



Cisco IOS XR ソフトウェアでのオブジェクト トラッキングの実装

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。IPSec のアプリケーションについては、『Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide』を参照してください。

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、[関連資料](#)、(15 ページ) を参照してください。設定作業の実行中に出てくるその他のコマンドのマニュアルを特定するには、オンラインで『Cisco ASR 9000 Series Aggregation Services Router Commands Master List』内を検索してください。

表 1: Cisco IOS XR ソフトウェア コンテンツのオブジェクトトラッキング実装の機能履歴

リリース	変更内容
リリース 3.9.0	変更なし。
リリース 4.2.1	この機能が導入されました。

このモジュールの構成は、次のとおりです。

- [オブジェクトトラッキングを実装するための前提条件](#), 2 ページ
- [オブジェクトトラッキングについて](#), 2 ページ
- [オブジェクトトラッキングの実装方法](#), 2 ページ
- [オブジェクトトラッキングの設定例](#), 13 ページ
- [その他の参考資料](#), 15 ページ

オブジェクトトラッキング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。

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。

インターフェイスがアップまたはダウンしているかどうかのトラッキング

1～10秒の差分で、インターフェイスのステータスがアップ状態かダウン状態かどうか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この任意のタスクを実行します。

トラッキング対象オブジェクトの状態がダウンに変更された場合、トラッキング オブジェクト (IPSec の場合、これはサービス仮想インターフェイス (SVI)) はダウンになり、次の結果が生じます。

- SVI 上のすべての既存のトンネルが解除されます。
- 新しいトンネルはこの SVI で確立できません。
- スタティックでもダイナミックでも、SVI を指しているすべてのルートは、逆ルート注入 (RRI) を含み削除されます。

手順の概要

1. **configure**
2. **track track-name**
3. (任意) **delay {up seconds|down seconds}**
4. 次のいずれかのコマンドを使用します。
 - **end**
 - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RSP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	track track-name 例： RP/0/RSP0/CPU0:router(config)# track track1	トラッキング コンフィギュレーション モードを開始します。 • <i>track-name</i> : トラッキングするオブジェクトの名前を指定します。
ステップ 3	delay {up seconds down seconds} 例： RP/0/RSP0/CPU0:router(config-track)# delay up 10	(任意) オブジェクトがアップ状態でもダウン状態でも、トラッキング中に発生する可能性のある遅延をスケジュールします。
ステップ 4	次のいずれかのコマンドを使用します。 • end • commit	設定変更を保存します。 • end コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before

	コマンドまたはアクション	目的
	<p>例 :</p> <pre>RP/0/RSP0/CPU0:router (config-track) # end</pre> <p>または</p> <pre>RP/0/RSP0/CPU0:router (config-track) # commit</pre>	<pre>exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> ◦ yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、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。 ◦ no 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。 ◦ cancel 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。 <ul style="list-style-type: none"> • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。

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。

トラッキング対象オブジェクトは、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、アップしていると見なされます。

トラッキング対象オブジェクトの設定後、そのステート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ト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。

手順の概要

1. **configure**
2. **track** *track-name*
3. **type line-protocol state**
4. **interface** *type interface-path-id*
5. (任意) **delay** {*up seconds*|*down seconds*}
6. **interface** { *service-gre numeric-name* | *service-ipsec numeric-name*}
7. **line-protocol track** *object-name*
8. 次のいずれかのコマンドを使用します。
 - **end**
 - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RSP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	track <i>track-name</i> 例： RP/0/RSP0/CPU0:router(config)# track track1	トラッキング コンフィギュレーション モードを開始します。 • <i>track-name</i> : トラッキングするオブジェクトの名前を指定します。
ステップ 3	type line-protocol state 例： RP/0/RSP0/CPU0: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	interface <i>type interface-path-id</i> 例： RP/0/RSP0/CPU0:router(config-track-line-prot)#interface atm 0/2/0/0.1	インターフェイス コンフィギュレーション モードを開始します。 • <i>type</i> : インターフェイスの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。 • <i>interface-path-id</i> : 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。

	コマンドまたはアクション	目的
		<p>(注) ルータに現在設定されているすべてのインターフェイスのリストを表示するには、show interfaces コマンドを使用します。</p> <p>(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。</p>
ステップ 5	<p>delay {up seconds down seconds}</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track)# delay up 10</pre>	<p>(任意)</p> <p>オブジェクトがアップ状態でもダウン状態でも、トラッキング中に発生する可能性のある遅延をスケジュールします。</p>
ステップ 6	<p>interface { service-gre numeric-name service-ipsec numeric-name}</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track)# interface service-ipsec 23</pre>	<p>service-ipsec インターフェイス モードを入力します。このインターフェイスモードで、service-ipsec インターフェイスをトラッキングが必要なステートのインターフェイスと関連づけます。たとえば、ATM など選択したインターフェイスのステートがダウンまたはアップになった場合、service-ipsec インターフェイスのステートはこれに続きます。</p> <ul style="list-style-type: none"> • numeric-name : service-ipsec インターフェイスの数値名で 1 ~ 65535 から指定できます。 <p>(注) service-gre インターフェイスをインターフェイスオブジェクトとしてトラッキングできますが、他のインターフェイスオブジェクトのステートをモニタする手段としては現在サポートされていません。</p>
ステップ 7	<p>line-protocol track object-name</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-if)# line-protocol track track12</pre>	<p>特定のトラッキングを IPSec インターフェイスや GRE インターフェイスに関連付けます。トラッキングのステートが変わると、インターフェイスのステートも変わります。</p>
ステップ 8	<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"> • end • commit 	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 <p>Uncommitted changes found, commit them</p>

	コマンドまたはアクション	目的
	<p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre> <p>または</p> <pre>RP/0/RSP0/CPU0:router(config-track)# commit</pre>	<pre>before exiting(yes/no/cancel)?</pre> <pre>[cancel]:</pre> <ul style="list-style-type: none"> ◦ yes と入力すると、実行コンフィギュレーションファイルに変更が保存され、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。 ◦ no 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。 ◦ cancel 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。 <ul style="list-style-type: none"> • 実行コンフィギュレーションファイルに変更を保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

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ています。

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。

手順の概要

1. **configure**
2. **track track-name**
3. **type route reachability**
4. 次のいずれかのコマンドを使用します。
 - **vrf vrf-table-name**
 - **route ipv4 IP-prefix/mask**
5. (任意) **delay {up seconds|down seconds}**
6. **interface { service-gre numeric-name | service-ipsec numeric-name}**
7. **line-protocol track object-name**
8. 次のいずれかのコマンドを使用します。
 - **end**
 - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例 : RP/0/RSP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	track track-name 例 : RP/0/RSP0/CPU0:router(config)# track track1	トラッキング コンフィギュレーション モードを開始します。 • track-name : トラッキングするオブジェクトの名前を指定します。
ステップ 3	type route reachability 例 : RP/0/RSP0/CPU0: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。 <ul style="list-style-type: none"> • vrf vrf-table-name • route ipv4 IP-prefix/mask 	トラッキング対象になるように、お使いのルータ タイプに応じて、次のいずれかで構成できる IP ルートのタイプを設定します。 <ul style="list-style-type: none"> • vrf-table-name : VRF テーブル名。

	コマンドまたはアクション	目的
	<p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track-route)# vrf vrf-table-4</pre> <p>または</p> <pre>RP/0/RSP0/CPU0:router(config-track-route)# route ipv4 10.56.8.10/16</pre>	<ul style="list-style-type: none"> • <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	<p>delay {up seconds down seconds}</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track)# delay up 10</pre>	<p>(任意)</p> <p>オブジェクトがアップ状態でもダウン状態でも、トラッキング中に発生する可能性のある遅延をスケジュールします。</p>
ステップ 6	<p>interface { service-gre numeric-name service-ipsec numeric-name }</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-track)# interface service-ipsec 23</pre>	<p>service-ipsec インターフェイスモードを入力します。このインターフェイスモードで、service-ipsec インターフェイスをトラッキングが必要なステートのインターフェイスと関連付けます。たとえば、ATM など選択したインターフェイスのステートがダウンまたはアップになった場合、service-ipsec インターフェイスのステートはこれに続きます。</p> <ul style="list-style-type: none"> • <i>numeric-name</i> : service-ipsec インターフェイスの数値名で 1 ~ 65535 から指定できます。 <p>(注) service-gre インターフェイスをインターフェイスオブジェクトとしてトラッキングできますが、他のインターフェイスオブジェクトのステートをモニタする手段としては現在サポートされていません。</p>
ステップ 7	<p>line-protocol track object-name</p> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-if)# line-protocol track track1</pre>	<p>Associates トラッキングを IPSec インターフェイスまたは GRE インターフェイスと関連付けます。トラッキングのステートが変わると、インターフェイスのステートも変わります。</p>
ステップ 8	<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"> • end • commit <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-if)# end</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> ◦ yes と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーション

	コマンドまたはアクション	目的
	または <pre>RP/0/RSP0/CPU0:router(config-if)# commit</pre>	<p>セッションが終了して、ルータがEXECモードに戻ります。</p> <ul style="list-style-type: none"> ◦ no 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。 ◦ cancel 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。 <p>• 実行コンフィギュレーションファイルに変更を保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。</p>

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。

手順の概要

1. **configure**
2. **track track-name**
3. **type list boolean { and | or }**
4. **object object-name [not]**
5. (任意) **delay {up seconds|down seconds}**
6. **interface { service-gre numeric-name | service-ipsec numeric-name }**
7. **line-protocol track object-name**
8. 次のいずれかのコマンドを使用します。
 - **end**
 - **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure 例： RP/0/RSP0/CPU0:router# configure	グローバル コンフィギュレーション モードを開始します。
ステップ 2	track track-name 例： RP/0/RSP0/CPU0:router(config)# track track1	トラッキング コンフィギュレーション モードを開始します。 • track-name : トラッキングするオブジェクトの名前を指定します。
ステップ 3	type list boolean { and or } 例： RP/0/RSP0/CPU0: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。 • boolean : トラッキング対象リストの状態がブール計算に基づくことを指定します。 • and : すべてのオブジェクトがアップである場合にリストがアップになり、1つ以上のオブジェクトがダウンの場合にリストがダウンになることを指定します。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。 • or : 少なくとも1つのオブジェクトがアップの場合にリストがアップになることを指定します。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

	コマンドまたはアクション	目的
		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。
ステップ 4	object <i>object-name</i> [not] 例： <pre>RP/0/RSP0/CPU0:router(config-track-list)# object 3 not</pre>	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。 <ul style="list-style-type: none"> • <i>object-name</i> : トラッキングするオブジェクトの名前。 • not : オブジェクトの状態を否定します。
ステップ 5	delay { up <i>seconds</i> down <i>seconds</i> } 例： <pre>RP/0/RSP0/CPU0:router(config-track)# delay up 10</pre>	(任意) オブジェクトがアップ状態でもダウン状態でも、トラッキング中に発生する可能性のある遅延をスケジュールします。
ステップ 6	interface { service-gre <i>numeric-name</i> service-ipsec <i>numeric-name</i> } 例： <pre>RP/0/RSP0/CPU0:router(config-track)# interface service-ipsec 23</pre>	service-ipsec インターフェイス モードを入力します。このインターフェイス モードで、service-ipsec インターフェイスをトラッキングが必要なステートのインターフェイスと関連づけます。たとえば、ATM など選択したインターフェイスのステートがダウンまたはアップになった場合、service-ipsec インターフェイスのステートはこれに続きます。 <ul style="list-style-type: none"> • <i>numeric-name</i> : service-ipsec インターフェイスの数値名で 1 ~ 65535 から指定できます。 (注) service-gre インターフェイスをインターフェイス オブジェクトとしてトラッキングできますが、他のインターフェイスオブジェクトのステートをモニタする手段としては現在サポートされていません。
ステップ 7	line-protocol track <i>object-name</i> 例： <pre>RP/0/RSP0/CPU0:router(config-if)# line-protocol track track1</pre>	トラッキングを IP Sec または GRE インターフェイスに関連付けます。トラッキングのステートが変わると、インターフェイスのステートも変わります。
ステップ 8	次のいずれかのコマンドを使用します。 <ul style="list-style-type: none"> • end • commit 例： <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre>	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 <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
	または RP/0/RSP0/CPU0:router (config-track) # commit	<ul style="list-style-type: none"> ◦ yes と入力すると、実行コンフィギュレーションファイルに変更が保存され、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。 ◦ no 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。 ◦ cancel 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。 <ul style="list-style-type: none"> • 実行コンフィギュレーションファイルに変更を保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

オブジェクトトラッキングの設定例

IPSec における多様なシナリオでオブジェクトトラッキングを使用する方法を示した例については、『*Cisco ASR 9000 Series Aggregation Services Router System Monitoring Configuration Guide*』の「*Implementing IPSec Network Security on Cisco IOS XR Software*」モジュールを参照してください。

インターフェイスがアップまたはダウンしているかどうかの追跡：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追跡：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 GigabitEthernet 0/0/0/3 を通過していきます。

```
track IPSec1
  type line-protocol state
  interface gigabitethernet0/0/0/3
  !
interface service-ipsec 1
```

```

ipv4 address 70.0.0.1 255.255.255.0
profile vrf1_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
!
```

インターフェイスのライン プロトコル ステータスの表示 : 例

この例では、前の例を実行した後の **show track** コマンドからの出力を表示します。

```

RP/0/RSP0/CPU0:router# show 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追跡 : 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の宛先が、ネットワーク **7.0.0.0/24** 内になります。このトラッキング手順では、ルーティングプロトコルプレフィックスの状態に従って、ルーティングテーブルに変更があった場合に信号を発信します。

```

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 : 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が、インターフェイス **GigabitEthernet 0/0/0/3** およびインターフェイス **ATM 0/2/0/0.1** を通過していきます。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。これには、ブール演算式 **AND** を使用しています。

```

track LIST2
  type list boolean and
  object IPSec1
  object IPSec2
  object PREFIX1
  !
track IPSec1
  type line-protocol state
  interface GigabitEthernet0/0/0/3
  !
```

```

track IPsec2
  type line-protocol state
  interface ATM0/2/0.1
  !
track PREFIX1
  type route reachability
  route ipv4 7.0.0.0/24
  !
interface service-ipsec1
  vrf 1
  ipv4 address 70.0.0.2 255.255.255.0
  profile vrf_1_ipsec
  line-protocol track LIST2
  tunnel source 80.0.0.2
  tunnel destination 80.0.0.1
  service-location preferred-active 0/2/0
  !

```

その他の参考資料

次の項では、IPSec ネットワーク セキュリティ用のオブジェクトトラッキングの実装に関する参考資料を示します。

関連資料

関連項目	参照先
IPSec ネットワーク セキュリティ コマンド	『Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide』の「IPSec Network Security Commands on Cisco ASR 9000 シリーズ ルータ」モジュール
インターネット キー交換 (IKE) セキュリティ プロトコル コマンド	『Cisco ASR 9000 Series Aggregation Services Router System Security Command Reference』の「Internet Key Exchange Security Protocol Commands on Cisco ASR 9000 シリーズ ルータ」モジュール
IP-Sec 関連のオブジェクトトラッキング コマンド	『Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference』

標準

標準	タイトル
この機能によってサポートされる新しい標準または変更された標準はありません。またこの機能による既存標準のサポートに変更はありません。	—

MIB

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

RFC

RFC	タイトル
RFC 2401	『 <i>Security Architecture for the Internet Protocol</i> 』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカルサポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html