステップ 6	<b>show license platform summary</b> 例： RP/0//CPU0:router#show license platform summary	登録ステータスを表示し、一般的なライセンス モデルまたはボルテックス ライセンス モデルでの必須、詳細、およびトラッキング ライセンスの消費に関する詳細情報を提供します。
ステップ 7	<b>show license platform detail</b> 例： RP/0//CPU0:router#show license platform detail	一般的なモデルとボルテックス モデルの両方で、特定のプラットフォームで消費される可能性のあるライセンスの詳細を表示します。また、特定のライセンスの現在および次回の消費数も表示されます。アクティブなモデルの情報（ライセンスモデルが一般的かボルテックスか）を表示します。
ステップ 8	<b>show call-home smart-licensing statistics</b>	Smart Call Home を使用したスマート ライセンス マネージャとシスコ バックエンド間の通信について統計情報を表示します。通信が失敗したりドロップされる場合は、Call Home の設定に誤りがないか確認してください。

次に、**show call-home smart-licensing statistics** コマンドの出力例を示します。

```
RP/0//CPU0:router#show call-home smart-licensing statistics
Success: Successfully sent and response received.
Failed : Failed to send or response indicated error occurred.
Inqueue: In queue waiting to be sent.
Dropped: Dropped due to incorrect call-home configuration.

Msg Subtype      Success Failed  Inqueue Dropped Last-sent (GMT-07:00)
-----
ENTITLEMENT      2         0         0         0    2014-04-24 18:24:34
REGISTRATION      1         0         0         0    2014-04-25 03:53:57
ACKNOWLEDGEMENT  1         0         0         0    2014-04-23 19:21:21
RENEW             1         0         0         0    2014-04-23 19:21:11
DEREGISTRATION   1         0         0         0    2014-04-25 03:31:35
```

## スマート ライセンス登録の更新

一般に、登録は6ヵ月ごとに自動更新されます。登録をオンデマンドの手動更新にするには、このオプションを使用します。これにより、次の登録更新サイクルまで6ヵ月待機する代わりに、このコマンドを実行してライセンスのステータスをすぐに確認できます。

## 始める前に

スマートライセンスを更新するには、次の条件が満たされていることを確認する必要があります。

- スマートライセンスが有効になっている。
- デバイスが登録されている。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>license smart renew {auth   id}</b> 例： <pre>RP/0/RP0/CPU0:#license smart renew auth Tue Apr 22 09:12:37.086 PST  license smart renew auth: Authorization process is in progress. Please check the syslog for the authorization status and result.</pre>	Cisco Smart Licensing を使用して ID または認証を更新します。ID 証明書の更新が失敗した場合は、製品インスタンスが未確認状態に移行し、評価期間の消費が開始されます。  (注) 認証期間は、スマートライセンスシステムによって 30 日ごとに更新されます。ライセンスが「承認済み」または「コンプライアンス違反」(OOC) にある限り、認証期間が更新されます。猶予期間は、承認期間が過ぎると開始されます。猶予期間中、または猶予期間が「期限切れ」状態になると、システムは引き続き認証期間の更新を試行します。再試行に成功すると、新しい認証期間が開始されます。

## スマートライセンスの登録解除

デバイスがインベントリから移された場合、再導入のために別の場所に出荷された場合、または返品許可 (RMA) プロセスを使用して交換のためにシスコに返送された場合は、登録解除オプションを使用してデバイスの登録をキャンセルできます。デバイスの登録をキャンセルするには、次の手順を実行します。

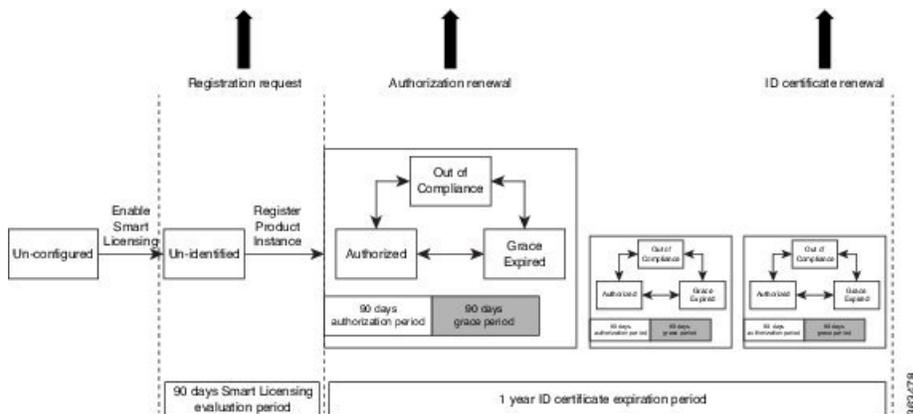
## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>license smart deregister</b> 例：	デバイスの登録をキャンセルし、30 日間の評価モードに送信します。プラットフォーム

コマンドまたはアクション	目的
<pre>RP/0//CPU0 #license smart deregister license smart deregister: Success License command "license smart deregister " completed successfully.</pre>	<p>フォームのすべてのスマートライセンス資格と証明書が削除されます。</p> <p>(注) 製品インスタンスはシスコのライセンスクラウドサービスから登録解除されましたが、スマートライセンスは引き続き有効になっています。</p>

## スマートライセンスのワークフロー

スマートライセンスのワークフローを、次のフローチャートに示します。



## ライセンス、製品インスタンス、および登録トークン

### ライセンス

ライセンスは、すべてのシスコ製品に必要です。シスコ製品のすべてのライセンスは、製品によって異なる次の2つのタイプのいずれかになります。

- 永久ライセンス：期限切れがないライセンス。
- 有効期限付きライセンス：一定期間（1年、3年、または購入した任意の期間）後に自動的に期限切れになるライセンス。

すべての製品ライセンスは、バーチャルアカウントに存在します。

### 製品インスタンス (Product Instances)

製品インスタンスとは、製品インスタンス登録トークン（または登録トークン）を使用して登録される、一意のデバイス ID (UDI) を使用した個々のデバイスです。1つの登録トークンで1つの製品のインスタンスをいくつでも登録できます。各製品インスタンスで、同じバーチャルアカウント内に存在する1つ以上のライセンスを使用できます。製品インスタンスは、特定

の更新期間中 Cisco Smart Software Manager サーバに定期的に接続する必要があります。製品インスタンスが接続に失敗するとライセンス不足であるとマークされますが、そのライセンスは引き続き使用されます。製品インスタンスを削除すると、そのライセンスがリリースされ、バーチャルアカウント内で使用可能になります。

#### 製品インスタンスの登録トークン

製品は、登録するまで登録トークンが必要です。登録トークンは、エンタープライズアカウントに関連付けられた製品インスタンス登録トークンテーブルに格納されます。製品を登録すると登録トークンは不要になるため、そのテーブルから取り消したり、削除したりしても影響はありません。登録トークンの有効日数は、1～365 日間に設定できます。

## 仮想アカウント

スマートライセンスでは、Smart Software Manager ポータル内で複数のライセンス プールまたはバーチャルアカウントを作成できます。[Virtual Accounts] オプションを使用して、組織の1つのセクションが別のセクションのライセンスを使用できないように、ライセンスをコストセンターに関連付けられた個別のバンドルに集約できます。たとえば、会社を地域で分ける場合、地域ごとにバーチャルアカウントを作成して、その地域のライセンスおよび製品インスタンスを保持できます。

注文プロセス中に別のライセンスを指定しない限り、新しいライセンスと製品インスタンスはすべて Smart Software Manager のデフォルトのバーチャルアカウントに配置されます。デフォルトのアカウントに配置されると、必要に応じて他のアカウントに転送できるようになります。これは必要なアクセス権限がある場合に限りです。

Smart Software Manager ポータル (<https://tools.cisco.com/rhodui/index>) を使用して、ライセンスプールを作成するか、ライセンスを転送します。

## コンプライアンス レポート

スマートライセンス契約の条件に規定されているとおり、インベントリとライセンスに関するコンプライアンスデータが含まれたレポートが定期的に自動送信されます。これらのレポートには、次の3つのいずれかの形式が使用されます。

- **定期的な記録**：この記録は、特定の時点で保存された関連性のあるインベントリデータを使用して、定期的（設定可能）に生成されます。このレポートは、アーカイブ用として Cisco Cloud 内に保存されます。
- **手動による記録**：この記録は、特定の時点で保存された関連性のあるインベントリデータを使用して手動で生成できます。このレポートは、アーカイブ用として Cisco Cloud 内に保存されます。
- **コンプライアンス違反の警告レポート**：このレポートは、ライセンスコンプライアンスに関するイベントが発生したときに、自動または手動で生成されます。このレポートには完全なインベントリ データは含まれておらず、特定のソフトウェアライセンスに対して不足している権限のみが記載されています。

これらのレポートは、Smart Software Manager ポータル (<https://tools.cisco.com/rhodui/index>) から確認できます。

