プロトコル認証

この章は、次の項で構成されています。

- COOP （1 ページ）
- EIGRP （3 ページ）

COOP

概要

マッピング情報（ロケーションおよび ID）をスパイン プロキシに伝達するために、Council of Oracles Protocol（COOP）を使用します。リーフ スイッチは、Zero Message Queue（ZMQ）を使用して、エンドポイントアドレス情報をスパインスイッチ「Oracle」に転送します。スパイン ノードで実行している COOP によって、すべてのスパイン ノードが一貫性のあるエンドポイント アドレスとロケーション情報のコピーを維持することができ、さらに、ロケーション マッピング データベースに対するエンドポイント ID の分散ハッシュ テーブル（DHT）レポジトリを維持することができます。

COOPデータ パス通信は、セキュアな接続を介した転送を優先します。COOP は悪意のあるトラフィック インジェクションから COOP メッセージを保護するために MD5 オプションの活用が強化されます。APIC コントローラおよびスイッチは、COOP プロトコル認証をサポートします。

COOP プロトコルは 2 つの ZMQ 認証モードをサポートするために強化されています：ストリクトおよび互換性。

- ストリクト モード：COOP では MD5 認証済みの ZMQ 接続のみを許可します。
- 互換性モード；COOP ではメッセージの転送に MD5 認証接続と非認証 ZMQ 接続の両方を許可します。

Cisco APIC で COOP を使用する

Cisco Application Centric Infrastructure（ACI）ファブリック上で COOP ゼロ メッセージ キュー（ZMQ）認証サポートを行うため、Application Policy Infrastructure Controller（APIC）では MD5 パスワードおよび COOP セキュア モードをサポートします。

COOP ZMQ 認証タイプの設定：新しい管理対象オブジェクトの `coop: AuthP` は、データ管理エンジン（DME）データベース（DME）/COOP に追加されます。属性タイプのデフォルト値は「互換性」ですが、ユーザーには「厳密」タイプ設定を行うオプションがあります。

COOP ZMQ 認証 MD5 パスワード：APIC では管理対象オブジェクト（`fabric:SecurityToken`）提供し、MD5 パスワードに使用する属性が含まれます。「トークン」と呼ばれるこの管理対象オブジェクト内の属性は、1 時間ごとに変更される文字列です。COOP は、DME から通知を受け取り、ZMQ 認証のパスワードを更新します。この属性トークンの値は表示されません。

注意事項と制約事項

次の注意事項と制約事項に従ってください。

- ACI ファブリックのアップグレード中は、すべてのスイッチがアップグレードされるまで、COOP 厳格モードが許可されません。この保護は、早期に厳格なモードを有効にすることでトリガされる可能性がある、予期しない COOP 接続の拒否を防ぎます。

APIC GUI を使用した COOP 認証の設定

ステップ 1 メニュー バーで、**[System]** > **[System Settings]** の順に選択します。

ステップ 2 **[ナビゲーション]** ペインで **[COOP グループ]** をクリックします。

ステップ 3 **[作業]** ペインの **[タイプ]** フィールドにある **[ポリシー プロパティ]** 領域で、**[互換性のあるタイプ]** および **[ストリクト タイプ]** オプションから希望のタイプを選択します。

ステップ 4 [Submit] をクリックします。
これにより、COOP 認証ポリシー設定を完了します。

Cisco NX OS スタイル CLI を使用した COOP 認証の設定

ストリクト モード オプションを使用して、COOP 認証ポリシーを設定します。

例：

```
apic1# configure
apic1(config)# coop-fabric
apic1(config-coop-fabric)# authentication type ?
compatible  Compatible type
```

```
strict      Strict type
apic101-apic1(config-coop-fabric)# authentication type strict
```

REST API を使用した COOP 認証の設定

COOP 認証ポリシーを設定します。

例では、ストリクト モードが選択されます。

例：

```
https://172.23.53.xx/api/node/mo/uni/fabric/pol-default.xml

<coopPol type="strict">
</coopPol>
```

EIGRP

概要

EIGRP は、リンクステート プロトコルの機能にディスタンス ベクトル プロトコルの利点を組み合わせたプロトコルです。EIGRP は、定期的に Hello メッセージを送信してネイバーを探索します。EIGRP は、新規ネイバーを検出すると、すべてのローカル EIGRP ルートおよびルート メトリックに対する 1 回限りの更新を送信します。受信側の EIGRP ルータは、受信したメトリックと、その新規ネイバーにローカルで割り当てられたリンクのコストに基づいて、ルート ディスタンスを計算します。この最初の全面的なルート テーブルの更新後は、ルート変更の影響を受けるネイバーにのみ、差分更新が EIGRP により送信されます。この処理により、コンバージェンスにかかる時間が短縮され、EIGRP が使用する帯域幅が最小限になります。

Cisco APIC では、EIGRP 認証でルートマップのキーチェーンのインフラストラクチャが MD5 認証に使用されます。2 つの EIGRP ピア間で認証を設定するには 2 つのパラメータが必要になります。パラメータは次のとおりです。

- モード
- Keychain

注意事項と制約事項

次の注意事項と制約事項に従ってください。

- MD5 認証のみサポートされます。キーチェーンは、RPM で設定されているキーチェーン名です。

- 2 つの EIGRP ピア間で認証の不一致がある場合は、ネイバーシップのフラッピングが発生します。フラッピングの理由は **show eigrp internal event-history syslog** で確認できます。

APIC GUI を使用した EIGRP 認証の設定

ステップ 1 メニュー バーで、**[Tenant]*[tenant-name]*** を選択します。

ステップ 2 [Navigation] ペインで、**[Policies]** > **[Protocol]** > **[EIGRP]** を展開します。

ステップ 3 [EIGRP] を展開し、[EIGRP KeyChains] を右クリックして [Create Keychain Policy] を開き、次の操作を行います。

a) [Name] フィールドにポリシーの名前を入力します。
b) [KeyID] フィールドに、キー ID 番号を入力します。
c) [Preshared key] フィールドに、事前共有キーの情報を入力します。
d) オプション。[Start Time] フィールドと [End Time] フィールドに、時間を入力します。

ステップ 4 [Navigation] ペインで、[EIGRP Interface] を右クリックし、次の操作を行います。

a) [Authentication] フィールドで、ボックスをクリックして有効にします。
b) [Key Chain Policy] フィールドで、ドロップダウン リストから作成したポリシーを選択し、[Submit] をクリックします。

NX-OS CLI を使用した EIGRP 認証の設定

ステップ 1 テナントで、キーチェーン ポリシーとキーポリシーを設定します。

例：

```
tenant T1
keychain-policy KeyChainPol
key-policy 2
```

ステップ 2 オプション。開始時刻を設定します。

例：

```
startime 2018-11-01T08:39:27.000+00:00
exit
```

ステップ 3 APIC からリーフ設定を開始します。インターフェイスでの認証を有効にし、キーチェーン ポリシーを設定します。

例：

```
IFC1(config-leaf)# show run
# Command: show running-config leaf 104
# Time: Thu Nov 8 12:05:45 2018
leaf 104
interface ethernet 1/2.45
```

```
vrf member tenant T1 vrf V1 l3out L3Out
ip router eigrp authentication keychain-policy KeyChainPol
ip router eigrp authentication enable
!
ipv6 router eigrp authentication keychain-policy KeyChainPol
ipv6 router eigrp authentication enable
exit
```

ステップ 4 EIGRP の設定を確認するには、次の手順を実行します。

例：

```
fav-blr4-ls-leaf4# show ip eigrp interfaces eth1/2.17
EIGRP interfaces for process 1 VRF T1:V1
Xmit Queue Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable SRTT Un/Reliable Flow Timer Routes
eth1/2.17 0 0/0 0 0/0 50 0
Hello interval is 5 sec
Holdtime interval is 15 sec
Next xmit serial: 0
Un/reliable mcasts: 0/3 Un/reliable ucasts: 6/4
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 1
Retransmissions sent: 0 Out-of-sequence rcvd: 0
Classic/wide metric peers: 0/0
Authentication mode is md5, key-chain is T1:KeyChainPol
ifav-blr4-ls-leaf4#
```

ステップ 5 スイッチでトラブルシューティングを行う場合は、次の CLI を使用できます。EIGRP 認証は、IPv4 と IPv6 の両方のアドレス ファミリでサポートされています。

例：

```
(none)# show ip eigrp interface vrf all
EIGRP interfaces for process 100 VRF pepsi
Xmit Queue Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable SRTT Un/Reliable Flow Timer Routes
eth1/1 1 0/0 207 0/0 828 0
Hello interval is 10 sec
Holdtime interval is 15 sec
Next xmit serial: 0
Un/reliable mcasts: 0/7 Un/reliable ucasts: 21/18
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 0
Retransmissions sent: 4 Out-of-sequence rcvd: 2
Classic/wide metric peers: 0/1
Authentication mode is md5, key-chain is eigrp-auth

(none)# show ipv6 eigrp interface vrf pepsi
IPv6-EIGRP interfaces for process 100 VRF pepsi
Xmit Queue Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable SRTT Un/Reliable Flow Timer Routes
eth1/1 0 0/0 0 0/0 0 0
Hello interval is 10 sec
Holdtime interval is 15 sec
Next xmit serial: 0
Un/reliable mcasts: 0/0 Un/reliable ucasts: 0/0
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 0
Retransmissions sent: 0 Out-of-sequence rcvd: 0
Classic/wide metric peers: 0/0
Authentication mode is md5, key-chain is eigrp-auth
```

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。