



# NETCONF エージェント

- [NETCONF エージェントについて \(1 ページ\)](#)
- [注意事項と制約事項 \(3 ページ\)](#)
- [NETCONF エージェントの構成 \(3 ページ\)](#)
- [NETCONF エージェントの使用 \(4 ページ\)](#)
- [NETCONF エージェントのトラブルシューティング \(8 ページ\)](#)

## NETCONF エージェントについて

Cisco NX-OS NETCONF エージェントは、クライアント要求とサーバー応答を XML でエンコードされた YANG モデルの形式でセキュアに転送するクライアント側インターフェイスです。

NETCONF エージェントは、候補構成機能をサポートしています。候補構成データストアは、候補構成と、実行構成を変更せずに行った変更を一時的に保持します。候補構成をコミットおよび確認する時には、候補構成でデバイスをいつアップデートするかを選択できます。

変更を確認しなかった場合、非永続的 NETCONF クライアントセッションを終了した場合、または変更をコミットした後にコミットをキャンセルした場合は、変更を確認しないと、システムタイマーがタイムアウトし、変更がロールバックされます。

永続トークンを使用して確認コミット操作を開始すると、NETCONF クライアントセッションは永続プロセスになります。永続プロセスでは、NETCONF クライアントセッションを終了しても自動ロールバックは呼び出されず、一致する永続トークンがなければ変更をロールバックできません。

Cisco NX-OS NETCONF は、次の構成機能をサポートします：

- 書き込み可能実行機能  
`urn:ietf:params:netconf:capability:writable-running:1.0`
- エラー時ロールバック機能  
`urn:ietf:params:netconf:capability:rollback-on-error:1.0`
- 候補構成機能  
`urn:ietf:params:netconf:capability:candidate:1.0`

- 検証機能

```
urn:ietf:params:netconf:capability:validate:1.1
```

- 確認済みコミット機能

```
urn:ietf:params:netconf:capability:confirmed-commit:1.1
```

新しいセッションが開始されると、NETCONF エージェントは<hello>メッセージを送信し、その機能をアドバタイズします。次の例では、NETCONF エージェントが<hello>メッセージをクライアントに送信します。

```
<?xml version="1.0" encoding="UTF-8"?>
<hello>

  <capabilities>

    <capability>urn:ietf:params:netconf:base:1.0</capability>
    <capability>urn:ietf:params:netconf:base:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:writable-running:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:rollback-on-error:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:candidate:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:validate:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:confirmed-commit:1.1</capability>

    <capability>http://cisco.com/rs/yang/cisco-nx-os-device?revision=2017-04-06&module=cisco-nx-os-device&deviations=cisco-nx-os-device&deviations</capability>

  </capabilities>

  <session-id>1438752697</session-id>
</hello>
```

Cisco NX-OS NETCONF エージェントは、次の NETCONF プロトコル操作をサポートしています。

- get
- get-config
- edit-config
- close-session
- kill-session

候補構成は、次の NETCONF プロトコル操作をサポートします。

- <source> または <target> としての候補構成の操作。
  - get-config
  - edit-config
  - copy-config
  - ロック
  - unlock
  - 検証

- <source> または <target> として候補構成を明示的に指定する必要のない候補構成の操作。
  - commit
  - cancel-commit
  - discard-changes



(注) delete-config 操作は許可されません。

## 注意事項と制約事項

NETCONF エージェントには、次の注意事項と制限事項があります。

- NETCONF は、RFC 6536 で指定されている拡張ロールベース アクセス コントロール (RBAC) をサポートしていません。「network-admin」ロールを持つユーザーのみが NETCONF エージェントへのアクセスを許可されます。
- NETCONF では、RPM がポート 830 でインストールされている必要があります。

## NETCONF エージェントの構成

NETCONF エージェントは、構成ファイル (/etc/mtx.conf) の [netconf] セクションで、次のオプションの構成パラメータをサポートします。

| パラメータ               | 説明  |
|---------------------|---|
| <b>idle_timeout</b> | <p>(オプション) アイドル状態のクライアントセッションが切断されるまでのタイムアウトを分単位で指定します。</p> <p>デフォルト値は 5 分です。</p> <p>値を 0 に設定するとタイムアウトが無効になります。</p> |
| <b>limit</b>        | <p>(オプション) 同時クライアントセッションの最大数を指定します。</p> <p>デフォルト値は 5 セッションです。</p> <p>指定できる範囲は 1 ~ 10 です。</p>                        |

次に、構成ファイルの [netconf] セクションの例を示します。

```
[netconf]
```

```
mtxadapter=/opt/mtx/lib/libmtxadapternetconf.1.0.1.so
idle_timeout=10
limit=1
```

変更した構成ファイルを有効にするには、CLI コマンド `[no] feature netconf` を使用して NETCONF エージェントを再起動し、無効にしてから再度有効にする必要があります。

## NETCONF エージェントの使用

### 一般的なコマンド

NETCONF エージェントは、コマンド `[no] feature netconf` で有効または無効にします。

### 候補構成データストアの初期化

候補構成は、実行構成の内容でのみ初期化できます。候補の構成データストアを初期化するには、`candidate` をターゲット、`running` を送信元として、SSH で Copy-Config 要求を送信します。

### 候補構成での読み取りと書き込みの実行

候補構成から読み取るには、`candidate` を送信元として、SSH で Get-Config 要求を送信します。

候補構成の内容を書き込むには、`candidate` を送信元として、SSH で Edit-Config 要求を送信します。

### NETCONF 候補構成ワークフロー

候補構成ワークフローは次のとおりです。

- 候補構成ファイルを編集します。
- 候補構成を検証します。
- 実行構成に変更をコミットします。

### 例：SSH セッション

この例では、SSH クライアントを使用してセッションを開始し、SSH クライアントを使用して Edit-Config および Get 要求を送信します。

```
client-host % ssh -s admin@172.19.193.152 -p 830 netconf
client-host % ssh -s admin@172.19.193.152 -p 830 netconf
User Access Verification
Password:
<?xml version="1.0" encoding="UTF-8"?>
<hello>
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
    <capability>urn:ietf:params:netconf:base:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:writable-running:1.0</capability>
```

```

    <capability>urn:ietf:params:netconf:capability:rollback-on-error:1.0</capability>

    <capability>urn:ietf:params:netconf:capability:candidate:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:validate:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:confirmed-commit:1.1</capability>

<capability>http://cisco.com/ns/yang/cisco-nx-os-device?revision=2017-04-06&module=cisco-nx-os-device&deviations=cisco-nx-os-device-deviations</capability>

    </capabilities>
    <session-id>1912037714</session-id>
</hello>
]]>]]><hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <capabilities>
        <capability>urn:ietf:params:netconf:base:1.1</capability>
    </capabilities>
</hello>
]]>]]>
#794
<rpc message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <System xmlns="http://cisco.com/ns/yang/cisco-nx-os-device">
                <bgp-items>
                    <inst-items>
                        <dom-items>
                            <Dom-list>
                                <name>default</name>
                                <rtrId>2.2.2.2</rtrId>
                            </Dom-list>
                        </dom-items>
                    </inst-items>
                </bgp-items>
            </System>
        </config>
    </edit-config>
</rpc>
##

#190
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

##

#511
<rpc message-id="109"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get-config>
    <source>
        <running/>
    </source>
    <filter type="subtree">
        <System xmlns="http://cisco.com/ns/yang/cisco-nx-os-device">
            <bgp-items>

```

```

        <inst-items>
          <dom-items>
            <Dom-list/>
          </dom-items>
        </inst-items>
      </bgp-items>
    </System>
  </filter>
</get-config>
</rpc>
##

#996
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="109"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <System>
      <bgp-items>
        <inst-items>
          <dom-items>
            <Dom-list>
              <name>default</name>
              <always>disabled</always>
              <bestPathIntvl>300</bestPathIntvl>
              <holdIntvl>180</holdIntvl>
              <kaIntvl>60</kaIntvl>
              <maxAsLimit>0</maxAsLimit>
              <pfxPeerTimeout>30</pfxPeerTimeout>
              <pfxPeerWaitTime>90</pfxPeerWaitTime>
              <reConnIntvl>60</reConnIntvl>
              <rtrId>2.2.2.2</rtrId>
            </Dom-list>
          </dom-items>
        </inst-items>
      </bgp-items>
    </System>
  </data>
</rpc-reply>
##

```

edit-config の操作属性は、指定した操作が実行される構成内のポイントを特定することにご注意ください。操作属性が指定されていない場合、構成は既存の構成データストアにマージされます。操作属性には、次の値を指定できます。

- create
- merge
- delete

次の例は、実行中の構成からインターフェイス Ethernet0/0 の構成を削除する方法を示しています。

```

xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <default-operation>none</default-operation>
  </config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">

```

```

<top xmlns="http://example.com/schema/1.2/config">
  <interface xc:operation="delete">
    <name>Ethernet0/0</name>
  </interface>
</top>
</config>
</edit-config>
</rpc>]]>]]>

```

## エラーメッセージ

要求がエラーになった場合、応答ペイロードにはエラーが含まれます。

### シスコが定義したエラー

シスコによって定義されているエラーは次のとおりです。

| シスコが定義したエラー                             | 説明   |
|---|--|
| unknown-error-cond (不明なエラー条件)           | 不明なエラーが発生しました。                                     |
| n-y-i                                   | 要求された操作はサポートされていません。<br>(not-yet-implemented、未実装)。 |
| namespace-not-found (名前空間が見つからない)       | 要求ペイロードでのエラー。                                      |
| namespace-already-exists (名前空間がすでに存在する) | 要求ペイロードでのエラー。                                      |
| object-not-found (オブジェクトが見つからない)        | 要求ペイロードでのエラー。                                      |
| object-not-container (オブジェクトはコンテナではない)  | 要求ペイロードでのエラー。                                      |
| object-not-property (オブジェクトはプロパティではない)  | 要求ペイロードでのエラー。                                      |
| no-property-in-object (オブジェクトにプロパティがない) | 要求ペイロードでのエラー。                                      |
| invalid-dn (無効な dn)                     | 内部エラーが発生しました。                                      |
| invalid-arg (無効な引数)                     | 内部エラーが発生しました。                                      |
| already-exists (すでに存在)                  | 要求ペイロードでのエラー。                                      |
| container-not-found (コンテナが見つからない)       | 要求ペイロードでのエラー。                                      |
| container-already-exists (コンテナはすでに存在する) | 要求ペイロードでのエラー。                                      |
| property-not-found (プロパティが見つからない)       | 要求ペイロードでのエラー。                                      |
| property-already-exists (プロパティはすでに存在する) | 要求ペイロードでのエラー。                                      |

|                        |               |
|------------------------|---------------|
| 形式異常                   | 要求ペイロードでのエラー。 |
| alloc-failed (割り当てに失敗) | 内部エラーが発生しました。 |
| sigint (秘密情報収集)        | 内部エラーが発生しました。 |
| not-initialized (未初期化) | 内部エラーが発生しました。 |
| inappropriate (不適切)    | 内部エラーが発生しました。 |

次に、無効な IP アドレス値を報告する NETCONF エラー応答ペイロードの例を示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="320" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <rpc-error>
    <error-type>Protocol</error-type>
    <error-tag>operation-failed</error-tag>
    <error-severity>Error</error-severity>
    <error-message xml:lang="en">Property Merge (set property) Failed: operation-failed
value=500.500.500.500</error-message>
    <error-path>/config/System/bgp-items/inst-items/dom-items/Dom-list/rtrId</error-path>
  </rpc-error>
</rpc-reply>
```

## NETCONF エージェントのトラブルシューティング

### 接続のトラブルシューティング

- クライアントシステムから、スイッチの管理ポートに **ping** を実行して、スイッチが到達可能であることを確認します。
  - スwitchの **bash** シェルで、**service netconf status** コマンドを実行してエージェントのステータスを確認します。
  - XML 管理インターフェイス (xmlagent と呼ばれる) というものがあります。これは NETCONF エージェントとはまったく異なりますが、よく混同されます。サーバーが正しい NETCONF メッセージで応答しない場合は、正しいポート 830 に接続していて、サーバーから正しい **<hello>** メッセージ (「NETCONF セッションの確立」セクションに示されているものと同様) を受信していることを確認します。
  - **/volatile/netconf-internal-log** ファイルの内容を表示することで、**Bash** シェルからの NETCONF エージェントのデバッグ情報を表示できます。**Bash** シェルを有効にするには、**feature bash** コマンドを使用します。**Bash** シェルを有効にした後、**run bash** コマンドを使用して **Bash** シェルを開始します。詳細については、このドキュメントの **Bash** の章を参照してください。
- 注** : **debug netconf** コマンドは、NETCONF エージェント動作のデバッグには使用できません。これらのデバッグコマンドは、NETCONF エージェント関連のログを出力しません。



## 翻訳について

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