



# raw ソケット トランスポート

この章は、次の項で構成されています。

- [raw ソケット トランスポートに関する情報 \(1 ページ\)](#)
- [前提条件 \(4 ページ\)](#)
- [注意事項と制約事項 \(5 ページ\)](#)
- [raw ソケット機能強化 \(5 ページ\)](#)
- [デフォルト設定 \(5 ページ\)](#)
- [raw ソケット トランスポートの設定 \(5 ページ\)](#)
- [raw ソケットキープアライブ設定 CLI \(12 ページ\)](#)
- [raw ソケットトランスポートに対する YANG データモデルのサポート \(12 ページ\)](#)
- [設定の確認 \(14 ページ\)](#)
- [設定例 \(14 ページ\)](#)

## raw ソケット トランスポートに関する情報

raw ソケットトランスポートは、ライフラインアプリケーションの IP ネットワークを介して、1 つのシリアルインターフェイスから別のシリアルインターフェイスに文字のストリームを転送します。

このドキュメントでは、IR1101 の raw ソケットトランスポートについて解説し、raw ソケットトランスポート コマンドに関する参照セクションを示します。

raw ソケットは、IP ネットワークを介してシリアルデータを転送するための方法です。この機能は、リモートターミナルの単位 (RTU) から遠隔監視制御・情報取得 (SCADA) データを転送するのに使用できます。このメソッドは、ブロックシリアルトンネル (BSTUN) プロトコルに代わるものです。

raw ソケットトランスポートは、トランスポートプロトコルとして TCP または UDP をサポートします。インターフェイスはいずれかのプロトコルを使用するように設定できますが、両方を同時に使用することはできません。TCP トランスポートは、データの確認応答と連続配信を必要とする制御アプリケーションなどのアプリケーションに適しています。回線 SEL リレーなど、遅延の影響を受けやすいアプリケーションに対して、UDP トランスポートは TCP よりも高速なシリアルデータ転送を提供します。

raw ソケット トランスポートでは、非同期シリアルインターフェイスに対して以下がサポートされています。

- 組み込みの自動 TCP 接続再試行メカニズムを装備したトランスポート プロトコルとしての TCP。
- 最大 32 の TCP セッション。
- サーバ、クライアント、またはその両方の組み合わせとしてのインターフェイス設定。
- サーバインターフェイス 1 つ、ただし複数のクライアントあり。
- VRF 認識。ルータは、バーチャルプライベート ネットワーク (VPN) 仮想ルーティングおよび転送 (VRF) インターフェイスを介して接続されたサーバ ホストに raw ソケット トランスポート トラフィックを送信できるようになります。

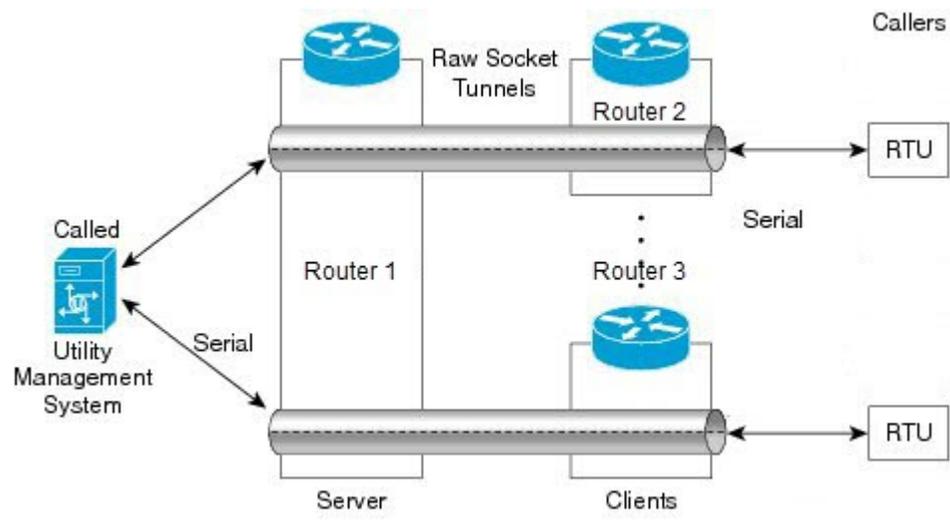
この項では、次のトピックについて取り上げます。

## TCP トランスポート

TCP raw ソケット トランスポートは、クライアント/サーバモデルを使用します。1 つの非同期シリアル回線で、最大 1 つのサーバと複数のクライアントを設定できます。クライアントモードでは、IR1101 は raw ソケット サーバ (他の IR1101 ルータまたはサードパーティ製デバイスが可能) に対して最大 32 の TCP セッションを開始できます。

次の図は、raw ソケットの TCP 設定の例を示しています。この例では、複数の IR1101 ルータを含む IP ネットワークを介して、RTU とライフライン管理システムの間でシリアルデータが転送されています。1 つの IR1101 ルータ (ルータ 1) は raw ソケット サーバとして機能し、raw ソケット クライアントとして設定された他の IR1101 ルータ (ルータ 2 とルータ 3) からの TCP 接続要求をリッスンします。

raw ソケット クライアントは RTU からシリアルデータのストリームを受信し、そのバッファにこのデータを蓄積してから、ユーザ指定の packetsize 基準に基づいてデータをパケットに格納します。raw ソケット クライアントは、raw ソケット サーバとの TCP 接続を開始し、IP ネットワークを介してパケット化データを raw ソケット サーバに送信します。これにより、パケットからシリアルデータが取得され、それがシリアルインターフェイスに送信され、ライフライン管理システムに送信されます。



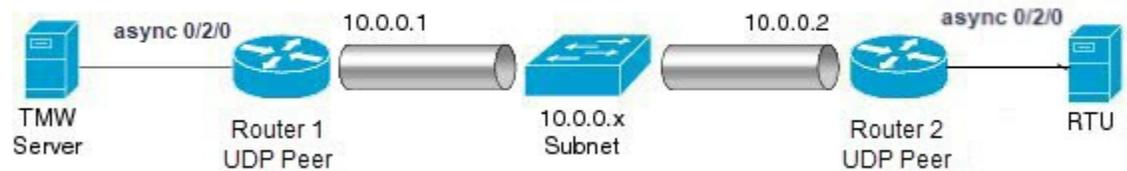
(注) ルータのシリアルリンク インターフェイスをサーバとして設定する場合、インターフェイスのピアはクライアントルータのシリアルリンクインターフェイスであり、その逆も同様です。

## UDP トランスポート

UDP トランスポートは、ピアツーピア モデルを使用しています。非同期シリアル回線には複数の UDP 接続を設定できます。

次の図は、raw ソケットの UDP 設定の例を示しています。この例では、シリアルデータは、raw ソケット UDP 対向として設定されている 2 つのルータ（ルータ 1 は IR1101、ルータ 2 は IR807）を含む IP ネットワークを介して RTU とライフライン管理システム間で転送されています。

この例では、raw ソケット UDP 対向は RTU からシリアルデータのストリームを受信し、そのバッファにこのデータを蓄積してから、ユーザ指定の packetsize に基づいてデータをパケットに格納します。raw ソケット UDP 対向は、IP ネットワークを介してパケット化データをもう一方の端末の raw ソケットピアに送信します。これにより、パケットからシリアルデータが取得され、それがシリアルインターフェイスに送信され、ライフライン管理システムに送信されます。



## シリアル データ 処理

デフォルトのシリアルプロトコル、非同期通信プロトコルが使用されている場合、raw ソケット対向が受信するシリアルデータ ストリームは、次の基準に基づいてパケット化されます。

- **Packet length** : IR1101 がシリアルデータを対向に送信する引き金となるパケット長を指定できます。IR1101 がバッファ内にこの量のデータを収集すると、蓄積されたデータをパケット化して raw ソケット対向に転送します。
- **Packet-timer value** : パケット タイマーは、IR1101 がストリーム内の次の文字を受信するまで待機する時間を指定します。パケット タイマーの期限終了までに文字が受信されない場合、IR1101 がバッファ内で累積したデータはパケット化され、raw ソケット対向に転送されます。
- **Special character** : IR1101 がバッファに蓄積されたデータをパケット化して raw ソケット対向に送信する引き金となる文字を指定することができます。特殊文字 (CR/LF など) を受信すると、IR1101 は蓄積されたデータをパケット化して raw ソケット対向に送信します。

処理オプションの設定については、[6 ページの「共通の raw ソケット ライン オプションの設定」の手順](#)を参照してください。

## VRF 対応 raw ソケット

VRF 対応 raw ソケット トランスポート 機能を使用すると、VRF を使用して raw ソケット トラフィックを分離し、シリアルデータを効率的に管理および制御することができます。VRF を設定したら、raw ソケット トランスポート 用に設定されたシリアル インターフェイスを VRF に関連付けることができます。設定例については、[raw ソケット VRF \(16 ページ\)](#) を参照してください。

## 前提条件

使用するネットワーク デバイスとインターフェイスを含め、ネットワーク内で raw ソケット トラフィックが転送される方法、ルータがシリアルデータをパケット化する方法、VRF を使用するかどうかを決定します。

## 注意事項と制約事項

通常、UDP トラフィックはネットワーク内のファイアウォールによってブロックされます。ネットワークにこうしたファイアウォールがある場合は、raw ソケット UDP トラフィックを許可するようにピンホールを設定してください。

## raw ソケット 機能強化

この機能強化により、ユーザーは書き込みソケットで利用できる最大再試行回数を入力できます。再試行回数の範囲は 1 ~ 1000 回です。デフォルトの再試行回数は 10 回です。この機能に対応するために、**raw-socket tcp max-retries <1-1000>** という新しい CLI が作成されました。<1-1000> は最大再試行回数です。

## デフォルト設定

機能	デフォルト設定
raw ソケット トランスポート	無効
パケット長	パケット長は設定されていません。
シリアルプロトコル	非同期通信プロトコル
パケット タイムアウト	15 ミリ秒
特殊文字	特殊文字は設定されていません。
raw ソケット モード	Best-effort モードはオフになっており、IR1101 ではサポートされていません。
TCP アイドル タイムアウト	5 分

## raw ソケット トランスポートの設定

この項では、次のトピックについて取り上げます。

## シリアルインターフェイスで raw ソケット トランスポート を有効化

IR1101 ルータで raw ソケット トランスポート を有効にするには、最初に非同期シリアル ポート を有効にし、そのポートに対して raw ソケット TCP または UDP カプセル化を有効にする必要があります。

### 始める前に

IR1101 のシリアル ポート が使用可能かどうかを確認します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure terminal</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>interface async0/slot /port</b>	async slot/port のインターフェイスのコマンドモードを開始します。
ステップ 3	<b>no ip address</b>	インターフェイスで IP 処理を無効にします。
ステップ 4	次のいずれかを実行します。 <ul style="list-style-type: none"> <li>• <b>encapsulation raw-tcp</b></li> <li>• <b>encapsulation raw-udp</b></li> </ul>	シリアル ポートの raw ソケット TCP カプセル化または UDP カプセル化を有効にします。

### 例：シリアルポートの有効化

次に、シリアルポート 0/2/0 を有効にし、そのポートで raw ソケット TCP カプセル化を有効にする例を示します。

```
router# configure terminal
router(config)# interface async0/2/0
router(config-if)# no ip address
router(config-if)# encapsulation raw-tcp
router(config-if)# exit
```

## 共通の raw ソケット ライン オプション の設定

回線上のすべての接続に共通するオプションを設定できます。共通のオプションは、TCP と UDP の両方に適用されます。

### 始める前に

[シリアルインターフェイスで raw ソケット トランスポート を有効化 \(6 ページ\)](#) の説明に従って、raw ソケット トランスポート を有効にします。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure terminal</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>line 0/slot /port</b>	シリアル スロット / ポートのライン コマンド モードを開始します。
ステップ 3	<b>raw-socket packet-length length</b>	IR1101 がシリアル データを対向に送信する引き金となるパケット長を指定します。IR1101 がバッファ内にこの大量のデータを蓄積すると、蓄積されたデータをパケット化して raw ソケット対向に転送します。  <i>length</i> : 2 ~ 1400 バイト。  デフォルトでは、パケット長トリガーは無効になっています。
ステップ 4	<b>raw-socket packet-timer timeout</b>	IR1101 がストリーム内の次の文字を受信するまで待機する最大時間をミリ秒単位で指定します。パケットタイマーの期限終了までに文字が受信されない場合、累積したデータはパケット化され、raw ソケット対向に転送されます。  <i>timeout</i> : 3 ~ 1000 ミリ秒。  デフォルトは 15 ミリ秒です。
ステップ 5	<b>raw-socket spec-char ascii_char</b>	バッファに蓄積されたデータをパケット化して raw ソケット対向に送信するように IR1101 をトリガーする文字を指定します。  <i>ascii_char</i> : 0 ~ 255。  デフォルトでは、特殊文字トリガーは無効になっています。

## 次のタスク

デフォルト値に戻すには、これらのコマンドの **no** 形式を使用します。

## 例：共通の Raw ソケット回線オプション

```
router# configure terminal
router(config)# line 0/2/0
router(config-line)# raw-socket packet-length 32
router(config-line)# raw-socket packet-timer 500
router(config-line)# raw-socket special-char 3
```

## raw ソケット TCP の設定

raw ソケット TCP カプセル化を有効にした後、TCP サーバまたはクライアントを設定します。

### raw ソケット TCP サーバの設定

始める前に

シリアルインターフェイスで raw ソケット トランスポート を有効化 (6 ページ) の説明に従って、そのポートのシリアル ポートおよび raw ソケット TCP カプセル化を有効にします。

手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure terminal</b>	コンフィギュレーション モードに入ります。
ステップ 2	<b>line 0/slot /port</b>	シリアル スロット / ポートのライン コマンド モードを開始します。
ステップ 3	<b>raw-socket tcp server port [ip_address ]</b>	非同期回線インターフェイスの raw ソケット トランスポート TCP サーバを起動します。raw ソケット サーバ モードでは、IR1101 は raw ソケット クライアントからの着信接続要求をリスンします。  <i>port</i> : サーバがリスンするポート番号。  <i>ip_address</i> : (任意) サーバが接続要求をリスンするローカル IP アドレス。
ステップ 4	<b>raw-socket tcp idle-timeout session_timeout</b>	非同期回線インターフェイスの raw ソケット トランスポート TCP セッション タイムアウトを設定します。この間隔でクライアントとサーバの間でデータが転送されない場合、TCP セッションは終了します。その後、クライアントはサーバとの TCP セッションの再確立を自動的に試みます。  このタイムアウト設定は、この特定の回線のすべての raw ソケット トランスポート TCP セッションに適用されます。  <i>session_timeout</i> : 設定されているセッションアイドルタイムアウト (分)。デフォルトは 5 分です。

次のタスク

raw ソケット TCP サーバを削除するには、**no raw-socket tcp server** コマンドを使用します。

## 例：Raw ソケット TCP サーバー

次の例は、非同期シリアル回線の raw ソケット TCP サーバを設定する方法を示しています。TCP サーバは、ローカル ポート 4000 およびローカル IP アドレス 10.0.0.1 で TCP クライアント接続要求を待ち受けます。raw ソケット TCP サーバと TCP クライアントのいずれかの間で 10 分間データが交換されない場合、TCP セッションが終了し、raw ソケット クライアントは raw ソケット サーバとのセッションの再確立を試行します。

```
router# configure terminal

router(config)# line 0/2/0
router(config-line)# raw-socket tcp server 4000 10.0.0.1
router(config-line)# raw-socket tcp idle-timeout 10
router(config-line)# exit
router(config)#
```

## raw ソケット TCP クライアントの設定

## 始める前に

シリアルインターフェイスで raw ソケットトランスポートを有効化 (6 ページ) の説明に従って、そのポートのシリアルポートおよび raw ソケット TCP カプセル化を有効にします。

## 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure terminal</b>	コンフィギュレーション モードに入ります。
ステップ 2	<b>line 0/slot /port</b>	シリアル スロット / ポートのライン コマンド モードを開始します。
ステップ 3	<b>raw-socket tcp client dest_ip_address dest_port</b> [local_ip_address ] [local_port ]	raw ソケット トランスポート TCP クライアント セッションの設定を指定します。  <i>dest_ip_address</i> : リモート raw ソケット サーバの宛先 IP アドレス。  <i>dest_port</i> : リモートサーバへの TCP 接続に使用する宛先ポート番号。  <i>local_ip_address</i> : (任意) クライアントがバインドできるローカル IP アドレス。  <i>local_port</i> : (任意) クライアントがバインドできるローカルポート番号。
ステップ 4	<b>raw-socket tcp idle-timeout session_timeout</b>	非同期回線インターフェイスの raw ソケット トランスポート TCP セッション タイムアウトを設定します。この間隔でクライアントとサーバの間でデータが転送されない場合は、TCP セッションが閉じられ

## 例 : Raw ソケット TCP クライアント

	コマンドまたはアクション	目的
		<p>ます。その後、クライアントはサーバとの TCP セッションの再確立を自動的に試みます。</p> <p>このタイムアウト設定は、この特定の回線のすべての raw ソケット トランスポート TCP セッションに適用されます。</p> <p><i>session_timeout</i> : 設定されているセッションアイドルタイムアウト (分)。デフォルトは 5 分です。</p>
ステップ 5	<code>raw-socket tcp keepalive interval</code>	<p>非同期回線インターフェイスの raw ソケット トランスポート TCP セッションのキープアライブインターバルを設定します。ルータは、設定された間隔に基づいてキープアライブメッセージを送信します。たとえば、セルラー インターフェイスを介して raw TCP トラフィックを送信するとき、この間隔の設定が必要な場合があります。</p> <p><i>interval</i> : 現在設定されているキープアライブインターバル (秒単位)。範囲は 1 ~ 864000 秒です。デフォルト値は 1 秒です。</p>

## 次のタスク

raw ソケット TCP クライアントを削除するには、**no raw-socket tcp client** コマンドを使用します。

## 例 : Raw ソケット TCP クライアント

次の例は、非同期シリアル回線の raw ソケット TCP クライアントを設定する方法を示しています。raw ソケットクライアントとして機能する IR1101 (ルータ) は、raw ソケットサーバとの TCP セッションを開始し、パケット化されたシリアルデータをそのサーバに転送します。ルータは、バッファ内のシリアルデータのストリームを収集します。バッファに 827 バイト蓄積されると、ルータはデータをパケット化して raw ソケットサーバに転送します。ルータと raw ソケットサーバの間で 10 分間データが交換がされないと、raw ソケットサーバとの TCP セッションが終了し、ルータは raw ソケットサーバとのセッションの再確立を試みます。

```
router# configure terminal
router(config)# line 0/2/0
router(config-line)# raw-socket tcp client 10.0.0.1 4000
router(config-line)# raw-socket packet-length 827
router(config-line)# raw-socket tcp idle-timeout 10
router(config-line)# exit
router(config)#
```

## raw ソケット UDP ピアツーピア接続の設定

raw ソケット UDP カプセル化および共通回線オプションを有効にした後、raw ソケット UDP ピアツーピア接続を設定します。接続の一端のローカルポートは、もう一方の端の宛先ポートである必要があります。

### 始める前に

シリアルインターフェイスでraw ソケットトランスポートを有効化 (6 ページ) の説明に従って、そのポートのシリアルポートおよびraw ソケット UDP カプセル化を有効にします。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>configure terminal</b>	コンフィギュレーション モードに入ります。
ステップ 2	<b>line 0/slot /port</b>	シリアル スロット / ポートのライン コマンド モードを開始します。
ステップ 3	<b>raw-socket udp connection</b> <i>dest_ip_address dest_port</i> <i>local_port</i> [ <i>local_ip_address</i> ]	raw ソケット トランスポート UDP 接続を指定します。  <i>dest_ip_address</i> : UDP 接続に使用する宛先 IP アドレス  <i>dest_port</i> : UDP 接続に使用する宛先ポート番号。  <i>local_port</i> : UDP 接続のローカル ポート番号。  <i>local_ip_address</i> : (任意) UDP 接続のローカル IP アドレス。

### 次のタスク

raw ソケット UDP 接続を削除するには、**no raw-socket udp connection** コマンドを使用します。

## 例 : Raw ソケット UDP 接続

以下は、ルータ A (ローカル IP アドレス 192.168.0.8) とルータ B (ローカル IP アドレス 192.168.0.2) との間に raw ソケット UDP 接続を設定する方法を示しています。

### ルータ A

```
router# configure terminal
router(config)# line 0/2/0
router(config-line)# raw-socket udp connection 192.168.0.2 5000 7000
router(config-line)# exit
router(config)#
```

## ルータ B

```
router# configure terminal
router(config)# line 0/2/0
router(config-line)# raw-socket udp connection 192.168.0.8 7000 5000
router(config-line)# exit
router(config)#
```

# raw ソケットキープアライブ設定 CLI

非同期インターフェイスの raw ソケットキープアライブは、従来の IOS プラットフォームに存在していた機能の 1 つです。17.10.1a の一部として、この機能は IOS-XE ベースのプラットフォームに拡張されます。次の構文を持つ新しい CLI が raw ソケットの下に追加されます。

```
Router(config-line)#raw-socket tcp keepalive interval
```

## CLI の変更

17.10.1a 以降の IOS-XE プラットフォームでは、CLI の修正があり、追加の CLI が raw ソケットの一部として追加されました。

修正は、**raw-socket idle timeout** コマンドに対するものです。以前の設定では分のみを使用していましたが、分と秒に基づいてタイムアウトを設定するオプションが追加されました。

```
Router(config-line)# raw-socket tcp idle-timeout [0-1440] [<0-59> | cr]
```

追加の CLI は、raw ソケット TCP クライアントをクリアするためのものです。コマンドの構文は **clear raw-socket line** [1-145/tty/x/y/z] です。例：

```
Router# clear raw-socket line 0/2/0
```



- 
- (注) **clear raw-socket line** を開始すると、**show raw-socket tcp sessions** コマンドから raw ソケットクライアントの raw ソケットセッションがクリアされます。接続は、TCP ハンドシェイク後に再確立されます。これは、TCP 接続インターフェイスで **shut/no shut** を実行することで達成できます。
- 

# raw ソケット トランスポート に対する YANG データモデルのサポート

リリース 17.2.1 では、追加の YANG データモデルに対するサポートが追加されています。これらの追加モデルには、raw ソケット トランスポートが含まれています。

YANG データモデルは次の URL で確認できます。

<https://github.com/YangModels/yang/tree/master/vendor/cisco/xe/1721>

メインの Cisco-IOS-XE ネイティブモデルに属する raw ソケットには 2 つの機能モジュールがあります。その内容は次のとおりです。

- Cisco-IOS-XE-rawsocket.yang

このモジュールには raw ソケットトランスポートのコンフィギュレーションコマンドの YANG 定義のコレクションが含まれています。

次に、このモジュールに対応する CLI コマンドを示します。

```
# encapsulation raw-tcp
# encapsulation raw-udp
# raw-socket packet-length <length>
# raw-socket packet-timer <timer>
# raw-socket special-char <value>
# raw-socket tcp server <port> <ip>
# raw-socket tcp idle-timeout <value>
# raw-socket tcp client <dest-ip> <dest-port>
# raw-socket tcp idle-timeout <timeout>
# raw-socket tcp tcp-session <value>
# raw-socket tcp dscp <value>
# raw-socket udp connection <dest-ip> <dest-port> <local_port>
```

- Cisco-IOS-XE-rawsocket-oper.yang

このモジュールには、raw ソケットトランスポートの運用データの YANG 定義のコレクションが含まれています。

次に、このモジュールに対応する CLI コマンドを示します。

```
# show raw udp statistics
# show raw tcp statistics
# show raw tcp session
# show raw udp session
# show raw tcp session local
# show raw udp session local
```

次に、依存モジュールのリストを示します。

- Cisco-IOS-XE-native
- Cisco-IOS-XE-features
- ietf-inet-types
- Cisco-IOS-XE-interfaces
- Cisco-IOS-XE-ip
- Cisco-IOS-XE-vlan
- ietf-yang-types @ (すべてのリビジョン)
- cisco-semver

## 設定の確認

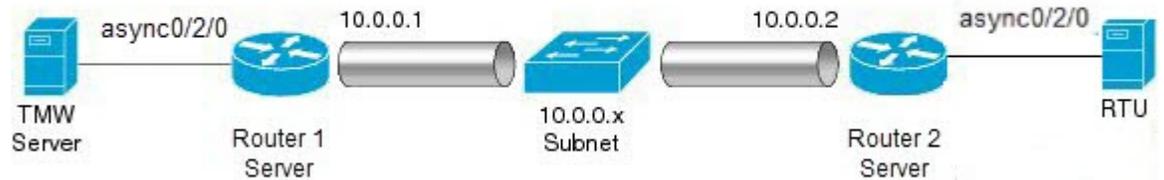
コマンド	目的
<b>show running-config</b>	アクティブな機能とその設定を含む IR1101 の設定を表示します。
<b>show raw-socket tcp detail</b>	raw ソケット トランスポート TCP アクティビティに関する情報を表示します。
<b>show raw-socket tcp sessions</b>	raw ソケット トランスポート TCP セッションに関する情報を表示します。
<b>show raw-socket tcp statistics</b>	各非同期シリアル回線の raw ソケット トランスポート TCP 統計情報を表示します。
<b>show raw-socket udp detail</b>	raw ソケット トランスポート UDP アクティビティに関する情報を表示します。
<b>show raw-socket udp sessions</b>	raw ソケット トランスポート UDP セッションに関する情報を表示します。
<b>show raw-socket udp statistics</b>	各非同期シリアル回線の raw ソケット トランスポート UDP 統計情報を表示します。
<b>clear raw-socket statistics</b>	特定の TTY インターフェイスまたはすべての非同期シリアル回線の raw ソケット トランスポート 統計情報をクリアします。

## 設定例

次のセクションでは、raw ソケット トランスポートの設定例を示します。

### raw ソケット TCP

次の例は、ある IR1101 ルータ（ルータ 1）がサーバとして機能し、別の IR809（ルータ 2）がクライアントとして機能する raw ソケット トランスポートの設定を示しています。



次の表は、上図で強調表示されているサーバーとクライアントIR1101の設定を示しています。

IR1101 サーバ設定	IR807 クライアント設定
<pre> ... interface async0/2/0 no ip address <b>encapsulation raw-tcp</b> ! ... line 0/2/0 <b>raw-socket tcp server 5000 10.0.0.1</b>  <b>raw-socket packet-timer 3</b> <b>raw-socket tcp idle-timeout 5</b> ... </pre>	<pre> ... interface async0 no ip address <b>encapsulation raw-tcp</b> ! interface async1 no ip address <b>encapsulation raw-tcp</b> ! ... line 1 <b>raw-socket tcp client 10.0.0.1 5000 10.0.0.2 9000</b>  <b>raw-socket packet-length 32</b> <b>raw-socket tcp idle-timeout 5</b> line 2 <b>raw-socket tcp client 10.0.0.1 5000 10.0.0.2 9001</b>  <b>raw-socket packet-length 32</b> <b>raw-socket tcp idle-timeout 5</b> </pre>

## raw ソケット UDP

次の例は、2つの IR1101 ルータ間の raw ソケット UDP 接続の設定例を示しています。

### Router1 から

```

interface GigabitEthernet0/1
ip address 192.168.0.8 255.255.255.0
duplex auto
speed auto
interface async0/2/0
no ip address
encapsulation raw-udp
line 0/2/0
raw-socket udp connection 192.168.0.2 2 2

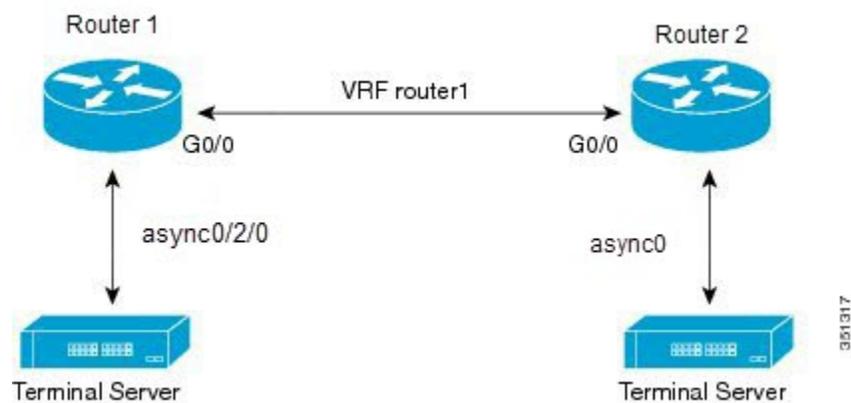
```

**Router2 から**

```
interface GigabitEthernet0/1
ip address 192.168.0.2 255.255.255.0
load-interval 60
duplex auto
speed auto
no keepalive
interface async0/2/0
no ip address
encapsulation raw-udp
line 0/2/0
raw-socket udp connection 192.168.0.8 2 2
```

**raw ソケット VRF**

次の例は、raw ソケット トランスポート用に設定された 2 台のルータが VRF を介して接続する raw ソケット VRF の設定を示しています。Router1 は raw ソケット TCP サーバとして機能する IR1101 で、Router2 は raw ソケット TCP クライアントとして機能する IR807 です。



以下は、上図に示す Router1 と Router2 の設定です。

**Router1 の設定**

ルータに VRF を定義：

```
vrf definition router1
 rd 100:1
 route-target export 100:3
 route-target import 100:3
 !
 address-family ipv4
 exit-address-family
```

インターフェイスに VRF 設定を適用：

```
interface GigabitEthernet0/0
 vrf forwarding router1
 ip address 100.100.100.2 255.255.255.0
```

```
duplex auto
speed auto
```

シリアル インターフェイスに raw TCP を適用 :

```
interface async0/2/0
vrf forwarding router1
no ip address
encapsulation raw-tcp
```

回線に raw TCP を適用 :

```
line 0/2/0
raw-socket tcp server 5000 4.4.4.4
```

## Router2 の設定

ルータに VRF を定義 :

```
vrf definition router1
rd 100:1
route-target export 100:3
route-target import 100:3
!
address-family ipv4
exit-address-family
```

インターフェイスに VRF 設定を適用 :

```
interface GigabitEthernet0/0
vrf forwarding router1
ip address 100.100.100.1 255.255.255.0
duplex auto
speed auto
```

シリアル インターフェイスに raw TCP を適用 :

```
interface async0
vrf forwarding router1
no ip address
encapsulation raw-tcp
```

回線に raw-tcp を適用 :

```
line 1
raw-socket tcp client 4.4.4.4 5000
```



## 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。