



## 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 \(1 ページ\)](#)
- [管理機能の設定方法 \(2 ページ\)](#)
- [管理機能の設定例 \(3 ページ\)](#)

## 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
```

