Cisco Unified SIP Proxy モジュールの設定

論理ネットワークの設定

Cisco Unified SIP Proxy 上の各インターフェイスは、論理ネットワークと関連付けられます。論理ネットワークは、サーバ グループ、リッスン ポイント、その他のプロパティの編成に使用されます。SIP メッセージは、メッセージが到達するネットワークと関連付けられます。

手順の概要

1. **cusp**
2. **configure**
3. **sip network** *network*
4. **end network**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**sip network** *network* **例：** `se-10-0-0-0(cusp-config)> sip network service-provider`	ネットワークを作成し、ネットワーク コマンド モードにします。この場合、作成されるネットワークの名前は「service provider」です。
ステップ 4	**end network** **例：** `se-10-0-0-0(cusp-config-network)> end network`	ネットワーク コマンド モードを終了します。

例

次の例では、「service-provider」という名前のネットワークを作成する方法を示します。

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip network service-provider
se-10-0-0-0(cusp-config-network)> end network
```

トリガー条件の設定

トリガー条件を作成すると、Cisco Unified SIP Proxy はさまざまな呼び出しフローに対して適切な動作で応答できます。一般的に、呼び出しフローが複雑であるほど複雑なトリガーが必要です。

手順の概要

1. **cusp**
2. **configure**
3. **trigger condition** *trigger-condition-name*

4. **sequence** *sequence-number*
5. （オプション）**in-network** *network-name*
6. （オプション）**mid-dialog**
7. end sequence
8. end trigger condition

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**trigger condition** *trigger-condition-name* **例：** `se-10-0-0-0(cusp-config)> trigger condition call-from-service-provider`	トリガー条件を作成し、トリガー コマンド モードにします。この場合、作成されるトリガーの名前は「call-from-service-provider」です。
ステップ 4	**sequence** *sequence-number* **例：** `se-10-0-0-0(cusp-config-trigger)> sequence 1`	指定した数字のシーケンスを作成し、トリガー シーケンス コマンド モードにします。この数字は、トリガーが評価される順番を示します。この場合、作成されるトリガーのシーケンス番号は 1 です。
ステップ 5	**in-network** *network-name* **例：** `se-10-0-0-0(cusp-config-trigger-seq)> in-network service-provider`	オプション。トリガー条件の着信ネットワーク名を指定します。この場合、着信ネットワークは「service-provider」ネットワークです。
ステップ 6	**mid-dialog** **例：** `se-10-0-0-0(cusp-config-trigger-seq)> mid-dialog`	オプション。mid-dialog メッセージのルーティング ポリシーをバイパスする特殊なトリガーです。
ステップ 7	**end sequence** **例：** `se-10-0-0-0(cusp-config-trigger-seq)> end sequence`	トリガー シーケンス コマンド モードを終了します。
ステップ 8	**end trigger condition** **例：** `se-10-0-0-0(cusp-config-trigger)> end trigger condition`	トリガー コマンド モードを終了します。

例

このサンプルでは、Cisco Unified SIP Proxy は呼び出しが入ってきたネットワークに基づいて対処を行うだけなので、トリガーは単純です。

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> trigger condition call-from-service-provider
se-10-0-0-0(cusp-config-trigger)> sequence 1
se-10-0-0-0(cusp-config-trigger-seq)> in-network service-provider
se-10-0-0-0(cusp-config-trigger-seq)> end sequence
se-10-0-0-0(cusp-config-trigger)> end trigger condition

se-10-0-0-0(cusp-config)> trigger condition mid-dialog
se-10-0-0-0(cusp-config-trigger)> sequence 1
se-10-0-0-0(cusp-config-trigger-seq)> mid-dialog
se-10-0-0-0(cusp-config-trigger-seq)> end sequence
se-10-0-0-0(cusp-config-trigger)> end trigger condition
```

サーバ グループの設定

サーバ グループについて

サーバ グループは、Cisco Unified SIP Proxy が各ネットワークで通信を行う要素を定義します。使用されるサーバ グループ名は、発信要求の SIP URI に挿入されます。Cisco Unified Communications Manager などの一部のデバイスでは、処理を行う前に要求の URI を検証します。つまり、これを利用できるようにするには、場合によって完全修飾ドメイン名（FQDN）を使ってエンド デバイスを設定する必要があります。

個別の各要素の 2 つのフィールド（q-value および weight）は、要素のプライオリティとロード バランシングを指定するために使用されるので重要です。呼び出しは q-value に基づいて特定の要素にルーティングされます。最も高い q-value を持つ要素は、そのサーバ グループにルーティングされたすべてのトラフィックを受信します。複数の要素が同じ q-value を持つ場合、トラフィックは、使用されているロードバランシング オプションに基づいて各要素に分散されます。デフォルトでは call-id に基づいてロード バランシングが行われますが、weight も使用できます。weight を使用する場合、ある要素が受信するトラフィックの割合は、その要素の weight を、q-value の weight が同じ稼動中の要素の合計で割った割合に等しくなります。これらの weight の合計は 100 に等しい必要はありません。weight と q-value を変えることで、さまざまなプライオリティやロードバランシング方式を設定できます。

手順の概要

1. **cusp**
2. **configure**
3. **server-group sip group** *server-group-name network*

4. **element ip-address** *ipaddress port* **{udp | tcp | tls} [q-value** *q-value***] [weight** *weight***]**
5. **lb-type {global | highest-q | request-uri | call-id | to-uri | weight }**
6. **end server-group**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**server-group sip group** *server-group-name network* **例：** `se-10-0-0-0(cusp-config)> server-group sip group sp.example.com service-provider`	SIP サーバ グループを作成し、サーバ グループ コマンド モードを開始します。この場合、作成されるサーバ グループの名前は「sp.example.com」です。「sp.example.com」は、「service-provider」という名前のネットワークを使用します。
ステップ 4	**element ip-address** *ipaddress port* **{udp | tcp | tls} [q-value** *q-value***] [weight** *weight***]** **例：** `se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.3 5060 tls q-value 1.0 weight 100`	SIP サーバ グループの IP 要素を作成し、この SIP サーバ グループの特性を決定します。 **（注）** このコマンドは、複数回入力できます。
ステップ 5	**lb-type {global | highest-q | request-uri | call-id | to-uri | weight }** **例：** `se-10-0-0-0(cusp-config-sg)> lb-type weight`	SIP サーバ グループのロード バランシング アルゴリズムを設定します。この例では、同じ q-value を持つ他の要素の重みに対して、その重みに比例して要素が選択されることを指定します。
ステップ 6	**end server-group** **例：** `se-10-0-0-0(cusp-config-sg)> end server-group`	サーバ グループ コマンド モードを終了します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> server-group sip group sp.example.com service-provider
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.3 5060 tls q-value 1.0 weight
100
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.4 5060 tls q-value 1.0 weight
50
```

```
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.5 5060 tls q-value 1.0 weight
50
se-10-0-0-0(cusp-config-sg)> lb-type weight
se-10-0-0-0(cusp-config-sg)> end server-group
```

ルート テーブルの設定

ルート テーブルについて

SIP 要求を適切な宛先へ送るには、ルート テーブルを設定する必要があります。各ルート テーブルは、ルックアップ ポリシーに基づいて照合するキーのセットで構成されています。たとえば、各キーはダイヤルされた電話番号の市外局番を表す場合があります。

手順の概要

1. **cusp**
2. **configure**
3. **route table** *table-name*
4. **key** *key* **response** *response-code*
5. **key** *key* **target-destination** *target-destination network*
6. **end route table**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**route table** *table-name* **例：** `se-10-0-0-0(cusp-config)> route table service-provider-table`	ルート テーブルを作成し、ルート テーブル コマンド モードを開始します。この場合、「service-provider-table」という名前のルート テーブルが作成されます。
ステップ 4	**key** *key* **response** *response-code* **例：** `se-10-0-0-0(cusp-config-rt)> key * response 404`	応答コードを検索キーに割り当てます。この例では、「404」の応答がすべてに割り当てられます。
ステップ 5	**key** *key* **target-destination** *target-destination network* **例：** `se-10-0-0-0(cusp-config-rt)> key 510 target-destination cube-sp.example.com cube-sp`	宛先要素の key 部分を指定した値に置き換えます。 **（注）** このコマンドは、複数回入力できます。
ステップ 6	**end route table** **例：** `se-10-0-0-0(cusp-config-rt)> end route table`	ルート テーブル コマンド モードを終了します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> route table service-provider-table
se-10-0-0-0(cusp-config-rt)> key * response 404
se-10-0-0-0(cusp-config-rt)> key 510 target-destination cube-sp.example.com cube-sp
se-10-0-0-0(cusp-config-rt)> end route table
```

正規化ポリシーの設定

正規化ポリシーは、互換性がないネットワークを考慮して SIP メッセージを変更します。この場合、サービス プロバイダーがエスケープ シーケンスの「91」を処理できないため、request-uri と TO ヘッダーからシーケンスを削除する必要があります。

手順の概要

1. **cusp**
2. **configure**
3. **policy normalization** *policy_name*

4. **uri-component update request-uri {user | host | host-port | phone | uri} {all |** *match-string***}** *replace-string*
5. **uri-component update header {first | last | all} {user | host | host-port | phone | uri} {all |** *match-string} replace-string*
6. **end policy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**policy normalization** *policy-name* **例：** `se-10-0-0-0(cusp-config)> policy normalization outgoing-norm-policy`	正規化ポリシーを作成し、ポリシー正規化コマンド モードを開始します。この例では、正規化ポリシーの名前を「outgoing-norm-policy」にします。
ステップ 4	**uri-component update request-uri {user | host | host-port | phone | uri} {all |** *match-string***}** *replace-string* **例：** `se-10-0-0-0(cusp-config-norm)> uri-component update request-uri user ^91 ""`	request-URI に含まれる URI コンポーネント フィールドを更新する正規化ポリシー手順を設定します。
ステップ 5	**uri-component update header {first | last | all} {user | host | host-port | phone | uri} {all |** *match-string} replace-string* **例：** `se-10-0-0-0(cusp-config-norm)> uri-component update TO all user ^91 ""`	ソース メッセージのヘッダーに含まれる URI コンポーネント フィールドを更新する正規化ポリシー手順を設定します。
ステップ 6	**end policy** **例：** `se-10-0-0-0(cusp-config-norm)> end policy`	ポリシー正規化コマンド モードを終了します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> policy normalization outgoing-norm-policy
```

```
se-10-0-0-0(cusp-config-norm)> uri-component update request-uri user ^91 ""
se-10-0-0-0(cusp-config-norm)> uri-component update TO all user ^91 ""
se-10-0-0-0(cusp-config-norm)> end policy
```

ルックアップ ポリシーの設定

ルックアップ ポリシーによって、ルート テーブル内のキーの使われ方が決まります。各キーは、ダイヤルされる電話番号の先頭を表します。これは、各ポリシーが、request-uri のユーザ コンポーネントをルート テーブル内のキーに対して照合するための記述であるためです。request-uri のユーザ コンポーネントは、呼び出される電話番号です。照合に使用されるルールはプレフィックスで、これはルート テーブル内の最も長いプレフィックス マッチが使用されることを意味します。したがって、ダイヤルされた番号が 510-1XX-XXXX である場合、呼び出しは cme.example.com サーバ グループへ送られます。ダイヤルされた番号が 510-XXX-XXXX である場合、呼び出しは cucm.example.com サーバ グループへ送られます。以下のサンプルの 4 つのポリシーは、それぞれが特定のテーブルを参照することを除いて同一です。

手順の概要

1. **cusp**
2. **configure**
3. **policy lookup** *policy-name*
4. **sequence** *sequence-number*
5. **rule {exact | prefix | subdomain | subnet | fixed** *length*} **[case-insensitive]**
6. **end sequence**
7. **end policy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**policy lookup** *policy-name* **例：** se-10-0-0-0(cusp-config)> **policy lookup service-provider-policy**	指定した名前のポリシーを作成し、ポリシー ルックアップ コマンド モードを開始します。この場合、作成されるポリシーの名前は「service-provider-policy」です。
ステップ 4	**sequence** *sequence-number* **例：** se-10-0-0-0(cusp-config-lookup)> **sequence 1**	指定した数字のシーケンスを作成し、ポリシー ルックアップ シーケンス コマンド モードを開始します。シーケンスは、その数字の順番に従って実行されます。
ステップ 5	**rule {exact | prefix | subdomain | subnet | fixed** *length***} [case-insensitive]** **例：** se-10-0-0-0(cusp-config-lookup-seq)> **rule prefix**	ルックアップ ポリシーのルーティング アルゴリズムを決定するルールを作成します。 この場合、最も長いプレフィクスの一致をルックアップ ポリシーで検索することを指定するルールが作成されます。
ステップ 6	**end sequence** **例：** se-10-0-0-0(cusp-config-lookup-seq)> **end sequence**	ポリシー ルックアップ シーケンス コマンド モードを終了します。
ステップ 7	**end policy** **例：** se-10-0-0-0(cusp-config-lookup)> **end policy**	ポリシー ルックアップ コマンド モードを終了します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> policy lookup service-provider-policy
se-10-0-0-0(cusp-config-lookup)> sequence 1 service-provider-table request-uri
uri-component user
se-10-0-0-0(cusp-config-lookup-seq)> rule prefix
se-10-0-0-0(cusp-config-lookup-seq)> end sequence
se-10-0-0-0(cusp-config-lookup)> end policy
```

ルーティング トリガーの設定

ルーティング トリガーは、トリガー条件をルックアップ ポリシーと相互に関連付けます。照合される対応条件によって、単一のポリシーが選択されます。条件はシーケンス番号の昇順で評価されます。ポリシー ステップが mid-dialog メッセージでスキップされるように、mid-dialog 条件が最初に評価されます。以下の設定に基づき、INVITE メッセージが正常にルーティングされた後、それに続くすべてのメッセージ（mid-dialog）はルーティング ポリシーをバイパスします。

手順の概要

1. **cusp**
2. **configure**
3. **trigger routing sequence** *sequence-number* **{by-pass | policy** *policy***} [condition** *trigger-condition***]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**trigger routing sequence** *sequence-number* **{by-pass | policy** *policy***} [condition** *trigger-condition***]** **例：** `se-10-0-0-0(cusp-config)> trigger routing sequence 2 policy service-provider-policy condition call-from-service-provider`	ルーティング ポリシーをトリガー条件と関連付けます。 この例では、2 番目のシーケンスは、以前に作成した「service-provider-policy」というポリシーと、以前に作成した「call-from-service-provider」というトリガーに従います。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> trigger routing sequence 1 by-pass condition mid-dialog
se-10-0-0-0(cusp-config)> trigger routing sequence 2 policy service-provider-policy
condition call-from-service-provider
se-10-0-0-0(cusp-config)> trigger routing sequence 3 policy cube-sp-policy condition
call-from-cube-sp
se-10-0-0-0(cusp-config)> trigger routing sequence 4 policy cube-es-policy condition
call-from-cube-es
se-10-0-0-0(cusp-config)> trigger routing sequence 5 policy enterprise-policy condition
call-from-enterprise
```

正規化トリガーの設定

正規化トリガーは、トリガー条件を正規化ポリシーと相互に関連付けます。トリガーには、ルーティングの前に発生する pre-normalization と、ルーティングの後に発生する post-normalization の 2 種類があります。ルーティング ポリシーと同様に、特殊なポリシーは mid-dialog メッセージでの正規化をバイパスします。

- 「手順の概要」（P.28）

手順の概要

1. **cusp**
2. **configure**
3. **trigger pre-normalization sequence** *sequence-number* {**by-pass | policy** *policy*} [**condition** *trigger-condition*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**trigger pre-normalization sequence** *sequence-number* **{by-pass | policy** *policy*} [**condition** *trigger-condition*] **例：** `se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 2 policy outgoing-norm-policy condition call-from-cube-sp`	着信 SIP メッセージの正規化前アルゴリズムを正規化ポリシーに設定します。 この例では、2 番目のシーケンスは、以前に作成した「outgoing-norm-policy」というポリシーと、以前に作成した「call-from-cube-sp」というトリガーに従います。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 1 by-pass condition
mid-dialog
se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 2 policy outgoing-norm-policy
condition call-from-cube-sp
```

リッスン ポートとレコードルート ポートの設定

各ネットワークのリッスン ポートとレコードルート ポートを設定する必要があります。リッスン ポートとレコードルート ポートでは、Cisco Unified SIP Proxy モジュールの実際のアドレスが使用されます。**sip record-route** コマンドは、発信要求内に record-route ヘッダーを挿入します。**sip listen** コマンドは、Cisco Unified SIP Proxy がそのポートで受信要求を受け付けられるようにします。

手順の概要

1. **cusp**
2. **configure**
3. **sip record-route** *network_name* **{tcp | tls | udp}** *ip_address [port]*
4. **sip listen** *network_name* **{tcp | tls | udp}** *ip_address port*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**sip record-route** *network_name* **{tcp | tls | udp}** *ip_address [port]* **例：** `se-10-0-0-0(cusp-config)> sip record-route service-provider udp 10.10.10.99 5060`	SIP ネットワークのレコードルーティングをイネーブルにします。 この例では、「service-provider」ネットワークはレコードルート コンフィギュレーションに関連付けられ、Record-Route ヘッダー フィールドに入力される IP アドレスは「10.10.10.99」、Record-Route ヘッダー フィールドに入力されるポートは 5060 です。
ステップ 4	**sip listen** *network_name* **{tcp | tls | udp}** *ip_address port* **例：** `se-10-0-0-0(cusp-config)> sip listen service-provider udp 10.10.10.99 5060`	特定の SIP ネットワーク、ホスト、およびポート上の SIP トラフィックをリッスンするリスナーを作成します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip record-route service-provider udp 10.10.10.99 5060
se-10-0-0-0(cusp-config)> sip listen service-provider udp 10.10.10.99 5060
```

ホスト名の設定

アップストリーム要素がネットワーク内の 2 つの Cisco Unified SIP Proxy をルーティングするために DNS SRV を使用している場合、この 2 つの Cisco Unified SIP Proxy が同じ FQDN を持つように設定する必要があります。これを行うには、両方の Cisco Unified SIP Proxy の Cisco Unified SIP Proxy コンフィギュレーション モードで **sip alias** コマンドを入力します。

手順の概要

1. **cusp**
2. **configure**
3. **sip alias** *hostname*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy 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	**sip alias** *hostname* **例：** `se-10-0-0-0(cusp-config)> sip alias myhost`	このインスタンスのホスト名を設定します。

例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip alias myhost
```

トランスポート レイヤ セキュリティ（TLS）の設定

署名付き証明書の作成とインポート

Cisco Unified SIP Proxy では、TLS、伝送制御プロトコル（TCP）、およびユーザ データグラム プロトコル（UDP）がサポートされています。TLS 接続の確立には署名付き証明書による認証が必要なため、いくつか追加の手順が必要です。

前提条件

証明書要求をエクスポートするには、FTP サーバか HTTP が必要です。

手順の概要

1. **configure terminal**
2. **crypto key generate [rsa {label** *label-name* **| modulus** *modulus-size*} **| default]**
3. **crypto key certreq label** *label-name* **url {ftp: | http:}**
4. **crypto key import rsa label** *label-name* **{der url {ftp: | http: } | pem { terminal | url {ftp: | http: }} [default]**
5. **crypto key import cer label mykey url ftp:**

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 terminal** **例：** se-10-0-0-0# **configure terminal**	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	**crypto key generate [rsa {label** *label-name* **| modulus** *modulus-size***} | default]** **例：** se-10-0-0-0(config)> **crypto key generate rsa label mykey modulus 512 default**	RSA 秘密キーを作成します。
ステップ 3	**crypto key certreq label** *label-name* **url {ftp: | http:}** **例：** se-10-0-0-0(config)> **crypto key certreq label mykey url ftp:**	署名する証明書要求を作成します。
ステップ 4	**crypto key import rsa label** *label-name* **{der url {ftp: | http: } | pem { terminal | url {ftp: | http: }} [default]** **例：** se-10-0-0-0(config)> **crypto key import trustcacert label rootCA url ftp:**	証明書要求に署名した後、要求への署名に使用した信頼済み認証局（CA）をインポートします。
ステップ 5	**crypto key import rsa label** *label-name* **{der url {ftp: | http: } | pem { terminal | url {ftp: | http: }} [default]** **例：** se-10-0-0-0(config)> **crypto key import cer label mykey url ftp:**	ルート CA をインポートした後、署名付き証明書をインポートします。

署名付き証明書の作成例

```
se-10-0-0-0# configure terminal
se-10-0-0-0(config)> crypto key generate rsa label mykey modulus 512 default
Key generation in progress. Please wait...
The label name for the key is mykey

se-10-0-0-0(config)> crypto key certreq label mykey url ftp:
Address or name of remote host? 192.168.202.216
Username (ENTER if none)? anonymous
Password (not shown)?
Destination path? netmod/mykey.csr
Uploading CSR file succeed

se-10-0-0-0(config)> crypto key import trustcacert label rootCA url ftp:
Import certificate file...
Address or name of remote host? 192.168.202.216
Source filename? netmod/rootCA/cacert.pem
1212 bytes received.
```

```
se-10-0-0-0(config)> crypto key import cer label mykey url ftp:
Import certificate file...
Address or name of remote host? 192.168.202.216
Source filename? netmod/mycert.cer
952 bytes received.
Import succeeded
```

次の作業

- TLS ピア要素のいずれかに使用する、信頼済み CA 証明書をインポートします。

Cisco Unified SIP Proxy 上での TLS の作成

証明書をインポートしたら、TLS 接続を有効にする必要があります。セキュリティを強化する場合は、信頼済みピアのリストを作成できます。このリストを作成すると、指定したピアからの接続だけを受け付けます。ピアのホスト名エントリは、証明書内にあるピアの subjectAltName である必要があります。subjectAltName が証明書内で使用されていない場合は、ピアのホスト名エントリは CN である必要があります。

- 「手順の概要」（P.33）
- 「手順の詳細」（P.33）
- 「TLS の設定例」（P.34）

手順の概要

1. **cusp**
2. **configure**
3. **sip tls**
4. **sip tls trusted-peer** {*peer's-hostname*}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cusp** **例：** `se-10-0-0-0> cusp`	Cisco Unified SIP Proxy EXEC モードを開始します。
ステップ 2	**configure** **例：** `se-10-0-0-0(cusp)> configure`	Cisco Unified SIP Proxy コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**sip tls** **例：** se-10-0-0-0(cusp-config)> **sip tls**	インターネット経由のセキュアな通信を提供する、他の SIP エンティティによる SIP TLS 接続の使用をイネーブルにします。
ステップ 4	**sip tls trusted-peer** {*peer's-hostname*} **例：** se-10-0-0-0(cusp-config)> **sip tls trusted-peer example.com**	信頼済みピアのリストを作成します。

TLS の設定例

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip tls
se-10-0-0-0(cusp-config)> sip tls trusted-peer example.com
```

設定の確定

ここで設定を確定する必要があります。設定を確定する目的は 2 つあります。設定をアクティブにすることとその保持のためです。

- 現在有効な設定を表示するには、**show configuration active** コマンドを入力します。
- 変更を確定した後で有効になる設定を表示するには、**show configuration candidate** コマンドを入力します。
- このサンプルの設定を確定するには、次のコマンドを入力します。

  se-10-0-0-0(cusp-config)> **commit**