Cisco APIC トラブルシューティングツールを使用します

この章では、発生する可能性のある問題のトラブルシューティングに一般的に使用されるツールと方法を紹介します。これらのツールは、トラフィックの監視、デバッグ、およびトラフィック ドロップ、誤ルーティング、ブロックされたパス、アップリンク障害などの問題の検出に役立ちます。この章で説明するツールの概要については、以下のツールを参照してください。

- **[ACL契約許可と拒否ログ（ACL Contract Permit and Deny Logs）]** — パケットのロギングをイネーブル化。もしくは、契約許可ルールとパケットのロギングかタブー契約拒否ルールのためにフローがドロップされているために送信が許可されているフローをイネーブル化。
- **[アトミック カウンタ（Atomic Counters）]** —ドロップ検出のフローの間のトラフィックの統計を収集することをイネーブル化。ファブリックのミスルーティングの統計を収集。クイック デバッギングとアプリケーション接続問題の隔離のイネーブル化。
- **[デジタル オプティカル モニタリング（Digital Optical Monitoring）]**：物理インターフェイスに関するデジタル オプティカル モニタリング（DOM）統計を表示できます。
- **正常性スコア**：ネットワーク階層をドリルダウンして障害を特定の管理対象オブジェクト（MO）に分離することにより、パフォーマンスの問題を分離できます。
- **ポート トラッキング**：アップリンクの障害を検出するために、リーフ スイッチとスパイン スイッチ間のリンクのステータスをモニタできます。
- **SNMP** — Simple Network Management Protocol (SNMP) は、個々のホスト（APIC またはその他のホスト）をリモートでモニタし、特定のノードの状態を確認できます。
- **SPAN**—Switchport Analizer（SPAN）は、詳細なトラブルシューティングの実行または特定のアプリケーションホストからトラフィックのサンプルを取得し、プロアクティブなモニタリングと分析を行うことができます。
- **[統計（Statistics）]**—監視対象オブジェクトのリアルタイム測定が提供されます。統計の表示により、トレンド分析とトラブルシューティングの実行が可能になります。

- **Syslog** — 送信されるメッセージの重大度の最小値、syslog メッセージに含める項目、および syslog の接続先を指定できます。フォーマットは、NX-OS CLI フォーマットで表示することもできます。
- **[トレースルート（Traceroute）]** — パケットが接続先に移動するときに実際にたどるルートを探すことできます。
- **[トラブルシューティング ウィザード（Troubleshooting Wizard）]** - 管理者は、2 つのエンドポイントを選択することで指定できる特定の時間枠内に発生する問題のトラブルシューティングを行うことができます。
- **[構成の同期の問題（Configuration Sync Issues）]** — Cisco APIC のトランザクションがまだ同期されていないかどうかを確認できます。

この章は、次の項で構成されています。

ACL コントラクトおよび拒否ログの有効化および表示

ACL 契約の許可および拒否ログについて

契約ルールのトラフィック フローをログ記録および監視するには、契約の許可ルールのため送信されることが許可されたパケットまたはフローのログを有効化および表示するか、契約の許可ルールのためドロップされたフローのログを有効化および表示できます。

- 禁止契約拒否ルール
- 契約の件名でアクションを拒否する
- 契約または件名の例外

- ACI ファブリックの ACL コントラクト許可は、EX または FX で終わる名前の Nexus 9000 シリーズ スイッチ、およびそれ以降のすべてのモデルでのみサポートされます。たとえば、N9K-C93180LC-EX や N9K-C9336C-FX のように指定してください。
- ACI ファブリックでのログの拒否は、すべてのプラットフォームでサポートされています。
- 管理契約のフィルタでログ directive を使用することはサポートされていません。ログ directive を設定すると、ゾーン分割ルールの展開エラーが発生します。

標準および禁止契約と件名についての詳細は、『*CiscoApplication Centric Infrastructure Fundamentals*』および『*Cisco APIC Basic Configuration Guide*』を参照してください。

ACL 許可および拒否ログ出力に含まれる EPG データ

Cisco APIC、リリース 3.2(1) まで、ACL 許可および拒否ログでは、記録されている契約に関連付けられた EPG を識別していませんでした。リリース 3.2(1) では、送信元 EPG と 送信先 EPG が ACI 許可および拒否ログの出力に追加されます。ACL 許可および拒否ログには、次の制限を持つ関連 EPG を含めます。

- ネットワーク内の EPG の配置によっては、ログの EPG データを使用できない場合があります。
- 設定の変更が発生するとき、ログデータが期限切れになっている可能性があります。安定した状態では、ログ データは正確です。

ログが次にフォーカスされているとき、許可および拒否ログの EPG データは最も正確になります。

- 入力 TOR で入力ポリシーがインストールされており、出力 TOR で出力ポリシーがインストールされている場合の EPG から EPG へのフロー。
- 境界リーフ TOR で 1 個のポリシーが適用され、非 BL TOR で他のポリシーが適用されている場合の EPG から L3Out へのフロー。

ログ出力の EPG は、共有サービス（共有 L3Outs を含む）で使用される uSeg Epg または Epg ではサポートされていません。

GUI を使用してACL 契約の許可とロギングの拒否を有効にする

次の手順では、GUI を使用してACL 契約の許可とロギングの拒否を有効にする方法を表示します。

（注） 許可ロギングを含むテナントは、EPG が関連する VRF を含むテナントです。これは必ずしも EPG と同じテナントや関連する契約である必要はありません。

手順

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Contracts] を展開し、 [Standard] を右クリックして [Create Contract] を選択します。

ステップ 3 [Create Contract] ダイアログボックスで、次の作業を実行します。

a) [Name] フィールドに、契約の名前を入力します。

b) [Scope] フィールドで、そのスコープ（[VRF]、[Tenant]、または [Global]）を選択します

c) オプション。契約に適用するターゲット DSCP または QoS クラスを設定します。

d) [+] アイコンをクリックして、[Subject] を展開します。

ステップ 4 [Create Contract Subject] ダイアログボックスで、次の操作を実行します。

ステップ 5 件名の名前と詳細な説明を入力します。

ステップ 6 オプション。ターゲット DSCP のドロップダウン リストから、件名に適用する DSCP を選択します。

ステップ 7 契約を両方向でなくコンシューマからプロバイダの方向にのみ適用するのでない限り、[Apply Both Directions] はオンにしたままにしておきます。

ステップ 8 [Apply Both Directions] をチェックしてない場合 [Reverse Filter Ports] をチェックしたままにして、ルールがプロバイダから消費者に適用されるようにレイヤ 4 ソースと宛先ポートを交換します。

ステップ 9 [+] アイコンをクリックして、[Filters] を展開します。

ステップ 10 [Name] ドロップダウン リストで、たとえば、**arp**、**default**、**est**、**icmp** などオプションを選択するか、以前設定したフィルタを選択します。

ステップ 11 [Directives] ドロップダウン リストで、[log] をクリックします。

ステップ 12 （任意） この件名で実行するアクションを [Deny] に変更します（またはアクションをデフォルトの [Permit] のままにします。

Directive：ログ有効化により、この件名のアクションが [Permit] になっている場合、ACL は件名と契約により制御されているフローとパケットを追跡します。この件名のアクションが [Deny] の場合、ACL の拒否ログはフローとパケットを追跡します。

ステップ 13 （任意） 件名の優先順位を設定します。

ステップ 14 [Update] をクリックします。

ステップ 15 [OK] をクリックします。

ステップ 16 **[送信（Submit）]** をクリックします。
ロギングがこの契約に対して有効になります。

NX-OS CLI を使用した ACL 契約許可ロギングの有効化

次の例は、NX-OS CLI を使用して契約許可ロギングを有効にする方法を示しています。

手順

ステップ 1 契約許可ルールにより送信できたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <permit>
subject <subject Name>
access-group <access-list> <in/out/both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract Logicmp type permit
apic1(config-tenant-contract)# subject icmp
apic1(config-tenant-contract-subj)# access-group arp both log
```

ステップ 2 許可ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group arp both log` コマンドを使用します。

REST API を使用した ACL 契約許可ロギングの有効化

次の例は、REST API を使用して許可および拒否ロギングを有効にする方法を示しています。この例では、ACL の許可を設定し、件名 Permit 設定し、設定されたアクションを拒否するには、契約のロギングを拒否します。

手順

この設定では、次の例のように XML で post を送信します。

例：

```
<vzBrCP dn="uni/tn-Tenant64/brc-C64" name="C64" scope="context">
    <vzSubj consMatchT="AtleastOne" name="HTTPSsbj" provMatchT="AtleastOne"
revFltPorts="yes" rn="subj-HTTPSsbj">
         <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-PerHTTPS" tDn="uni/tn-Tenant64/flt-PerHTTPS" tRn="flt-PerHTTPS"
tnVzFilterName="PerHTTPS"/>
    </vzSubj>
    <vzSubj consMatchT="AtleastOne" name="httpSbj" provMatchT="AtleastOne"
revFltPorts="yes" rn="subj-httpSbj">
        <vzRsSubjFiltAtt action="deny" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-httpFilter" tDn="uni/tn-Tenant64/flt-httpFilter" tRn="flt-httpFilter"
 tnVzFilterName="httpFilter"/>
    </vzSubj>
   <vzSubj consMatchT="AtleastOne" name="subj64" provMatchT="AtleastOne" revFltPorts="yes"
 rn="subj-subj64">
        <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes"
```

```
priorityOverride="default"
rn="rssubjFiltAtt-icmp" tDn="uni/tn-common/flt-icmp" tRn="flt-icmp" tnVzFilterName="icmp"/>

    </vzSubj>
</vzBrCP>
```

GUI を使用した禁止契約拒否ロギングの有効化

次の手順は、GUI を使用して禁止コントラクトの拒否ロギングを有効にする方法を示しています。

手順

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Contracts] を展開します。

ステップ 3 [Taboos] を右クリックし、[Create Taboo Contract] を選択します。

ステップ 4 [Create Taboo Contract] ダイアログ ボックスで、次の操作を実行して禁止契約を指定します。

a) [Name] フィールドに、契約の名前を入力します。
b) オプション。[Description] フィールドに、禁止契約の説明を入力します。
c) [+] アイコンをクリックして、[Subject] を展開します。

ステップ 5 [Create Taboo Contract Subject] ダイアログ ボックスで、次の操作を実行します。

a) [Specify Identity of Subject] 領域に、名前と説明（オプション）を入力します。
b) [+] アイコンをクリックして、[Filters] を展開します。
c) [Name] ドロップダウン リストから、**<tenant_name>/arp**、 **<tenant_name>/default**、**<tenant_name>/est**、**<tenant_name>/icmp** などのデフォルト値のいずれかを選択し、以前作成したフィルタか [Create Filter] を選択します。

（注） [Specify Filter Identity] 領域で [Create Filter] を選択した場合、次の操作を実行して、ACL 拒否ルールの基準を指定します。

1. 名前とオプションの説明を入力します。
2. [Entries] を展開し、ルールの名前を入力して、拒否するトラフィックを定義する条件を選択します。
3. [Directives] ドロップダウンリストで [log] を選択します。
4. [Update] をクリックします。
5. [OK] をクリックします。

ステップ 6 **[送信（Submit）]** をクリックします。
ロギングがこの禁止契約に対して有効になります。

NX-OS CLI を使用した禁止契約拒否ロギングの有効化

次の例は、NX-OS CLI を使用して禁止契約拒否ロギングを有効にする方法を示しています。

手順

ステップ 1 禁止契約拒否ルールのためにドロップされたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <deny>
subject <subject Name>
access-group <access-list> <both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract dropFTP type deny
apic1(config-tenant-contract)# subject dropftp
apic1(config-tenant-contract-subj)# access-group ftp both log
```

ステップ 2 拒否ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group https both log` コマンドを使用します。

REST API を使用した禁止契約拒否ロギングの有効化

次の例は、REST API を使用して禁止契約拒否ロギングを有効にする方法を示しています。

手順

タブー契約を設定するロギングを拒否する、次の例のように XML で post を送信します。

例：

```
<vzTaboo dn="uni/tn-Tenant64/taboo-TCtrctPrefix" name="TCtrctPrefix" scope="context">
    <vzTSubj name="PrefSubj" rn="tsubj-PrefSubj"">
        <vzRsDenyRule directives="log" forceResolve="yes" rn="rsdenyRule-default"
tCl="vzFilter"
tDn="uni/tn-common/flt-default" tRn="flt-default"/>
    </vzTSubj>
</vzTaboo>
```

GUI を使用した ACL 許可および拒否ログの表示

次の手順は、GUI を使用して、トラフィック フローの ACL 許可および拒否ログを（有効になっていれば）表示する方法を示しています。

手順

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Tenant <tenant name>] をクリックします。

ステップ 3 **Tenants <tenant name> [Work]** ペインで、**[Operational]** タブをクリックします。

ステップ 4 [Operational] タブの下で、[Flows] タブをクリックします。
[Flows] タブの下で、いずれかのタブをクリックして、レイヤ 2 許可ログ（[L2 Permit]）、レイヤ 3 許可ログ（[L3 Permit]）、レイヤ 2 拒否ログ（[L2 Drop]）、またはレイヤ 3 拒否ログ（[L3 Drop]）のログ データを表示します。各タブで、トラフィックがフローしていれば、ACL ロギング データを表示できます。データ ポイントは、ログ タイプと ACL ルールに応じて異なります。たとえば、[L3 Permit] ログおよび [L3 Deny] ログには次のデータ ポイントが含まれます。

- VRF
- Alias
- 送信元 IP アドレス
- 宛先 IP アドレス
- プロトコル
- 送信元ポート
- 宛先ポート
- 送信元 MAC アドレス
- 宛先 MAC アドレス
- Node
- 送信元インターフェイス
- VRF Encap
- 送信元 EPG
- 宛先 EPG
- 送信元 PC タグ
- 宛先 PC タグ

（注） また、[Flows] タブの横の [Packets] タブを使用して、シグニチャ、送信元、および宛先が同じであるパケットのグループ（最大 10 個）の ACL ログにアクセスできます。送信されたりドロップされたりするパケットのタイプを確認できます。

REST API を使用した ACL 許可および拒否ログ

次の例は、REST API を使用して、トラフィック フローのレイヤ 2 拒否ログ データを表示する方法を示しています。次の MO を使用してクエリを送信することができます。

- acllogDropL2Flow
- acllogPermitL2Flow
- acllogDropL3Flow
- acllogPermitL3Flow
- acllogDropL2Pkt
- acllogPermitL2Pkt
- acllogDropL3Pkt
- acllogPermitL3Pkt

始める前に

ACL 契約許可および拒否ログのデータを表示する前に、許可または拒否ロギングを有効にする必要があります。

手順

レイヤ 3 ドロップ ログ データを表示するには、REST API を使用して次のクエリを送信します。

```
GET https://apic-ip-address/api/class/acllogDropL3Flow
```

例：

次の例では、サンプル出力をいくつか示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="2">
    <acllogPermitL3Flow childAction=""
dn="topology/pod-1/node-101/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]


-dip-[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-

        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
 dstMacAddr="00:00:12:00:00:25"
```

```
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn=""
protocol="udp" srcEpgName="unknown"
        srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0"
srcMacAddr="00:00:15:00:00:28" srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
    <acllogPermitL3Flow childAction=""
dn="topology/pod-1/node-102/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]-dip-


[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-

        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
 dstMacAddr="00:00:12:00:00:25"
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn=""
protocol="udp" srcEpgName="unknown"
        srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0"
srcMacAddr="00:00:15:00:00:28" srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
</imdata>
```

NX-OS CLI を使用した ACL 許可および拒否ログの表示

次の手順は、NX-OS スタイル CLI **show acllog** コマンドを使用して ACL ログの詳細を表示する方法を示しています。

レイヤ 3 コマンドの構文は、**show acllog {permit | deny} l3 {pkt | flow} tenant <tenant_name> vrf <vrf_name> srcip <source_ip> dstip <destination_ip> srcport <source_port> dstport <destination_port> protocol <protocol> srcintf <source_interface> start-time <startTime> end-time <endTime> detail** です。

レイヤ 2 コマンドの構文は、**show acllog {permit | deny} l2 {flow | pkt} tenant <tenant_name> vrf <VRF_name> srcintf <source_interface> vlan <VLAN_number> detail** です。

（注） **show acllog** コマンドの完全な構文は、第二世代 Cisco Nexus 9000 シリーズ スイッチ（N9K-C93180LC-EX など名前の最後に EX または FX がつく。もしくはそれ以降のシリーズ）および Cisco APIC リリース 3.2 以降でのみ使用できます。第一世代のスイッチ（名前の最後に EX または FX が付かない）または 3.2 以前の Cisco APIC リリースでは、使用可能な構文は上記の通りです。

Cisco APIC 3.2 以降では、追加のキーワードが **detail** keyword:**[dstEpgName <destination_EPG_name>| dstmac <destination_MAC_address> | dstpctag <destination_PCTag> | srcEpgName <source_EPG_name> | srcmac <source_MAC_address> | srcpctag <source_PCTag>]** とともにコマンドの両方のバージョンに追加されます。

手順

ステップ 1 次の例では、**show acllog drop l3 flow tenant common vrf default detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l3 flow tenant common vrf default detail
SrcPcTag  : 49153
DstPcTag  : 32773
SrcEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg6
DstEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg5
SrcIp     : 16.0.2.10
DstIp     : 19.0.2.10
Protocol  : udp
SrcPort   : 17459
DstPort   : 8721
SrcMAC    : 00:00:15:00:00:28
DstMAC    : 00:00:12:00:00:25
Node      : 101
SrcIntf   : port-channel5
VrfEncap  : VXLAN: 2097153
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 2 次の例では、**show acllog deny l2 flow tenant common vrf tsw0connctx0 detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l2 flow tenant common vrf tsw0connctx0 detail
SrcPcTag DstPcTag   SrcEPG           DstEPG          SrcMAC            DstMAC
Node   SrcIntf   vlan
--------- --------- ----------------  --------------  ------------------
------------------   -----   ---------  ------
 32773    49153   uni/tn-TSW       uni/tn-TSW  00:00:11:00:00:11  11:00:32:00:00:33
101     port-     2
                  _Tenant0/ap-     _Tenant0/ap-
       channel8
                  tsw0AP0/epg-     tsw0AP0/epg-
                  tsw0ctx0BD0epg5  tsw0ctx0BD0epg6
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 3 次の例では、**show acllog permit l3 pkt tenant <tenant name> vrf <vrf name> [detail]** コマンドを使用して、送信された一般的な VRF ACL レイヤ 3 許可パケットに関する詳細情報を表示する方法を示しています。

```
apic1# show acllog permit l3 pkt tenant common vrf default detail acllog permit l3 packets
 detail:
srcIp      : 10.2.0.19
dstIp      : 10.2.0.16
protocol   : udp
srcPort    : 13124
dstPort    : 4386
srcIntf    : port-channel5
vrfEncap   : VXLAN: 2097153
pktLen     : 112
srcMacAddr : 00:00:15:00:00:28
```

```
dstMacAddr : 00:00:12:00:00:25
timeStamp  : 2015-03-17T21:31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 4 次の例では、**show acllog permit l2 pkt tenant <tenant name> vrf <vrf name> srcintf <s interface>** コマンドを使用して、インターフェイス ポートチャネル 15 から送信されたデフォルトの VRF レイヤ 2 パケットに関する情報を表示する方法を示しています。

```
apic1# show acllog permit l2 pkt tenant common vrf default srcintf port-channel5

acllog permit L2 Packets
      Node          srcIntf       pktLen     timeStamp
 --------------  --------------  --------  --------------
                 port-channel5      1      2015-03-17T21:
                                           31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

統計情報の収集にアトミック カウンタ ポリシーを使用する

アトミック カウンタ ポリシーを使用すると、エンドポイント、エンドポイント グループ、外部インターフェイス、および IP アドレスの組み合わせ間のトラフィックに関する統計を収集できます。収集された情報で、ファブリック内のドロップや誤ったルーティングを検出できるため、迅速なデバッグを実行し、アプリケーション接続の問題を切り分けることができます。

アトミック カウンタ

アトミック カウンタは、ファブリック内のエンドポイント、EPG、またはアプリケーション間の接続のトラブルシューティングに役立ちます。ユーザー レポート アプリケーションが遅くなったり、2 つのエンドポイント間のトラフィック損失を監視するためにアトミック カウンタが必要になる場合があります。アトミック カウンタが提供する機能の 1 つは、トラブル チケットを予防的な監視モードにする機能です。たとえば、問題が断続的であり、オペレーターがチケットをアクティブに処理しているときに発生するとは限りません。

アトミック カウンタは、ファブリックでのパケット損失の検出に役立ち、接続の問題の原因をすばやく特定できます。アトミック カウンタでは、ファブリックで NTP を有効にする必要があります。

リーフ間（TEP 間）のアトミック カウンタは次を提供できます。

- ドロップ、承認および超過パケットのカウント
- 最後の 30 秒などの短期間のデータ収集、5 分、15 分、またはそれ以上の長期間のデータ収集

- スパイン トラフィックごとの詳細（TEP、リーフ、またはVPCの数が64未満の場合に使用可能）
- 継続的なモニタリング

リーフ間（TEP間）アトミック カウンタは累積であり、クリアできません。ただし、30秒のアトミック カウンタは30秒間隔でリセットされるため、断続的な問題や再発する問題の分離に使用できます。

テナントのアトミック カウンタは次を提供できます。

- ドロップ、承認および超過パケットを含む、ファブリック全体のトラフィックのアプリケーション固有カウンタ
- モードは次を含みます。
- エンドポイントからエンドポイントへのMACアドレス、またはエンドポイントからエンドポイントへのIPアドレス。1つのターゲット エンドポイントに複数のIPアドレスが関連付けられている可能性があることに注意してください。
- オプションのドリル ダウン付きのEPGツーEPG
- EPGからエンドポイント
- EPGから*（任意）
- エンドポイントから外部IPアドレス

（注） アトミック カウンタは、2つのエンドポイント間のパケット量を追跡し、これを測定値として使用します。これらは、ハードウェア レベルでのドロップやエラー カウンタを考慮していません。

ドロップされたパケットは、送信元が送信したよりも接続先が受信したパケットが少ない場合に計算されます。

超過パケットは、送信元が送信したよりも接続先が受信したパケットの方が多い場合に計算されます。

アトミック カウンタに関する注意事項および制約事項

- アトミックカウンタの使用は、エンドポイントが異なるテナントまたは同じテナント内の異なるコンテキスト（VRF）にある場合はサポートされません。
- Cisco APIC リリース3.1(2m)以降では、ファブリックのライフタイム内のパスで統計情報が生成されなかった場合、そのパスに対するアトミック カウンタは生成されません。また、**[トラフィック マップ（Traffic Map）]**（**[可視化（Visualization）]**タブにあるもので、**[操作（Operations）]** > **[可視化（Visualization）]**をCisco APIC GUIで選択する）には、すべてのパスではなく、アクティブなパス、つまりファブリックの寿命のいずれかの時点で、トラフィックがあったパスだけが表示されます。
- IPアドレスが学習されない純粋なレイヤ2設定（IPアドレスは0.0.0.0）では、エンドポイント/EPG間およびEPG/エンドポイント間のアトミック カウンタ ポリシーはサポートされません。この場合、エンドポイント間およびEPG間のポリシーはサポートされます。

外部ポリシーは学習されたIPアドレスが必要なVirtual Routing and Forwarding（VRF）ベースであり、サポートされます。

- アトミック カウンタの送信元または宛先がエンドポイントである場合、そのエンドポイントはスタティックではなくダイナミックである必要があります。ダイナミック エンドポイント（fv:CEp）とは異なり、スタティック エンドポイント（fv:StCEp）にはアトミック カウンタに必要な子オブジェクト（fv:RsCEpToPathEp）がありません。
- 中継トポロジでは、リーフ スイッチはすべてのスパイン スイッチを使用したフル メッシュにはなく、リーフ間（TEP 間）のカウンタは予期どおりに動作しません。
- リーフ間（TEP 間）アトミック カウンタの場合、トンネル数がハードウェア制限を上回ると、システムはモードをトレール モードからパス モードに変更し、ユーザにはスパインごとのトラフィックは表示されなくなります。
- アトミック カウンタはスパイン プロキシ トラフィックはカウントしません。
- ファブリックに入る前、またはリーフ ポートに転送される前にドロップされたパケット、アトミック カウンタによって無視されます。
- ハイパーバイザで切り替えられるパケット（同じポート グループとホスト）はカウントされません。
- アトミック カウンタには、アクティブなファブリック ネットワーク タイム プロトコル（NTP）ポリシーが必要です。
- アトミック カウンタはIPv6の送信元と宛先で動作しますが、IPv4アドレスとIPv6アドレスを混在させて送信元IP アドレスと宛先IP アドレスを設定することはできません。
- 送信元または宛先としてfvCEpを使用して設定されたアトミック カウンタ ポリシーでは、fvCEp管理対象オブジェクトに存在するMACアドレスおよびIPアドレスからのトラフィックと、両者へのトラフィックだけがカウントされます。fvCEpの管理対象オブジェクトでIPアドレス フィールドが空の場合、そのMACアドレスとの間で送受信されるすべてのトラフィックがIP アドレスに関係なくカウントされます。Cisco APIC が fvCEp について複数のIPアドレスを学習している場合、前述のように、fvCEp管理対象オブジェクト自体にある1 つのIP アドレスのみがカウントされます。特定のIP アドレスとの送受信に関連したアトミック カウンタ ポリシーを設定するには、送信元または宛先として fvIp 管理対象オブジェクトを使用します。
- fvCEp の背後に fvIp が存在する場合は、fvCEp ベースのポリシーではなく fvIP ベースのポリシーを追加する必要があります。
- エンドポイントが同じ EPG に属している場合、IPv6 ヘッダーを持つレイヤ 2 ブリッジド トラフィックの、それらのエンドポイント間でのアトミック カウンタ統計は報告されません。
- EPG または ESG から L3Out EPG に流れるトラフィックに対してアトミック カウンタが機能するには、すべてのプレフィックスとマッチさせるため、0/0 ではなく 0/1 および 128/1 を使用して L3Out EPG を設定します。

アトミック カウンタの構成

手順

ステップ 1 メニュー バーで、[Tenants] をクリックします。

ステップ 2 サブメニュー バーで、必要なテナントをクリックします。

ステップ 3 **Navigation** ウィンドウで、テナントを展開し、**Policies** を展開し、それから **Troubleshoot** を展開します。

ステップ 4 **Troubleshoot** の下で、**Atomic Counter Policy** を展開し、トラフィック トポロジを選択します。

エンドポイントの組み合わせ、エンドポイント グループ、外部インターフェイスおよび IP アドレス間のトラフィックを測定できます。

ステップ 5 必要なトポロジを右クリックして、**Add** *topology* **Policy** を選択し、**Add Policy** ダイアログボックスを開きます。

ステップ 6 [Add Policy] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドにポリシーの名前を入力します。

b) トラフィックの送信元の識別情報を選択するか、入力します。

必要な識別情報のソース（エンドポイント、エンドポイントのグループ、外部インターフェイス、または IP アドレス）によって異なります。

c) トラフィックの宛先の識別情報を選択するか、入力します。

d) （任意）（任意）[Filters] テーブルで + アイコンをクリックし、カウントするトラフィックのフィルタリングを指定します。

表示される [Create Atomic Counter Filter] ダイアログボックスで、IP プロトコル番号（たとえば TCP=6）によるフィルタリング、および送信元と宛先の IP ポート番号によるフィルタリングを指定できます。

e) [Submit] をクリックし、アトミック カウンタ ポリシーを保存します。

ステップ 7 [Navigation] ペインで、選択したトポロジの下の新しいアトミック カウンタ ポリシーを選択します。

ポリシー設定が [Work] ペインに表示されます。

ステップ 8 [Work] ペインで [Operational] タブをクリックし、[Traffic] サブタブをクリックして、アトミック カウンタの統計情報を表示します。

アトミック カウンタのイネーブル化

アトミック カウンターを使用してファブリック内のドロップと誤ルーティングを検出し、アプリケーション接続の問題の迅速なデバッグと分離を可能にするには、次のいずれかのタイプの 1 つ以上のテナント アトミック カウンタ ポリシーを作成します。

- EP_to_EP — エンドポイントからエンドポイント（ **dbgacEpToEp**）

- EP_to_EPG — エンドポイントからエンドポイント グループ（**dbgacEpToEpg** ）
- EP_to_Ext — 外部 IP アドレスへのエンドポイント（**dbgacEpToExt**）
- EPG_to_EP — エンドポイント グループからエンドポイントへ（**dbgacEpgToEp**）
- EPG_to_EPG—エンドポイントグループからエンドポイントグループへ（**dbgacEpgToEpg**）
- EPG_to_IP — エンドポイント グループから IP アドレス（**dbgacEpgToIp**）
- Ext_to_EP — エンドポイントへの外部 IP アドレス（**dbgacExtToEp**）
- IP_to_EPG — エンドポイント グループへの IP アドレス（**dbgacIpToEpg**）
- Any_to_EP — 任意からエンドポイント（**dbgacAnyToEp**）
- EP_to_Any — エンドポイントから任意（**dbgacEpToAny**）

手順

ステップ 1 REST API を使用して EP_to_EP ポリシーを作成するには、次の例のような XML を使用します。

例：

```
<dbgacEpToEp name="EP_to_EP_Policy" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/acEpToEp-EP_to_EP_Policy" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EP_Filter" ownerTag="" ownerKey="" descr=""
srcPort="https" prot="tcp" dstPort="https"/>
</dbgacEpToEp>
```

ステップ 2 REST API を使用して EP_to_EPG ポリシーを作成するには、次の例のような XML を使用します。

例：

```
<dbgacEpToEpg name="EP_to_EPG_Pol" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/epToEpg-EP_to_EPG_Pol" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EPG_Filter" ownerTag="" ownerKey="" descr=""
srcPort="http" prot="tcp" dstPort="http"/>
<dbgacRsToAbsEpg tDn="uni/tn-Tenant64/ap-VRF64_app_prof/epg-EPG64"/>
</dbgacEpToEpg>
```

REST API とアトミック カウンタを一緒に使用したトラブルシューティング

手順

ステップ 1 ファブリック内に展開されたエンドポイント間アトミック カウンターのリストと、ドロップされたパケットの統計情報やパケット数などの関連する詳細を取得するには、次の例のように XML で **dbgEpToEpTsIt** クラスを使用します。

例：

```
https://apic-ip-address/api/node/class/dbgEpToEpRslt.xml
```

ステップ 2 外部 IP からエンドポイントへのアトミック カウンターと関連する詳細のリストを取得するには、次の例のように、XML で **dbgacExtToEp** クラスを使用します。

例：

```
https://apic-ip-address/api/node/class/dbgExtToEpRslt.xml
```

デジタル オプティカル モニタリング（DOM） 統計をイネーブル化と表示

リアルタイムのデジタル オプティカル モニタリング（DOM）データは SFP、SFP+、および XFP から定期的に収集され、警告およびアラームのしきい値テーブル値と比較されます。収集された DOM データは、トランシーバ送信バイアス電流、トランシーバ送信電力、トランシーバ受信電力、およびトランシーバ電源電圧です。

GUI を使用したデジタルオプティカルモニタリング（DOM）をイネーブル化

物理インターフェイスに関するデジタル オプティカル モニタリング（DOM）統計を表示する前に、ポリシー グループに関連付けられたスイッチ ポリシーを使用して、リーフインターフェイスまたはスパイン インターフェイスで DOM を有効にします。

GUI を使用して DOM を有効にするには：

手順

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

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[モニタリング（Monitoring）]** > **[ファブリック ノード コントロール（Fabric Node Controls）]**を展開します。

ステップ 3 **[ファブリック ノード コントロール（Fabric Node Controls）]**を展開して、既存のポリシーのリストを表示します。

ステップ 4 **[ワーク（Work）]** ペインで **[アクション（ACTIONS）]** ドロップダウン メニューをクリックして、**[ファブリック ノード コントロールを作成（Create Fabric Node Control）]**を選択します。

[ファブリック ノード コントロールを作成（Create Fabric Node Controls）] ダイアログ ボックスは、表示されます。

ステップ 5 **[ファブリック ノード コントロールを作成（Create Fabric Node Control）]** ダイアログボックスで、次の操作を実行します：

a) [Name] フィールドにポリシーの名前を入力します。

b) （省略可）**[説明]** フィールドに、ポリシーの説明を入力します。

c) **[DOM を有効にする（Enable DOM）]** の横にあるボックスにチェックを入れます。

ステップ 6 **[送信]** をクリックしてポリシーを作成します。

これで、次の手順で説明するように、このポリシーをポリシー グループとプロファイルに関連付けることができます。

ステップ 7 **[ナビゲーション（Navigation）]** ウィンドウで **[スイッチポリシー（Switch Policies）][ポリシー グループ（Policy Groups）]** を展開します。

ステップ 8 **[ワーク（Work）]** ペインで、**[アクション（ACTIONS）]** ドロップダウン メニューをクリックし、**[リーフ スイッチ ポリシー グループを作成（Create Leaf Switch Policy Group）]** (スパインの場合は、**[スパイン スイッチ ポリシー グループを作成（Create Spine Switch Policy Group）]**) を選択します。

[リーフ スイッチ ポリシー グループの作成（Create Leaf Switch Policy Group）] または **[スパイン スイッチ ポリシー グループの作成（Create Spine Switch Policy Group）]** ダイアログ ボックスが表示されます。

ステップ 9 ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドにポリシー グループの名前を入力します。

b) **[ノード コントロール ポリシー（Node Control Policy）]** ドロップダウン メニューから、既存のポリシー (先ほど作成したものなど) を選択するか、**[ファブリック ノード コントロールを作成（Create Fabric Node Control）]** を選択して新しいポリシーを選択します。

c) [送信（Submit）] をクリックします。

ステップ 10 次のように、作成したポリシー グループをスイッチにアタッチします。

a) **[ナビゲーション（Navigation）]** ペインで、**[スイッチ ポリシー（Switch Policies）]** > **[プロファイル（Profiles）]** を展開します。

b) **[ワーク（Work）]**ペインで、**[アクション（ACTIONS）]**ドロップダウン メニューをクリックし、必要に応じて **[リーフ スイッチ プロファイルを作成（Create Leaf Switch Profile）]** または **[スパイン スイッチ プロファイルを作成（Create Spine Switch Profile）]** を選択します。

c) ダイアログボックスの中で、**[名前（Name）]** フィールドにプロファイルのための名前を入力します。field.

d) **[スイッチの関連付け（Switch Associations）]** で、プロファイルに関連付けるスイッチの名前を追加します。
e) **[ブロック（Block）]** プルダウン メニューから、該当するスイッチの横にあるボックスをオンにします。
f) **[ポリシー グループ（Policy Group）]** プルダウン メニューから、前に作成したポリシー グループを選択します。
g) **[アップデート（Update）]** をクリックし、**[送信（Submit）]** をクリックします。

REST API を使用したデジタル オプティカル モニタリング（DOM）をイネーブル化

物理インターフェイスに関するデジタル オプティカル モニタリング（DOM）統計を表示する前に、インターフェイスで DOM を有効にします。

REST API を使用して DOM を有効にするには：

手順

ステップ 1 次の例のように、ファブリック ノード制御ポリシー（fabricNodeControlPolicy）を作成します。

```
<fabricNodeControl dn="uni/fabric/nodecontrol-testdom" name="testdom" control="1"
rn="nodecontrol-testdom" status="created" />
```

ステップ 2 次のように、ファブリック ノード制御ポリシーをポリシー グループに関連付けます。

```
<?xml version="1.0" encoding="UTF-8" ?>
 <fabricLeNodePGrp dn="uni/fabric/funcprof/lenodepgrp-nodegrp2" name="nodegrp2"
rn="lenodepgrp-nodegrp2" status="created,modified" >

    <fabricRsMonInstFabricPol tnMonFabricPolName="default" status="created,modified" />
    <fabricRsNodeCtrl tnFabricNodeControlName="testdom" status="created,modified" />

</fabricLeNodePGrp>
```

ステップ 3 次のように、ポリシー グループをスイッチに関連付けます（次の例では、スイッチは 103 です）。

```
<?xml version="1.0" encoding="UTF-8" ?>
 <fabricLeafP>
  <attributes>
   <dn>uni/fabric/leprof-leafSwitchProfile</dn>
   <name>leafSwitchProfile</name>
   <rn>leprof-leafSwitchProfile</rn>
   <status>created,modified</status>
  </attributes>
  <children>
   <fabricLeafS>
    <attributes>
     <dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typ-range</dn>
     <type>range</type>
```

```
     <name>test</name>
     <rn>leaves-test-typ-range</rn>
     <status>created,modified</status>
    </attributes>
    <children>
     <fabricNodeBlk>
      <attributes>

<dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typ-range/nodeblk-09533c1d228097da</dn>

       <from_>103</from_>
       <to_>103</to_>
       <name>09533c1d228097da</name>
       <rn>nodeblk-09533c1d228097da</rn>
       <status>created,modified</status>
      </attributes>
     </fabricNodeBlk>
    </children>
    <children>
     <fabricRsLeNodePGrp>
      <attributes>
       <tDn>uni/fabric/funcprof/lenodepgrp-nodegrp2</tDn>
       <status>created</status>
      </attributes>
     </fabricRsLeNodePGrp>
    </children>
   </fabricLeafS>
  </children>
 </fabricLeafP>
```

デジタル オプティカル モニタリング（DOM） 統計と GUI を一緒に表示

GUI を使用して DOM 統計を表示するには：

始める前に

インターフェイスの DOM 統計を表示するには、事前にインターフェイスのデジタル オプティカル モニタリング（DOM）統計を有効にしておく必要があります。

手順

ステップ 1 メニューバーから**[ファブリック（Frabric）]** and **[インベントリ（Inventory）]**を選択します。

ステップ 2 ナビゲーション ウィンドウで、調査している物理インターフェイスがあるポッドおよびリーフ ノードを展開します。

ステップ 3 **[インターフェイス（Interface）]** を展開します。

ステップ 4 **[物理 インターフェイス（Physical Interfaces）]**を拡大します。

ステップ 5 調査している物理インターフェイスを展開します。

ステップ 6 **[DOM 統計（DOM Stats）]** を選択します。

インターフェイスの DOM 統計が表示されます。

REST API によるデジタル オプティカル モニタリング（DOM）を使用したトラブルシューティング

XML REST API クエリを使用して DOM 統計を表示するには：

始める前に

インターフェイスの DOM 統計を表示する前に、インターフェイスでデジタル オプティカル モニタリング（DOM）を有効にしておく必要があります。

手順

次の例は、REST API クエリを使用して、ノード 104 の eth1 / 25 の物理インターフェイスで DOM 統計を表示する方法を示しています：

```
GET
https://apic-ip-address/api/node/mo/topology/pod-1/node-104/sys/phys-[eth1/25]/phys/domstats.xml?
query-target=children&target-subtree-class=ethpmDOMRxPwrStats&subscription=yes
```

次の応答が返されます：

```
response : {
   "totalCount":"1",
      "subscriptionId":"72057611234705430",
          "imdata":[
{"ethpmDOMRxPwrStats":{
   "attributes":{
      "alert":"none",
      "childAction":"",
      "dn":"topology/pod-1/node-104/sys/phys[eth1/25]/phys/domstats/rxpower",
      "hiAlarm":"0.158490",
      "hiWarn":"0.079430",
      "loAlarm":"0.001050",
      "loWarn":"0.002630",
      "modTs":"never",
      "status":"",
       "value":"0.139170"}}}]}
```

正常性スコアの概要を表示

APICは、ポリシーモデルを使用してデータを正常性スコアに組み入れます。正常性スコアはインフラストラクチャ、アプリケーション、またはサービスなどさまざまなエリアで集約できます。正常性スコアを使用すると、ネットワーク階層をドリルダウンして障害を特定の管理対象オブジェクト（MO）に分離することにより、パフォーマンスの問題を分離できます。アプ

リケーションの状態（テナントごと）またはリーフ スイッチの状態（ポッドごと）を表示することで、ネットワークの状態を表示できます。

正常性スコア、エラー、正常性 スコアの計算については、『*Cisco APIC Fundamentals Guide*』を参照してください。

ヘルス スコアのタイプ

APIC は次の正常性 スコアのタイプをサポートします。

- システム — ネットワーク全体の正常性を要約します。
- リーフ—ネットワークのリーフ スイッチの正常性を要約します。リーフ正常性には、ファン トレイ、電源、および CPU を含むスイッチのハードウェア正常性が含まれます。
- テナント — テナントとテナントのアプリケーションの正常性を要約します。

正常性 スコアによるフィルタ処理

次のツールを使用して、正常性 スコアをフィルタ処理できます。

- 正常性 スクロール バー：正常性 スクロール バーを使って、どのオブジェクトを表示するかを指定できます。スコアを下げれば、正常性スコアの低いオブジェクトだけ見ることができます。
- 劣化した正常性スコアの表示：劣化した正常性スコアを表示するには、ギア アイコンをクリックし、**[劣化した正常性スコアのみを表示（Show only degraded health score）]** を選択します。

テナントの正常性の表示

アプリケーションの健全性を表示するには、メニュー バーで **[テナント（Tenants）]** > *[tenant-name]* をクリックし、次に **[ナビゲーション（Navigation）]** ペインでテナント名をクリックします。GUI がアプリケーションや EPG を含むテナントの正常性の要約を表示します。テナントの構成をドリルダウンするには、正常性 スコアをダブルクリックします。

健全性の要約の場合は、**[仕事（Work）]** ペインの **[正常性（Health）]** タブをクリックします。ネットワークのこの表示が正常性 スコアとネットワーク上の MO 間の関係を示すので、パフォーマンスの問題を分離し、解決することができます。たとえば、テナントのコンテクストの管理オブジェクトの共通シーケンスは、**[テナント（Tenant）]** > **[アプリケーション プロファイル（Application profile）]** > **[アプリケーション EPG（Application EPG）]** > **[EPP]** > **[ファブリックの場所（Fabric location）]** > **[EPG からパス アタッチメント（ EPG to Path Attachment ）]** > **[ネットワーク パス エンドポイント（Network Path Endpoint）]** > **[集約インターフェイス（Aggregation Interface）]** > **[集約されたインターフェイス（Aggregated Interface）]** > **[集約されたメンバー インターフェイス（Aggregated Member Interface）]** となります。

ファブリックの正常性の表示

ファブリックの正常性を表示するには、メニューバーの**[ファブリック（Fabric）]**をクリックします。**[ナビゲーション（navigation）]**のペインで、ポッドを選択します。GUI は、ノードを含むポッドの正常性の要約を表示します。ファブリック構成の一部をドリル ダウンするには、正常性 スコアをダブルクリックします。

健全性の要約の場合は、**[仕事（work）]**ペインの**[正常性（Health）]**タブをクリックします。ネットワークのこの表示が正常性 スコアとネットワーク上の MO 間の関係を示すので、パフォーマンスの問題を分離し、解決することができます。たとえば、ファブリックのコンテキストにおける管理対象オブジェクトの共通シーケンスは、**[ポッド（Pod）]> [リーフ（Leaf）] > [シャーシ（Chassis）] > [ファン トレイ スロット（Fan tray slot）] > [回線モジュールのスロット（Line module slot）] >[回線モジュール（Line module ）] > [ファブリック ポート（Fabric Port）] > [レイヤ 1 物理インターフェイス構成（Layer 1 Physical Interface Configuration）] > [物理インターフェイス実行時間状態（Physical Interface Runtime State）です。]**

（注） 物理ネットワークの問題など、ファブリックの問題は、MOが直接関連するとテナントのパフォーマンスに影響を及ぼすことがあります。

Visore での MO 正常性の表示

Visore で MO の正常性を表示するには、**H** アイコンをクリックします。

次の MO を使って、正常性情報を表示します。

- 正常性：Inst
- 正常性：NodeInst
- オブザーバ：ノード
- オブザーバ：Pod

Visoreに関する詳細情報については、『*Cisco Application Centric Infrastructure Fundamentals*』ガイドを参照してください。

ログを使用する正常性 スコアのデバッグ

次のログ ファイルを使用して、APIC の正常性 スコアをデバッグできます。

- svc_ifc_eventmgr.log
- svc_ifc_observer.log

ログを使用して正常性 スコアをデバッグする場合、次の項目を確認してください：

- syslog （エラーまたはイベント）の送信元を確認します。

- syslog ポリシーが APIC で構成されているかどうかを確認します。
- syslogポリシータイプおよびシビラティ（重大度）が正しく設定されているかどうかを確認します。
- コンソール、ファイル、RemoteDest、または教授の syslog 接続先を指定できます。RemoteDestの場合、syslogサーバーが実行中であり、到達可能であることを確認します。

エラーの表示

次の手順では、障害情報を表示する場所について説明します。

手順

ステップ 1 障害ウィンドウに移動します。

- システム障害（System Faults）：メニュー バーから、**[システム（System）]** > **[障害（Faults）]** をクリックします。
- テナント障害（Tenant Faults）：メニュー バーから、
  1. **[テナント（Tenants）]** > *[tenant-name]* をクリックします。
  2. **[ナビゲーション（Navigation）]** ペインで、**[テナント（Tenant）]** *[*テナント名（*tenant name*）*]* をクリックします。
  3. **[作業（Work）]** ペインで、**[障害（Faults）]** タブをクリックします。
- ファブリック障害（Fabric Faults）：メニューバーから
  1. **[ファブリック（Fabric）]** > **[インベントリ（Inventory）]** をクリックします。
  2. **[ナビゲーション（Navigation）]** ペインで、**ポッド**をクリックします。
  3. **[作業（Work）]** ペインで、**[障害（Faults）]** タブをクリックします。

障害のリストが要約表に表示されます。

ステップ 2 障害をダブルクリックします。

ファブリック テーブルとシステム テーブルが変更され、クリックした障害の障害コードに一致する障害が表示されます。

a) ファブリックまたはシステムの障害から、サマリーテーブルの障害をダブルクリックして詳細を表示します。

[障害のプロパティ（Fault Properties）] ダイアログが表示され、次のタブが表示されます。

- **一般（General）**：以下を表示します。
  - **プロパティ（Properties）**：サマリー テーブルにある情報が含まれます

- **詳細（Details）**：サマリー テーブルで見つかった障害情報、発生数、変更セット、および選択した障害の元、以前、および最高の重大度レベルが含まれます。

- **トラブルシューティング（Troubleshooting）**：次のとおり、表示します。
  - **トラブルシューティング（Troubleshooting）**：障害の説明と推奨されるアクションを含むトラブルシューティング情報が含まれていま
  - **監査ログ（Audit log）**：障害が発生する前にユーザーが開始したイベントの履歴を表示できるツール。指定した分数ごとに履歴が一覧表示されます。ドロップダウン矢印をクリックして、分数を調整できます。

- **履歴** — 影響を受けるオブジェクトの履歴情報を表示します

アップリンク障害検出のためのポート トラッキング をイネーブル化

このセクションでは、GUI、NX-OS CLI、および REST API を使用してポート トラッキングを有効にする方法について説明します。

ファブリックポートの障害検出のためのポートトラッキングポリシー

ファブリック ポートの障害検出は、ポート トラッキング システム設定で有効にすることができます。ポートトラッキングポリシーは、リーフスイッチとスパインスイッチ間のファブリック ポート、およびティア 1 リーフスイッチとティア 2 リーフスイッチ間のポートのステータスを監視します。有効なポート トラッキングポリシーがトリガーされると、リーフ スイッチは、EPGによって導入されたスイッチ上のすべてのアクセスインターフェイスをダウンさせます。

[ポート トラッキングがトリガーされたときに APIC ポートを含める（Include APIC ports when port tracking is triggered）] オプションを有効にした場合、リーフスイッチがすべてのファブリック ポートへの接続を失うと（つまり、ファブリック ポートが 0 になると）、ポート トラッキングは Cisco Application Policy Infrastructure Controller（APIC）ポートを無効にします。Cisco APIC がファブリックに対してデュアルまたはマルチホームの場合にのみ、この機能を有効にしてください。Cisco APIC ポートを停止すると、デュアルホームの Cisco APIC の場合にセカンダリ ポートに切り替えるのに役立ちます。

（注） ポートトラッキングの設定は、**[システム（System）]> >[システム設定（System Settings）] > > [ポート トラッキング（Port Tracking）]** で行えます。

ポート トラッキング ポリシーは、ポリシーをトリガーするファブリック ポート接続の数と、指定されたファブリック ポートの数を超えた後にリーフスイッチアクセスポートをバックアップするための遅延タイマーを指定します。

次の例は、ポート トラッキング ポリシーの動作を示しています。

- ポート トラッキング ポリシーは、ポリシーをトリガーする各リーフ スイッチのアクティブなファブリック ポート接続のしきい値が 2 であると指定しています。
- ポート トラッキング ポリシーは、リーフ スイッチからスパイン スイッチへのアクティブなファブリック ポート接続の数が 2 に低下したときにトリガーされます。
- 各リーフスイッチは、そのファブリック ポート接続を監視し、ポリシーで指定されたしきい値に従ってポート トラッキング ポリシーをトリガーします。
- ファブリック ポート接続が復旧すると、リーフ スイッチは遅延タイマーの設定時間が経過するのを待ってから、アクセス ポートを復旧します。これにより、トラフィックがリーフスイッチ アクセス ポートで再開可能になる前に、ファブリックが再コンバージェンスする時間が与えられます。大規模なファブリックでは、遅延タイマーをより長い時間に設定する必要がある場合があります。

（注） このポリシーを構成するときは注意してください。ポート トラッキングをトリガーする、アクティブなスパイン ポートの数に関するポート トラッキング設定が高すぎる場合、すべてのリーフ スイッチ アクセス ポートがダウンします。

GUI を使用したポート トラッキングの構成

この手順では、GUIを使用してポート トラッキング機能を使用する方法について説明します。

手順

ステップ 1 **[システム（System）]**メニューから、**[システム設定（System Settings）]**を選択します。

ステップ 2 ナビゲーションウィンドウから**[ポート トラッキング（Port Tracking）]**を選択します。

ステップ 3 **[ポート トラッキング状態（Port tracking state）]** の横にある **[オン（on）]** を選択して、ポート トラッキング機能をオンにします。

ステップ 4 プロパティのポート トラッキング状態の横にある **[オフ（off）]** を選択して、ポート トラッキング機能をオフにします。

ステップ 5 （任意） **[遅延復元タイマー（Delay restore timer）]**をデフォルト（120 秒）からリセットします。

ステップ 6 ポート トラッキングがトリガーされる前に稼働している現用系スパイン リンクの最大数（0 ～ 12 の任意の構成値）を入力します。

ステップ 7 **[送信（Submit）]** をクリックして、目的のポート トラッキング構成をファブリック上のすべてのスイッチにプッシュします。

NX-OS CLI を使用したポート トラッキング

この手順では、NX-OS CLI を使用してポート トラッキング機能を使用する方法について説明します。

手順

ステップ 1 次のようにポート トラッキング機能をオンにします。

例：

```
apic1# show porttrack
Configuration
Admin State                   : on
Bringup Delay(s)              : 120
Bringdown # Fabric Links up   : 0
```

ステップ 2 次のように、ポート トラッキング機能をオフにします。

例：

```
apic1# show porttrack
Configuration
Admin State                   : off
Bringup Delay(s)              : 120
Bringdown # Fabric Links up   : 0
```

REST API を使用した ポート トラッキング

始める前に

この手順では、REST API を使用してポート トラッキング機能を使用する方法について説明します。

手順

ステップ 1 次のように REST API を使用してポート トラッキング機能をオンにします（**[管理状態：on（admin state: on）]**）。

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="on">

</infraPortTrackPol>
```

```
</infraInfra>
</polUni>
```

ステップ 2 次のように REST API を使用してポート トラッキング機能をオフにします（**[管理状態：off（admin state: off）]**）

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="off">

</infraPortTrackPol>
</infraInfra>
</polUni>
```

デバイスのモニタリングおよび管理用 SNMP の構成

このセクションでは、GUI を使用して SNMP を構成する方法について説明します。

SNMP について

Cisco Application Centric Infrastructure（ACI）は、管理情報ベース（MIB）と通知（トラップ）を含む広範な SNMPv1、v2、および v3 のサポートを提供します。SNMP 標準では、Cisco ACI ファブリックを管理しモニタリングする各 MIB をサポートするサードパーティ製アプリケーションを使用できます。

SNMPv3 はさらに広範なセキュリティ機能を提供します。各 SNMPv3 デバイスで SNMP サービスを有効または無効にするように選択できます。また、各デバイスで SNMP v1 および v2 要求の処理方法を設定できます。

5.1(1) リリース以降、SNMPv3 は Secure Hash Algorithm-2（SHA-2）認証タイプをサポートします。

SNMP の使用方法の詳細については、『*Cisco ACI MIB Quick Reference*』を参照してください。

Cisco ACI での SNMP アクセスのサポート

（注） Cisco Application Centric Infrastructure（ACI）でサポートされる MIB の完全なリストについては、 http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/list/mib-support.htmlを参照してください。

Cisco ACI での SNMP サポートは次のとおりです。

- SNMP 読み取りクエリー（Get、Next、Bulk、Walk）は、リーフおよびスパイン スイッチと Cisco Application Policy Infrastructure Controller（APIC）によってサポートされます。

- SNMP 書き込みコマンド（Set）は、リーフおよびスパイン スイッチまたは Cisco APIC によってサポートされません。
- SNMP トラップ（v1、v2c、および v3）は、リーフおよびスパイン スイッチと Cisco APIC によってサポートされます。

（注） Cisco ACI は最大 10 個のトラップ レシーバをサポートします。

- SNMPv3 は、リーフおよびスパイン スイッチと Cisco APIC によってサポートされます。
- Cisco APIC IPv6 アドレスを使用した SNMP はサポートされていません。

表 *1 : Cisco APIC* リリースでの *SNMP* サポートの変更

リリース	説明
1.2(2)	SNMP トラップの宛先として IPv6 サポートを追加。
1.2(1)	Cisco APIC コントローラの SNMP サポートを追加。以前のリリースでは、リーフおよびスパイン スイッチについてのみ SNMP がサポートされています。

GUI による SNMP ポリシーの設定

この手順では、ACI スイッチの SNMP ポリシーを設定し、有効にします。

始める前に

SNMP 通信を有効にするには、以下の設定が必要です。

- アウトオブバンド コントラクトを設定して SNMP トラフィックを許可します。SNMP トラフィックは、通常、SNMP 要求に UDP ポート 161 を使用します。
- 'mgmt' テナントで APIC アウトオブバンド IP アドレスを設定します。アウトオブバンド アドレスは APIC セットアップ時に設定されますが、アウトオブバンド コントラクトを有効にするには 'mgmt' テナントでアドレスを明示的に設定する必要があります。

手順

ステップ 1 メニュー バーで、[Fabric] をクリックします。
ステップ 2 サブメニュー バーで、[Fabric Policies] をクリックします。
ステップ 3 **[Navigation]** ペインで、**[Pod Policies]** を展開します。
ステップ 4 **[Pod Policies]** の下で **[Policies]** を展開します。

ステップ 5 **[SNMP]** を右クリックし、**[Create SNMP Policy]** を選択します。

新しい SNMP ポリシーを作成する代わりに、次の手順で示されるものと同じ方法で **[default]** ポリシー フィールドを編集できます。

ステップ 6 SNMP ポリシーのダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、SNMP ポリシーの名前を入力します。

b) **[Admin State]** フィールドで、**[Enabled]** を選択します。

c) （任意） **[SNMP v3 Users]** テーブルで **[+]** アイコンをクリックし、**名前**を入力して、ユーザの認証データを入力し、**[Update]** をクリックします。

この手順は SNMPv3 アクセスが必要な場合のみ実行します。

d) **[コミュニティ ポリシー（Community Policies）]** テーブルで **[+]** アイコンをクリックし、**[名前（Name）]** を入力して、**[更新（Update）]** をクリックします。

コミュニティポリシー名の最大長は32文字です。名前には、アンダースコア（_）、ハイフン（-）、またはピリオド（。）の文字、数字、および特殊文字のみを使用できます。名前に @ 記号を含めることはできません。

e) **[Trap Forward Servers]** テーブルで、**[+]**アイコンをクリックし、外部サーバの**[IP Address]** を入力し、**[Update]**をクリックします。

ステップ 7 必須: 許可された SNMP 管理ステーションを設定するには、SNMP ポリシーのダイアログボックスで、次の操作を実行します。

a) **[Client Group Policies]** テーブルで **[+]** アイコンをクリックし、**[Create SNMP Client Group Profile]** ダイアログボックスを開きます。

b) **[Name]** フィールドに、SNMP クライアント グループのプロファイル名を入力します。

c) **[Associated Management EPG]** ドロップダウン リストから管理 EPG を選択します。

d) **[Client Entries]** テーブルで **[+]** アイコンをクリックします。

e) **[Name]** フィールドにクライアントの名前を入力し、**[Address]** のフィールドにクライアントの IP アドレスを入力して、**[Update]** をクリックします。

（注） SNMP 管理ステーションが SNMPv3 を使用して APIC と接続する場合、APIC は SNMP クライアント グループのプロファイルに指定されたクライアント IP アドレスを強制しません。SNMPv3 の場合、管理ステーションが **[Client Entries]** リストに含まれている必要がありますが、SNMPv3 クレデンシャルのみでアクセス可能なため、IP アドレスが一致している必要はありません。

ステップ 8 [OK] をクリックします。

ステップ 9 **[送信（Submit）]** をクリックします。

ステップ 10 **[Pod Policies]** の下で **[Policy Groups]** を展開して、ポリシー グループを選択するか、または **[Policy Groups]** を右クリックし、**[Create POD Policy Group]** を選択します。

新しいポッド ポリシー グループを作成することも、既存のグループを使用することもできます。ポッド ポリシー グループには、SNMP ポリシーに加えて他のポッド ポリシーを含めることができます。

ステップ 11 ポッド ポリシー グループのダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、ポッド ポリシー グループの名前を入力します。

b) **[SNMP Policy]** ドロップダウン リストから、設定した SNMP ポリシーを選択して、**[Submit]** をクリックします。

ステップ 12 **[Pod Policies]** の下で **[Profiles]** を展開し、**[default]** をクリックします。

ステップ 13 **[Work]** ペインで、**[Fabric Policy Group]** ドロップダウン リストから、作成したポッド ポリシー グループを選択します。

ステップ 14 [送信（Submit）] をクリックします。

ステップ 15 [OK] をクリックします。

GUI による SNMP トラップ通知先の設定

この手順では、SNMP トラップ通知を受信する SNMP マネージャのホスト情報を設定します。

（注） ACI は最大 10 個のトラップ レシーバをサポートします。10 個より多く設定すると、一部では通知が受信されません。

手順

ステップ 1 メニュー バーで、[Admin] をクリックします。

ステップ 2 サブメニュー バーで、[External Data Collectors] をクリックします。

ステップ 3 **[Navigation]** ペインで、**[Monitoring Destinations]** を展開します。

ステップ 4 **[SNMP]** を右クリックし、**[Create SNMP Monitoring Destination Group]** を選択します。

ステップ 5 **[Create SNMP Monitoring Destination Group]** ダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、SNMP 通知先の名前を入力し、**[Next]** をクリックします。

b) **[Create Destinations]** テーブルで **[+]** アイコンをクリックし、**[Create SNMP Trap Destination]** ダイアログボックスを開きます。

c) **[ホスト名/IP（Host Name/IP）]** フィールドに、IPv4 または IPv6 アドレスまたは宛先ホストの完全修飾ドメイン名を入力します。

d) 通知先の**ポート**番号と SNMP **バージョン**を選択します。

e) SNMP v1 または v2c 通知先の場合、**[Security Name]** として設定したコミュニティ名の 1 つを入力し、**[v3 Security Level]** として **[noauth]** を選択します。

SNMP v1 または v2c セキュリティ名の最大長は 32 文字です。名前には、アンダースコア（_）、ハイフン（-）、またはピリオド（。）の文字、数字、および特殊文字のみを使用できます。SNMP v2c の場合、@ 記号も使用できます。

f) SNMP v3 通知先の場合、**[Security Name]** として設定したユーザ名の 1 つを入力し、必要な **[v3 Security Level]** を選択します。

SNMP v3 セキュリティ名の最大長は 32 文字です。名前は大文字または小文字で始まる必要があり、文字、数字、およびアンダースコア（_）、ハイフン（-）、ピリオド（。）、または@記号の特殊文字のみを使用できます。

g) **[Management EPG]** ドロップダウン リストから管理 EPG を選択します。
h) [**OK**] をクリックします。
i) **[終了]** をクリックします。

GUI による SNMP トラップ ソースの設定

この手順では、ファブリック内のソース オブジェクトを選択して有効にし、SNMP トラップ通知を生成します。

手順

ステップ 1 メニュー バーで、[Fabric] をクリックします。
ステップ 2 サブメニュー バーで、[Fabric Policies] をクリックします。
ステップ 3 [Navigation] ペインで、[Monitoring Policies] を展開します。

共通ポリシー、**デフォルト** ポリシーで SNMP ソースを作成することも、または新しいモニタリング ポリシーを作成することもできます。

ステップ 4 必要なモニタリング ポリシーを展開し、[Callhome/SNMP/Syslog] を選択します。

[Common Policy] を選択する場合は、[Common Policy] を右クリックして、[Create SNMP Source] を選択し、そのダイアログボックスで次の手順に従ってください。

ステップ 5 [Work] ペインで、[Monitoring Object] ドロップダウン リストから [ALL] を選択します。
ステップ 6 [Source Type] ドロップダウン リストから、[SNMP] を選択します。
ステップ 7 テーブルで + アイコンをクリックし、[Create SNMP Source] ダイアログボックスを開きます。
ステップ 8 [Create SNMP Source] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、SNMP ポリシーの名前を入力します。
b) [Dest Group] ドロップダウン リストから、通知を送信する既存の宛先を選択するか、または [Create SNMP Monitoring Destination Group] を選択して、新しい宛先を作成します。

SNMP の通知先グループを作成する手順は、別項で説明します。

c) [送信（Submit）] をクリックします。

SNMP を使用したシステムのモニタリング

個々のホスト（APIC またはその他のホスト）をリモートでモニタし、特定のノードの状態を確認できます。

SNMP を使用してシステムの CPU とメモリの使用状況をチェックし、CPU のスパイクが発生しているかどうかを確認できます。SNMP（ネットワーク管理システム）は、SNMP クライアントを使用して APIC の情報にアクセスし、情報を取得します。

リモートでシステムにアクセスして、情報がネットワーク管理システムのコンテキストに属するものかどうかを確認し、CPU またはメモリの使用量が多すぎないか、またはシステムやパフォーマンスの問題が発生しているかどうかを調べることができます。問題の原因がわかると、システムの正常性をチェックし、メモリまたは CPU の使用量が多すぎないかどうかを確認できます。

詳細については、「*Cisco ACI MIB Quick Reference Manual*」を参照してください。

トラフィック モニタリングの SPAN の構成

このセクションでは、SPAN のガイドラインと制約事項をリストし、SPAN セッションの構成方法について説明します。

SPAN の概要

スイッチドポートアナライザ（SPAN）ユーティリティを使って、詳細なトラブルシューティングの実行または特定のアプリケーションホストからトラフィックのサンプルを取得し、プロアクティブなモニタリングと分析を行うことができます。

SPANは1つ以上のポート、VLAN、またはエンドポイントグループ（EPG）からのトラフィックをコピーし、ネットワークアナライザによる分析のためにコピーしたトラフィックを1つ以上の送信先に送信します。このプロセスはどの接続デバイスも中断せず、ハードウェア内で実施されるので不要な CPU 負荷を防ぎます。

SPANセッションはソースが受信したトラフィック（入力トラフィック）、ソースから送信したトラフィック（出力トラフィック）、またはその両方をモニタリングするように設定できます。デフォルトでは、SPAN はすべてのトラフィックをモニタリングしますが、選択したトラフィックだけをモニタリングするようにフィルタを設定できます。

テナントまたはスイッチで SPAN を設定できます。スイッチ上で設定する場合、SPAN をファブリック ポリシーまたはアクセス ポリシーとして設定できます。

APIC は、SPAN（ERSPAN）のカプセル化されたリモート拡張をサポートします。

リリース 4.1(1i) 以降、次の機能がサポートされるようになりました。

- 送信元とポートチャネルが同じスイッチ上でローカルである限り、宛先として静的ポートチャネルを使用した、ローカル SPAN に対するサポート。

（注） APIC リリース 4.1(1i) 以降を実行していて、宛先として静的ポートチャネルを設定した後、4.1(1i) より前のリリースにダウングレードすると、これが原因でSPANセッションが管理者無効状態になります。この機能は、リリース 4.1.(1i) より前には利用できませんでした。機能への影響はありません。

- レイヤ 3 インターフェイス フィルタリングを使用して送信元 SPAN を設定するときに、レイヤ 3 インターフェイスの IP プレフィックスを含める必要がなくなりました。
- 1 つ以上のフィルタ エントリのグループであるフィルタ グループ設定のサポート。フィルタ グループを使用すれば、受信したパケットを SPAN を使用して分析する必要があるかどうかを判断するために使用される一致基準が指定できます。
- ASIC の入力での転送が原因でドロップされたパケットをキャプチャし、事前設定された SPAN 宛先に送信する SPAN-on-drop 機能。SPAN-on-drop 設定には、アクセス ポートを SPAN 送信元として使用するアクセス ドロップ、ファブリック ポートを SPAN 送信元として使用するファブリック ドロップ、およびノード上のすべてのポートを SPAN 送信元として使用するグローバル ドロップの 3 種類があります。SPAN-on-drop は、通常の SPAN を使用し (CLI、GUI、および REST API 経由) とトラブルシューティング SPAN を使用して (CLI および REST API のみを経由) 設定されます。この機能の設定の詳細については、*GUI* を使用した *SPAN* の設定、*NX-OS* スタイル *CLI* を使用した *SPAN* の設定、および *REST API* を使用した *SPAN* の設定を参照してください。

マルチノード SPAN

APIC のトラフィックのモニタリング ポリシーは、各アプリケーション グループのすべてのメンバーと彼らが接続する場所を追跡するために、適切な範囲にポリシーのスパンを広げることが可能です。メンバーが移動すると、APIC は新しいリーフにポリシーを自動的にプッシュします。たとえば、エンドポイントが新しいリーフ スイッチに VMotion により移動すると、スパンの設定は自動的に調整されます。

ACI ファブリックは、カプセル化リモート SPAN（ERSPAN）形式の次の 2 つの拡張をサポートします。

- アクセスまたはテナント SPAN：VLAN をフィルタとして使用するかどうかにかかわらず、リーフ スイッチのフロント パネル ポートに対して実行されます。リーフ スイッチの Broadcom Trident 2 ASIC は、ERSPAN タイプ 1 形式とはわずかに異なるバージョンをサポートします。上記で参照したドキュメントで定義されている ERSPAN タイプ 1 フォーマットとは、GRE ヘッダーが 4 バイトのみであり、シーケンス フィールドがないという点で異なっています。GREヘッダーは常に次のようにエンコードされます–0x000088be。0x88be は ERSPAN タイプ 2 を示していますが、フィールドの残りの 2 バイトにより、これは 4 バイトの GRE ヘッダーを持つ ERSPAN タイプ 1 パケットとして識別されます。
- ファブリック SPAN：リーフ スイッチの Northstar ASIC により、またはスパイン スイッチの Alpine ASIC により実行されます。これらの ASIC は ERSPAN タイプ 2 および 3 フォー

マットをサポートしていますが、ACI ファブリックは現在、ファブリック SPAN の ERSPAN タイプ 2 のみをサポートしています。これについては、上記のベースライン ドキュメントに記載されています。

ERSPAN ヘッダーの説明については、次の URL にある IETF インターネット ドラフトを参照してください。https://tools.ietf.org/html/draft-foschiano-erspan-00

SPAN の注意事項と制約事項

（注） 多くのガイドラインと制約事項は、スイッチが第 1 世代スイッチか第 2 世代スイッチかによって異なります。スイッチの生成は次のように定義されます。

- 第 1 世代スイッチは、スイッチ名の末尾に「EX」、「FX」、「FX2」などのサフィックスがないことで識別されます（N9K-9312TX など）。
- 第 2 世代スイッチは、スイッチ名の末尾に「EX」、「FX」、「FX2」などのサフィックスが付いています。

- サポートされる SPAN のタイプはさまざまです。
  - 第 1 世代のスイッチの場合、テナントおよびアクセス SPAN は、カプセル化された SPAN（ERSPAN）タイプ I を使用します（Cisco Application Policy Infrastructure Controller（APIC）GUI のバージョン 1 オプション）。
  - 第 2 世代スイッチの場合、テナントおよびアクセス SPAN は、カプセル化された SPAN（ERSPAN）タイプ II（Cisco APIC GUI のバージョン 2 オプション）を使用します。
  - ファブリック SPAN は ERSPAN タイプ II を使用します。

  リリース 5.2(3) 以降、ERSPAN は IPv6 宛先をサポートしています。

  uSeg EPG または ESG は、SPAN 送信元 EPG として使用できません。これは、SPAN 送信元フィルタが VLAN ID に基づいているためです。したがって、エンドポイントが uSeg EPG または ESG に分類されている場合でも、その VLAN が SPAN 送信元 EPG の VLAN である場合、エンドポイントからのトラフィックはミラーリングされます。
- ERSPAN セッションを構成するときに、SPAN 送信元に GOLF VRF インスタンス内のスパイン スイッチからの接続先とインターフェイスが含まれている場合、L3Out プレフィックスが間違った BGP ネクストホップで GOLF ルータに送信され、GOLF からその L3Out への接続が切断されます。
- SPAN 送信元として l3extLIfP のレイヤ 3 サブインターフェイスを指定することはできません。外部ソースからのトラフィックをモニタリングするためにはポート全体を使用します。
- FEX インターフェイスのローカル SPAN では、FEX インターフェイスは SPAN 送信元としてのみ使用でき、SPAN 宛先としては使用できません。

- 第 1 世代スイッチでは、レイヤ 3 スイッチド トラフィックに対して Tx SPAN は機能しません。
- 第 2 世代のスイッチでは、トラフィックがレイヤ 2 またはレイヤ 3 のどちらでスイッチングされているかにかかわらず、Tx SPAN は機能しません。

Rx SPAN に制限はありません。

FEX ファブリック ポートチャネル（NIF）の SPAN の場合、メンバー インターフェイスは第 1 世代リーフ スイッチの SPAN 送信元インターフェイスとしてサポートされます。

（注） 第 2 世代スイッチで FEX ファブリック ポートチャネル（NIF）メンバー インターフェイスを SPAN 送信元インターフェイスとして設定することもできますが、これは Cisco APIC リリース 4.1 より前のリリースではサポートされていません。

ERSPAN ヘッダーについては、IETF の Internet Draft（https://tools.ietf.org/html/draft-foschiano-erspan-00）を参照してください。

- ERSPAN 宛先 IP アドレスは、エンドポイントとしてファブリックで学習する必要があります。
- SPAN は IPv6 トラフィックをサポートします。
- ポート チャネルまたは vPC の個別ポート メンバーは送信元として設定されます。ポート チャネル、vPC、または vPC コンポーネントを SPAN セッションの送信元として使用します。
- 宛先 EPG が削除されるか使用できない場合、ERSPAN 送信元グループで障害は発生しません。
- SPAN フィルタは、第 2 世代のリーフ スイッチでのみサポートされます。

  アクセス SPAN 送信元は、特定の時点で次のいずれかのフィルタのみをサポートします。
  - EPG
  - 外部ルーティング（L3Out）
- L3Out フィルタを使用してアクセス SPAN 送信元を展開する場合は、L3Out が一致するインターフェイスにも展開されていることを確認します。
  - L3Out がポートに展開されている場合、SPAN 送信元は同じポートに展開する必要があります。
  - L3Out が PC に展開されている場合、SPAN 送信元は同じ PC に展開する必要があります。

- L3Out が vPC に展開されている場合、SPAN 送信元は同じ vPC に展開する必要があります。

- L3Out ルーテッド インターフェイスおよびルーテッド サブインターフェイスはポートまたはPCに導入できますが、L3OutSVIはポート、PC、またはvPCに導入できます。L3Out フィルタを使用する SPAN 送信元は、それに応じて展開する必要があります。
- L3Out フィルタは、ファブリック SPAN またはテナント SPAN セッションではサポートされません。
- EPG ブリッジ ドメインの [L3 設定（L3 Configuration）] タブで正しい L3Out を選択する必要があります。そうしないと、基本的な L3Out のパケット フローが機能しません。
- カプセル化値は、ルーテッドサブインターフェイスおよびSVIには必須ですが、ルーテッドインターフェイスには適用されません。L3Outサブインターフェイスまたは SVI カプセル化値は、EPG カプセル化値とは異なる必要があります。

  SPAN セッション内で EPG フィルタが有効になっている場合、中継、つまり tx 方向のインターフェイスから送信される ARP パケットはスパンされません。
- 次の場合、SPAN フィルタはサポートされません。
  - ファブリック ポート
  - ファブリックおよびテナント SPAN セッション
  - スパイン スイッチ
- 公式にサポートされているよりも多くの L4 ポート範囲を追加しようとしても、L4 ポート範囲フィルタ エントリは追加されません。
- SPAN 送信元グループ レベルまたは個々の SPAN 送信元レベルで、サポートされているフィルタ エントリより多くのエントリを関連付けようとすると、SPANセッションは起動しません。
- 公式にサポートされているよりも多くのフィルタ エントリを追加または削除すると、削除されたフィルタ エントリは TCAM に残ります。
- アクティブな SPAN セッションの最大数や、SPAN フィルタ制限など、SPAN 関連の制限については、『*Verified Scalability Guide for Cisco ACI*』ドキュメントを参照してください。
- SPAN-on-drop 機能では、次の注意事項と制限事項が適用されます。
  - SPAN-on-drop 機能は、第 2 世代リーフ スイッチでサポートされます。
  - SPAN-on-drop 機能は、LUX ブロック内の転送ドロップがあるパケットのみをキャプチャします。これは、入力での転送ドロップ パケットをキャプチャします。SPAN-on-drop 機能は、BMX（バッファ）ドロップおよび RWX（出力）ドロップをキャプチャできません。

- トラブルシューティング CLI を使用して SPAN-on-drop と Cisco APICを有効にして宛先として SPAN セッションを作成する場合、100 MBの データがキャプチャされるとセッションは無効になります。
- モジュラ シャーシでは、SPAN-on-drop 機能はライン カードでドロップされたパケットに対してのみ機能します。ファブリック カードでドロップされたパケットはスパンされません。
- SPAN-on-drop ACL と他の SPAN ACL はマージされません。SPAN-on-drop セッションが ACL ベースの SPAN とともにインターフェイスで設定されている場合、そのインターフェイスでドロップされたパケットは SPAN-on-drop セッションにのみ送信されます。
- SPAN on drop と SPAN ACL を同じセッションで設定することはできません。
- アクセスまたはファブリック ポート ドロップ セッションとグローバル ドロップ セッションが設定されている場合、アクセスまたはファブリック ポート ドロップ セッションがグローバル ドロップ セッションよりも優先されます。
- TCAM でサポートされるフィルタ エントリの数=（M * S1 * 1 + N * S2 * 2）+（S3 * 2）。これは、rx SPAN または tx SPAN に個別に適用されます。現在この式に従うと、tx または rx SPAN でサポートされる最大フィルタ エントリは各方向で 480 です（また、フィルタ グループ アソシエーション（S3 = 0 を意味する）なしで、16 個のポート範囲を含む他の送信元が設定されていない場合））。フィルタ エントリの数が最大許容数を超えると、障害が発生します。フィルタエントリでレイヤ4ポート範囲を指定できることに注意してください。ただし、16個のレイヤ4ポートが単一のフィルタ エントリとしてハードウェアにプログラムされます。

（注）
- M = IPv4 フィルタの数
- S1 = IPv4 フィルタを使用した送信元の数
- N = IPv6 フィルタの数
- S2 = IPv6 フィルタを使用した送信元の数
- S3=フィルタ グループが関連付けられていない送信元の数

- PC または vPC の LACP ポリシーで MAC ピニングを設定すると、PC メンバー ポートは LACP 個別ポートモードになり、PC は動作しません。したがって、このような PC での SPAN 送信元設定は失敗し、「No operating src / dst」障害が生成されます。MAC ピニング モードが設定されている場合、SPAN は個々のポートでのみ設定できます。
- Cisco Application Centric Infrastructure（ACI）リーフ スイッチで受信されたパケットは、スパン インターフェイスが入力インターフェイスと出力インターフェイスの両方で設定されている場合でも、一度だけスパンされます。

- ルーテッド外部 SPAN 送信元フィルタを使用すると、Tx 方向のユニキャストのみが表示されます。Rx 方向では、ユニキャスト、ブロードキャスト、およびマルチキャストを確認できます。

- L3Out フィルタは、送信マルチキャスト SPAN ではサポートされません。L3Out は、入力 ACL フィルタでは sclass / dclass の組み合わせとして表されるため、ユニキャスト トラフィックのみを照合できます。送信マルチキャスト トラフィックは、ポートおよびポート チャネルでのみスパンできます。

- ポート チャネル インターフェイスを SPAN 宛先として使用できるのは、-EX 以降のスイッチだけです。

- SPAN フィルタ（5 タプル フィルタ）が適用されている場合、同じ送信元インターフェイスで複数の SPAN セッションを設定することはできません。

  リーフ スイッチのローカル SPAN 宛先ポートは、着信トラフィックを予期しません。レイヤ 2 インターフェイス ポリシーを設定し、**VLAN 範囲**プロパティを**グローバル範囲**ではなく**ポート ローカル範囲**に設定することで、スイッチが着信 SPAN 宛先ポート トラフィックをドロップするようにできます。このポリシーを SPAN 宛先ポートに適用します。レイヤ2インターフェイスポリシーを設定するには、GUI で次の場所に移動します。**[ファブリック（Fabric）]**>**[アクセス ポリシー（Access Policies）]**>**[ポリシー（Policies）]**>**[インターフェイス（Interface）]** > **[L2 インターフェイス（L2 Interface）]**

  特定のパケットに SPAN を設定すると、SPAN はそのパケットに対して 1 回だけサポートされます。最初の SSN の Rx の SPAN によってトラフィックが選択された場合、2 番目の SSN の Tx の SPAN によってトラフィックが再度選択されることはありません。したがって、SPAN セッションの入力ポートと出力ポートが単一のスイッチ上にある場合、SPAN セッションのキャプチャは一方向のみです。SPAN セッションは双方向トラフィックを表示できません。

- フィルタ グループに設定された SPAN ACL フィルタは、アクセス インターフェイスから出力されるブロードキャスト、不明ユニキャスト、およびマルチキャスト（BUM）トラフィックをフィルタリングしません。出力方向の SPAN ACL は、ユニキャスト IPv4 または IPv6 トラフィックに対してのみ機能します。

  SPAN 宛先をローカル ポートとして設定する場合、EPG はそのインターフェイスに展開できません。

  リーフ スイッチでは、VRF フィルタを持つ SPAN 送信元は、VRF インスタンスの下のすべての通常のブリッジ ドメインとすべてのレイヤ 3 SVI にマッチします。

  スパイン スイッチでは、VRF を持つ SPAN 送信元は、設定された VRF VNID トラフィックのみにマッチします。また、ブリッジ ドメイン フィルタは、ブリッジ ドメイン VNID トラフィックのみにマッチします。

- 独自の SPAN 拡張フィルタ エントリを作成する場合、拡張フィルタ エントリの管理対象オブジェクトを識別するために、**__UI__AUTO__CONFIG__DEFAULT__EXTENDED__MO** をオブジェクト名として使用することはできません。

GUI を使用した SPAN の設定

Cisco APIC GUI を使用したテナント SPAN セッションの設定

SPAN は、スイッチまたはテナントで設定できます。このセクションでは、Cisco APIC GUI を使用して、複製された送信元パケットをリモート トラフィック アナライザに転送するようにテナントの SPAN ポリシーを設定する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。フィールドを理解し、有効な値を決定するには、ダイアログ ボックスの右上隅にあるヘルプアイコン（**?**）をクリックしてヘルプ ファイルを表示します。

手順

ステップ 1 メニュー バーで、[Tenants] をクリックします。

ステップ 2 サブメニュー バーで、送信元エンドポイントを含むテナントをクリックします。

ステップ 3 **[ナビゲーション（Navigation）]** ペインでテナントを展開し、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）] を展開して、** > **[SPAN]** を展開します。

[SPAN] に表示される 2 つのノード：**[SPAN 宛先グループ（SPAN Destination Groups）]** と **[SPAN 送信元グループ（SPAN Source Groups）]**。

ステップ 4 **[ナビゲーション（Navigation）]**の下で **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Group）]** を選択します。**[Create SPAN Source Group]** ダイアログが表示されます。

ステップ 5 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスの必須フィールドに適切な値を入力します。

ステップ 6 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成]** ダイアログ ボックスを開きます。

ステップ 7 **[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 8 SPAN送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 9 **[リモート場所の作成（Create Remote Location）]** ダイアログのフィールドに値を入力したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 送信先のトラフィック アナライザを使用して、SPAN 送信元 EPGからのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

APIC GUI を使用した SPAN フィルタ グループの設定

手順

ステップ 1 メニュー バーで **[ファブリック（Fabric）]** をクリックし、サブメニュー バーで **[アクセス ポリシー（ Access Policies）]**をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** を展開し、**[SPAN]** を展開します

ステップ 3 **[SPAN]** の下で **[SPAN フィルタ グループ（SPAN Filter Groups）]** を右クリックし、**[SPAN フィルタ グループの作成（Create SPAN Filter Group）]** を選択します。
[フィルタ グループの作成（Create Filter Group）] ダイアログボックスが表示されます。

ステップ 4 SPAN フィルタ グループの名前を入力します。**[フィルタ エントリ（Filter Entries）]** テーブルで、**[+]** をクリックし、次のフィールドに値を入力します。

- **[送信元 IP プレフィックス（Source IP Prefix）]** : *IP* アドレス/マスクの形式で送信元 IP アドレスを入力します。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- **[最初の送信元ポート（First Source Port）]** 最初の送信元レイヤー 4 ポートを入力します。このフィールドは、**[最後の送信元ポート（Last Source Port）]** フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[最後の送信元ポート（Last Source Port）]** 最後の送信元レイヤー 4 ポートを入力します。このフィールドは、**[最初の送信元ポート（First Source Port）]** フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[宛先 IP プレフィックス（Destination IP Prefix）]** : *IP* アドレス/マスクの形式で宛先 IP アドレスを入力します。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- **[最初の宛先ポート（First Destination Port）]** : 最初の宛先レイヤー 4 ポートを入力します。このフィールドは、**[最後の宛先ポート（Last Destination Port）]** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[最後の宛先ポート（Last Destination Port）]** : 最後の宛先レイヤー 4 ポートを入力します。このフィールドは、**[最初の宛先ポート（First Destination Port）]** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[IP プロトコル（IP Protocol）]** : IP プロトコルを入力します。値 **0** は、このフィールドで **任意の**エントリを指定するために使用します。

- **[拡張フィルタ エントリ（Extended Filter Entries）]** テーブルで、**[+]** をクリックし、次のフィールドに値を入力します。
  - **[名前（Name）]:** 拡張フィルタ エントリの名前を入力します。
  - **[最初の DSCP（DSCP From）]**：DSCP 値を入力します。このフィールドは、**[最後の DSCP（DSCP To）]** フィールドとともに、DSCP 値をフィルタリングする範囲を指定します。
  - **[最後ののDSCP（DSCP To）]**：DSCP 値を入力します。このフィールドは、**[最初の DSCP（DSCP From）]** フィールドとともに、DSCP 値をフィルタリングする範囲を指定します。
  - **[最初の Dot1P（Dot1P From）]**：Dot1P 値を入力します。このフィールドは、**[最後の Dot1P（Dot1P To）]** フィールドとともに、Dot1P 値をフィルタリングする範囲を指定します。
  - **[最後の Dot1P（Dot1P To）]**：Dot1P 値を入力します。このフィールドは、**[最初の Dot1P（Dot1P From）]** フィールドとともに、Dot1P 値をフィルタリングする範囲を指定します。

    送信元ポートと宛先ポートの範囲、または DSCP と Dot1P の範囲の値を指定できます。送信元ポートと宛先ポートの範囲、および DSCP と Dot1P の範囲の両方を指定すると、障害が表示されます。

    DSCP または Dot1P は、出力方向ではサポートされていません。方向として **[両方（Both）]** を選択した場合、DSCP または Dot1P のいずれかが入力方向のみでサポートされ、出力方向ではサポートされません。
  - **[TCP フラグ（TCP Flags）]** ドロップダウンリストで、**TCPフラグ**を選択します。

    TCP フラグを設定できるのは、フィルタ グループのドロップダウン リストで **[未指定（Unspecified）]** または **[TCP]** を **[IP プロトコル（IP Protocol）]** として選択した場合だけです。
  - **[パケット タイプ（Packet Type）]**：パケット タイプを選択します。**[ルート/スイッチ（Routed/Switched）]**、**[ルート（Routed）]**、または **[スイッチのみ（Switched Only）]** のいずれかを選択します。

ステップ 5 このフォームの各フィールドに適切な値を入力したら、**[更新（Update）]** をクリックし、**[送信（Submit）]** をクリックします。

NX-OS スタイルの CLI を使用した拡張フィルタによる SPAN フィルタの設定

次の例は、CLI を使用して SPAN フィルタと拡張フィルタを設定する方法を示しています。

手順

CLI を使用して SPAN フィルタと拡張フィルタを設定するには：

例：

```
apic1(config-monitor-access-filtergrp-filter-extended-filters)# show run
# Command: show running-config monitor access filter-group filtergroup1 filter dstaddr
192.168.10.1 srcaddr 192.168.10.100 extended-filters ext1
# Time: Wed May 11 11:25:23 2022
  monitor access filter-group filtergroup1
    filter srcaddr 192.168.10.100 dstaddr 192.168.10.1
      extended-filters ext1
        dscp from CS0 to 4
        dot1p from 1 to 5
        forwarding-type switched
        tcp-flag ack off
        tcp-flag fin off
        tcp-flag rst on
        exit
      exit
    exit
apic1#
```

REST API を使用した拡張フィルタによる SPAN フィルタの設定

次の例は、REST API を使用して SPAN フィルタを設定する方法を示しています。

手順

Rest API を使用して SPAN フィルタを設定するには:

例：

```
URL: {{apic-host}}/api/node/mo/.xml
BODY:
<polUni>
   <infraInfra dn="uni/infra">
      <spanSrcGrp adminSt="enabled" descr="" dn="uni/infra/srcgrp-local1" nameAlias=""
ownerKey=""
       ownerTag="">
          <spanRsSrcGrpToFilterGrp tDn="uni/infra/filtergrp-two" />
         <spanSrc descr="" dir="both" name="src1" nameAlias="" ownerKey="" ownerTag="">

          <spanRsSrcToPathEp tDn="topology/pod-1/paths-101/pathep-[eth1/15]" />
          </spanSrc>
          <spanSpanLbl descr="" name="dest1" nameAlias="" ownerKey="" ownerTag="" tag=
           "yellow-green" />
       </spanSrcGrp>
       <spanDestGrp annotation="" descr="" dn="uni/infra/destgrp-dest1" nameAlias=""
ownerKey=""
        ownerTag="">
          <spanDest annotation="" descr="" name="destg" nameAlias="" ownerKey=""
ownerTag="">
         <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-101/pathep-

           [eth1/7]" />
```

```
            </spanDest>
        </spanDestGrp>
        <spanFilterGrp name="two">
           <spanFilterEntry name="udp_two" ipProto="udp" srcAddr="1002::1/64"
dstAddr="1001::1/64"
            srcPortFrom="1" srcPortTo="2" dstPortFrom="1" dstPortTo="2">
              <spanExtendedFltEntry name="arun1" dscpFrom="0" dscpTo="10" dot1pFrom="0"
 dot1pTo="7"
                tcpFlags="128" v6FlowLabel="1522" forwardingVal="switched" />
            </spanFilterEntry>
        </spanFilterGrp>
    </infraInfra>
</polUni>
```

APIC GUI を使用したアクセス SPAN ポリシーの設定

この手順では、Cisco APIC GUI を使用してアクセス SPAN ポリシーを設定します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

手順

ステップ 1 メニューバーで、**[ファブリック（Fabric）]** > **[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスを開いて、必須のフィールドに適切な値を入力します。

ステップ 6 [Create SPAN Source] ダイアログ ボックスで、[Add Source Access Paths] を展開して、ソース パスを指定します。

[送信元をパスに関連付ける（Associate Source to Path）] ダイアログ ボックスが表示されます。

ステップ 7 **[送信元をパスに関連付ける（Associate Source to Path）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 8 送信元とパスの関連付けが完了したら、**[OK]** をクリックします。

[SPAN 送信元の作成（Create SPAN Source）] ダイアログ ボックスに戻ります。

ステップ 9 SPAN 送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 10 SPAN 送信元グループの設定が完了したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

Cisco APIC GUI を使用したファブリック SPAN ポリシーの設定

このセクションでは、Cisco APIC GUI を使用してファブリック SPAN ポリシーを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

手順

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[ファブリック ポリシー（Fabric Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成]** ダイアログ ボックスを開きます。

ステップ 6 **[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 7 完了したら、[OK] をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 8 **[リモート場所の作成（Create Remote Location）]** ダイアログのフィールドに値を入力したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

APIC GUI を使用した外部アクセス用のレイヤ 3 EPG SPAN セッションの設定

この手順は、Cisco APIC GUI を使用して外部アクセス用のレイヤ 3 EPG SPAN ポリシーを設定する方法を示しています。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

手順

ステップ 1 メニューバーで、**[ファブリック（Fabric）]**>**[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[フィルタ グループ（Filter Group）]** フィールドで、フィルタ グループを選択または作成します。

詳細については、「APIC GUI を使用した SPAN フィルタ グループの設定 （41 ページ）」を参照してください。

ステップ 6 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスを開きいて、以下の操作を実行します。

a) 送信元ポリシーの**[名前（Name）]** を入力します。
b) トラフィック フローの **[方向（Direction）]** オプションを選択します。
c) (オプション) **[ドロップ パケットのスパニング（Span Drop Packets）]** チェックボックスをクリックしてチェックマークを付けます。オンにすると、SPAN-on-drop 機能が有効になります。
d) 外部アクセスの場合は、**[外部にルーティング（Routed Outside）]**（**[タイプ（Type）]** フィールド）をクリックします。

（注） 外部アクセスで **[外部にルーティング（Routed Outside）]** を選択した場合、**[名前（Name）]**、**[アドレス（Address）]**、および **[Encap]** フィールドが表示されて、**[L3 Outside]** を設定できるようになります。

e) **[送信元アクセス パスの追加（Add Source Access Paths）]** を展開して、送信元パスを指定します。

[送信元をパスに関連付ける（Associate Source to Path）] ダイアログ ボックスが表示されます。

f) **[送信元をパスに関連付ける（Associate Source to Path）]** ダイアログ ボックスのフィールドに適切な値を入力します。

g) 送信元とパスの関連付けが完了したら、**[OK]** をクリックします。

[SPAN 送信元の作成（Create SPAN Source）] ダイアログ ボックスに戻ります。

h) SPAN 送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 7 SPAN 送信元グループの設定が完了したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

Cisco APIC GUI を使用したアクセス SPAN ポリシーの宛先グループの設定

このセクションでは、Cisco APIC GUI を使用して、アクセス SPAN ポリシーの宛先グループを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

SPAN宛先グループと送信元を作成すれば、SPAN宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

手順

ステップ 1 メニューバーで、**[ファブリック（Fabric）]** > **[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 宛先グループ（SPAN Destination Groups）]** を右クリックして、**[SPAN 宛先グループの作成（Create SPAN Destination Groups）]** を選択します。
[Create SPAN Destination Group] ダイアログが表示されます。

ステップ 4 **[SPAN 宛先グループの作成（Create SPAN Destination Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 完了したら、[送信（Submit）] をクリックします。

宛先グループが作成されます。

Cisco APIC GUI を使用したファブリック SPAN ポリシーの宛先グループの設定

このセクションでは、Cisco APIC GUI を使用して、ファブリック SPAN ポリシーの宛先グループを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

SPAN宛先グループと送信元を作成すれば、SPAN宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

手順

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[ファブリック ポリシー（Fabric Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 宛先グループ（SPAN Destination Groups）]** を右クリックして、**[SPAN 宛先グループの作成（Create SPAN Destination Groups）]** を選択します。
[Create SPAN Destination Group] ダイアログが表示されます。

ステップ 4 **[SPAN 宛先グループの作成（Create SPAN Destination Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 完了したら、[送信（Submit）] をクリックします。

宛先グループが作成されます。

次のタスク

まだ作成していない場合は、ファブリック SPAN ポリシーの送信元を設定します。

NX-OS Style CLI を使用した SPAN の設定

NX-OS スタイルの CLI を使用したアクセス モードでのローカル SPAN の設定

これは、アクセス リーフ ノードにローカルな従来の SPAN 設定です。1 つ以上のアクセス ポートまたはポート チャネルから発信されたトラフィックをモニタリングし、同じリーフ ノードにローカルな宛先ポートに送信できます。

手順

ステップ 1 **configure terminal**

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。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session mySession
```

ステップ 3 [**no**] **description** *text*

このアクセス モニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-access)# description "This is my SPAN session"
```

ステップ 4 [**no**] **destination interface ethernet** *slot/port* **leaf** *node-id*

宛先インターフェイスを指定します。宛先インターフェイスを FEX ポートにすることはできません。

例：

```
apic1(config-monitor-access)# destination interface ethernet 1/2 leaf 101
```

ステップ 5 [**no**] **source interface ethernet** {[*fex/*]*slot/port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface ethernet 1/2 leaf 101
```

ステップ 6 **drop enable**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-access-source)# drop enable
```

ステップ 7 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# direction tx
```

ステップ 8 [**no**] **filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
```

ステップ 9 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 10 [**no**] **destination interface port-channel** *port-channel-name-list* **leaf** *node-id*

宛先インターフェイスを指定します。宛先インターフェイスを FEX ポートにすることはできません。

（注） リリース 4.1(1) 以降、コマンド例に示すように、宛先インターフェイスとしてスタティック ポート チャネルを使用できるようになりました。

例：

```
apic1(config-monitor-access)# destination interface port-channel pc1 leaf 101
```

ステップ 11 [**no**] **source interface port-channel** *port-channel-name-list* **leaf** *node-id* [**fex** *fex-id*]

送信元インターフェイス ポート チャネルを指定します。

（トラフィックの方向とフィルタ設定を入力します。ここには表示されていません）。

例：

```
apic1(config-monitor-access)# source interface port-channel pc5 leaf 101
```

ステップ 12 [**no**] **filter tenant** *tenant-name* **l3out** *L3Out-name* **vlan** *interface-VLAN*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

（注） リリース 4.1(1) 以降、例に示すように、L3Out インターフェイス フィルタリングを設定するときに IP プレフィックスを指定する必要がなくなりました。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 l3out l3out1 vlan 2820
```

ステップ 13 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-access)# no shut
```

例

この例は、ローカル アクセス モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my SPAN session"
apic1(config-monitor-access)# destination interface ethernet 1/2 leaf 101
apic1(config-monitor-access)# source interface ethernet 1/1 leaf 101
apic1(config-monitor-access)# drop enable
apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
# Command: show running-config monitor access session mySession
# Time: Fri Nov 6 23:55:35 2015
  monitor access session mySession
    description "This is my SPAN session"
    destination interface eth 1/2 leaf 101
    source interface eth 1/1 leaf 101
      direction tx
      filter tenant t1 application app1 epg epg
      exit
    exit
```

NX-OS スタイルの CLI を使用した SPAN フィルタ グループの設定

次の手順では、SPAN フィルタ グループとフィルタ エントリを設定する方法について説明します。

手順

ステップ 1 **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。

例：

```
apic1# configure
```

ステップ 2 [**no**] **monitor access filter-group** *filtergroup-name*

アクセス モニタリング フィルタ グループ設定を作成します。

例：

```
apic1(config)# monitor access filter-group filtergroup1
```

ステップ 3 [**no**] **filter srcaddress** *source-address* **dstaddress** *destination-address* **srcport-from** *source-from-port* **srcport-to** *source-to-port* **dstport-from** *destination-from-port* **dstport-to** *destination-to-port* **ipproto** *IP-protocol*

フィルタ グループのフィルタ エントリを設定します。ここで、

- *source-address* は、*IP* アドレス/マスク 形式の送信元 IP アドレスです。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- *destination-address* は、*IP* アドレス/マスク形式の宛先 IP アドレスです。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- *source-from-port* は、最初の送信元レイヤ 4 ポートです。このフィールドは、**srcport-to** フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *source-to-port* は、最後の送信元レイヤ 4 ポートです。このフィールドは、**srcport-from** フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *destination-from-port* は、最初の宛先レイヤ 4 ポートです。このフィールドは、**dstport-to** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *destination-to-port* は、最後の宛先レイヤ 4 ポートです。このフィールドは、**dstport-from** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *IP-protocol* は IP プロトコルです。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。

例：

```
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from
 0 srcport-to 0 dstport-from 0 dstport-to 0 ipproto 20
```

ステップ 4 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-fltgrp)# exit
```

ステップ 5 **exit**

グローバル コンフィギュレーション モードを終了します。

例：

```
apic1(config)# exit
```

例

この例は、SPAN フィルタ グループとフィルタ エントリを設定する方法を示しています。

```
apic1# configure
apic1(config)# monitor access filter-group filtergroup1
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from
 0 srcport-to 0 dstport-from 0 dstport-to 0 ipproto 20
apic1(config-monitor-fltgrp)# exit
apic1(config)# exit
```

NX-OS スタイルの CLI を使用した SPAN フィルタ グループの関連付け

次の手順では、フィルタ グループを SPAN 送信元グループに関連付ける方法について説明します。

手順

ステップ 1 **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。

例：

```
apic1# configure
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session session1
```

ステップ 3 **filter-group** *filtergroup-name*

フィルタ グループを関連付けます。

例：

```
apic1(config-monitor-access)# filter-group filtergroup1
```

ステップ 4 **no filter-group**

必要に応じて、フィルタ グループの関連付けを解除します。

例：

```
apic1(config-monitor-access)# no filter-group
```

ステップ 5 [**no**] **source interface ethernet** {[*fex/*]*slot/port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface ethernet 1/9 leaf 101
```

ステップ 6 **filter-group** *filtergroup-name*

フィルタ グループを SPAN 送信元に関連付けます。

例：

```
apic1(config-monitor-access-source)# filter-group filtergroup2
```

ステップ 7 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 8 **no filter-group**

必要に応じて、SPAN 送信元からフィルタ グループの関連付けを解除します。

例：

```
apic1(config-monitor-access-source)# no filter-group
```

ステップ 9 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-access)# exit
```

ステップ 10 **exit**

グローバル コンフィギュレーション モードを終了します。

例：

```
apic1(config)# exit
```

例

この例は、フィルタ グループを関連付ける方法を示しています。

```
apic1# configure
apic1(config)# monitor access session session1
apic1(config-monitor-access)# filter-group filtergroup1
apic1(config-monitor-access)# source interface ethernet 1/9 leaf 101
apic1(config-monitor-access-source)# filter-group filtergroup2
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access-source)# no filter-group
apic1(config-monitor-access)# exit
```

```
apic1(config)# exit
```

NX-OS スタイルの CLI を使用したアクセス モードでの ERSPAN の設定

ACI ファブリックでは、アクセス モードの ERSPAN 設定を使用して、1 つ以上のリーフ ノードのアクセス ポート、ポート チャネル、および vPC から発信されたトラフィックを監視できます。

ERSPAN セッションの場合、宛先は常にエンドポイント グループ (EPG) で、これらはファブリック内のどこにでも展開できます。監視対象のトラフィックは、どこであれ、EPG が移動した場所である宛先に転送されます。

手順

ステップ 1 **configure terminal**

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。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session mySession
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-access)# description "This is my access ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーションモードを開始します。

例：

```
apic1(config-monitor-access)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-access-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-access-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-access-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-access-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-access-dest)#
```

ステップ 10 [**no**] **source interface ethernet** {[*fex/*]*slot/port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface eth 1/2 leaf 101
```

ステップ 11 [**no**] **source interface port-channel** *port-channel-name-list* **leaf** *node-id* [**fex** *fex-id*]

送信元インターフェイスのポートチャネルを指定します。

例：

```
apic1(config-monitor-access)# source interface port-channel pc1 leaf 101
```

ステップ 12 [**no**] **source interface vpc** *vpc-name-list* **leaf** *node-id1 node-id2* [**fex** *fex-id1 fex-id2*]

送信元インターフェイス vPC を指定します。

例：

```
apic1(config-monitor-access)# source interface vpc pc1 leaf 101 102
```

ステップ 13 **drop enable**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-access-source)# drop enable
```

ステップ 14 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# direction tx
```

ステップ 15 [**no**] **filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
```

ステップ 16 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 17 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-access)# no shut
```

例

この例は、ERSPAN アクセス モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my access ERSPAN session"
apic1(config-monitor-access)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-access-dest)# erspan-id 100
apic1(config-monitor-access-dest)# ip dscp 42
apic1(config-monitor-access-dest)# ip ttl 16
apic1(config-monitor-access-dest)# mtu 9216
apic1(config-monitor-access-dest)# exit
apic1(config-monitor-access)# source interface eth 1/1 leaf 101
apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)#drop enable
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
```

```
# Command: show running-config monitor access session mySession
# Time: Fri Nov 6 23:55:35 2015
  monitor access session mySession
    description "This is my ERSPAN session"
    source interface eth 1/1 leaf 101
      direction tx
      filter tenant t1 application app1 epg epg1
      exit
    destination tenant t1 application app1 epg epg1 destination-ip 192.0.20.123
source-ip-prefix 10.0.20.1
      ip dscp 42
      ip ttl 16
      erspan-id 9216
      mtu 9216
      exit
    exit
```

この例は、モニタリング送信元としてポート チャネルを設定する方法を示しています。

```
apic1(config-monitor-access)# source interface port-channel pc3 leaf 105
```

この例は、モニタリング送信元として vPC の 1 つのレッグを設定する方法を示しています。

```
apic1(config-monitor-access)# source interface port-channel vpc3 leaf 105
```

次の例は、FEX 101 からのポートの範囲をモニタリング送信元として設定する方法を示しています。

```
apic1(config-monitor-access)# source interface eth 101/1/1-2 leaf 105
```

NX-OS スタイルの CLI を使用したファブリック モードでの ERSPAN の設定

ACI ファブリックでは、ファブリック モードの ERSPAN 設定を使用して、リーフ ノードまたはスパイン ノードの 1 つ以上のファブリック ポートから発信されたトラフィックをモニタリングできます。ローカル SPAN はファブリック モードではサポートされていません。

ERSPAN セッションの場合、宛先は常にエンドポイント グループ (EPG) で、これらはファブリック内のどこにでも展開できます。監視対象のトラフィックは、どこであれ、EPGが移動した場所である宛先に転送されます。ファブリック モードでは、ファブリック ポートのみが送信元として許可されますが、リーフ スイッチとスパイン スイッチの両方が許可されます。

手順

ステップ 1 **configure terminal**

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。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor fabric session** *session-name*

ファブリック モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor fabric session mySession
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-fabric)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-fabric-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-fabric-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-fabric-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-fabric-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-fabric-dest)#
```

ステップ 10 [**no**] **source interface ethernet** {*slot/port* | *port-range*} **switch** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-fabric)# source interface eth 1/2 switch 101
```

ステップ 11 **drop enable**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-fabric-source)# drop enable
```

ステップ 12 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-fabric-source)# direction tx
```

ステップ 13 [**no**] **filter tenant** *tenant-name* **bd** *bd-name*

ブリッジ ドメインでトラフィックをフィルタリングします。

例：

```
apic1(config-monitor-fabric-source)# filter tenant t1 bd bd1
```

ステップ 14 [**no**] **filter tenant** *tenant-name* **vrf** *vrf-name*

VRF でトラフィックをフィルタリングします。

例：

```
apic1(config-monitor-fabric-source)# filter tenant t1 vrf vrf1
```

ステップ 15 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-fabric-source)# exit
```

ステップ 16 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-fabric)# no shut
```

例

この例は、ERSPANファブリック モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor fabric session mySession
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
apic1(config-monitor-fabric)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-fabric-dest)# erspan-id 100
apic1(config-monitor-fabric-dest)# ip dscp 42
apic1(config-monitor-fabric-dest)# ip ttl 16
apic1(config-monitor-fabric-dest)# mtu 9216
apic1(config-monitor-fabric-dest)# exit
apic1(config-monitor-fabric)# source interface eth 1/1 switch 101
apic1(config-monitor-fabric-source)# drop enable
apic1(config-monitor-fabric-source)# direction tx
apic1(config-monitor-fabric-source)# filter tenant t1 bd bd1
apic1(config-monitor-fabric-source)# filter tenant t1 vrf vrf1
apic1(config-monitor-fabric-source)# exit
apic1(config-monitor-fabric)# no shut
```

NX-OS スタイルの CLI を使用したテナント モードでの ERSPAN の設定

ACI ファブリックでは、テナント モードの ERSPAN 設定を使用して、テナント内のエンドポイント グループから発信されたトラフィックをモニタリングできます。

テナント モードでは、送信元 EPG から発信されたトラフィックは、同じテナント内の宛先 EPG に送信されます。送信元または宛先の EPG がファブリック内で移動しても、トラフィックのモニタリングには影響しません。

手順

ステップ 1 **configure terminal**

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。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor tenant** *tenant-name* **session** *session-name*

テナント モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor tenant session mySession
```

ステップ 3 [**no**] **description** *text*

このアクセス モニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-tenant)# description "This is my tenant ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-tenant)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-tenant-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-tenant-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-tenant-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-tenant-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-tenant-dest)#
```

ステップ 10 [**no**] **source application** *application-name* **epg** *epg-name*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-tenant)# source application app2 epg epg5
```

ステップ 11 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-tenant-source)# direction tx
```

ステップ 12 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-tenant-source)# exit
```

ステップ 13 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-tenant)# no shut
```

例

この例は、ERSPAN テナント モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-tenant)# description "This is my tenant ERSPAN session"
apic1(config-monitor-tenant)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-tenant-dest)# erspan-id 100
apic1(config-monitor-tenant-dest)# ip dscp 42
apic1(config-monitor-tenant-dest)# ip ttl 16
apic1(config-monitor-tenant-dest)# mtu 9216
apic1(config-monitor-tenant-dest)# exit
apic1(config-monitor-tenant)# source application app2 epg epg5
apic1(config-monitor-tenant-source)# direction tx
apic1(config-monitor-tenant-source)# exit
apic1(config-monitor-tenant)# no shut
```

NX-OS スタイルの CLI を使用したグローバル SPAN-On-Drop セッションの設定

このセクションでは、ノード上のすべてのポートを SPAN 送信元とするグローバル ドロップを作成する方法を示します。

手順

ステップ 1 **configure terminal**

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。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor fabric session** *session-name*

ファブリック モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor fabric session Spine301-GD-SOD
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
```

ステップ 4 **source global-drop switch**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-fabric)# source global-drop switch
```

ステップ 5 [**no**] **destination tenant** *tenant-name***application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーションモードを開始します。

例：

```
apic1(config-monitor-fabric-dest)# destination tenant ERSPAN application A1 epg E1
destination-ip 165.10.10.155 source-ip-prefix 22.22.22.22
```

例

次に、SPAN-on-Drop セッションを設定する例を示します。

```
apic1# configure terminal
apic1(config)# monitor fabric session Spine301-GD-SOD
apic1(config-monitor-fabric)# source global-drop switch
apic1(config-monitor-fabric)# destination tenant ERSPAN application A1 epg E1
destination-ip 179.10.10.179 source-ip-prefix 31.31.31.31
```

REST API を使用した SPAN の構成

REST API を使用した ERSPAN 宛先のファブリック宛先グループの設定

このセクションでは、REST API を使用して、ERSPAN 宛先のファブリック宛先グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

ERSPAN 宛先のファブリック宛先グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
    <spanDest annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
    <spanRsDestEpg annotation="" dscp="unspecified" finalIp="0.0.0.0" flowId="1"
ip="179.10.10.179"
     mtu="1518"srcIpPrefix="20.20.20.2" tDn="uni/tn-ERSPAN/ap-A1/epg-E1" ttl="64"
ver="ver2"
     verEnforced="no"/>
    </spanDest>
</spanDestGrp>
```

REST API を使用したグローバル ドロップ送信元グループの設定

このセクションでは、REST API を使用してグローバル ドロップ送信元グループを構成することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

グローバル ドロップ送信元グループを構成します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Spine-402-GD-SOD" nameAlias="">

   <spanSrc annotation="" descr="" dir="both" name="402" nameAlias=""  spanOnDrop="yes">

    <spanRsSrcToNode annotation="" tDn="topology/pod-1/node-402"/>
    </spanSrc><spanSpanLbl annotation="" descr="" name="402-dst-179" nameAlias=""
tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した SPAN 宛先としてのリーフ ポートの設定

このセクションでは、REST API を使用してリーフ ポートを SPAN 宛先として設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

リーフ ポートを SPAN 宛先として設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
    <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
    <spanRsDestPathEp annotation="" mtu="1518"
tDn="topology/pod-1/paths-301/pathep-[eth1/18]"/>
    </spanDest>
</spanDestGrp>
```

REST API を使用した SPAN アクセス送信元グループの設定

このセクションでは、REST API を使用して SPAN アクセス ソース グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

SPAN アクセス送信元グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr=""  name="Test-Src2" nameAlias=""
ownerKey=""
 ownerTag="">
    <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey=""
ownerTag=""
     spanOnDrop="yes">
    <spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/1]"/>
    </spanSrc>
   <spanSpanLbl annotation="" descr="" name="Dest1" nameAlias="" ownerKey="" ownerTag=""

     tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した SPAN ファブリック送信元グループの設定

このセクションでは、REST API を使用して SPAN ファブリック送信元グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

SPAN ファブリック送信元グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Test-Src2" nameAlias=""
ownerKey=""
 ownerTag="">
    <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey=""
     ownerTag="" spanOnDrop="yes">
    <spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/51]"/>
    </spanSrc>
   <spanSpanLbl annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag=""

     tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した ERSPAN 宛先のアクセス宛先グループの設定

このセクションでは、REST API を使用して、ERSPAN 宛先のアクセス宛先グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

手順

ERSPAN 宛先のアクセス宛先グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
  ownerTag="">
    <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
     ownerTag="">
    <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-301/pathep-
     [eth1/18]"/>
    </spanDest>
</spanDestGrp>
```

統計を使用

統計は、観察されたオブジェクトのリアルタイムの測定値を提供し、傾向分析とトラブルシューティングを可能にします。統計収集は、継続的またはオンデマンドの収集用に構成でき、累計カウンタとゲージで収集できます。

ポリシーは、収集する統計の種類、間隔、実行するアクションを定義します。たとえば、入力VLAN でドロップされたパケットのしきい値が毎秒 1000 を超えた場合、EPG 上で 1 つの障害を生成するようにポリシーを構成できます。

統計データは、インターフェイス、VLAN、EPG、アプリケーションプロファイル、ACL ルール、テナント、内部APICプロセスなどのさまざまなソースから収集されます。統計は、5分、15 分、1 時間、1 日、1 週間、1 か月、1 四半期、または 1 年のサンプリング間隔でデータを蓄積します。短い間隔で収集されて蓄積されたデータが、長い間隔で収集されるデータのソースになります。さまざまな統計情報プロパティを利用でき、最終値、累計、周期、変化のレート、トレンド、最大、最小と平均などがあります。収集/保持時間は構成可能です。ポリシーは、統計をシステムの現在の状態から収集するか、履歴として蓄積するか、またはその両方を行うかを指定できます。たとえば、ポリシーは、履歴統計を 1 時間にわたって 5 分間隔で収集するように指定できます。1 時間は移動ウィンドウです。1 時間が経過すると、次の 5 分間の統計が追加され、最初の 5 分間に収集されたデータは破棄されます。

（注） 5 分粒度のサンプル レコードの最大数は 12 サンプル（1 時間の統計）に制限されます。他のすべてのサンプル間隔は、1,000 サンプル レコードに制限されます。たとえば、1 時間粒度の統計は 41 日間まで保持できます。

GUI での統計情報の表示

アプリケーション プロファイル、物理インターフェイス、ブリッジ ドメイン、ファブリック ノードなど、APIC GUI を使用して、多数のオブジェクトの統計情報を表示できます。GUI で統計情報を表示するには、**ナビゲーション**ペインでオブジェクトを選択し、[STATS]タブをクリックします。

インターフェイスの統計情報を表示する手順は、次のとおりです。

手順

ステップ 1 メニュー バーで、**[ファブリック（Fabric）] >[インベントリ（Inventory）]** を選択します。

ステップ 2 **[ナビゲーション（navigation）]** のペインで、ポッドを選択します。

ステップ 3 ポッドを展開し、スイッチを展開します。

ステップ 4 **[ナビゲーション（Navigation）]** ペインで、**[インターフェイス（Interfaces）]** を展開し、**eth1/1** を選択します。

ステップ 5 **[作業（Work）]** ペインで、**[STATS]** タブを選択します。

APIC はインターフェイス統計情報を表示します。

例

次のタスク

[作業（Work）] ペインの次のアイコンを使用して、APIC での統計情報の表示方法を管理できます。

- Refresh：統計情報を手動で更新します。
- Show Table View：表とチャートの表示を切り替えます。
- Start または Stop Stats：統計情報の自動更新を有効または無効にします。
- Select Stats：表示するカウンタとサンプルのインターバルを指定します。
- Download Object as XML：XML 形式でオブジェクトをダウンロードします。
- Measurement Type（ギア アイコン）：統計情報の測定タイプを指定します。オプションとして累積値、定期値、平均値、傾向値があります。

スイッチの統計情報コマンド

次のコマンドを使って、ACI リーフ スイッチの統計情報を表示できます。

コマンド	目的
レガシー Cisco Nexus の **[表示（show）]** / **[クリア（clear）]** コマンド	詳細については、『*Cisco Nexus 9000 Series NX-OS Configuration Guide*』を参照してください。

コマンド	目的
[プラットフォーム内部カウンタ ポートを表示（show platform internal counters port）] [*port_num* | **detail** | **nz** | {**internal** [**nz** | *int_port_num*]}]	スパイン ポート統計情報を表示します。 • *port_num* — スロットのない前面ポート番号。 • **[詳細（detail）]** — SNMP、クラス、および転送の統計を返します。 • **nz** — ゼロ以外の値のみを表示します。 • **[内部（internal）]**—内部ポートの統計情報を表示します。 • *int_port_num*—内部論理ポート番号。たとえば、BCM-0/97 の場合は、97 と入力します。 （注） リンクがリセットされると、スイッチのカウンターがゼロになります。カウンターリセットの条件には以下のものがあります。 • 偶発的なリンクのリセット • 手動で有効化されたポート(ポートが無効化された後)
[プラットフォーム内部カウンタ vlan を表示（show platform internal counters vlan）][*hw_vlan_id*	VLAN 統計情報を表示します。
[プラットフォーム内部カウンタ tep を表示（show platform internal counters tep）][*tunnel_id*]	TEP 統計情報の表示
[プラットフォーム内部カウンタ フローを表示（show platform internal counters flow）][*rule_id* | {**dump** [*asic inst*] | [**slice direction** | **index** *hw_index*]}]	フロー統計情報を表示します。
[プラットフォーム内部カウンターポートをクリア（clear platform internal counters port）][*port_num* | {**internal** [*int_port_num*]}]	ポート統計情報を消去します。
[プラットフォーム内部カウンタ vlan をクリア（clear platform internal counters vlan）][*hw_vlan_id*]	VLAN カウンタを消去します。

コマンド	目的
[プラットフォーム内部統計ロギングレベルをデバッグ（debug platform internal stats logging level）] *log_level*	デバッグ ロギング レベルを設定
[プラットフォーム内部統計ロギングをデバッグ（debug platform internal stats logging）] {**err**|**trace**|**flow**}	デバッグのロギング タイプを設定します。

GUI を使用する統計情報しきい値の管理

手順

ステップ 1 メニュー バーで、[Fabric] > [Fabric Policies] を選択します。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで + をクリックし、**[モニタリング ポリシー（Monitoring Policies）]** を展開します。

ステップ 3 **[ナビゲーション（Navigation）]** ペインで、モニタリング ポリシー名（デフォルトなど）を拡張します。

ステップ 4 **[統計収集ポリシー（Stats Collection Policies）]** をクリックします。

ステップ 5 **[統計収集ポリシー（Stats Collection Policies）]** ウィンドウで、しきい値を設定する **[モニタリング オブジェクト（Monitoring Object）]** および **[統計タイプ（Stat Type）]** を選択します。

ステップ 6 **[作業（Work）]** ペインで、**[構成しきい値（CONFIG THRESHOLDS）]** の下の + をアイコンをクリックします。

ステップ 7 **[コレクションのためのしきい値（THRESHOLDS FOR COLLECTION）]** ウィンドウで + をクリックし、しきい値を追加します。

ステップ 8 **[プロパティを選択（Choose a Property）]** ウィンドウで、統計タイプを選択します。

ステップ 9 **[統計しきい値を編集（EDIT STATS THRESHOLD）]** ウィンドウで、次のしきい値を指定します。

- 標準値 — カウンタの有効値。
- しきい値の方向 — しきい値が最大値または最小値かどうかを示します。
- 上昇値（クリティカル、メジャー、マイナー、注意）—値がしきい値を上回った場合にトリガーされます。
- 下降値（クリティカル、メジャー、マイナー、注意）—値がしきい値を下回った場合にトリガーされます。

ステップ 10 上昇および下降しきい値の設定値、リセット値を指定できます。設定値はエラーがトリガーされるタイミングを指定します。リセット値はエラーが消去されるタイミングを指定します。

ステップ 11 しきい値を保存するには、**[送信する（SUBMIT）]** をクリックします。

ステップ 12　**[コレクションのためのしきい値（THRESHOLDS FOR COLLECTION）]** ウィンドウで、**[閉じる（CLOSE）]** をクリックします。

統計情報に関するトラブルシューティングのシナリオ

次の表で、Cisco APIC に共通する統計情報に関するトラブルシューティングのシナリオを要約します。

問題	ソリューション
APICは、設定されたモニタリングポリシーを適用しません。	モニタリング ポリシーが適用されていても、APIC が統計情報の収集やトリガーしきい値に対する操作など、対応するアクションを実行しないと問題が発生します。問題を解決するには、次の手順に従ってください。 • monPolDn が正しいモニタリング ポリシーを指していることを確認します。 • セレクタが正しく設定され、エラーがないことを確認します。 • テナントのオブジェクトの場合は、モニタリングポリシーとの関係を確認します。
設定した一部の統計情報が見つからない。	問題を解決するには、次の手順に従ってください。 • モニタリングポリシーおよび収集ポリシー内でデフォルトによって無効になっている統計情報を確認します。 • 収集ポリシーを確認し、統計情報がデフォルトで無効になっているか、または特定のインターバルで無効になっているかを識別します。 • 統計ポリシーを確認し、統計情報がデフォルトで無効になっているか、または特定のインターバルで無効になっているかを識別します。 （注）　ファブリック ヘルスの統計情報を除き、5 分間の統計情報がスイッチに保存され、スイッチがリブートされると失われます。

問題	ソリューション
統計情報や履歴を設定した期間保持できない	問題を解決するには、次の手順に従ってください。 • 収集設定を確認してください。モニタリングポリシーの最上位レベルで設定されていると、特定のオブジェクトまたは統計タイプでは、統計情報が無効になる場合があります。 • モニタリングオブジェクトに割り当てられた収集ポリシーを確認します。ポリシーが存在するのを確認し、管理状態および履歴保持の値を確認します。 • 統計タイプが正しく設定されていることを確認します。
設定されたインターバルにわたって保持されない統計情報がある。	設定が履歴記録サイズの最大値を超えていないかどうか確認します。制限は次のとおりです。 • 5 分間の細かさでのスイッチ統計情報は 12 サンプル（5分間の細かさの統計情報の 1 時分）に限られています。 • 1000 サンプルの厳しい制限があります。たとえば、1 時間の細かさの統計情報は 41 日間まで保持できます。
エクスポートポリシーは設定されるが、APIC が統計情報をエクスポートしない。	問題を解決するには、次の手順に従ってください。 • 送信先ポリシーの状態オブジェクトを確認します。 • 統計をエクスポートするノードでエクスポートステータスのオブジェクトをチェックし、エクスポートステータスと詳細のプロパティを確認してください。集約された EPG 統計は APIC ノードから 15 分ごとにエクスポートされます。その他の統計は、送信元ノードから 5 分ごとにエクスポートされます。たとえば、EPG が 2 つのリーフ スイッチに展開され、EPG アグリゲーション パーツをエクスポートするように設定されている場合、それらのパーツは 5 分ごとにノードからエクスポートされます。 • 構成がエクスポートポリシーの最大数を超えていないかどうかを確認します。統計のエクスポートポリシーの最大数は、テナントの数とほぼ同じです。 （注） 各テナントは複数の統計エクスポートポリシーを持つことができ、複数のテナントが同じエクスポートポリシーを共有できますが、ポリシーの合計数はテナントの数とほぼ同数に制限されます。

問題	ソリューション
5 分間統計が変動する	APIC システムは、約 10 秒ごとにサンプリングされた統計を 5 分ごとに報告します。データが収集されるときにわずかな時間差があるため、5 分間で取得されるサンプルの数は異なる場合があります。その結果、統計情報が少し長い、または短い期間を表す場合があります。これは想定されている動作です。
一部の履歴統計情報が見つからない。	詳しくは、「統計情報の消去」を参照してください。

統計情報の消去

APIC とスイッチは次のように統計情報を消去します。

- スイッチ — スイッチは次のように統計情報を消去します。
  - スイッチの 5 分間の統計情報は、5 分間カウンタ値が報告されないと消去されます。この状況はポリシーによってオブジェクトが削除される、または統計情報が無効化されるときに起こる場合があります。
  - 統計が 1 時間以上欠落している場合、粒度の大きい統計はパージされます。これは、次の場合に発生する可能性があります。
    - 統計情報がポリシーによって無効化されている。
    - スイッチが 1 時間以上 APIC から切断されている。
  - スイッチは 5 分後に削除されたオブジェクトの統計情報を消去します。オブジェクトがこの時間内に再作成されると、統計カウントは未変更のままになります。
  - 無効化されたオブジェクト統計情報は 5 分後に削除されます。
  - 統計情報レポートが 5 分間無効化されるなど、システム状態が変化すると、このスイッチによって統計情報が消去されます。
- APIC — APIC はインターフェイス、EPG、温度センサーと正常性統計情報を含むオブジェクトを 1 時間後に消去します。

Syslog のソースと宛先の指定

このセクションでは、syslog 宛先グループ、syslog ソースを作成する方法、および REST API を使用して syslog を NX-OS CLI 形式で表示できるようにする方法について説明します。

Syslog について

稼働中、シスコ アプリケーション セントリック インフラストラクチャ（ACI）システムでの障害またはイベントは、コンソール、ローカル ファイル、および別のシステム上のロギング サーバへのシステム ログ（syslog）の送信をトリガーできます。システム ログ メッセージには、通常、障害またはイベントに関する情報のサブセットが含まれます。システム ログ メッセージには、監査ログとセッション ログのエントリを含めることもできます。

（注） APIC およびファブリック ノードが生成できる syslog メッセージのリストについては、http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/ACI_SysMsg.html を参照してください。

多くのシステム ログ メッセージは、ユーザが実行している処理、あるいはユーザが設定または管理しているオブジェクトに固有のものです。これらのメッセージには次のようなものがあります。

- 情報メッセージ。実行している処理のヘルプおよびヒントを提供します。
- 警告メッセージ。ユーザが設定または管理しているオブジェクト（ユーザ アカウントやサービス プロファイルなど）に関連するシステム エラーの情報を提供します。

システム ログ メッセージを受信してモニタするためには、syslog 宛先（コンソール、ローカル ファイル、または syslog サーバを実行している 1 つ以上のリモート ホスト）を指定する必要があります。また、コンソールに表示されるか、ファイルまたはホストによってキャプチャされるメッセージのシビラティ（重大度）の最小値を指定できます。syslog メッセージを受信するローカル ファイルは `/var/log/external/messages` です。

Syslog 送信元は、オブジェクト モニタリング ポリシーを適用できる任意のオブジェクトにすることができます。送信されるメッセージのシビラティ（重大度）の最小値、syslog メッセージに含める項目、および syslog の宛先を指定できます。

Syslog の表示形式を NX-OS スタイル形式に変更できます。

これらのシステム メッセージを生成する障害またはイベントの詳細は、『*Cisco APIC Faults, Events, and System Messages Management Guide*』で説明しています。システム ログ メッセージのリストについては『*Cisco ACI System Messages Reference Guide*』を参照してください。

（注） システム ログ メッセージは、必ずしもシステムに問題があることを示しているとは限りません。単に情報を通知するだけのメッセージもありますし、通信回線、内部ハードウェア、またはシステム ソフトウェアに関する問題点の診断に役立つメッセージもあります。

Syslog の宛先および宛先グループの作成

この手順では、ロギングおよび評価用の syslog データの宛先を設定します。syslog データは、コンソール、ローカル ファイル、または宛先グループ内の 1 つまたは複数の syslog サーバにエクスポートできます。

手順

ステップ 1 メニュー バーで、[Admin] をクリックします。

ステップ 2 サブメニュー バーで、[External Data Collectors] をクリックします。

ステップ 3 [Navigation] ペインで、[Monitoring Destinations] を展開します。

ステップ 4 [Syslog] を右クリックし、[Create Syslog Monitoring Destination Group] を選択します。

ステップ 5 [Create Syslog Monitoring Destination Group] ダイアログボックスで、次の操作を実行します。

a) グループおよびプロファイルの [Name] フィールドに、モニタリングの宛先グループおよびプロファイルの名前を入力します。

b) グループおよびプロファイルの [Format] フィールドで、Syslog メッセージの形式を選択します。

デフォルトは [aci]、または RFC 5424 準拠のメッセージ形式ですが、NX-OS スタイル形式に設定することもできます。

c) グループおよびプロファイルの [Admin State] ドロップダウン リストで、[enabled] を選択します。

d) ローカル ファイルへの syslog メッセージの送信を有効にするには、[Local File Destination] の [Admin State] ドロップダウン リストから [enabled] を選択し、[Local File Destination] の [Severity] ドロップダウン リストからシビラティ（重大度）の最小値を選択します。

syslog メッセージを受信するローカル ファイルは `/var/log/external/messages` です。

e) コンソールへの syslog メッセージの送信を有効にするには、[Console Destination] の [Admin State] ドロップダウン リストから [enabled] を選択し、[Console Destination] の [Severity] ドロップダウン リストからシビラティ（重大度）の最小値を選択します。

f) [Next] をクリックします。

g) [Create Remote Destinations] 領域で、[+] をクリックしてリモート宛先を追加します。

注意 指定した DNS サーバがインバンド接続を介して到達可能に設定されている場合、リモート syslog 宛先のホスト名解決に失敗するリスクがあります。この問題を回避するには、IP アドレスを使用して syslog サーバを設定します。ホスト名を使用する場合は、アウトオブバンド インターフェイス経由で DNS サーバに到達できることを確認します。

ステップ 6 [Create Syslog Remote Destination] ダイアログボックスで、次の操作を実行します。

a) [Host] フィールドに、送信先ホストの IP アドレスまたは完全修飾ドメイン名を入力します。

b) （任意） [Name] フィールドに、宛先ホストの名前を入力します。

c) [Admin State] フィールドで、[enabled] オプション ボタンをクリックします。
d) （任意） 最小シビラティ（重大度）、**[シビラティ（重大度）（Severity）]**、**[ポート（Port）]** 番号、および syslog **[ファシリティ（Facility）]**を選択します。

[ファシリティ（Facility）]は、メッセージを生成したプロセスを示すためにオプションで使用できる番号で、受信側でのメッセージの処理方法を決定するために使用できます。

e) 5.2（3）以降のリリースでは、**[トランスポート（Transport）]** フィールドで、メッセージに使用するトランスポート プロトコルを選択します。

- リリース 5.2(4) より前のリリースでは、メッセージに使用するトランスポート プロトコルとして **tcp** または **udp** を選択します。
- 5.2(4) リリース以降では、メッセージに使用するトランスポートプロトコルのオプションとして、**ssl** も選択できるようになりました。この機能を使用すると、（クライアントとして機能している）ACI スイッチが、ロギングにセキュアな接続をサポートする（サーバーとして機能している）リモート Syslog サーバーに対してセキュアな暗号化されたアウトバウンド接続を確立できるようになります。認証と暗号化により、この機能では、セキュリティ保護されていないネットワーク上でもセキュアな通信を実現できます。

  メッセージに使用するトランスポート プロトコルとして **ssl** を選択した場合は、必要な SSL 証明書もアップロードする必要があることに注意してください。**[認証局の作成（Create Certificate Authority）]** ウィンドウに移動して、必要な SSL 証明書をアップロードできます。

  [管理（Admin）] > **[AAA]** > **[セキュリティ（Security）]** > **[公開キー管理（Public Key Management）]** > **[認証局（Certificate Authorities）]**を選択し、その後**[アクション（Actions）]** > **[認証局の作成（Create Certificate Authority）]**を選択します。

トランスポート プロトコルのデフォルト オプションは **udp** です。

f) [Management EPG] ドロップダウン リストから管理エンドポイント グループを選択します。
g) [OK] をクリックします。

ステップ 7 （任意） リモート宛先グループにリモート宛先を追加するには、もう一度 [+] をクリックし、[Create Syslog Remote Destination] ダイアログボックスの手順を繰り返します。

ステップ 8 **[終了]** をクリックします。

Syslog 送信元の作成

Syslog 送信元は、オブジェクト モニタリング ポリシーを適用できる任意のオブジェクトにすることができます。

始める前に

syslog モニタリング宛先グループを作成します。

手順

ステップ 1 メニュー バーおよびナビゲーション フレームから、関心領域の [Monitoring Policies] メニューに移動します。

テナント、ファブリック、およびアクセスのモニタリング ポリシーを設定できます。

ステップ 2 [Monitoring Policies] を展開し、モニタリング ポリシーを選択して展開します。

[Fabric] > [Fabric Policies] > [Monitoring Policies] > [Common Policy] の下に、基本モニタリング ポリシーがあります。このポリシーは、すべての障害とイベントに適用され、ファブリック内のすべてのノードとコントローラに自動的に導入されます。または、スコープが限定された既存のポリシーを指定することもできます。

ステップ 3 モニタリング ポリシーの下で、[Callhome/SNMP/Syslog] をクリックします。

ステップ 4 [Work] ペインで、[Source Type] ドロップダウン リストから [Syslog] を選択します。

ステップ 5 [Monitoring Object] リストから、モニタ対象の管理対象オブジェクトを選択します。

目的のオブジェクトがリストに表示されない場合は、次の手順に従います。

a) [Monitoring Object] ドロップダウン リストの右側にある [Edit] アイコンをクリックします。
b) [Select Monitoring Package] ドロップダウン リストから、オブジェクト クラス パッケージを選択します。
c) モニタ対象の各オブジェクトのチェックボックスをオンにします。
d) [Submit] をクリックします。

ステップ 6 テナント モニタリング ポリシーでは、[All] ではなく特定のオブジェクトを選択すると、[Scope] 選択が表示されます。

[Scope] フィールドで、オプション ボタンを選択して、このオブジェクトに関して送信するシステム ログ メッセージを指定します。

- [all]：このオブジェクトに関連するすべてのイベントと障害を送信します。
- [specific event]：このオブジェクトに関連する指定されたイベントのみを送信します。[Event] ドロップダウン リストからイベント ポリシーを選択します。
- [specific fault]：このオブジェクトに関連する指定された障害のみを送信します。[Fault] ドロップダウン リストから障害ポリシーを選択します。

ステップ 7 [+] をクリックして syslog 送信元を作成します。

ステップ 8 [Create Syslog Source] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、syslog 送信元の名前を入力します。
b) [Min Severity] ドロップダウン リストから、送信するシステム ログ メッセージのシビラティ（重大度）の最小値を選択します。
c) [Include] フィールドで、送信するメッセージ タイプのチェックボックスをオンにします。
d) [Dest Group] ドロップダウン リストから、システム ログ メッセージの送信先の syslog 宛先グループを選択します。

e) [Submit] をクリックします。

ステップ 9 （任意） syslog 送信元を追加するには、もう一度 [+] をクリックし、[Create Syslog Source] ダイアログボックスの手順を繰り返します。

REST API を使用した NX-OS CLI 形式での Syslog 表示の有効化

デフォルトで Syslog 形式は RFC 5424 に準拠しています。次の例のように、Syslog のデフォルト表示を NX-OS タイプ形式に変更できます。

```
apic1# moquery -c "syslogRemoteDest"
Total Objects shown: 1

# syslog.RemoteDest
host                : 172.23.49.77
adminState          : enabled
childAction         :
descr               :
dn                  : uni/fabric/slgroup-syslog-mpod/rdst-172.23.49.77
epgDn               :
format              : nxos
forwardingFacility : local7
ip                  :
lcOwn               : local
modTs               : 2016-05-17T16:51:57.231-07:00
monPolDn            : uni/fabric/monfab-default
name                : syslog-dest
operState           : unknown
port                : 514
rn                  : rdst-172.23.49.77
severity            : information
status              :
uid                 : 15374
vrfId               : 0
vrfName             :
```

NX-OS タイプ形式で Syslog を表示できるようにするには、REST API を使用して次の手順を実行します。

手順

ステップ 1 次の例に示すように、NX-OS タイプ形式での Syslog の表示を有効にします。

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="nxos">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

syslogGroup は Syslog モニタリングの宛先グループ、**sysLogRemoteDest** は事前に設定した Syslog サーバの名前、**host** は事前に設定した Syslog サーバの IP アドレスです。

ステップ 2 次の例に示すように、Syslog 形式をデフォルトの RFC 5424 形式に戻します。

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="aci">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

Traceroute を使用したパスの検出と接続性のテスト

このセクションでは、tracerouteの注意事項と制限事項をリストし、エンドポイント間でtracerouteを実行する方法について説明します。

トレースルートの概要

トレースルートツールは、パケットが送信先に移動するときに実際に通るルートを検出するために使用されます。tracerouteでは、ホップごとに使用されるパスが識別され、双方向で各ホップにタイムスタンプが付けられます。tracerouteを使用すると、発信元のデバイスと送信先に最も近いデバイスの間のパスに沿ってポート接続をテストできます。送信先に到達できない場合は、パス検出によってパスが障害ポイントまで追跡されます。

テナントのエンドポイントから開始されたトレースルートは、入力リーフのスイッチに表示される中間ホップとしてデフォルト ゲートウェイを示します。

トレースルートでは、次のようなさまざまなモードがサポートされています。

- エンドポイント間、リーフ間（トンネル エンドポイント、または TEP 間）
- エンドポイントから外部 IP
- 外部 IP からエンドポイント
- 外部 IP 間

トレースルートはファブリック全体のすべてのパスを検出し、外部エンドポイントの出口を検出します。パスが妨げられているかどうかを発見するのに役立ちます。

Windows および Linux トレースルートについて

traceroute コマンドを使用すると、パケットが通過した一連のホップを返すことで、特定の送信元からパケットが接続先に到達するまでのパスを判断できます。このユーティリティは、ホスト オペレーティング システム（Linux や Microsoft（MS）Windows）に付属しています。

送信元デバイス (ホスト、またはホストとして機能するルータなど) で **traceroute** *ip-address* コマンドを実行すると、指定された最大値まで増加する存続可能時間 (TTL) 値を持つ IP パケットが宛先に送信されます。ホップカウント。これはデフォルトで 30 です。通常、宛先へのパスにある各ルータは、これらのパケットを転送している間、TTL フィールドを 1 単位だけ減らします。パスの途中にあるルータが TTL = 1 のパケットを見つけると、インターネット制御

メッセージ プロトコル (ICMP) の「時間超過」メッセージでソースに応答します。このメッセージは、パケットがその特定のルータをホップとして通過することを送信元に知らせます。

（注） 以下の Linux および Windows セクションで説明するように、さまざまなオペレーティング システムで **traceroute** コマンドを実装する方法にはいくつかの違いがあります。

Linux

最初のユーザー データグラム プロトコル（UDP）データグラム プローブの TTL は、1（または拡張 **traceroute** コマンドでユーザーが指定した最小 TTL）に設定されます。初期データグラム プローブの宛先 UDP ポートは 33434（または拡張 **traceroute** コマンド出力で指定されたとおり）に設定されています。拡張 **traceroute** コマンドは、通常の **traceroute** コマンドのバリエーションであり、TTL や宛先ポート番号などの **traceroute** 操作で使用されるパラメーターのデフォルト値を変更できます。初期データグラム プローブのソース UDP ポートはランダム化されており、論理演算子 OR と 0x8000 が含まれています（最小ソース ポートが 0x8000 であることを保証します）。これらの手順は、UDPデータグラムが起動されたときに何が起こるかを示しています。

（注） パラメータは構成可能です。この例は、n = 1 で始まり、n = 3 で終わります。

1. UDP データグラムは、TTL = 1、宛先 UDP ポート = 33434、および送信元ポートがランダム化された状態で配信されます。
2. UDP 宛先ポートが増分され、送信元 UDP ポートがランダム化され、2 番目のデータグラムが配信されます。
3. ステップ 2 は、最大 3 つのプローブに対して（または拡張 **traceroute** コマンド出力で要求される回数）繰り返されます。送信されたプローブごとに、宛先ホストへの段階的なパスを構築するために使用される「TTL 超過」メッセージを受信します。
4. ICMP の「時間超過」メッセージを受信すると、TTL が増分され、このサイクルが増分の宛先ポート番号で繰り返されます。次のいずれかのメッセージを受け取ることもできます。
   - ホストに到達したことを示す ICMP タイプ 3、コード 3 (「接続先到達不能」、「ポート到達不能」) メッセージ。
   - 「ホスト到達不能」、「ネット到達不能」、「最大 TTL 超過」、または「タイムアウト」タイプのメッセージ。これは、プローブが再送信されたことを意味します。

Cisco ルータは、ランダムな送信元ポートと増分の宛て先ポートを使用して UDP プローブ パケットを送信します (異なるプローブを区別するため)。Cisco ルータは、UDP/ICMP パケットを受信した送信元に ICMP メッセージ「時間超過」を送信します。

Linux **traceroute** コマンドは、Cisco ルータの実装に似ています。ただし、固定送信元ポートを使用します。**traceroute** コマンドの *-n* オプションは、ネーム サーバーへの要求を回避するために使用されます。

（注） UCS サーバーの CIMC コントローラは、UDP ベースの traceroute メッセージに応答しません。ICMP ベースの traceroute にのみ応答します。デフォルトでは、Windows traceroute は ICMP ベースのメッセージを送信します。Linux（および Mac）traceroute は、デフォルトで UDP ベースのメッセージを送信します。-I（大文字の i）オプションを使用すると、Linux（および Mac）の traceroute は ICMP ベースのメッセージを送信します。

Linux の traceroute がデフォルトであるため、ACI ネットワークのトラブルシューティングで traceroute を Cisco APIC に送信する必要がある場合は、Windows の traceroute を使用するか、ICMP ベースの traceroute を指定する必要があります。

Windows

MS Windows の **tracert** コマンドは、UDP データグラムの代わりに ICMP エコー要求データグラムをプローブとして使用します。ICMP エコー要求は、TTL を増分して起動され、上記と同じ動作が発生します。ICMP エコー要求データグラムを使用する意義は、最終ホップが宛先ホストからの ICMP 「到達不能」メッセージの応答に依存しないことです。代わりに、ICMP エコー応答メッセージに依存します。

コマンド構文は次のとおりです。

```
tracert [-d] [-h maximum_hops] [-j computer-list] [-w timeout] target_name
```

次の表に、コマンド パラメータについての説明を記載します。

表 2:

パラメータ	説明
-d	アドレスをコンピュータ名に解決しないように指定します。
-h maximum_hops	ターゲットを検索する最大ホップ カウントを指定します。
-j computer-list	computer-list に沿ったルーズなソース ルートを指定します。
-w timeout	各応答のタイムアウトで指定されたミリ秒数を待機します。
target_name	ターゲット コンピュータの名前。

トレースルートの注意事項および制約事項

- トレースルートの送信元または宛先がエンドポイントである場合、そのエンドポイントはスタティックではなくダイナミックである必要があります。ダイナミックエンドポイント（fv:CEp）とは異なり、スタティックエンドポイント（fv:StCEp）にはトレースルートに必要な子オブジェクト（fv:RsCEpToPathEp）がありません。
- トレースルートはIPv6の送信元と宛先で動作しますが、IPv4アドレスとIPv6アドレスを混在させて送信元IPアドレスと宛先IPアドレスを設定することはできません。
- トレースルート関連の制限については、『*Verified Scalability Guide for Cisco ACI*』ドキュメントを参照してください。
- エンドポイントを新しいMACアドレス（トレースルートポリシーを設定する際に指定したMACアドレスと異なる）のToRスイッチに移動すると、トレースルートポリシーでそのエンドポイントに「missing-target」と表示されます。この場合は、新しいMACアドレスを指定して新しいトレースルートポリシーを設定する必要があります。
- ポリシーベースのリダイレクト機能を含むフローに対してトレースルートを実行する場合、パケットがサービスデバイスからリーフスイッチに送信されるときに、リーフスイッチが存続時間（TTL）期限切れメッセージを送信するために使用するIPアドレスが、常に、サービスデバイスのブリッジドメインのスイッチ仮想インターフェイス（SVI）のIPアドレスです。この動作は表面的なものであり、トラフィックが予期された経路をとっていないことを示すものではありません。

エンドポイント間での **traceroute** の実行

手順

ステップ 1 メニューバーで、[Tenants] をクリックします。

ステップ 2 サブメニューバーで、送信元エンドポイントを含むテナントをクリックします。

ステップ 3 **[ナビゲーション]** ペインでテナントを展開し、**[ポリシー]** > **[トラブルシューティング]** を展開します。

ステップ 4 [Troubleshoot] で次のトレースルートポリシーのいずれかを右クリックします。

- [Endpoint-to-Endpoint Traceroute Policies] を右クリックして [Create Endpoint-to-Endpoint Traceroute Policy] を選択する
- [Endpoint-to-External-IP Traceroute Policies] を右クリックして [Create Endpoint-to-External-IP Traceroute Policy] を選択する
- [External-IP-to-Endpoint Traceroute Policies] を右クリックして [Create External-IP-to-Endpoint Traceroute Policy] を選択する
- [External-IP-to-External-IP Traceroute Policies] を右クリックして [Create External-IP-to-External-IP Traceroute Policy] を選択する

ステップ 5 ダイアログボックスのフィールドに適切な値を入力し、[Submit] をクリックします。

（注） フィールドの説明については、ダイアログボックスの右上隅にあるヘルプ アイコン（[?]）をクリックしてください。

ステップ 6 **[Navigation]** ペインまたは **[Traceroute Policies]** テーブルで、traceroute ポリシーをクリックします。

トレースルート ポリシーが [Work] ペインに表示されます。

ステップ 7 [Work] ペインで [Operational] タブをクリックし、[Source Endpoints] タブ、[Results] タブの順にクリックします。

ステップ 8 **[Traceroute Results]** テーブルで、追跡に使用された単数または複数のパスを確認します。

（注）
- 複数のパスが、送信元ノードから宛先ノードへの移動に使用されている場合があります。
- **[Name]** 列など、1 つまたは複数の列の幅を広げると確認しやすくなります。

トラブルシューティング ウィザードを使用

トラブルシューティング ウィザードを使用すると、ネットワークの動作を理解して可視化できるため、問題が発生した場合にネットワークに関する懸念を緩和できます。たとえば、2 つのエンドポイントで断続的なパケット損失が発生していて、その理由がわからない場合があります。トラブルシューティング ウィザードを使用すると、問題を評価することができるため、この問題のある動作の原因と思われる各マシンにログオンしなくても、問題を効果的に解決できます。

このウィザードを使用すると、管理ユーザは、選択した送信元と接続先の特定の時間枠に発生する問題のトラブルシューティングを行うことができます。デバッグを実行する時間枠を定義でき、TAC に送信できるトラブルシューティング レポートを生成できます。

トラブルシューティング ウィザードの開始

トラブルシューティング ウィザードの使用を開始する前に、管理ユーザとしてログオンする必要があります。次に、送信元と接続先を指定し、トラブルシューティング セッションの時間枠を選択する必要があります。時間枠は、イベント、障害レコード、展開レコード、監査ログ、および統計を取得するために使用されます。

トラブルシューティング ウィザードの画面をナビゲートするときには、いつでも、画面の右上にある **[プリント（Print）]** アイコン（  ）をクリックし、スクリーンショットを撮ってプ

リンタに送信するか、 PDF として保存することができます。画面の表示を変更するために使用できるズームインおよびズームアウトアイコン（  ）もあります。

（注）

- **[レポートの生成（Generate Report）]** または **[送信（Submit）]** をクリックした後は、送信元と接続先を変更できません。入力した送信元と接続先の情報を変更する場合は、現在のセッションを削除して、新しいセッションを開始する必要があります。
- **[送信（Submit）]** をクリックした後は、ウィザードの最初のページで説明と時間枠を変更することはできません。
- トラブルシューティング ウィザードで静的 IP アドレス エンドポイントを使用することはできません。
- 指定するエンドポイントはすべて、EPG の下にある必要があります。

トラブルシューティング セッション情報を設定するには、次の手順を実行します。

手順

ステップ 1 **[オペレーション（Operations）]**> > **[可視性とトラブルシューティング（Visibility & Troubleshooting）]** を選択します。

[可視性とトラブルシューティング（Visibility & Troubleshooting）] 画面が表示されます。

ステップ 2 **[セッション名（Session Name）]** フィールドで、ドロップダウンリストを使用して既存のトラブルシューティング セッションを選択するか、名前を入力して新しいセッションを作成します。

ステップ 3 **[セッション タイプ（Session Type）]** ドロップダウンリストから目的のセッション タイプを選択します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint）]**：送信元と接続先は両方とも内部エンドポイントです。

  同じテナントから送信元エンドポイントと接続先エンドポイントを選択する必要があります。そうしなかった場合、このドキュメントで後述するように、トラブルシューティング機能の一部が影響を受ける可能性があります。このセッションタイプでは、両方のエンドポイントが同じリーフスイッチのセットに接続している場合、アトミック カウンタを使用できません。

- **[エンドポイントから外部 IP（Endpoint to External IP）]**：送信元は内部エンドポイントであり、接続先は外部 IP アドレスです。
- **[外部 IP からエンドポイント（External IP to Endpoint）]**：送信元は外部 IP アドレスであり、接続先は内部エンドポイントです。

- **[外部 IP から外部 IP（External IP to External IP）]**：送信元と接続先は両方とも外部 IP アドレスです。3.2(6) リリース以降、このタイプを選択できます。このセッション タイプでは、トレースルート、アトミック カウンタ、または遅延を使用できません。

ステップ 4 （任意）**[説明（Description）]** フィールドに説明を入力し、追加情報を入力します。

ステップ 5 **[送信元（Source）]** エリアに送信元情報を入力します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint）]** または **[エンドポイントから外部 IP（Endpoint to External IP）]** のセッション タイプを選択した場合は、MAC、IPv4、または IPv6 アドレス、または VM 名を入力して、**[検索（Search）]** をクリックします。

  セッション タイプが **[エンドポイントからエンドポイント（Endpoint to Endpoint）]** であり、両方のエンドポイントの MAC アドレスにそれらから学習された IP アドレスがない場合にのみ、MAC アドレスを入力できます。

  選択に役立つ詳細情報を含む 1 つ以上の行を表示するボックスが表示されます。各行は、入力した IP アドレス（**[IP]** 列）が特定のエンドポイント グループ（**[EPG]** 列）にあり、特定のテナント（**[テナント（Tenant）]** 列内）にある、特定のアプリケーション（**[アプリケーション（Application）]** 列）に属していることを示しています。リーフスイッチ番号、FEX 番号、およびポートの詳細は、**[学習した場所（Learned At）]** 列に表示されます。

- **[外部 IP からエンドポイント（External IP to Endpoint）]** のセッション タイプを選択した場合は、外部 IP アドレスを入力します。

- **[外部 IP から外部 IP（External IP to External IP）]** へのセッション タイプを選択した場合は、外部レイヤ 3 外部ネットワークの外部 IP アドレスと識別名を入力します。

ステップ 6 **[接続先（Destination）]** エリアに接続先情報を入力します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint））]** または**[外部 IP からエンドポイント（External IP to Endpoint）]** のセッション タイプを選択した場合は、MAC、IPv4、または IPv6 アドレス、または VM 名を入力して、**[検索（Search）]** をクリックします。

  セッション タイプが **[エンドポイントからエンドポイント（Endpoint to Endpoint）]** であり、両方のエンドポイントの MAC アドレスにそれらから学習された IP アドレスがない場合にのみ、MAC アドレスを入力できます。

  選択に役立つ詳細情報を含む 1 つ以上の行を表示するボックスが表示されます。各行は、入力した IP アドレス（**[IP]** 列）が特定のエンドポイント グループ（**[EPG]** 列）にあり、特定のテナント（**[テナント（Tenant）]** 列内）にある、特定のアプリケーション（**[アプリケーション（Application）]** 列）に属していることを示しています。リーフスイッチ番号、FEX 番号、およびポートの詳細は、**[学習した場所（Learned At）]** 列に表示されます。

- **[エンドポイントから外部 IP（Endpoint to External IP）]** のセッション タイプを選択した場合は、外部 IP アドレスを入力します。

- **[外部 IP から外部 IP（External IP to External IP）]** へのセッション タイプを選択した場合は、外部レイヤ 3 外部ネットワークの外部 IP アドレスと識別名を入力します。

ステップ 7 **[タイム ウィンドウ（Time Window）]** エリアで、タイム ウィンドウを指定します。

[タイム ウィンドウ（Time Window）] は、過去の特定の時間枠に発生した問題をデバッグするために使用され、イベント、すべてのレコード、展開レコード、監査ログ、および統計を取得するために使用されます。2つのウィンドウセットがあります。1つはすべてのレコード用で、もう 1 つは個々のリーフスイッチ（またはノード）用です。

デフォルトでは、**[最新（Latest Minutes）]** フィールドで指定した任意の（時間の）分数に基づいて、ローリング タイム ウィンドウを指定できます。デフォルトは 240 分です。セッションには、セッションを作成した時刻より前に指定した過去（分）のデータが含まれます。

[固定時間を使用（Use fixed time）] ボックスにチェックを入れると、**[開始（From）]** および **[終了（To）]** フィールドでセッションの固定時間ウィンドウを指定できます。セッションには、**[開始（From）]** から **[終了（To）]** 時刻までのデータが含まれます。

ステップ 8 **[送信（Submit）]** をクリックして、トラブルシューティング セッションを開始します。

しばらくすると、トラブルシューティング セッションのトポロジ図が表示されます。

トラブルシューティング レポートの生成

トラブルシューティング レポートは、JSON、XML、PDF、HTML などのいくつかのフォーマットで生成できます。フォーマットを選択すると、レポートをダウンロードする（またはレポートのダウンロードをスケジュールする）ことができ、オフライン分析に使用するか、サポート ケースを作成できるように TAC に送信することができます。

トラブルシューティングに関するレポートを生成するには、次をします：

手順

ステップ 1 画面の右下隅にある **[レポートを生成（GENERATE REPORT）]** をクリックします。
[レポート ジェネレータ（Report Generator）] ダイアログボックスが表示されます。

ステップ 2 [レポート形式] ドロップダウン メニューから出力フォーマット（**XML、HTML、JSON、**または **PDF**）を選択します。

ステップ 3 レポートのダウンロードをすぐに実行するようにスケジュールする場合は、**[今すぐ（Now）]**> **[送信（SUBMIT）]** をクリックします。
レポートが生成されると、レポートの入手先を示す **[情報（Information）]** ボックスが表示されます。

ステップ 4 レポートの生成を後でスケジュールするには、**[スケジューラを使用（Use a scheduler）]** > **[スケジューラ（Scheduler）]** ドロップダウン メニュークリックして、存在するスケジュールまた

は、**[スケジューラを作成（Create Scheduler）]** をクリックして新しいスケジューラを作成してそれを選択します。を選択します。
[トリガー スケジュールの作成（CREATE TRIGGER SCHEDULE）] ダイアログが表示されます。

ステップ 5 **[名前（Name）]**、**[説明（Description）]** (オプション)、および **[スケジュール ウィンドウ（Schedule Windows）]** フィールドに情報を入力します。

（注） **[スケジューラ（SCHEDULER）]** の使用方法の詳細については、オンライン ヘルプを参照してください。

ステップ 6 [SUBMIT] をクリックします。

レポートの生成には、ファブリックのサイズと障害またはイベントの数に応じて、数分から最大 10 分かかります。レポートの生成中はステータス メッセージが表示されます。トラブルシューティング レポートを取得して表示するには、**[生成されたレポートを表示（SHOW GENERATED REPORTS）]** をクリックします。

[必要な認証（Authentication Required）] ウィンドウで、サーバーの資格情報 **([ユーザー名（User Name）]**と**[パスワード（Password）])** を入力します。次に、トラブルシューティング レポートがシステムにローカルにダウンロードされます。

[すべてのレポート（ALL REPORTS）] ウィンドウが表示され、今、トリガーしたものを含む、生成されたすべてのレポートのリストが表示されます。そこから、選択した出力ファイル フォーマットに応じて、リンクをクリックしてレポートをダウンロードするか、すぐに表示することができます（たとえば、ファイルが PDF の場合、ブラウザですぐに開くことができます）。

トラブルシューティング ウィザードのトポロジについて

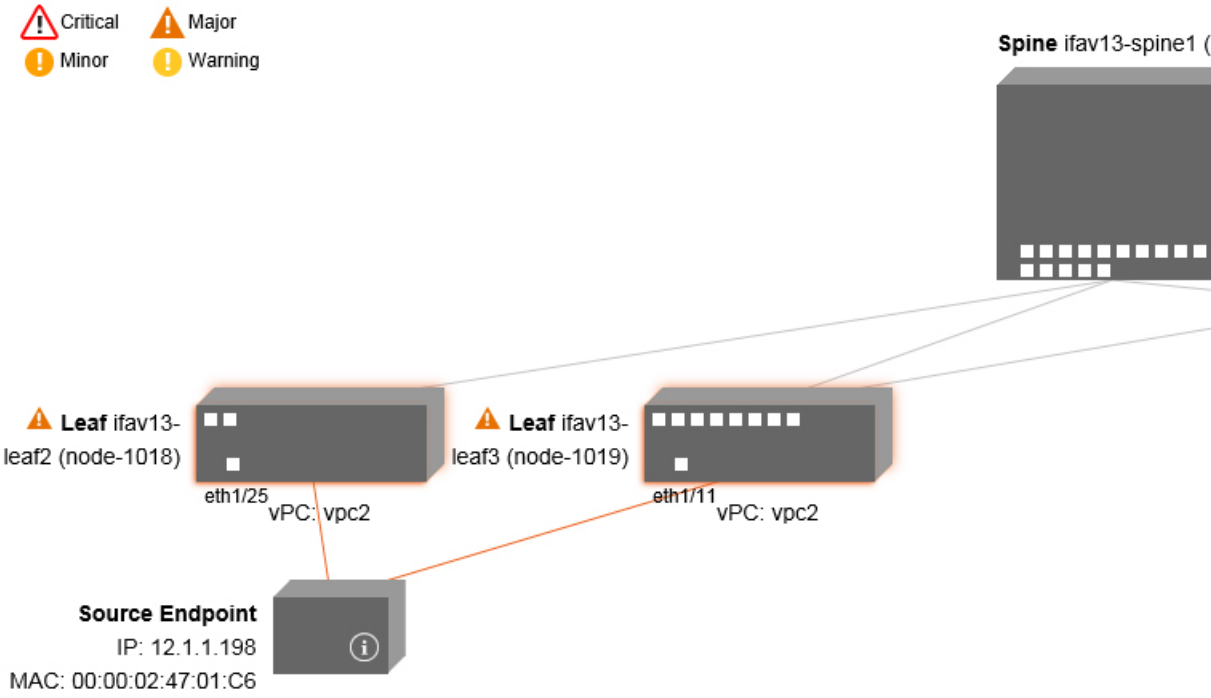
このセクションでは、トラブルシューティング ウィザードのトポロジについて説明します。トポロジは、送信元と接続先がどのようにファブリックに接続されているか、送信元から接続先までのネットワーク パス、および中間スイッチが何であるかを示しています。

次のウィザード トポロジ ダイアグラムに示すように、送信元はトポロジの左側に表示され、接続先は右側に表示されます。

（注） このウィザード トポロジには、送信元から接続先へのトラフィックに関係するデバイスのリーフ スイッチ、スパイン スイッチ、および FEX のみが表示されます。ただし、他の多くのリーフスイッチ（数十または数百のリーフスイッチと他の多くのスパイン スイッチ）が存在する場合があります。

このトポロジには、リンク、ポート、およびデバイスも表示されます。アイコンにカーソルを合わせると、送信元または接続先が属するテナント、それが属するアプリケーション、使用しているトラフィックのカプセル化（VLAN など）が表示されます。

画面の左側に色の凡例があり（次のように表示されます）、トポロジ図の各色に関連付けられたシビラティ（重大度）レベル（たとえば、クリティカルとマイナー）を説明します。



トポロジ内のボックスやポートなどの項目にカーソルを合わせると、より詳細な情報が表示されます。ポートまたはリンクに色が付いている場合は、トラブルシューティングが必要な問題があることを意味します。たとえば、色が赤またはオレンジの場合、これはポートまたはリンクに障害があることを示しています。色が白の場合、障害はありません。リンクで円の中に数字がある場合は、同じ2つのノード間で円の色によって示されるシビラティ（重大度）の障害の影響を受けている並列リンクの数を示しています。ポートにカーソルを合わせると、送信元に接続されているポートを確認できます。

リーフスイッチを右クリックすると、スイッチのコンソールにアクセスできます。そのデバイスにログインできるポップアップ ウィンドウが表示されます。

（注）

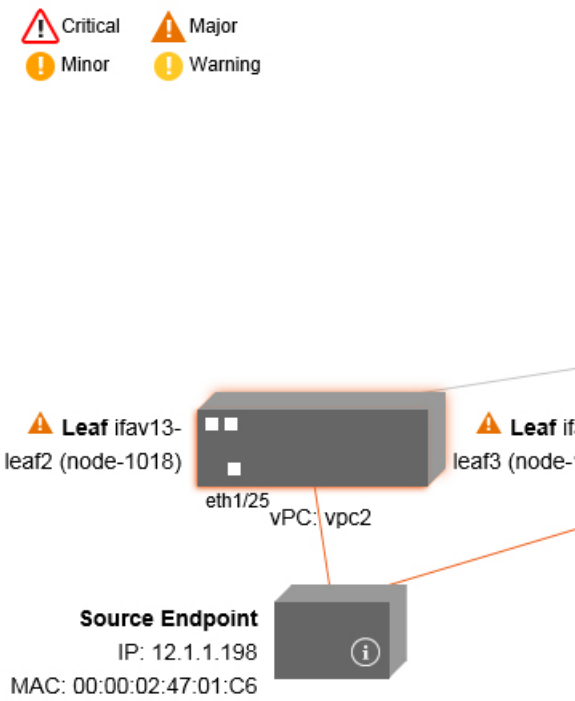
- レイヤ 4 からレイヤ 7 のサービス（ファイアウォールとロードバランサ）がある場合、それらもトポロジに表示されます。
- ロードバランサを使用するトポロジの場合、接続先は仮想IP（VIP）アドレスであることが想定されます。
- 送信元または接続先が ESX サーバーの背後にある場合、ESX はトポロジに表示されます。

障害トラブルシューティング画面の使用

この手順では、トラブルシューティング ウィザードの障害の使用方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**[ナビゲーション（Navigation）]** ペインで **[障害 （Faults）]** をクリックして、**[障害（Faults）]** トラブルシューティング画面の使用を開始します。	**[障害 （Faults）]** 画面には、以前に選択した送信元と接続先を接続するトポロジと、見つかった障害が表示されます。指定された通信の障害のみが表示されます。障害がある場合は常に、重大度を伝えるために特定の色で強調表示されます。画面上部の色の凡例を参照して、各色に関連付けられた重大度レベルを理解してください。白いボックスは、その特定のエリアにトラブルシューティングする問題がないことを示しています。 このトポロジには、トラブルシューティング セッションに関連するリーフ スイッチ、スパイン スイッチ、およびFEXも表示されます。リーフ スイッチ、スパイン スイッチ、FEXなどの項目にカーソルを合わせるか、障害をクリックすると、分析のためのより詳細な情報が表示されます。

	コマンドまたはアクション	目的
		
ステップ 2	障害をクリックすると、分析のためのより詳細な情報を含む **[ドロップ統計（Drop Stats）]**、**[連絡先ドロップ（Contract Drops）]**、および **[トラフィック統計（Traffic Stats）]** タブのあるダイアログボックスが表示されます。	

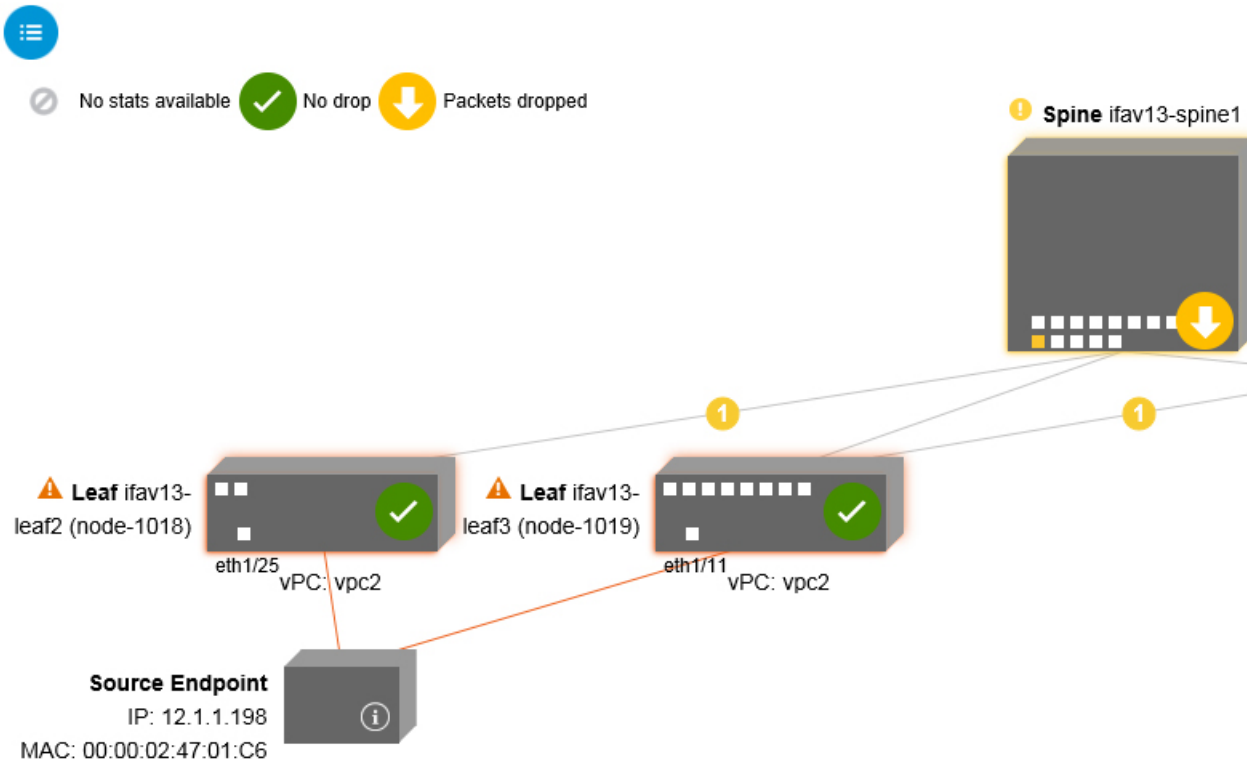
関連トピック

ドロップ / 統計トラブルシューティング画面の使用 （91 ページ）

ドロップ / 統計トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[ドロップ / 統計（Drop/Stats）]** をクリックして、**[ドロップ / 統計（Drop/Stats）]**のトラブルシューティング画面の使用を開始します。

[ドロップ / 統計（Drop/Stats）] ウィンドウには、ドロップからのすべての統計情報を含むトポロジが表示されるため、ドロップが存在するかどうかを明確に確認できます。ドロップ画像をクリックすると、分析のための詳細情報が表示されます。



ドロップ画像をクリックすると、**[ドロップ / 統計（Drop/Stats）]** 画面の上部に 3 つのタブがあり、表示される統計はその特定のリーフまたはスイッチにローカライズされます。

3 つの統計タブは次のとおりです。

- **[ドロップ統計（DROP STATS）]**

  このタブには、ドロップカウンターの統計が表示されます。さまざまなレベルでドロップされるパケットがここに表示されます。

（注） デフォルトでは、値がゼロのカウンターは非表示になっていますが、ユーザーはすべての値を表示することを選択できます。

- **[契約の落ち込み（CONTRACT DROPS）]**

  このタブには、個々のパケットログ（ACL ログ）である、発生したコントラクトドロップのリストが表示され、**[送信元インターフェイス、送信元 IP アドレス、送信元ポート、宛先 IP アドレス、宛先ポート（Source Interface, Source IP address, Source Port, Destination IP address, Destination Port,）]** と **[プロトコル（Protocol）]** などの各パケットの情報が表示されます。

（注） すべてのパケットがここに表示されるわけではありません。

- **[トラフィック 統計情報（TRAFFIC STATS）]**

  このタブには、進行中のトラフィックを示す統計が表示されます。これらは、転送されたパケットの数です。

（注） デフォルトでは、値がゼロのカウンターは非表示になっていますが、ユーザーはすべての値を表示することを選択できます。

画面の左上隅にあるすべてアイコン（  ）をクリックして、すべての管理対象オブジェクトのすべての統計を一度に表示することもできます。

ゼロまたはゼロ以外のドロップを選択するオプションもあります。**[値がゼロの統計を表示（Show stats with zero values）]**（画面の左上隅にあります）のチェックボックスをオンにすると、既存のすべてのドロップを表示できます。**[時間、影響を受けたオブジェクト、統計（Time、Affected Object、Stats）]**、および **[値（Value）]** のフィールドには、すべてのゼロ値のデータが入力されます。

[ゼロ値の統計を表示（Show stats with zero values）] ボックスをチェックしない場合、ゼロ以外のドロップで結果が表示されます。

（注） **[すべて（All）]** アイコンをクリックした場合も、同じロジックが適用されます。3つすべてのタブ（**[ドロップ統計（DROP STATS）]**、**[契約ドロップ（CONTRACT DROPS）]**、および **[トラフィック統計（TRAFFIC STATS）]**）も使用でき、同じタイプの情報が表示されます。

関連トピック

契約トラブルシューティング画面の使用 （93 ページ）

契約トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[コントラクト（Contracts）]** をクリックして、**[コントラクト（Contracts）]** のトラブルシューティング画面の使用を開始します。

[コントラクト（Contracts）] のトラブルシューティング画面には、送信元から宛先、および宛先から送信元に適用可能な契約が表示されます。

青いテーブルの見出しの各行は、フィルタを示しています。各フィルタの下には、特定のリーフまたはスイッチの複数のフィルタ エントリ（**プロトコル、L4 Src、L4 宛先、TCP フラグ、アクション、ノード**、および**ヒット**）を示す複数の行があります。

証明書アイコンにカーソルを合わせると、コントラクト名とコントラクトフィルタ名が表示されます。各青いテーブルの見出し行（またはフィルタ）の右側に表示されるテキストは、コントラクトのタイプを示します。次に例を示します。

- Epg から Epg
- BD 許可
- あらゆる状況に対応
- コンテキスト拒否

これらのコントラクトは、送信元から宛先へ、および宛先から送信元へと分類されます。

（注） 各フィルタに表示されるヒットは累積的です（つまり、特定のリーフごとに、そのコントラクトヒット、コントラクトフィルタ、またはルールの合計ヒットが表示されます）。統計は（1）分ごとに自動的に更新されます。

情報（  ）アイコンにカーソルを合わせると、ポリシー情報を取得できます。また、参照されている EPG を確認することもできます。

（注） エンドポイント間にコントラクトがない場合、これは **[契約データがありません（There is no contract）]** ポップアップで示されます。

関連トピック

イベント トラブルシューティング画面の使用 （94 ページ）

イベント トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[イベントと監査（Events and Audits）]** をクリックして、**[イベントと監査（Events and Audits）]** トラブルシューティング画面の使用を開始します。

個々のリーフまたはスパインスイッチをクリックすると、その個々のイベントに関するより詳細な情報を表示できます。

[イベント（EVENTS）] と **[導入記録（DEPLOYMENT RECORDS）]** の 2 つのタブを使用できます。

- **[イベント（EVENTS）]** は、システム（物理インターフェースや VLANS など）で発生した変更のイベントレコードを表示します。特定のリーフごとに個別のイベントがリストされています。これらのイベントは、**[重大度、影響を受けるオブジェクト、作成時間、原**

因（**Severity, Affected Object, Creation Time, Cause**）]、および**[説明（Description）]**に基づいて並べ替えることができます。

- **[導入記録（DEPLOYMENT RECORDS）]**は、物理インターフェイス、VLAN、VXLAN、および L3 CTX でのポリシーの展開を示しています。これらのレコードは、epg のために VLAN がリーフに配置された時刻を示しています。

[すべての変更（All Changes）] 画面の **[すべて（All）]** アイコン（  ）をクリックすると、指定した時間間隔（またはトラブルシューティング セッション）中に発生した変更を示すすべてのイベントを表示できます。

[すべての変更（All Changes）] 画面には、次の 3 つのタブがあります。

- **[監査（AUDITS）]**

  監査にはリーフ アソシエーションがないため、**[すべての変更（All Changes）]** 画面でのみ使用できます。

- **[イベント（EVENTS）]**（上記）
- **[展開記録（DEPLOYMENT RECORDS）]**（上記）

関連トピック

トレースルート トラブルシューティング画面の使用 （95 ページ）

トレースルート トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[トレースルート（Traceroute）]** をクリックして、**[トレースルート（Traceroute）]** トラブルシューティング画面の使用を開始します。

トラブルシューティングのために traceroute を作成して実行するには、次の手順を実行します。

1. **[トレースルート（Traceroute）]** ダイアログボックスで、**[接続先ポート（Destination Port）]** ドロップダウン リストで、接続先ポートを選択します。
2. **[Protocol（プロトコル）]** プルダウン メニューからプロトコルを選択します。サポートされているオプションは次のとおりです。
   - **icmp**：このプロトコルは単方向であり、ソース リーフから接続先エンドポイントのみへの traceroute を実行します。
   - **tcp** ：このプロトコルは、**udp** プロトコルについて上で説明したように、双方向でもあります。
   - **udp** ：このプロトコルは双方向であり、ソース リーフから宛先エンドポイントへの traceroute を実行し、次に接続先リーフからソース エンドポイントへの traceroute を実行します。

（注） IPv4 でサポートされているプロトコルは、UDP、TCP、および ICMP のみです。IPv6 の場合、UDP のみがサポートされます。

3. traceroute を作成したら、**[再生（Play）]**（または Start）ボタンをクリックして traceroute を開始します。

（注） **[再生（Play）]** ボタンを押すと、システム上にポリシーが作成され、**警告**メッセージが表示されます。

4. **[OK]** をクリックして続行すると、traceroute の実行が開始されます。
5. **[停止（Stop）]** ボタンをクリックして、traceroute を終了します。

（注） **[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

traceroute が完了すると、起動された場所と結果が表示されます。**トレースルートの結果（Traceroute Results）** の隣には、traceroute が起動された場所（ソースから接続先へ、または接続先からソースへ）を示すプルダウン メニューがあります。

結果は、**実行時間、Traceroute ステータス、接続先ポート**、および**プロトコル**の情報を含む **[トレースルート（Traceroute）]** ダイアログにも表示されます。

結果は、緑および/または赤の矢印で表されます。緑の矢印は、traceroute プローブに応答したパス内の各ノードを表すために使用されます。赤い矢印の始点は、トレースルートプローブに応答した最後のノードであるため、パスが終了する場所を表します。traceroute を起動する方向を選択しません。代わりに、traceroute は常にセッションに対して開始されます。セッションが次の場合：

- EP から外部 IP または外部 IP から EP の場合、traceroute は常に EP から外部 IP に起動されます。
- EP から EP およびプロトコルが ICMP である場合、traceroute は常に送信元から接続先へ開始します。
- EP から EP およびプロトコルは UDP/TCP で、traceroute は常に双方向です。

（注）

- **[トレースルートの結果（Traceroute Results）]**ドロップダウン メニューを使用して、上記のシナリオ #3 の各方向の結果を表示/視覚化できます。シナリオ #1 と #2 では、常にグレー表示されます。
- **トレースルート ステータス（Traceroute Status）**が不完全と表示される場合、これは、データの一部が戻ってくるのをまだ待っていることを意味します。**トレースルート ステータス（Traceroute Status）**が**完了（Complete）**と表示されている場合は、実際には完了しています。

関連トピック

アトミック カウンタ トラブルシューティング画面を使用 （97 ページ）

アトミック カウンタ トラブルシューティング画面を使用

[ナビゲーション（Navigation）] ペインの **[アトミック カウンタ（Atomic Counter）]** をクリックして、**[アトミック カウンタ（Atomic Counter）]** のトラブルシューティング画面の使用を開始します。

Atomic Counter 画面は、送信元と接続先の情報を取得し、それに基づいてカウンター ポリシーを作成するために使用されます。2 つのエンドポイント間にアトミック カウンタ ポリシーを作成し、送信元から宛接続先、および接続先から送信元に行き来するトラフィックをモニタリングできます。通過するトラフィックの量を判断でき、特に、送信元と宛先のリーフ間で異常（パケットのドロップまたは超過）が報告されているかどうかを判断できます。

画面の上部に **[再生（Play）]** (または [開始]) および **[停止（Stop）]** ボタンがあるため、いつでもアトミック カウンタ ポリシーを開始および停止でき、送信されているパケットをカウントできます。

（注）

[再生（Play）]ボタンを押すと、システム上にポリシーが作成され、パケット カウンターが開始されます。**[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

結果は 2 つの異なるフォーマットで表示されます。それらは、要約を含む短いフォーマットで表示することも、長いフォーマットで表示することもできます **([展開（Expand）]** ボタンをクリックします)。簡潔なフォーマットと拡張されたフォーマットの両方で、両方の方向が示されます。拡張フォーマットでは、累積カウントと最新の 30 秒間隔ごとのカウントが表示されますが、簡易フォーマットでは、累積および最後の間隔のカウントのみが表示されます。

関連トピック

SPAN トラブルシューティング画面の使用 （98 ページ）

SPAN トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **SPAN** をクリックして、**SPAN** トラブルシューティング画面の使用を開始します。

この画面を使用して、双方向トラフィックをスパン（またはミラーリング）して、アナライザにリダイレクトできます。SPAN セッションでは、コピーを作成してアナライザに送信します。

このコピーは特定のホスト（アナライザーの IP アドレス）に送信され、Wireshark などのソフトウェア ツールを使用してパケットを表示できます。セッション情報には、送信元と接続先の情報、セッション タイプ、およびタイムスタンプの範囲があります。

（注） **[再生（Play）]** ボタンを押すと、システム上にポリシーが作成されます。**[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

（注） トラブルシューティング ウィザードの CLI コマンドのリストについては、『*Cisco APIC Command-Line Interface User Guide*』を参照してください。

Cisco APIC トラブルシューティング CLI を使用して SPAN セッションを作成

このセクションでは、Cisco APIC トラブルシューティング CLI を使用して SPAN セッションを作成する方法を示します。

手順

ステップ 1 **troubleshoot node session** <*session_name*> **nodename** <*node_id*>

ノードレベルのセッション（グローバル ドロップ）を作成するには：

例：

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301
```

ステップ 2 **troubleshoot node session** <*session_name*> **nodename** <*node_id*> **interface ethernet** <*interface*>

インターフェイス レベルのセッションを作成するには、次の手順を実行します：

例：

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301 interface eth1/3
```

ステップ 3 **troubleshoot node session** <*session_name*> **monitor destination** *apic_ip* **srcipprefix** <*ip_prefix*> **drop enable erspan-id**[optional]

接続先を Cisco APIC として指定し、ドロップ時に SPAN を有効にするには：

例：

```
apic1(config)# troubleshoot node  session 301-GD-APIC monitor destination apic srcipprefix
 13.13.13.13 drop enable
```

ステップ 4 **troubleshoot node session** <*session_name*> **monitor destination tenant** *tenant* **application** <*app*> **destip** <*dest_ip*>**srcipprefix**<*ip_prefix*>**drop enable erspan-id**[optional]

ERSPAN 接続先を指定し、ドロップ時に SPAN を有効にするには：

例：

```
apic1(config)# troubleshoot node  session 301-GD-APIC monitor destination tenant ERSPAN
 application A1 epg E1 destip 179.10.10.179 srcipprefix 31.31.13.31 drop enable
```

接続先として設定されているときに Cisco APIC で SPAN-on-drop パケットを確認するには、次の手順を実行します：

1. SPAN-on-drop セッションを無効にします。

   ```
   apic1(config)# no troubleshoot node session 301-GD-APIC monitor
   ```

2. drop-stats ディレクトリに移動し、DropPackets_*.pcap ファイルを確認します：**/data2/techsupport/troubleshoot/node/Session_name/span_capture/drop-stats/DropPackets_*.pcap**

L4 - L7 サービス検証済みシナリオ

トラブルシューティング ウィザードを使用すると、2 つのエンドポイントを提供し、それらのエンドポイント間の対応するトポロジを表示できます。トポロジ内の 2 つのエンドポイント間に L4 - L7 サービスが存在する場合、これらも表示できます。

このセクションでは、このリリースで検証された L4 から L7 のシナリオについて説明します。L4 ～ L7 サービス内では、トポロジの数が非常に多いため、ファイアウォール、ロードバランサ、およびそれぞれの組み合わせに対してさまざまな構成を使用できます。トポロジ内の 2 つのエンドポイント間にファイアウォールが存在する場合、トラブルシューティング ウィザードはファイアウォールデータとファイアウォールからリーフへの接続を取得します。2 つのエンドポイント間にロードバランサーが存在する場合、サーバーまでではなく、ロードバランサーまでの情報を取得して表示できます。

次の表は、トラブルシューティング ウィザードで検証された L4 - L7 サービス シナリオを示しています。

シナリオ	1	2	3	4	5	6
ノード数	1	1	2	1	1	2
デバイス	GoTo FW（vrf分割）	SLBに行く	GoTo、GoTo FW、SLB	FWゴースルー	SLB-GoTo	FW、SLB（GoThrough、GoTo）
[アームの数（Number of Arms）]	2	2	2	2	2	2

シナリオ	1	2	3	4	5	6
コンシューマ	EPG	EPG	EPG	L3Out	L3Out	L3Out
プロバイダー	EPG	EPG	EPG	EPG	EPG	EPG
Device Type	VM	VM	VM	physical	physical	physical
[コントラクト範囲（Contract Scope）]	tenant	コンテキスト	コンテキスト	コンテキスト	コンテキスト	global
[コネクタ モード（Connector Mode）]	L2	L2	L2, L2	L3, L2	L3	L3 / L2,L3
[サービスの付加（Service Attach）]	BSW	BSW	DL / PC	通常の港	vPC	通常の港
[クライアント接続（Client Attach）]	FEX	FEX	FEX	レギュラーポート	レギュラーポート	通常の港
[サーバー接続（Server Attach）]	vPC	vPC	vPC	通常の港	通常の港	通常の港

エンドポイントからエンドポイントへの接続の API リスト

次のリストは、EPからEP（エンドポイントからエンドポイント）接続への使用可能なトラブルシューティング ウィザード API です：

インタラクティブ API

エンドポイント（ep）からエンドポイントへの対話型トラブルシューティング セッションを作成するには、**[対話型（interactive）]** API を使用します。モジュール名は **Troubleshooting.eptoeputils.topo** で、関数は **getTopo** です。対話型 API に必要な引数 **(req_args)** は **- session** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete

- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

createsession API

エンドポイント（ep）からエンドポイントへのトラブルシューティングセッションを作成するには、**createsession** API を使用します。モジュール名は **Troubleshooting.eptoeputils.session** で、関数は **createSession** です。

createsession API の必須引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻

- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
-action	traceroute/atomiccounter の start/stop/status など
- scheduler	
- srctenant	送信元エンドポイントのテナントの名前
- srcapp	送信元エンドポイントのアプリの名前
- srcepg	送信元エンドポイントのエンドポイント グループの名前
- dsttenant	接続先エンドポイントのテナントの名前
- dstapp	接続先エンドポイントのアプリの名前
- dstepg	接続先エンドポイントのエンドポイントの名前
- mode	内部で使用

変更セッション API

エンドポイント（ep）をエンドポイントのトラブルシューティング セッションに変更するには、**modifysession** API を使用します。モジュール名は **Troubleshooting.eptoeputils.topo** で、関数は **modifySession** です。

modifysession API に必要な引数（**req_args**）は、**[- セッション（- session）]**（セッション名）および **[- モード（- mode）]** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- starttime	トラブルシューティング セッションの開始時刻。
- endtime	トラブルシューティング セッションの終了時刻。
- latestmin	開始時刻（分単位）に始まるトラブルシューティング セッションのタイム ウインドウ
- 説明	セッションの説明

atomiccounter API

エンドポイント（ep）からエンドポイントへのアトミック カウンター セッションを作成するには、**atomiccounter API** を使用します。モジュール名は**Troubleshooting.eptoeputils.atomiccounter** で、関数は**manageAtomicCounterPols**です。

atomiccounter API に必要な引数 (**req_args**) は次のとおりです。

- - セッション
- - アクション
- - モード

（注） atomiccounter API にはオプションの引数（**opt_args**）はありません。

トレースルート API

API を使用してエンドポイント（ep）からエンドポイントのトレースルートセッションを作成するには、**[トレースルート（traceroute）]** API を使用します。モジュール名は **Troubleshooting.eptoeputils.traceroute** で、関数は **manageTraceroutePols** です。

トレースルート API に必要な引数（**req_args**）には、次のものがあります。

- - セッション（セッション名）
- - アクション（スタート/ ストップ/ ステータス）
- - モード

構文の説明

[オプションの引数（opt_args）（Optional Arguments (opt_args)）]	説明
- プロトコル	プロトコル名
- dstport	宛て先ポート名

スパン API

エンドポイント (ep) からエンドポイントまでのスパンのトラブルシューティング セッションを作成するには、**スパン** API を使用します。モジュール名は **Troubleshoot.eptoeputils.span** で、機能は **monitor** です。

スパン API の必要な引数（**req_args**）は、これらを含みます。

- - session（セッション名）

• - アクション (start/stop/status)

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティングセッションの開始時刻
- endtime	トラブルシューティングセッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティングセッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- srctenant	ソース エンドポイントのテナントの名前
- srcapp	送信元エンドポイントのアプリの名前
- srcepg	送信元エンドポイントのエンドポイント グループの名前

- dsttenant	接続先エンドポイントのテナント名
- dstapp	宛先エンドポイントのアプリの名前
- dstepg	宛先エンドポイントのエンドポイント グループの名前
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

generatereport API

API を使用してトラブルシュート レポートを生成するために**generatereport** API を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**generateReport**です。

generatereport API に必要な引数 (**req_args**) は、 **[- セッション（- session）]** (セッション名) および **[- モード（- mode）]**です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- 同梱	Obsolete
- フォーマット	生成するレポートのフォーマット

スケジュールレポート API

API を使用してトラブルシュート レポートの生成をスケジュールするために **schedulereport** API を使用します。モジュール名は **Troubleshooting.eptoeputils.report** で、関数は **scheduleReport** です。schedulereport API に必要な引数 (**req_args**) は **[- セッション（- session）]** です。

schedulereport API の必要な引数（**req_args**）は、これらを含みます：

- - session（セッション名）
- - スケジューラ（スケジューラ名）
- - mode

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻

- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- include	Obsolete
- format	生成するレポートのフォーマット
- action	traceroute/atomiccounter の start/stop/status など

getreportstatus API

API を使用して生成されたレポートのステータスを取得するには、**getreportstatus** API を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**getStatus**です。

getreportstatus API に必要な引数（**req_args** ）は次のとおりです。

- - セッション（session name）
- - sessionurl (session URL)
- - モード

（注） getreportstatus API にはオプションの引数 **(opt_args)** はありません。

getreportslist API

API を使用して生成されたレポートのリストを取得するには、**getreportslist API** を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**getReportsList**です。

getreportslist API に必要な引数（**req_args**）は、**[- セッション（- session）]**（session name）および**[- モード（ - mode）]**です。

（注） getreportslist API には、オプションの引数（**opt_args**）はありません。

getsessionslist API

API を使用してトラブルシューティング セッションのリストを取得するには、**getsessionslist API** を使用します。モジュール名は **Troubleshooting.eptoeputils.session** で、機能は **getSessions** です。

getsessionlist API の必須引数（**req_args**）は **[- モード（- mode）]** です。

（注） getsessionlist API には、オプションの引数（**opt_args**）はありません。

getsessiondetail API

API を使用してトラブルシューティング セッションに関する特定の詳細を取得するには、**getsessiondetai**l API を使用します。モジュール名は **Troubleshooting.eptoeputils.session** で、関数は **getSessionDetail** です。

getsessiondetail API に必要な引数 (**req_args**) は、**[- セッション（- session）]** (セッション名) および**[- モード（- mode）]**です。

（注） getsessiondetail API にはオプションの引数（**opt_args**）はありません。

deletesession API

API を使用して特定のトラブルシューティング セッションを削除するには、**deletesession** API を使用します。モジュール名は **Troubleshooting.eptoeputils.session** で、機能は **deleteSession** です。

deletesession API の必須引数 (**req_args**) は **[- セッション（- session）]** (セッション名) です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻

- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

clearreports API

API を使用して生成されたレポートのリストをクリアするには、**clearreports API** を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**clearReports**です。

clearreports API に必要な引数 (**req_args**) は、 **[- セッション（- session）]** (セッション名) および**[- モード（- mode）]**です。

（注） clearreports API にはオプションの引数（**opt_args**）はありません。

コントラクト API

API を使用してコントラクト情報を取得するには、**contracts** API を使用します。モジュール名は**Troubleshooting.eptoeputils.contracts**で、関数は**getContracts**です。

コントラクト API に必要な引数 （**req_args**）は、**[- セッション（- session）]**（セッション名）と**[- モード（- mode）]**です。

コントラクト API にはオプションの引数 (**opt_args**) はありません。

エンドポイントからレイヤ 3 外部接続の API リスト

次のリストは、EPからEP（エンドポイントからエンドポイント）接続への使用可能なトラブルシューティング ウィザード API です：

対話型 API

エンドポイント（ep）からレイヤー3（L3）への外部対話型トラブルシューティングセッションを作成するには、**[対話型（interactive）]** API を使用します。モジュール名は **Troubleshooting.epextutils.epext_topo**で、関数は**getTopo**です。対話型 API に必要な引数 **(req_args)** は、**[- セッション（- session）]**、**[- 含める（- include）]**、および**[- モード（- mode）]**です。

次の表にオプションの引数（**opt_args**）が表示されています：

構文の説明

[オプションの引数（opt_args）（Optional Arguments (opt_args)）]	説明
更新	

createsession API

API を使用してエンドポイント（Ep）からレイヤー 3（L3）への外部トラブルシューティング セッションを作成するには、**createsession** API を使用します。モジュール名は **Troubleshooting.epextutils.epextsession** で、関数は **createSession** です。createsession API の必須引数（**req_args**）は **- session（セッション名）** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- アクション	traceroute/atomiccounter の start/stop/status など

- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

変更セッション API

エンドポイント（Ep）をレイヤ3（L3）の外部トラブルシューティングセッションに変更するには、**modifysession API** を使用します。モジュール名は **Troubleshooting.epextutils.epextsession** で、関数は **modifySession** です。modifysession API の必須引数（**req_args**）は - session（**セッション名**）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット

- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

アトミックカウンタ API

エンドポイント（ep）からエンドポイントへのアトミック カウンタ セッションを作成するには、**atomiccounter API** を使用します。モジュール名は **Troubleshooting.epextutils.epext_ac** で、関数は **manageAtomicCounterPols** です。

atomiccounter API に必要な引数（**req_args**）は次のとおりです。

- - session（セッション名）
- - action（start/stop/status）

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）

- ui	内部で使用（無視）
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

traceroute API

API を使用してレイヤ 3 外部トレースルート トラブルシューティング セッションへのエンドポイント（ep）を作成するには、**[トレースルート（traceroute）]** API を使用します。モジュール名は **Troubleshooting.epextutils.epext_traceroute** で、関数は **manageTracerouteePols** です。

traceroute API に必要な引数（**req_args**）には、次のものがあります。

- - session（セッション名）
- - action（start/stop/status）

構文の説明

オプションの引数（**opt_args**）	説明
- protocol	プロトコル名
- dstport	宛先ポート名
- srcep	送信元エンドポイント
- dstep	宛先エンドポイント
- srcip	送信元 IP アドレス
- dstip	宛先 IP アドレス
- srcextip	送信元外部 IP アドレス
- dstIp	接続先外部 IP アドレス
- ui	内部で使用（無視）
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

スパン API

エンドポイント（Ep）からレイヤー 3（L3）への外部スパンのトラブルシューティング セッションを作成するには、**[スパン（span）]** API を使用します。モジュール名は **[Troubleshoot.epextutils.epext_span]**で、機能は**[モニタ（monitor）]**です。

スパン API の必要な引数（**req_args**）は、これらを含みます：

- - セッション（セッション名）
- - アクション（スタート/ ストップ/ ステータス）
- - モード

次のテーブルはオプションの引数（**opt_args**）とそれぞれの説明のリストです。

構文の説明

[オプションの引数（opt_args）（Optional Arguments (opt_args)）]	**説明**
- ポートリスト	ポートのリスト
- dstapic	接続先 APIC
- srcipprefix	送信元エンドポイントの IP アドレス プレフィックス
- flowid	[フローID（Flow ID）]
- dstepg	接続先 エンドポイント グループ
- dstip	接続先 エンドポイント IP アドレス
- analyser	???
- desttype	宛先タイプ（Destination type）
- spansrcports	スパン ソース ポート

generatereport API

API を使用してトラブルシュート レポートを生成するために**generatereport** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **generateReport** です。

generatereport API に必要な引数（**req_args**）は **[- セッション（- session）]**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	**説明**
- srcep	送信元エンドポイント名

- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

スケジュールレポート API

API を使用してトラブルシュート レポートの生成をスケジュールするには、**schedulereport API** を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **scheduleReport** です。schedulereport API に必要な引数（**req_args**）は **- session** です。

schedulereport API に必要な引数（**req_args**）には、以下のものが含まれます。

- - session（セッション名）
- - スケジューラ（スケジューラ名）

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント
- dstep	宛先エンドポイント
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用

- ctx	内部で使用

getreportstatus API

API を使用して生成されたレポートのステータスを取得するには、**getreportstatus** API を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**getStatus**です。

getreportstatus API に必要な引数（**req_args** ）は次のとおりです。

- - セッション（session name）
- - sessionurl (session URL)
- - モード

（注） getreportstatus API にはオプションの引数 (**opt_args**) はありません。

getreportslist API

API を使用して生成されたレポートのリストを取得するには、**getreportslist API** を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**getReportsList**です。

getreportslist API に必要な引数（**req_args**）は、 **[- セッション（- session）]**（session name）および**[- モード（ - mode）]**です。

（注） getreportslist API には、オプションの引数（**opt_args**）はありません。

getsessionslist API

API を使用してトラブルシューティング セッションのリストを取得するには、**getsessionslist** API を使用します。モジュール名は**Troubleshooting.epextutils.epextsession**で、関数は**getSessions**です。

（注） この API には必須の引数はありません。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- session	セッション名

- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

getsessiondetail API

API を使用してトラブルシューティング セッションに関する特定の詳細を取得するには、**getsessiondetai**l API を使用します。モジュール名は **Troubleshooting.epextutils.session** で、関数

は **getSessionDetail** です。getsessiondetail API の必須引数（**req_args**）は **[- セッション（-session）]** (セッション名) です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションの説明
- スケジューラ	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- 同梱	Obsolete
- フォーマット	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- アクション	traceroute / atomiccounter の開始 / 停止 / ステータスなど
- mode	内部で使用
- _dc	内部で使用

- ctx	内部で使用

deletesession API

API を使用して特定のトラブルシューティング セッションを削除するには、**deletesession** API を使用します。モジュール名は**Troubleshooting.epextutils.epextsession**で、関数は**deleteSession**です。

deletesession API に必要な引数（**req_args**）は、**[- セッション（- session）]** (セッション名) および **[- モード（- mode）]**です。

（注） deletesession API にはオプションの引数 **(opt_args)** はありません。

clearreports API

API を使用して生成されたレポートのリストをクリアするには、**clearreports API** を使用します。モジュール名は**Troubleshooting.eptoeputils.report**で、関数は**clearReports**です。

clearreports API に必要な引数 **(req_args)** は、**[- セッション（- session）]** (セッション名) および**[- モード（- mode）]**です。

（注） clearreports API にはオプションの引数（**opt_args**）はありません。

契約 API

API を使用して契約情報を取得するには、**[契約（contracts）]** APIを使用します。モジュール名は**troubleshoot.epextutils.epext_contracts**で、関数は**getContracts**です。コントラクト API に必要な引数（**req_args**）は **[- セッション（- session）]**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC

- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- epext	エンドポイントから外部へ
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用
- ui	内部で使用（無視）

ratelimit API

このセクションでは、**ratelimit** API に関する情報を提供します。モジュール名は **Troubleshooting.eptoeputils.ratelimit** で、機能は **[コントロール（control）]** です。ratelimit API に必要な引数（**req_args**）は **- action**（start/stop/status）です。

次の表に、オプションの引数 (opt_args) とそれぞれの説明を示します。

構文の説明

[オプションの引数（opt_args）（Optional Arguments (opt_args)）]	**説明**
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻。

- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻（分単位）に始まるトラブルシューティング セッションのタイム ウインドウ
- epext	エンドポイントから外部
- モード	社内で使用
- _dc	社内で使用
- ctx	社内で使用

13ext API

このセクションでは、**13ext** API に関する情報を提供します。モジュール名は **Troubleshooting.epextutils.l3ext** で、関数は **[実行（execute）]**されます。13ext API に必要な引数（**req_args**）は **[- アクション（- action）]**（start / stop / status）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- epext	エンドポイントから外部へ
- mode	内部で使用

構成同期問題をチェック中です

Cisco Application Centric Infrastructure（APIC）で要求（構成の変更など）を行うと、通常、変更が発生したことがすぐにわかります。ただし、Cisco APICで問題が発生した場合は、GUI でチェックして、まだ有効になっていないユーザー設定可能なオブジェクトに関連するトランザクションがあるかどうかを確認できます。パネルの情報を使用して、デバッグに役立てることができます。

Cisco APICGUI の **[構成オブジェクト保留中解像度（Configuration Objects Pending Resolution）]** パネルに、遅延があるかどうかが表示されます。

始める前に

手順

ステップ 1 Cisco APIC にログインします。

ステップ 2 画面の右上にある設定アイコン（歯車の記号）をクリックし、**[構成同期問題（Config Sync Issues）]** を選択します。

ステップ 3 **[構成オブジェクト保留中解像度（Configuration Objects Pending Resolution）]** パネルで、テーブルに何かがリストされているかどうかを確認します。

テーブルにエントリがない場合、同期の問題はありません。

ステップ 4 エントリがある場合は、テーブルの情報をキャプチャし、デバッグまたは Cisco サポートとの連携に使用します。

ユーザー アクティビティ の表示

管理者が Cisco APIC セットアップの変更に気付いた場合、管理者は**[ユーザー アクティビティ（User Activities）]**機能を使用して、ユーザーが実行したアクションの 2 週間の履歴を表示できます。履歴データには、アクションが発生したときのタイムスタンプ、アクションを実行したユーザー、ユーザーが実行したアクション、影響を受けるオブジェクト、および説明が含まれます。

ユーザ アクティビティをアクセス

[ユーザ アクティビティ（User Activies）] ウィンドウでは、Cisco APIC GUI で実行されたユーザー アクティビティの 2 週間の履歴を表示できます。

手順

ステップ 1 メニュー バーから、**[システム（System）]** > **[アクティブ セッション（Active Sessions）]** を選択します。

[アクティブ セッション（Active Session）] ウィンドウが表示されます。

ステップ 2 アクティブなセッションを右クリックし、**[ユーザ アクティビティ（User Activies）]** を選択します。

ユーザ アクティビティのリストが表示されます。

（注） フィールドの説明については、**[アクティブ セッション（Active Session）]**ウィンドウの右上隅のヘルプ アイコンをクリックして、ヘルプ ファイルを表示してください。

ステップ 3 最後のドロップダウン メニューの **[アクション（Actions in the last）]** をクリックして、ユーザ アクティビティを表示する履歴を選択します。

組み込み論理アナライザ モジュール

組み込み論理アナライザ モジュールについて

ELAM（組み込み論理アナライザ モジュール）は、シスコ ASIC の内部を調べ、パケットの転送方法を理解するためのエンジニアリング ツールです。ELAMは、転送パイプラインの中に組み込まれていて、パフォーマンスとコントロール プレーン リソースに影響を及ぼさずにリアルタイムでパケットをキャプチャできます。ELAM は、次の機能を実行できます。

- パケットが転送エンジンに到達したかどうかを判断する
- 受信したパケットのポートとVLANを指定
- パケットの表示（レイヤー 2 からレイヤー 4 のデータ）
- パケットが送信された場所で変更されていないかどうかを確認します

モジュラ スイッチの簡略化した出力で**ELAM** レポートを生成します

Cisco Application Policy Infrastructure Controller（APIC）4.2（1）リリースは、簡略した人間が読める ELAM 出力を紹介します。EX、FX か FX2 がスイッチ名の最後にあるスイッチ モデルしか簡略化した出力をサポートしません。モジュラ スイッチの場合は、次の手順を使用します。

手順

ステップ 1 パケット転送情報を収集するために ELAM ツールを実行します。正確なコマンドとパラメータはハードウェア次第です。

ステップ 2 オリジナルのフォーマットと簡略化したフォーマットのパケット転送情報ELAM レポートを作成するために **ereport** コマンドを実行します。

例：

```
module-1(DBG-elam-el6)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

================================================================
                    Trigger/Basic Information
================================================================
ELAM Report File    : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1(DBG-elam-el6)# exit
module-1(DBG-elam)# exit
module-1# exit

apic1-leaf11# cd /tmp/logs
apic1-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apic1-leaf11#
```

ELAM は、 /tmp/logs/ ディレクトリに出力ファイルを保存します。例の中では elam_2019-09-04-51m-13h-30s.txt ファイルは、オリジナル フォーマットの ELAM レポートです。 pretty_elam_2019-09-04-51m-13h-30s.txt ファイルは、簡略化されたフォーマットの ELAM レポートです。 ただし、簡易フォーマット ファイルは空になります。簡略化されたフォーマットでレポートを取得するには、追加の手順を実行する必要があります。

ステップ 3 元のフォーマットの ELAM レポートをスーパーバイザの /bootflash ディレクトリにアップロードします。

この例では、このレポートは elam_2019-09-04-51m-13h-30s.txt ファイルです。

ステップ 4 管理者として スーパーバイザ にログインします。

ステップ 5 ディレクトリを /tmp、または管理ユーザの書き込み権限を持つ任意のディレクトリに変更します。

例：

```
# cd /tmp
```

ステップ 6 元のフォーマットの ELAM レポートで **decode_elam_parser** コマンドを実行します。

例：

```
# decode_elam_parser /bootflash/elam_2019-09-04-51m-13h-30s.txt
```

decode_elam_parser コマンドは、簡略化された出力ファイルを現在のディレクトリに保存します。

固定したフォームファクタ スイッチの簡略化した出力でELAM レポートを生成します

Cisco Application Policy Infrastructure Controller（APIC）4.2（1）リリースは、簡略した人間が読める ELAM 出力を紹介します。EX、FX か FX2 がスイッチ名の最後にあるスイッチ モデルしか簡略化した出力をサポートしません。固定フォーム ファクタのリーフ スイッチとスパイン スイッチには、次の手順を使用します。

手順

ステップ 1 パケット転送情報を収集するために ELAM ツールを実行します。正確なコマンドとパラメータはハードウェア次第です。

ステップ 2 オリジナルのフォーマットと簡略化したフォーマットのパケット転送情報ELAM レポートを作成するために **ereport** コマンドを実行します。

例：

```
module-1(DBG-elam-insel6)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

===============================================================
                    Trigger/Basic Information
===============================================================
ELAM Report File    : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1(DBG-elam-insel6)# exit
module-1(DBG-elam)# exit
module-1# exit

apic1-leaf11# cd /tmp/logs
apic1-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apic1-leaf11#
```

ELAM は、 /tmp/logs/ ディレクトリに出力ファイルを保存します。. 例の中では elam_2019-09-04-51m-13h-30s.txt ファイルは、オリジナル フォーマットの ELAM レポートです。 pretty_elam_2019-09-04-51m-13h-30s.txt ファイルは、簡略化されたフォーマットの ELAM レポートです。

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。