GETVPN に関する一般的な問題のトラブルシューティング

内容

概要

このドキュメントでは、Group Encrypted Transport VPN（GETVPN）でよく見られる問題を解決するために収集すべきデバッグについて説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- GETVPN
- Syslog サーバの使用

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

背景情報：GETVPN トラブルシューティング ツール

GETVPN は、トラブルシューティング プロセスを容易にする多数のトラブルシューティング ツールのセットです。それぞれのトラブルシューティング タスクでどんなツールが使用可能か、またどんな場合にそのツールが適しているかを理解することが重要です。トラブルシューティングを行う際、実稼働環境に悪影響を及ぼさないように、最初の段階では最も影響の少ない方法を使用することをお勧めします。このプロセスを支援するために、よく使用されるいくつかのツールについてこの項で説明します。

コントロール プレーン デバッグ ツール

show コマンド

表示コマンドは、GETVPN 環境でのランタイム動作を表示するためによく使用されます。

Syslog

GETVPN には、重要なプロトコル イベントやエラー状態に関する多数の syslog メッセージがあります。デバッグを実行するには、その前に必ずこれを最初に調べてください。

Group Domain of Interpretation（GDOI）イベント トレース

この機能はバージョン 15.1(3)T で追加されました。イベント トレースは、重要な GDOI イベントと エラーに関する軽量の、常時オンになっているトレースです。例外状態のトレースバックが有効になっている出口パス トレースもあります。

GDOI 条件付きデバッグ

この機能はバージョン 15.1(3)T で追加されました。この機能により、ピア アドレスに基づいて特定のデバイスをフィルタリング付きでデバッグできます。特にキー サーバでは、可能な限りこれを常に使用してください。

グローバル暗号化および GDOI デバッグ

これらは、様々な GETVPM デバッグ機能すべてです。管理者は、大規模な環境でデバッグする際に注意が必要です。GDOI デバッグには、デバッグの詳細度を指定する 5 つのデバッグ レベルがあります。

```
GM1#debug crypto gdoi gm rekey ?
all-levels All levels
detail Detail level
error Error level
event Event level
packet Packet level
terse Terse level
```

デバッグレベル	得られる情報
エラー	エラー状態
Terse	ユーザおよびプロトコルの問題に関する重要なメッセージ
[Event]	キー再生成の送受信などのイベントおよび状態遷

	移
詳細（Outcall Billing Detail）	最も詳細なデバッグ メッセージ情報
パケット	詳細なパケット情報のダンプを含む
すべて	上記のすべて

データ プレーン デバッグ ツール

いくつかのデータ プレーン デバッグ ツールを次に示します。

- アクセス リスト
- IP Precedence アカウンティング
- NetFlow
- Interface Counters
- 暗号化カウンタ
- IP Cisco Express Forwarding（CEF）グローバルおよび機能ごとのドロップ カウンタ
- 組み込みパケット キャプチャ（EPC）
- データ プレーン デバッグ（IP パケットおよび CEF のデバッグ）

トラブルシュート

ロギング機能の準備および他のベスト プラクティス

トラブルシューティングを開始する前に、次の説明に従ってロギング機能を必ず準備してください。また、いくつかのベスト プラクティスも示します。

- ルータの空きメモリ量を確認し、[logging buffered debugging] **を大きな値（可能であれば 10 MB 以上）に設定します。**
- コンソール、モニタ、syslog サーバへのロギングを無効にします。
- バッファ再使用が原因でログが失われるのを防ぐために、**show log コマンドを使ってロギング バッファの内容を一定間隔（20 分 ~ 1 時間）ごとに取得します。**
- 何らかの現象が発生した場合、影響を受けたグループ メンバー（GM）およびキーサーバ（KS）から **show tech コマンドを入力します。また、グローバルに、および該当する Virtual Routing and Forwarding（VRF）ごとに show ip route コマンドの出力を必要に応じて調べます。**

- デバッグされるすべてのデバイス間でクロックを同期するために、ネットワーク タイム プロトコル（NTP）を使用します。デバッグ メッセージとログ メッセージに関してミリ秒（msec）タイムスタンプを次のように有効にします。

```
service timestamps debug datetime msec
service timestamps log datetime msec
```

- 表示コマンドの出力にタイムスタンプが付いていることを確認します。

```
Router#terminal exec prompt timestamp
```

- コントロール プレーン イベントまたはデータ プレーン カウンタに関する表示コマンド出力を収集するときには、必ず同じ出力を何度か反復して収集します。

IKE 確立のトラブルシューティング

登録プロセスが最初に開始するとき、GM と KS は GDOI トラフィックを保護するためにインターネット キー エクスチェンジ（IKE）セッションをネゴシエートします。

- GM で、IKE が正常に確立されていることを次のように確認します。

```
gm1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst src state conn-id status
172.16.1.9 172.16.1.1 GDOI_REKEY 1068 ACTIVE
172.16.1.1 172.16.1.9 GDOI_IDLE 1067 ACTIVE
```

注：登録の基本である GDOI_IDLE 状態は、初期登録後に必要でなくなるため、すぐにタイムアウトになって消えます。

- KS に次のように表示されます。

```
ks1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst src state conn-id status
172.16.1.1 172.16.1.9 GDOI_IDLE 1001 ACTIVE
```

注：キー再生成セッションは KS で必要なときにのみ表示されます。

この状態に達しない場合、次の手順を実行してください。

- 障害の原因を分析して理解するために、このコマンドの出力を確認します。
  `router# show crypto isakmp statistics`
- 前のステップで役立つ情報が得られない場合、通常の IKE デバッグを有効にすると、プロトコル レベルの情報が得られます。
  `router# debug crypto isakmp`

注：
* IKE は通常の UDP/500 ポートではなく UDP/848 で使用されます。
* このレベルで問題が見つかったら、KS および影響を受ける GM の両方のデバッグを提供してください。

- グループ キー再生成用に Rivest-Shamir-Adleman（RSA）シグニチャに依存するため、KS では RSA キーが設定されている**必要があり、グループ設定で指定された名前と同じでなければなりません。**

  これを確認するには、次のコマンドを入力します。

```
ks1# show crypto key mypubkey rsa
```

初期登録のトラブルシューティング

GM で、登録ステータスを確認するためにこのコマンドの出力を確認します。

```
gm1# show crypto gdoi | i Registration status
Registration status  : Registered
gm1#
```

Registered（登録済み）以外のステータスが出力に表示される場合、次のコマンドを入力します。

GM で、次のようにします：

- 暗号化が有効になっているインターフェイスをシャットダウンします。
  注意：アウトオブバンド管理が有効になっていると想定されます。

- これらのデバッグを有効にします：

```
gm1# debug crypto gdoi infra packet
gm1# debug crypto gdoi gm packet
```

- KS 側でデバッグを有効にします（次の項を参照）。

- KS デバッグの準備ができたら、暗号化が有効になっているインターフェイスのシャットダウンを解除し、登録を待ちます（そのプロセスを加速するには GM で clear crypto gdoi コマンドを発行します）。

KS で、次のようにします：

- KS での RSA キーの存在を次のように確認します。

```
ks1# show crypto key mypubkey rsa
```

- これらのデバッグを有効にします：

```
ks1# debug crypto gdoi infra packet
ks1# debug crypto gdoi ks packet
```

ポリシーに関連する問題のトラブルシューティング

(フェールクローズ ポリシーに関連して) 登録前に発生するポリシーの問題

この問題は GM のみに影響を与えるため、GM からこの出力を収集します：

```
gm1# show crypto ruleset
```

> **注**：Cisco IOS-XE では、パケットの分類がソフトウェアで行われないため、この出力は常に空です。

該当するデバイスからの **show tech コマンド出力に、必要な残りの情報が示されます。**

プッシュされるグローバル ポリシーに関連して、登録後に発生するポリシーの問題

通常、この問題は次の 2 つの形態で発生します。

- KS が GM にポリシーをプッシュできない。
- さまざまな GM にポリシーが部分的に適用される。

これらの問題をトラブルシュートするには、いずれの場合も次の手順を実行します。

1. 影響を受けている GM で、この出力を収集します。

```
gm1# show crypto gdoi acl
gm1# show crypto ruleset
```

2. GM でこれらのデバッグを有効にします。

```
gm1# debug crypto gdoi infra packet
gm1# debug crypto gdoi gm acls packet
```

3. 影響を受けている GM の登録先 KS で、この出力を収集します。

```
ks1# show crypto gdoi ks members
```

```
ks1# show crypto gdoi ks policy
```

注：GM の接続先である KS を識別するには、**show crypto gdoi group コマンドを入力します。**

4. 同じ KS で、これらのデバッグを有効にします。

```
ks1# debug crypto gdoi infra packet
ks1# debug crypto gdoi ks acls packet
```

5. GM でこのコマンドを使って GM の登録を強制的に実行します。

```
clear crypto gdoi
```

グローバル ポリシーとローカル オーバーライドのマージに関連して、登録後に発生するポリシーの問題

この問題が発生すると、通常、暗号化すべきでないとローカル ポリシーで指示されている暗号化パケット（またはその逆）が受信されたというメッセージが表示されます。この場合、前の項で必要となったすべてのデータに加えて show tech コマンド出力も必要になります。

キー再生成の問題のトラブルシューティング

GM で、次のようにします：

- これらのデバッグを収集します。

```
gm1# debug crypto gdoi infra packet
gm1# debug crypto gdoi gm packet
gm1# debug crypto gdoi gm rekey packet
```

- タイプ GDOI_REKEY の IKE セキュリティ アソシエーション（SA）が GM にまだ存在することを確認するために、このコマンドを入力します。

```
gm1# show crypto isakmp sa
```

KS で、次のようにします：

- **それぞれの KS から show crypto key mypubkey rsa コマンド出力を収集します。**それらのキーは同一であることが想定されます。

- KS で何が発生するかを表示するために、これらのデバッグを入力します。

```
ks1# debug crypto gdoi infra packet
ks1# debug crypto gdoi ks packet
ks1# debug crypto gdoi ks rekey packet
```

時間ベースのアンチリプレイ（TBAR）のトラブルシューティング

TBAR 機能では、グループ間で同じ時刻を維持する必要があるため、GM は疑似時刻クロックを常に再同期させる必要があります。キー再生成中（またはキー再生成が行われない場合は 2 時間おきに）これが実行されます。

> **注**：GM および KS 両方の出力とデバッグが適切に関連するよう、これらをすべて同時に収集する必要があります。

このレベルで発生する問題を調査するには、この出力を収集します。

- GM で、次のようにします：

```
gm1# show crypto gdoi
gm1# show crypto gdoi replay
```

- KS で、次のようにします：

```
ks1# show crypto gdoi ks members
ks1# show crypto gdoi ks replay
```

より動的な方法で TBAR 時刻維持を調査するには、これらのデバッグを有効にします。

- GM で、次のようにします：

```
gm1# debug crypto gdoi gm rekey packet
gm1# debug crypto gdoi replay packet (verbosity might need to be lowered)
```

- KS で、次のようにします：

```
ks1# debug crypto gdoi ks rekey packet
ks1# debug crypto gdoi replay packet (verbosity might need to be lowered)
```

Cisco IOS バージョン 15.2(3)T 以降、TBAR エラーを記録する機能が追加されたため、これらのエラーを特定しやすくなりました。GM で、TBAR エラーがあるかどうか確認するためにこのコマンドを使用します。

```
R103-GM#show crypto gdoi gm replay
Anti-replay Information For Group GETVPN:
 Timebased Replay:
   Replay Value             : 512.11 secs
   Input Packets            : 0        Output Packets           : 0
   Input Error Packets    : 0        Output Error Packets   : 0
   Time Sync Error          : 0        Max time delta           : 0.00secs

TBAR Error History (sampled at 10pak/min):
```

```
No TBAR errors detected
```

TBAR の問題をトラブルシュートする方法について、詳しくは時間ベースのアンチリプレイの障害を参照してください。

KS 冗長性のトラブルシューティング

Cooperative（COOP）は、KS 間通信を保護するために IKE セッションを確立します。このため IKE 確立に関してすでに説明したトラブルシューティング技法がここでも当てはまります。

COOP 固有のトラブルシューティングを行うには、該当するすべての KS でこのコマンドの出力を確認します。

```
ks# show crypto gdoi ks coop
```

> **注**：COOP KS の導入でよくある間違いは、すべての KS のグループ用に同じ RSA キー（秘密キーと公開キーの両方）をインポートし忘れることです。この場合、キー再生成中に問題が発生します。公開キーを確認して KS の間で比較するには、各 KS からの **show crypto key mypubkey rsa コマンド出力を比べます。**

プロトコル レベルのトラブルシューティングが必要な場合、該当するすべての KS でこのデバッグを有効にします。

```
ks# debug crypto gdoi ks coop packet
```

FAQ

「% Setting rekey authentication rejected」というエラー メッセージが表示されるのはなぜですか

以下の行が追加された後で KS を設定すると、このエラー メッセージが表示されます。

```
KS(gdoi-local-server)#rekey authentication mypubkey rsa GETVPN_KEYS
% Setting rekey authentication rejected.
```

通常、このエラー メッセージの原因は GETVPN_KEYS というキーが存在しないことです。これを修正するには、次のコマンドを使用して、正しいラベルのキーを作成します。

```
crypto key generate rsa mod <modulus> label <label_name>
```

> **注**：これが COOP 導入である場合はエクスポート可能（exportable）キーワードを末尾に追加し、別の KS で同じキーをインポートします。

ある GETVPN グループの KS として設定されたルータは、同じグループの GM としても機能できますか

いいえ。すべてのGETVPN導入には、同じグループのGMとして参加できない専用のKSが必要です。この機能がサポートされない理由は、GM 機能が追加された KS で暗号化、ルーティング、QoS などのあらゆる対話が発生する可能性があるという状況は、この重要なネットワーク デバイスの健全性にとって不適切であるためです。GETVPN 導入全体を機能させるには、これが常に利用可能でなければなりません。

関連情報

- Group Encrypted Transport VPN (GET VPN) - Cisco Systems
- テクニカル サポートとドキュメント – Cisco Systems